Chapter 7

Partial Differential Equation

CHAPTER 7 ..... **1**

PARTIAL DIFFERENTIAL EQUATION **1**

7.1 CHARACTERISTICS OF PDE 4
EXERCISE 7.1 8
7.2 NUMERICAL METHODS FOR PDES 8
7.2.1 Diffusion and Advective-Diffusion Equation *8*
7.2.2 Laplace Equation *23*
7.2.3 1st Order Wave Equation *32*
7.2.4 2nd Order Wave Equation *36*
7.2.5 Diffusion Equation in 2-D *39*
EXERCISE 7.2 42
7.3 STABILITY ANALYSIS 45
7.3.1 von Neumann Stability Analysis *49*
EXERCISE 7.3 54
7.4 SUMMARY 54

Batman goes paragliding. While flying around with his *Batwings*, he found out that some of the para-gliders were maneuvering better than him by using the thermals in the mountain terrain. So, he decided to learn paragliding to improve his flying and turning techniques. He realized with *Batmanesque* instinct that it is important to precisely locate the thermals. While studying the temperature distribution in atmosphere, he came across a number of partial differential equations. For example, the energy balance equation takes the form:

$$\rho C_v\left(\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}+w\frac{\partial T}{\partial z}\right)-\alpha\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}\right)=-p\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)+\delta\pm\sigma \quad (7.1)$$

where, six state variables are u, v, w, p, ρ and T. The u, v, and w are wind velocities in the directions x, y and z, respectively; p is the pressure; ρ is the density; and T is the temperature. The parameters and constants are heat capacity at constant volume C_v, thermal conductivity α, heat generated due to dissipation δ and any source or sink σ. In order to solve for six state variables, he needed five more equations. These are provided by continuity equation (one) and momentum equations (three) written for compressible Newtonian fluid in a gravitational field, and a constitutive relationship for ideal gas relating the pressure and density as $p\frac{m}{\rho}=RT$. *Batman* quickly learned all that atmospheric fluid mechanics. If you want to learn that too, you may consult

other books, for example Seinfeld (1986). However, in this book, we shall concentrate on numerical solution of these equations.

In the beginning, the full set of equations was too complicated even for the *Batman*. So, he needed some *spherical cow* approximations. With *Boussinesq approximations* and adiabatic equilibrium assumption, one obtains the temperature distribution equation as:

$$rC_p\left(\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+v\frac{\partial T}{\partial y}+w\frac{\partial T}{\partial z}\right)-a\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}\right)=\pm s \tag{7.2}$$

This is the equation for heat transfer in an incompressible fluid. Near the mountain ridge, he was interested in getting the lift along vertical planes parallel to the ridge. In these cases, he was only interested in thermal gradients along the vertical planes running parallel to the ridge. So, he could get rid of one of the horizontal dimension perpendicular to the ridge whenever he wanted to simulate the temperature contours near the ridge and obtain:

$$rC_p\left(\frac{\partial T}{\partial t}+u\frac{\partial T}{\partial x}+w\frac{\partial T}{\partial z}\right)-a\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial z^2}\right)=\pm s \tag{7.3}$$

Lastly, when he wanted to know only the vertical distribution of temperature as he rose from the ground, he only needed to keep the vertical dimension leading to:

$$rC_p\left(\frac{\partial T}{\partial t}+w\frac{\partial T}{\partial z}\right)-a\frac{\partial^2 T}{\partial z^2}=\pm s \tag{7.4}$$

If the flying condition was quiescent (no wind) for long time such that a steady state temperature distribution was established, the equations in 3-d and 2-d for such cases were:

$$\begin{aligned} &a\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}\right)=\pm s \\ &a\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial z^2}\right)=\mp s \end{aligned} \tag{7.5}$$

Furthermore, if there were no source or sink in the region for the heat energy, he needed only to solve the classic Laplace equations:

$$\begin{aligned} &\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}=0 \\ &\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial z^2}=0 \end{aligned} \tag{7.6}$$

These later equations must be looking familiar to the readers as they come across similar equations in many engineering disciplines. Advection-dispersion equations (7.2-4) and Laplace equations (7.5-6) are some of the most common PDEs in engineering applications. Having learned the equations, *Batman* now wants to

program his *Batwing* computer for their solutions so that the thermals or the temperature contours in the region are displayed on his screen as he is flying. He requires fast and accurate numerical algorithme to do that. And …… as you can guess, here he is with us, studiously learning again.

Before we venture into any numerical methods, let us recall some of the commonly used terms in PDE:

Order of a PDE is the highest order of the derivative present in the equation.

Degree of a PDE is the highest power to which the highest order term is raised in the equation.

A PDE of degree "one" with no term(s) containing the product(s) of dependent variable and its derivative(s) is *linear*. Otherwise, it is *nonlinear*.

A PDE is *Homogeneous* if it does not contain any term which is a constant or a function of only independent variables. *Nonhomogeneous* PDE on the other hand contains a term involving only the independent variable(s) or a constant.

Some examples of the categorization of PDEs are given below:

2nd Order, 1st Degree, Linear, Homogeneous PDE:

$$x^2 \frac{\partial^2 f}{\partial x \partial t} + e^{-t} \frac{\partial f}{\partial x} + f = 0 \tag{7.7}$$

1st Order, 2nd Degree, Nonlinear, Homogeneous PDE:

$$\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 = 0 \tag{7.8}$$

2nd Order, 1st Degree, Nonlinear, Non-homogeneous PDE:

$$\frac{\partial f}{\partial t} + f \frac{\partial f}{\partial x} - a \frac{\partial^2 f}{\partial x^2} = f(x,t) \tag{7.9}$$

2nd Order, 1st Degree, Linear, Non-homogeneous PDE:

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f(x,y) \tag{7.10}$$

First hurdle for the *Batman* was to recognize the natural boundary conditions for the specific problems and the necessity of boundary conditions for obtaining a meaningful solution for a PDE. In order to familiarize the readers, we shall first present brief review of the characteristic analysis of PDEs. Then, we shall venture into methods for solving some of the commonly encountered form of PDEs in the engineering problems. In this, we will only consider the methods for linear PDEs. We will show how to analyze the methods for stability and convergence. Lastly we

will address how one can use the methods for linear PDEs in order to solve non-linear problems.

7.1 Characteristics of PDE

Partial differential equations typically describe a physical process in the space-time continuum. Every physical process often have well defined boundaries in space and suitably chosen landmarks in time where the conditions are known. These serve as natural boundary and initial conditions. However, one often wonders whether it is possible to solve a PDE given a set of boundary or initial conditions. Conversely, where and what kind of boundary and initial conditions are required to solve a particular PDE in a certain space-time domain? The analysis of characteristics help us to understand this question. In this section, we will introduce the readers to the concept of characteristics and how it can help to determine the type and location of initial and boundary conditions required to solve a PDE. We shall limit ourselves to introduction because a detailed analysis is beyond the scope of an introductory numerical analysis book and there are many good texts available for detailed analysis of characteristics for various types of PDEs, e.g. Celia and Gray (1992).

The *characteristics* of a PDE are non-intersecting lines or planes or hyper-planes (depending on the number of independent variables) along which the information propagates and no information propagates across them. It is then easy to visualize that one must specify the initial and boundary conditions in such a way so that they cut across the characteristics and information can flow from every point of the boundary. Alternatively, if the boundary happens to run along the characteristics, no information will flow from this boundary into the domain which lies across.

Let us introduce the concept of characteristics with a first order PDE as follows:

$$\frac{\partial f}{\partial t} + u\frac{\partial f}{\partial x} = 0 \tag{7.11}$$

Recall the definition of characteristics along which the information propagates from the boundary. Let us denote these curves as $y(x,t) = K$, where, K is a constant. Therefore, the change along this curve is zero. Mathematically, this can be expressed as:

$$dy = 0 = y_t dt + y_x dx \quad \text{or} \quad \frac{dt}{dx} = -\frac{y_x}{y_t} \tag{7.12}$$

Small change of f can be expressed in terms of its partial derivative as:

$$df = f_t dt + f_x dx \tag{7.13}$$

If the change in f is considered along the curve $y(x,t) = K$, then using the relations in equations (7.11) and (7.12) in (7.13), one obtains,

$$\left(1+u\frac{y_x}{y_t}\right)f_x=\left.\frac{df}{dx}\right|_{y(x,t)=K} \tag{7.14}$$

If the above equation has a solution, that would mean specifying a condition along $y(x,t)=K$ would enable computation of f_x at any point. By definition of characteristics, this cannot be true if $y(x,t)=K$ is indeed a characteristic. This would mean that the coefficient on the left must be zero when the equation does not have a solution. This gives the equation of characteristics as:

$$\left(1+u\frac{y_x}{y_t}\right)=0 \text{ or } y_t+uy_x=0 \tag{7.15}$$

The above has solutions of $x - ut = K$, where K is a constant. In a time space domain, these are parallel straight lines with a slope of u. Interesting is the case of $K = 0$. Along this line, $x = ut$ or $t = x/u$. An initial condition which is specified for all x at $t = 0$ will influence the solution in the region $x > ut$. The boundary condition which is specified for all t at $x = 0$ will influence the solution in the region $t > x/u$. These regions are schematically shown in Figure 7.1. Therefore, both initial and boundary conditions are required to obtain solution in the complete space-time continuum. If any condition is specified along $x - ut = K$, we will not be able to obtain solution for this PDE at all times and at all locations as no information will propagate across these lines. As seen in Figure 7.1, the characteristics must intersect the curves along which the conditions are specified so as to pick up the information from these curves and propagate them into the domain while satisfying the PDE.

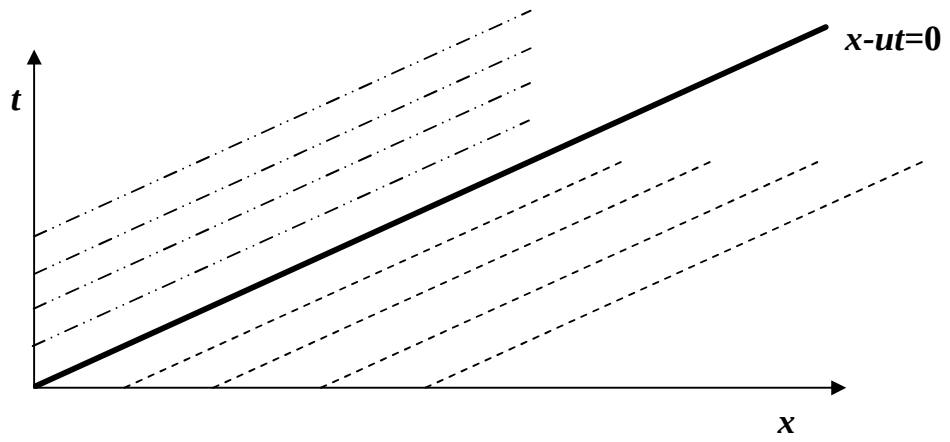


Figure 7.1: Influence of boundary and initial conditions in the solution domain. Initial condition is specified along x axis at t = 0. Dashed line shows the region of influence of the initial condition. Boundary condition is specified along t axis at x = 0. Region of influence of the boundary condition is shown by the chain dotted line.

Let us now consider a general 2nd order PDE with two independent variables:

$$a f_{xx} + 2b f_{xy} + g f_{yy} + q f_x + w f_y + r(f,x,y) = 0 \tag{7.16}$$

Recall in Chapter 6, we converted the higher order IVPs to first order by defining the lower order derivatives as new variables. Let us define similar variables to reduce the above PDE to first order. We define as follows:

$$x = f_x \text{ and } h = f_y \tag{7.17}$$

This automatically leads to $x_y = h_x$ (7.18)

Using these, the PDE can be written as:

$$a x_x + b x_y + b h_x + g h_y + q f_x + w f_y + r(f,x,y) = 0 \tag{7.19}$$

Similar to the characteristics analysis shown for the 1st order problem, let us once again consider $y(x,y) = K$ as the characteristics such that 7.12 is true. The changes in the three variables of this PDE can be expressed similar to equation (7.13):

$$df = f_x dx + f_y dy \tag{7.20}$$

$$dx = x_x dx + x_y dy \tag{7.21}$$

$$dh = h_x dx + h_y dy \tag{7.22}$$

By considering the changes along $y(x,y) = K$, one can derive a relation (7.23) similar to equation (7.12). Using this relation in equations (7.20-22), we obtain the relations in (7.24-26).

$$\frac{dy}{dx} = -\frac{y_x}{y_y} \tag{7.23}$$

$$f_x = f_y \frac{y_x}{y_y} + \left.\frac{df}{dx}\right|_{y(x,y)=K} \tag{7.24}$$

$$x_x = x_y \frac{y_x}{y_y} + \left.\frac{dx}{dx}\right|_{y(x,y)=K} \tag{7.25}$$

$$h_x = h_y \frac{y_x}{y_y} + \left.\frac{dh}{dx}\right|_{y(x,y)=K} \tag{7.26}$$

Now it is possible to eliminate the partial derivatives with respect to *x* from the equation (7.19) and only retain the partial derivatives with respect to *y* as unknowns. Substituting the relations (7.24-26) in the equation (7.19), we obtain:

$$\left(b + a\frac{y_x}{y_y}\right)x_y + \left(g + b\frac{y_x}{y_y}\right)h_y + \left(w + q\frac{y_x}{y_y}\right)f_y + r(f,x,y) = \\ -r(f,x,y) - a\left.\frac{dx}{dx}\right|_{y(x,y)=K} - b\left.\frac{dh}{dx}\right|_{y(x,y)=K} - q\left.\frac{df}{dx}\right|_{y(x,y)=K} \tag{7.27}$$

From 7.17 and 7.26, we can also write:

$$x_y - h_y\frac{y_x}{y_y} = \left.\frac{dh}{dx}\right|_{y(x,y)=K} \tag{7.28}$$

Lastly, the definition $f_y = h$ (7.29)

From equations 7.27-29, the partial derivatives of f, x, and h with respect to y can not be solved for if the determinant of the coefficient matrix is zero. This will define the characteristics as:

$$\begin{vmatrix} b + a\frac{y_x}{y_y} & g + b\frac{y_x}{y_y} & w + q\frac{y_x}{y_y} \\ 1 & -\frac{y_x}{y_y} & 0 \\ 0 & 0 & 1 \end{vmatrix} = 0 \tag{7.30}$$

This leads to the equation:

$$a\left(\frac{y_x}{y_y}\right)^2 + 2b\frac{y_x}{y_y} + g = 0 \tag{7.31}$$

The solution or the roots are:

$$\frac{y_x}{y_y} = \frac{-b \pm \sqrt{b^2 - ag}}{a} \tag{7.32}$$

In the equation 7.19, if we eliminated the partial derivatives with respect to y and solved for the partial derivatives with respect to x, we will obtain the same solution for the characteristics as equation 7.32 for $\frac{y_y}{y_x}$. The characteristics then depend on the determinant as follows:

$b^2 - ag > 0$: Two unique real characteristics and the PDE is called hyperbolic. Examples are second order wave equation where, $a = -c^2, g = 1, b = 0$.

$b^2 - ag = 0$: One unique real characteristics and the PDE is called parabolic. Examples are diffusion and advection diffusion equations where, $a = K, g = 0, b = 0$.

$b^2 - ag < 0$: Two complex conjugate characteristics and the PDE is elliptic. Example is Laplace equation where, $a = 1, g = 1, b = 0$.

From the exact values of a, b and g, one can determine the characteristics and ensure that the boundaries intersect them. Detail discussions of characteristics for specific equations are beyond the scope of this book. Interested reader may refer to Celia and Gray (1992) for elaborate presentations of boundary condition requirements based on characteristics analysis. In the following sections, we will develop methods for the solution of PDEs and demonstrate some analysis techniques for assuring convergence.

Exercise 7.1

1. Using equation (7.32) determine the characteristics of the Laplace equation. Draw the region indicating what type of boundary conditions are necessary to solve the equation.

2. Determine the characteristics of the advection-diffusion equation (7.34). Now set a = 0 and determine the characteristics again. This new sets of characteristics are termed as secondary characteristics of the equation. State under what circumstances the primary or secondary characteristics would influence the solution. Also identify the differences in the behaviour of these two sets of characteristics.

3. Set u = 0 and obtain the characteristics for pure diffusion problem.

7.2 Numerical Methods for PDEs

In this section, we shall consider PDEs commonly encountered in the engineering problems. In order to develop the numerical methods, we will draw heavily on the concepts developed in the earlier chapters. Readers are advised to familiarize themselves on these concepts before venturing onto solving PDEs. Specifically, we shall use concepts of ODEs developed in Chapter 6, finite difference approximation of derivatives developed in Chapter 5 and solution of linear systems of equation developed in Chapter 2. Goal is to obtain a meaningful solution at discrete location that satisfies the PDE as well as the boundary and initial conditions. PDEs will be first decomposed into ODEs and further onto linear algebraic equations. For the analysis, we will use the concept of Fourier representation of a function developed in Chapter 4. We divide the section into subsections covering different types of PDEs.

7.2.1 Diffusion and Advective-Diffusion Equation

In this section, we will develop numerical methods for the solution of two linear parabolic equations namely, diffusion equation and advective-diffusion (also called advection-dispersion) equation. These are shown below:

$$\frac{\partial f}{\partial t} = a \frac{\partial^2 f}{\partial x^2} \tag{7.33}$$

$$\frac{\partial f}{\partial t} + u\frac{\partial f}{\partial x} = a\frac{\partial^2 f}{\partial x^2} \tag{7.34}$$

where, u and a may be functions of the independent variables. For the solution, each equation also requires two boundary conditions and one initial condition. These equations contain two independent variables in the form of time (t) and space (x). These two independent variables have very different physical significance. For example, range of the space variable is the entire real line $(-\infty,\infty)$ while for most of the practical problems, the time only progresses forward from any arbitrary initial starting point, that is in $(0,\infty)$. Therefore, while one can use all the forward, backward and central difference approximations developed in Chapter 5 for derivatives involving the space variable, the methods for initial value problems developed in Chapter 6 need to be used for the derivative involving time variable. Since, space and time both are independent variables, it is possible to decouple the approximations in these two variables. In order to understand this, let us consider a finite domain (0, L) for the space variable x. For all approximations of the dependent variable, we shall use the subscript index for the space variable and the superscript index for the time variable. Therefore, the notation f_i^n would indicate the value of f at i^{th} location at n^{th} time step. In order to write difference approximations, we need to lay a grid on the domain (in other words, choose the points in the domain at which the approximate solutions will be obtained). Let us choose a finite number (m+1) of equally spaced points (Figure 7.2). Therefore, the length of each interval between two successive point is $\Delta x = L/m$.

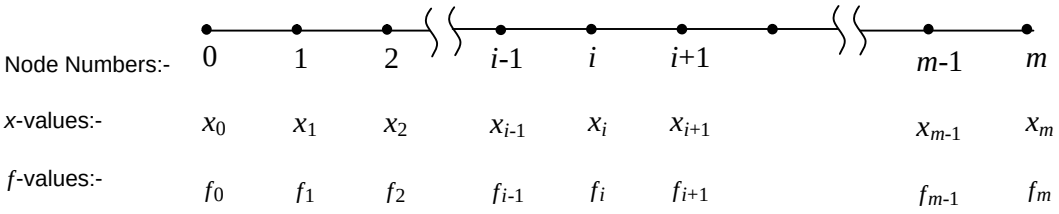


Figure 7.2: A grid for the spatial variable.

One can now write finite difference approximation for the space derivatives in the equations. If we choose 2nd order central difference scheme for approximations, we can write the following for each interior points (i) in the grid:

Diffusion Equation: $$\frac{df_i}{dt} = a_i\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2} \tag{7.35}$$

Advection Diffusion Equation: $$\frac{df_i}{dt} = -u_i\frac{f_{i+1} - f_{i-1}}{2\Delta x} + a_i\frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2} \tag{7.36}$$

Notice, that we converted the time derivatives into total derivative. This is because, we have already made discrete approximation of the space derivatives and each f_i is only a function of time. The partial differential equations are now converted to a system of coupled initial value problems with f_i's as dependent variables. In order to

observe this, let us divide the solution domain into four equal divisions such that we have 5 nodes (numbered 0-4), 2 at the boundaries (0 & 4) and 3 in the interiors (1, 2, 3). The time dependent variables are then f_0, f_1, f_2, f_3 and f_4. For illustration purpose, we shall also assume two boundary conditions as $f(0,t)=c_0$ and $f(L,t)=c_L$. These boundary conditions essentially mean, $f_0 = c_0$ and $f_4 = c_L$, that is, the values at two boundaries remain constant at all times. Later on we will discuss in detail various types of boundary conditions and how to incorporate them in the discrete form. At present, we shall write the discrete forms of eq (7.33-34) by incorporating the above boundary conditions:

Diffusion Equation:
$$\begin{bmatrix} \frac{df_1}{dt} \\ \frac{df_2}{dt} \\ \frac{df_3}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{2a_1}{\Delta x^2} & \frac{a_1}{\Delta x^2} & 0 \\ \frac{a_2}{\Delta x^2} & -\frac{2a_2}{\Delta x^2} & \frac{a_2}{\Delta x^2} \\ 0 & \frac{a_3}{\Delta x^2} & -\frac{2a_3}{\Delta x^2} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} + \begin{bmatrix} \frac{c_0 a_1}{\Delta x^2} \\ 0 \\ \frac{c_L a_3}{\Delta x^2} \end{bmatrix} \tag{7.37}$$

Advection-Diffusion Equation:
$$\begin{bmatrix} \frac{df_1}{dt} \\ \frac{df_2}{dt} \\ \frac{df_3}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{2a_1}{\Delta x^2} & -\frac{u_1}{2\Delta x}+\frac{a_1}{\Delta x^2} & 0 \\ \frac{u_2}{2\Delta x}+\frac{a_2}{\Delta x^2} & -\frac{2a_2}{\Delta x^2} & -\frac{u_2}{2\Delta x}+\frac{a_2}{\Delta x^2} \\ 0 & \frac{u_3}{2\Delta x}+\frac{a_3}{\Delta x^2} & -\frac{2a_3}{\Delta x^2} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} + \begin{bmatrix} \frac{u_1 c_0}{2\Delta x}+\frac{c_0 a_1}{\Delta x^2} \\ 0 \\ -\frac{u_3 c_L}{2\Delta x}+\frac{c_L a_3}{\Delta x^2} \end{bmatrix} \tag{7.38}$$

We can write both the equations in the form of:

$$\frac{\mathbf{d}\overline{f}}{\mathbf{dt}} = \mathbf{A}\overline{f} + \mathbf{b} \tag{7.39}$$

where, $\overline{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$. This constitutes a system of IVP (6.120) where initial condition vector is given by the initial condition of the PDE. Therefore, through discretization of space variable, we have reduced a PDE into a system of IVP. This is commonly termed as *semi-discretization* since we have discretized only one of the two independent variables in the original PDE. For time discretization, one can use any of the methods described in the previous chapter. For example, application of Euler Forward leads to eq (6.115) and Euler Backward yields (6.121). Instead of repeating the same equations here, let us write a weighted parameterized form that becomes Euler's methods or Trapezoidal method depending on the parameter value. Using the superscript for the time step index, we write the parameterized scheme for equation (7.39) as follows:

$$\overline{f}^{n+1} = \overline{f}^{n} + \Delta t\left[m\left\{\mathbf{A}^{n}\overline{f}^{n} + \mathbf{b}^{n}\right\} + (1-m)\left\{\mathbf{A}^{n+1}\overline{f}^{n+1} + \mathbf{b}^{n+1}\right\}\right] \tag{7.40}$$

This is generalization of Euler and Trapezoidal methods developed in Chapter 6 for the solution of ODEs. The slope function (Figure 6.3) is evaluated as the weighted average of two slopes, one at the present time and the other at the future time. The value of m is the weight for the slope at the present time (n) and (1-m) is the weight for slope at the future time (n+1). Therefore, giving full weight to the slope at the present time (m=1) leads to an explicit method. On the other hand, giving zero weight (m=1) at the present time leads to fully implicit method. For all the intermediate values of m,

the method is multi-step. For m = 1, ½ and 0, the above yields *Euler Forward*, *Trapezoidal* and *Euler Backward* method, respectively. In many literature, the reader may come across the above method written in the following form:

$$\overline{f}^{n+1} = \overline{f}^{n} + \Delta t\left[(1-q)\left\{\mathbf{A}^{n}\overline{f}^{n} + \mathbf{b}^{n}\right\} + q\left\{\mathbf{A}^{n+1}\overline{f}^{n+1} + \mathbf{b}^{n+1}\right\}\right] \tag{7.40a}$$

Both the methods are essentially same and, q and m are related by $q = 1 - m$. Some may find (7.40a) easier to remember intuitively since, q = 1 essentially puts full weightage at the end of the time step (n+1) and q = 0 puts full weightage at the beginning of the time step (n). In this chapter, we will present all analyses (truncation error, stability etc.) using (7.40). However, the readers only need to substitute $m = 1 - q$ in order to obtain equivalent expressions for (7.40a).

The coefficient matrix **A** and the vector **b** contain u and a, which may be functions of both the independent variables. Therefore, the time step index is also shown on **A** and **b** but these can be computed exactly for all time steps since they are functions of independent variables only. The system of linear algebraic equation to be solved at each time step is then given by:

$$\left[I - \Delta t(1-m)\mathbf{A}^{\mathbf{n+1}}\right]\overline{f}^{n+1} = \left[I + m\Delta t\mathbf{A}^{\mathbf{n}}\right]\overline{f}^{n} + \Delta t\left[(1-m)\mathbf{b}^{\mathbf{n+1}} + m\mathbf{b}^{\mathbf{n}}\right] \tag{7.41}$$

For m = 1, the coefficient matrix on the left hand side is identity and therefore does not require solution of any system of equation. This is expected for Euler Forward method. For all other values of m, the system of equation needs to be solved. If u and a are not functions of time, the coefficient matrix becomes constant and a decomposition such as *LU* or Thomas Algorithme (for tridiagonal matrices) saves substantial computation time.

Instead of discretization in two steps (first space and then time), we could have discretized the variable f in both time and space together. This is commonly termed as *full-discretization*. In this case, we will have to choose both space and time discretization schemes. For example, let us choose central difference (CD) scheme for spatial discretization and the weighted parameter (m) scheme for time variable. This we shall refer as *m-CD* scheme for PDE. Application of this scheme to the diffusion and advective-diffusion equations leads to following discrete equations for an interior node (i):

Diffusion Equation:

$$\frac{f_i^{n+1} - f_i^{n}}{\Delta t} = m a_i^{n}\frac{f_{i+1}^{n} - 2f_i^{n} + f_{i-1}^{n}}{\Delta x^2} + (1-m)a_i^{n+1}\frac{f_{i+1}^{n+1} - 2f_i^{n+1} + f_{i-1}^{n+1}}{\Delta x^2} \tag{7.42}$$

Advection Diffusion Equation:

$$\begin{aligned}\frac{f_i^{n+1} - f_i^{n}}{\Delta t} = m\left[-u_i^{n}\frac{f_{i+1}^{n} - f_{i-1}^{n}}{2\Delta x} + a_i^{n}\frac{f_{i+1}^{n} - 2f_i^{n} + f_{i-1}^{n}}{\Delta x^2}\right] + \\ (1-m)\left[-u_i^{n+1}\frac{f_{i+1}^{n+1} - f_{i-1}^{n+1}}{2\Delta x} + a_i^{n+1}\frac{f_{i+1}^{n+1} - 2f_i^{n+1} + f_{i-1}^{n+1}}{\Delta x^2}\right]\end{aligned} \tag{7.43}$$

We leave it to the reader to show that for a given problem, if the same scheme is applied for time and space discretization, both *semi-* and *full* discretization lead to exact same set of linear algebraic equations to solve. For $m = ½$, the numerical scheme is known as *Crank-Nicholson* method. We will now discuss, how to incorporate various types of boundary conditions into discrete equations. One commonly encounters three types of boundary conditions for PDEs. These are as follows:

1st Type or *Dirichlet* Condition:

The value of dependent variable is specified at the boundaries. The example boundary conditions considered in this section were of this type. Since, exact values are specified at the boundary, this can be directly associated with the boundary nodes and put in the discrete equations as in 7.37-38.

2nd Type or *Neuman* Condition:

The flux or gradient of the dependent variable may be specified at one or more boundary. For eq (7.33-34), this takes the form:

$$\left.\frac{dj}{dx}\right|_{(0,t) \text{ and/or } (L,t)} = c \tag{7.44}$$

3rd Type or *Robin* Condition:

A linear combination of dependent variable and its gradient may be specified at one or more boundary. For the PDEs in 7.33-34, this is of the form:

$$a\frac{dj}{dx} + bf = c \text{ at } (0,t) \text{ or } (L,t) \tag{7.45}$$

The coefficients *a*, *b* and *c* in the 2nd and 3rd type BC's can be functions of independent variables. The readers may have noticed by now that the space derivatives of the PDE are discretized using the *Direct methods* for BVP (section 6.5.2). Therefore, the 2nd and 3rd type boundary conditions can be incorporated in the discrete equations using either a ghost node or backward/forward difference similar to the boundary value problems. We demonstrate this in the example 7.1.

Example 7.1: Consider the 1-D Convection Diffusion equation:

$$\frac{\partial T}{\partial t} + u\frac{\partial T}{\partial x} = a\frac{\partial^2 T}{\partial x^2} \qquad 0 \le x \le 1$$

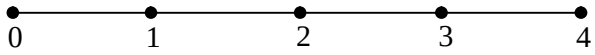
where,

$$u = 0.1; \quad a = 0.01; \quad T(0,t) = 0; \quad T(1,t) = 0; \quad T(x,0) = 50\sin(px)$$

Show solution for one time step using the following methods: (a) Euler Forward in time and Centeral Difference in space (EF-CD), (b) Euler Backward in time and Central Difference in space (EB-CD), (c) *Crank Nicholson* method, (d) A 2nd order R-

K method in time and Central Difference approximation in space. Use $\Delta x = 0.25$ and $\Delta t = 0.5$.

Solution: All the methods use Central Difference approximation in space. Let us first perform a semi-discretization in space using central difference and convert the PDE into a system of ODE. Using $\Delta x = 0.25$, five equidistant nodes in space are shown in the figure below:



For any node j, the discrete approximation of the equation is:

$$\frac{dT_j}{dt} + u\frac{T_{j+1} - T_{j-1}}{2\Delta x} = a\,\frac{T_{j+1} - 2T_j + T_{j-1}}{\Delta x^2}$$

$$\text{or } \frac{dT_j}{dt} = \left(\frac{u}{2\Delta x} + \frac{a}{\Delta x^2}\right)T_{j-1} + \left(-\frac{2a}{\Delta x^2}\right)T_j + \left(-\frac{u}{2\Delta x} + \frac{a}{\Delta x^2}\right)T_{j+1}$$

Using the values of the constants, one obtains:

$$\frac{dT_j}{dt} = 0.36T_{j-1} - 0.32T_j - 0.04T_{j+1}$$

Writing the above equations for T_1, T_2 and T_3, and using two boundary conditions, we obtain the following system of IVPs:

$$\frac{d\mathbf{T}}{dt} = \mathbf{AT}$$

$$\text{where, } \mathbf{T} = \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \text{ and } \mathbf{A} = \begin{bmatrix} -0.32 & -0.04 & 0 \\ 0.36 & -0.32 & -0.04 \\ 0 & 0.36 & -0.32 \end{bmatrix}$$

From the initial condition:

$$\mathbf{T}^0 = \begin{bmatrix} T_1^0 \\ T_2^0 \\ T_2^0 \end{bmatrix} = \begin{bmatrix} 35.3553 \\ 50.0 \\ 35.3553 \end{bmatrix}$$

(a) Euler Forward:

$$\mathbf{T^{n+1}} = [I + \Delta t\mathbf{A}]\mathbf{T^n}$$

$$\text{or } \mathbf{T}^{0.5} = \begin{bmatrix} T_1^{0.5} \\ T_2^{0.5} \\ T_2^{0.5} \end{bmatrix} = \begin{bmatrix} 0.84 & -0.02 & 0 \\ 0.18 & 0.84 & -0.02 \\ 0 & 0.18 & 0.84 \end{bmatrix} \begin{bmatrix} 35.3553 \\ 50.0 \\ 35.3553 \end{bmatrix} = \begin{bmatrix} 28.6985 \\ 47.6568 \\ 38.6985 \end{bmatrix}$$

(b) Euler Backward:

$$[I - \Delta t\mathbf{A}]\mathbf{T}^{\mathbf{n+1}} = \mathbf{T}^{\mathbf{n}}$$

$$\text{or } \begin{bmatrix} 1.16 & 0.02 & 0 \\ -0.18 & 1.16 & 0.02 \\ 0 & -0.18 & 1.16 \end{bmatrix} \begin{bmatrix} T_1^{0.5} \\ T_2^{0.5} \\ T_2^{0.5} \end{bmatrix} = \begin{bmatrix} 35.3553 \\ 50.0 \\ 35.3553 \end{bmatrix}$$

Solve using Thomas Algorithme to obtain:

$$\mathbf{T}^{0.5} = \begin{bmatrix} T_1^{0.5} \\ T_2^{0.5} \\ T_2^{0.5} \end{bmatrix} = \begin{bmatrix} 29.6674 \\ 47.0556 \\ 37.7804 \end{bmatrix}$$

(c) *Crank Nicholson* Scheme:

$$[I - 0.5\Delta t\mathbf{A}]\mathbf{T}^{\mathbf{n+1}} = [I + 0.5\Delta t\mathbf{A}]\mathbf{T}^{\mathbf{n}}$$

$$\begin{bmatrix} 1.08 & 0.01 & 0 \\ -0.09 & 1.08 & 0.02 \\ 0 & -0.09 & 1.08 \end{bmatrix} \begin{bmatrix} T_1^{0.5} \\ T_2^{0.5} \\ T_2^{0.5} \end{bmatrix} = \begin{bmatrix} 0.92 & -0.01 & 0 \\ 0.09 & 0.92 & -0.01 \\ 0 & 0.09 & 0.92 \end{bmatrix} \begin{bmatrix} 35.3553 \\ 50.0 \\ 35.3553 \end{bmatrix} = \begin{bmatrix} 32.0269 \\ 48.8284 \\ 37.0269 \end{bmatrix}$$

Solve using Thomas Algorithme.

$$\mathbf{T}^{0.5} = \begin{bmatrix} T_1^{0.5} \\ T_2^{0.5} \\ T_2^{0.5} \end{bmatrix} = \begin{bmatrix} 29.2166 \\ 47.2923 \\ 38.2252 \end{bmatrix}$$

(d) 2nd Order *Runge Kutta* Method:

Three different forms of 2nd order Runge Kutta method is shown in Table 6.6. Let us use *Heun's predictor corrector* form (2nd in Table 6.6). We leave the other two forms of 2nd Order R-K method for the readers to apply on this problem. Let us use subscripts *p* for the predictor and *c* for the corrector. Applying the predictor formula (6.107) on the problem leads to:

$$\mathbf{T}_p^{n+1} = [I + \Delta t\mathbf{A}]\mathbf{T}^n$$

Since, this is same as the Euler forward method, the solution of (a) is the predictor. The corrector is then given by:

$$\mathbf{T}_c^{n+1} = \mathbf{T}^n + \tfrac{\Delta t}{2}\left[\mathbf{A}\mathbf{T}_p^{n+1} + \mathbf{A}\mathbf{T}^n\right] = \mathbf{T}^n + \tfrac{\Delta t}{2}\mathbf{A}\left[\mathbf{T}_p^{n+1} + \mathbf{T}^n\right]$$

$$\mathbf{T}_c^{0.5} = \begin{bmatrix} T_{1c}^{0.5} \\ T_{2c}^{0.5} \\ T_{2c}^{0.5} \end{bmatrix} = \begin{bmatrix} 35.3553 \\ 50.0 \\ 35.3553 \end{bmatrix} + \begin{bmatrix} -0.08 & -0.01 & 0 \\ 0.09 & -0.08 & -0.01 \\ 0 & 0.09 & -0.08 \end{bmatrix} \left[\begin{Bmatrix} 28.6985 \\ 47.6568 \\ 38.6985 \end{Bmatrix} + \begin{Bmatrix} 35.3553 \\ 50.0 \\ 35.3553 \end{Bmatrix} \right] = \begin{bmatrix} 29.2544 \\ 47.2118 \\ 38.2201 \end{bmatrix}$$

We have seen how to discretize and obtain solution for a diffusion and advective-diffusion equation. Let us now try to look at the discretization along with the physics of the system. For any interior node *i*, the discrete equations for the PDEs (7.42-43) can be rearranged into following forms:

Diffusion Equation:

$$\begin{aligned} &\left[-(1-m)a_i^{n+1}\frac{\Delta t}{\Delta x^2}\right]f_{i+1}^{n+1} + \left[1+2(1-m)a_i^{n+1}\frac{\Delta t}{\Delta x^2}\right]f_i^{n+1} + \left[-(1-m)a_i^{n+1}\frac{\Delta t}{\Delta x^2}\right]f_{i-1}^{n+1} \\ &\qquad = \left[ma_i^n\frac{\Delta t}{\Delta x^2}\right]f_{i+1}^n + \left[1-2ma_i^n\frac{\Delta t}{\Delta x^2}\right]f_i^n + \left[ma_i^n\frac{\Delta t}{\Delta x^2}\right]f_{i-1}^n \end{aligned} \tag{7.46}$$

The above essentially shows that the coefficient matrix will always be symmetric-tri-diagonal unless disturbed by incorporation of boundary conditions. Moreover, if a is constant which is true more often than not, the coefficient matrix only depend on the weighting parameter m and the group $a\frac{\Delta t}{\Delta x^2}$.

Advection Diffusion Equation:

$$\begin{aligned} &\left[(1-m)\left(u_i^{n+1}\frac{\Delta t}{2\Delta x} - a_i^{n+1}\frac{\Delta t}{\Delta x^2}\right)\right]f_{i+1}^{n+1} + \left[1+2(1-m)a_i^{n+1}\frac{\Delta t}{\Delta x^2}\right]f_i^{n+1} + \left[(1-m)\left(-u_i^{n+1}\frac{\Delta t}{2\Delta x} - a_i^{n+1}\frac{\Delta t}{\Delta x^2}\right)\right]f_{i-1}^{n+1} \\ &\qquad = \left[m\left(-u_i^n\frac{\Delta t}{2\Delta x} + a_i^n\frac{\Delta t}{\Delta x^2}\right)\right]f_{i+1}^n + \left[1-2ma_i^n\frac{\Delta t}{\Delta x^2}\right]f_i^n + \left[m\left(u_i^n\frac{\Delta t}{2\Delta x} + a_i^n\frac{\Delta t}{\Delta x^2}\right)\right]f_{i-1}^n \end{aligned} \tag{7.47}$$

Once again, we observe that the resulting matrix is tri-diagonal but the symmetry is missing. The asymmetry in the coefficient matrix is the result of advection. For constant u and a, the coefficient matrix can be generated using values of m, $u\frac{\Delta t}{2\Delta x}$ and $a\frac{\Delta t}{\Delta x^2}$.

The *Peclet Number* in an advection-diffusion system is defined as the dimensionless number relating advection and diffusion in the system:

$P_e = \frac{uL}{a}$ where, u is the advection velocity, L is the characteristic length scale and a is the diffusion coefficient which is the thermal diffusion in the case of heat transfer and mass diffusion or dispersion in the case of mass transfer. In the case of numerical solution using the finite difference schemes described so far, we will be unable to get any information about the state variable at a spatial resolution less than the grid length

Δx. The solutions are obtainable only at the grid points. For values at any intermediate points, we will have to resort to interpolation. Therefore, using the spatial grid length as the characteristic length, we can define a dimensionless number similar to the Peclet number and we shall call it the *Grid Peclet Number* (P_g):

$$P_g = \frac{u\Delta x}{a} \tag{7.48}$$

For pure advection systems, another important dimensionless quantity is the CFL number named after Courant, Friedrich and Lewy. It is defined as:

$$C = u\frac{\Delta t}{\Delta x} \tag{7.49}$$

For constant u and a, both the discrete equations can now be expressed in terms of these two dimensionless numbers. For example, the discrete advection dispersion equation takes the following form:

$$\begin{aligned}\left[(1-m)\left(\frac{C}{2}-\frac{C}{P_g}\right)\right]f_{i+1}^{n+1} + \left[1+2(1-m)\frac{C}{P_g}\right]f_i^{n+1} + \left[(1-m)\left(-\frac{C}{2}-\frac{C}{P_g}\right)\right]f_{i-1}^{n+1} \\ = \left[m\left(-\frac{C}{2}+\frac{C}{P_g}\right)\right]f_{i+1}^{n} + \left[1-2m\frac{C}{P_g}\right]f_i^{n} + \left[m\left(\frac{C}{2}+\frac{C}{P_g}\right)\right]f_{i-1}^{n}\end{aligned} \tag{7.50}$$

Let us now analyze these schemes to understand their consistency and stability properties. We shall demonstrate this using *m-CD* scheme applied to the diffusion equation (7.46) with constant a.

$$\begin{aligned}\left[-(1-m)a\frac{\Delta t}{\Delta x^2}\right]f_{i+1}^{n+1} + \left[1+2(1-m)a\frac{\Delta t}{\Delta x^2}\right]f_i^{n+1} + \left[-(1-m)a\frac{\Delta t}{\Delta x^2}\right]f_{i-1}^{n+1} \\ = \left[ma\frac{\Delta t}{\Delta x^2}\right]f_{i+1}^{n} + \left[1-2ma\frac{\Delta t}{\Delta x^2}\right]f_i^{n} + \left[ma\frac{\Delta t}{\Delta x^2}\right]f_{i-1}^{n}\end{aligned} \tag{7.51}$$

We first write the following Taylor series expansions:

$$f_{i\pm1}^{n+1} = f_i^n + \left(\Delta t\frac{\partial}{\partial t} \pm \Delta x\frac{\partial}{\partial x}\right)f\Big|_i^n + \frac{1}{2!}\left(\Delta t\frac{\partial}{\partial t} \pm \Delta x\frac{\partial}{\partial x}\right)^2 f\Big|_i^n + \frac{1}{3!}\left(\Delta t\frac{\partial}{\partial t} \pm \Delta x\frac{\partial}{\partial x}\right)^3 f\Big|_i^n + HOT \tag{7.52}$$

$$f_{i\pm1}^{n} = f_i^n \pm \Delta x\frac{\partial f}{\partial x}\Big|_i^n + \frac{\Delta x^2}{2!}\frac{\partial^2 f}{\partial x^2}\Big|_i^n \pm \frac{\Delta x^3}{3!}\frac{\partial^3 f}{\partial x^3}\Big|_i^n + \frac{\Delta x^4}{4!}\frac{\partial^4 f}{\partial x^4}\Big|_i^n + HOT \tag{7.53}$$

$$f_i^{n+1} = f_i^n + \Delta t\frac{\partial f}{\partial t}\Big|_i^n + \frac{\Delta t^2}{2!}\frac{\partial^2 f}{\partial t^2}\Big|_i^n + \frac{\Delta t^3}{3!}\frac{\partial^3 f}{\partial t^3}\Big|_i^n + HOT \tag{7.54}$$

We can now put these expansions into the finite difference approximation (7.51) of the diffusion equation:

$$\left[-(1-m)a\frac{\Delta t}{\Delta x^2}\right]\left\{\begin{array}{l} f_i^n+\left(\Delta t\frac{\partial}{\partial t}+\Delta x\frac{\partial}{\partial x}\right)f\Big|_i^n+\frac{1}{2!}\left(\Delta t\frac{\partial}{\partial t}+\Delta x\frac{\partial}{\partial x}\right)^2 f\Big|_i^n+\frac{1}{3!}\left(\Delta t\frac{\partial}{\partial t}+\Delta x\frac{\partial}{\partial x}\right)^3 f\Big|_i^n \\ +\frac{1}{4!}\left(\Delta t\frac{\partial}{\partial t}+\Delta x\frac{\partial}{\partial x}\right)^4 f\Big|_i^n+HOT \end{array}\right\}$$

$$+\left[1+2(1-m)a\frac{\Delta t}{\Delta x^2}\right]\left\{f_i^n+\Delta t\frac{\partial f}{\partial t}\Big|_i^n+\frac{\Delta t^2}{2!}\frac{\partial^2 f}{\partial t^2}\Big|_i^n+\frac{\Delta t^3}{3!}\frac{\partial^3 f}{\partial t^3}\Big|_i^n+HOT\right\}+$$

$$\left[-(1-m)a\frac{\Delta t}{\Delta x^2}\right]\left\{\begin{array}{l} f_i^n+\left(\Delta t\frac{\partial}{\partial t}-\Delta x\frac{\partial}{\partial x}\right)f\Big|_i^n+\frac{1}{2!}\left(\Delta t\frac{\partial}{\partial t}-\Delta x\frac{\partial}{\partial x}\right)^2 f\Big|_i^n+\frac{1}{3!}\left(\Delta t\frac{\partial}{\partial t}-\Delta x\frac{\partial}{\partial x}\right)^3 f\Big|_i^n \\ +\frac{1}{4!}\left(\Delta t\frac{\partial}{\partial t}-\Delta x\frac{\partial}{\partial x}\right)^4 f\Big|_i^n+HOT \end{array}\right\}$$

$$=\left[ma\frac{\Delta t}{\Delta x^2}\right]\left\{f_i^n+\Delta x\frac{\partial f}{\partial x}\Big|_i^n+\frac{\Delta x^2}{2!}\frac{\partial^2 f}{\partial x^2}\Big|_i^n+\frac{\Delta x^3}{3!}\frac{\partial^3 f}{\partial x^3}\Big|_i^n+\frac{\Delta x^4}{4!}\frac{\partial^4 f}{\partial x^4}\Big|_i^n+HOT\right\}+\left[1-2ma\frac{\Delta t}{\Delta x^2}\right]f_i^n$$

$$+\left[ma\frac{\Delta t}{\Delta x^2}\right]\left\{f_i^n-\Delta x\frac{\partial f}{\partial x}\Big|_i^n+\frac{\Delta x^2}{2!}\frac{\partial^2 f}{\partial x^2}\Big|_i^n-\frac{\Delta x^3}{3!}\frac{\partial^3 f}{\partial x^3}\Big|_i^n+\frac{\Delta x^4}{4!}\frac{\partial^4 f}{\partial x^4}\Big|_i^n+HOT\right\} \tag{7.55}$$

Grouping similar terms and making algebraic manipulations, one obtains:

$$\frac{\partial f}{\partial t}\Big|_i^n-a\frac{\partial^2 f}{\partial x^2}\Big|_i^n=-\frac{\Delta t}{2}\frac{\partial^2 f}{\partial t^2}\Big|_i^n+(1-m)\Delta t a\frac{\partial^3 f}{\partial t\partial x^2}\Big|_i^n-\frac{\Delta t^2}{6}\frac{\partial^3 f}{\partial t^3}\Big|_i^n$$
$$+(1-m)\frac{\Delta t^2}{2}a\frac{\partial^4 f}{\partial t^2\partial x^2}\Big|_i^n+a\frac{\Delta x^2}{12}\frac{\partial^4 f}{\partial x^4}\Big|_i^n+HOT \tag{7.56}$$

The left side of this equation contains the original PDE (7.33). If the approximation was accurate, the right hand side would be zero. The leftover terms on the right are therefore the truncation error due to approximation. Using our standard definition of error, i.e. *Error = True – Approximation*, we obtain the truncation error as,

$$TE=-+\frac{\Delta t}{2}\frac{\partial^2 f}{\partial t^2}\Big|_i^n-(1-m)\Delta t a\frac{\partial^3 f}{\partial t\partial x^2}\Big|_i^n+\frac{\Delta t^2}{6}\frac{\partial^3 f}{\partial t^3}\Big|_i^n$$
$$-(1-m)\frac{\Delta t^2}{2}a\frac{\partial^4 f}{\partial t^2\partial x^2}\Big|_i^n-a\frac{\Delta x^2}{12}\frac{\partial^4 f}{\partial x^4}\Big|_i^n+HOT \tag{7.57}$$

Now, let us recognize an identity using the original PDE,

$$a\frac{\partial^3 f}{\partial t \partial x^2} = \frac{\partial}{\partial t}\left(a\frac{\partial^2 f}{\partial x^2}\right) = \frac{\partial}{\partial t}\left(\frac{\partial f}{\partial t}\right) = \frac{\partial^2 f}{\partial t^2} \tag{7.58}$$

Using this identity in (7.57) and retaining only similar terms (in derivative), we obtain:

$$TE = \left(\mu - \tfrac{1}{2}\right)\Delta t \frac{\partial^2 f}{\partial t^2}\bigg|_i^n - a\frac{\Delta x^2}{12}\frac{\partial^4 f}{\partial x^4}\bigg|_i^n + \frac{\Delta t^2}{6}\frac{\partial^3 f}{\partial t^3}\bigg|_i^n + HOT \tag{7.59}$$

The first term of the truncation error is 1st order in time (Δt) but it vanishes for $\mu = ½$. Rest of the terms are 2nd order or higher in both time and space. Therefore, the scheme is 2nd order in time and space for $\mu = ½$ and we denote it as $o\left(\Delta t^2, \Delta x^2\right)$. For all other values of μ, the scheme is 1st order in time and 2nd order in space, i.e., $o\left(\Delta t, \Delta x^2\right)$.

Let us try to interpret this from our knowledge of ODE methods. We have used the central difference scheme for the spatial derivatives, which is 2nd order. Therefore, it comes as no surprise that the accuracy in space is always 2nd order. For the time discretization, we have used a weighting factor m. For $\mu = 0$ and 1, we get the Euler methods (backward and forward, respectively). Since, both of these are 1st order methods, we obtain 1st order accuracy in time for PDE as well. For $\mu = ½$, the method becomes Trapezoidal which is known to be 2nd order. Therefore, the results are in the expected lines. However, in some cases, we can increase the order of truncation error for this scheme by judicial choice of time step size and the spatial grid size, which was not possible for the ODE. In order to see this, let us consider the following identity:

$$\frac{\partial^2 f}{\partial t^2} = \frac{\partial}{\partial t}\left(\frac{\partial f}{\partial t}\right) = \frac{\partial}{\partial t}\left(a\frac{\partial^2 f}{\partial x^2}\right) = a\frac{\partial^2}{\partial x^2}\left(\frac{\partial f}{\partial t}\right) = a\frac{\partial^2}{\partial x^2}\left(a\frac{\partial^2 f}{\partial x^2}\right) = a^2\frac{\partial^4 f}{\partial x^4} \tag{7.60}$$

Using this identity, the truncation error becomes:

$$TE = \left\{\left(\mu - \tfrac{1}{2}\right)\Delta t a^2 - a\frac{\Delta x^2}{12}\right\}\frac{\partial^4 f}{\partial x^4}\bigg|_i^n + \frac{\Delta t^2}{6}\frac{\partial^3 f}{\partial t^3}\bigg|_i^n + HOT \tag{7.61}$$

The first term on the right will vanish if the coefficient in the curly bracket is set to zero. This essentially increases the order of the method even when μ is not equal to ½. The required identity is,

$$a\frac{\Delta t}{\Delta x^2} = \frac{1}{12\left(\mu - \frac{1}{2}\right)} \tag{7.62}$$

Note that on the left we have a familiar term group namely the ratio C/P_g. This ratio can be suitably chosen (by adjusting Δt and Δx) for all values of $\mu > ½$. For $\mu = ½$, the right hand side is infinite. This can be achieved only if Δx is zero which is not possible. However for $\mu = ½$, the method is second order in time anyway (7.59) and therefore, no need to adjust these parameters. For $\mu < ½$, the ratio of C/P_g has to be

negative for the time and derivative time terms to cancel each other. Since, a, Δt and Δx all are greater than zero, negative value is not possible. Therefore, for fully implicit method like Euler Backward, it is not possible to cancel the time and space terms in order to achieve a higher order of accuracy. For Euler Forward method, m = 1 and the relation is,

$$a\frac{\Delta t}{\Delta x^2}=\frac{1}{6} \tag{7.63}$$

We shall see in the section 7.3 that the above condition is within the stability limit of the Euler Forward method for the diffusion equation and therefore, it is possible to achieve the condition. However, this puts more severe restriction than the stability on the choice of time step for any chosen Δx .

Let us now perform similar truncation analysis on the discrete advective-diffusion equation (7.47). For constant u and a, the equation is as follows:

$$\begin{aligned}\left[(1-m)\left(u\frac{\Delta t}{2\Delta x}-a\frac{\Delta t}{\Delta x^2}\right)\right]f_{i+1}^{n+1}+\left[1+2(1-m)a\frac{\Delta t}{\Delta x^2}\right]f_i^{n+1}+\left[(1-m)\left(-u\frac{\Delta t}{2\Delta x}-a\frac{\Delta t}{\Delta x^2}\right)\right]f_{i-1}^{n+1}\\ =\left[m\left(-u\frac{\Delta t}{2\Delta x}+a\frac{\Delta t}{\Delta x^2}\right)\right]f_{i+1}^{n}+\left[1-2ma\frac{\Delta t}{\Delta x^2}\right]f_i^{n}+\left[m\left(u\frac{\Delta t}{2\Delta x}+a\frac{\Delta t}{\Delta x^2}\right)\right]f_{i-1}^{n}\end{aligned} \tag{7.64}$$

Using the expansions of 7.53-54 and performing rigorous algebraic operations, we obtain,

$$\begin{aligned}\left.\frac{\partial f}{\partial t}\right|_i^n+u\left.\frac{\partial f}{\partial x}\right|_i^n-a\left.\frac{\partial^2 f}{\partial x^2}\right|_i^n=&-u^2\Delta t\left(m-\tfrac{1}{2}\right)\left.\frac{\partial^2 f}{\partial x^2}\right|_i^n+2ua\Delta t\left(m-\tfrac{1}{2}\right)\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n-a^2\Delta t\left(m-\tfrac{1}{2}\right)\left.\frac{\partial^4 f}{\partial x^4}\right|_i^n\\&+\frac{u^3\Delta t^2}{6}\left(\tfrac{m}{2}-\tfrac{1}{3}\right)\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n-\frac{u\Delta x^2}{6}\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n+HOT\end{aligned} \tag{7.65}$$

Therefore, the m-CD scheme is 1st order in time and second order in space except for m = ½ where all the 1st order time term disappears from the truncation error leaving the 2nd order as the highest order terms in the truncation error. For all values of m except ½, the 1st term on the right hand side is similar to the diffusion term on the left hand side containing 2nd order space derivative. In fact one can write the above equation in the following form:

$$\begin{aligned}\left.\frac{\partial f}{\partial t}\right|_i^n+u\left.\frac{\partial f}{\partial x}\right|_i^n-\left[a-u^2\Delta t\left(m-\tfrac{1}{2}\right)\right]\left.\frac{\partial^2 f}{\partial x^2}\right|_i^n=&2ua\Delta t\left(m-\tfrac{1}{2}\right)\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n-a^2\Delta t\left(m-\tfrac{1}{2}\right)\left.\frac{\partial^4 f}{\partial x^4}\right|_i^n\\&+\frac{u^3\Delta t^2}{6}\left(\tfrac{m}{2}-\tfrac{1}{3}\right)\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n-\frac{u\Delta x^2}{6}\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n+HOT\end{aligned} \tag{7.66}$$

This essentially shows that the effect of the highest order truncation error term is to increase or decrease the net diffusion (or dispersion) coefficient of the physical problem depending on the value of m. This extra diffusion (or dispersion) is purely a numerical artifact and will be called *numerical diffusion*. Simplest way to minimize

the effect of numerical diffusion is to use a modified artificial diffusion coefficient (a') in place of the original one. For the *m-CD* scheme, this modification rule is,

$$a' = a + u^2 \Delta t \left(m - \tfrac{1}{2}\right) \tag{7.67}$$

Use of this modified diffusion coefficient does not increase the order of accuracy of the method but it reproduces the effect of diffusion in the numerical solution equivalent to the original diffusion coefficient in analytical equation. In order to avoid using this artificial diffusion coefficient, one can change the numerical scheme such that the numerical diffusion does not appear in the truncation error. One such commonly used method is *upwind scheme*. In this, the first order spatial derivative with the velocity term *u* is discretized using a forward difference instead of central difference. This forward difference can be 1st, 2nd, or 3rd order. Higher the order, less is the numerical diffusion. We leave it to the readers to show this. We shall use the upwind scheme in section 7.2.3 for 1st order hyperbolic wave equation where the advection term has the opposite sign and backward difference will be used instead of the forward difference to achieve the same objective of reducing numerical diffusion. Let us illustrate the numerical diffusion with the help of an example.

Example 7.2. Solve the following 1-D advection diffusion equation using EF-CD scheme with original and modified diffusion coefficient. Also solve it using *Crank-Nicholson* scheme with original diffusion coefficient. Compare three results graphically at $t = 3$.

$$\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} = a\frac{\partial^2 C}{\partial x^2} \qquad 0 \le x \le 1$$

where,

$$u = 0.2; \quad a = 0.01; \quad C(0,t) = 1; \quad \frac{\P C}{\P x}(1,t) = 0; \quad C(x,0) = 0$$

Use $\Delta x = 0.2$ and $\Delta t = 0.5$

Solution: With $\Delta x = 0.2$, we have five equal interval in the domain which leads to 6 nodes including the boundary nodes. Let us number them 0-5 with node zero at x=0 and node 5 at x=1.

The equations for an interior node (i) can be formulated using (7.64) for both EF-CD and Crank Nicholson Schemes with appropriate values of *m*. Let us first formulate the equations for the EF-CD scheme, $m = 1$.

With original diffusion coefficient: $\dfrac{u\Delta t}{2\Delta x} = 0.25, \ \dfrac{a\Delta t}{\Delta x^2} = 0.125$

$$C_i^{n+1} = 0.375C_{i-1}^n + 0.75C_i^n - 0.125C_{i+1}^n$$

We can now write the following set of equations for the values at the interior nodes by using the boundary condition $C_0 = 1$

$$C_1^{n+1} = 0.75C_1^n - 0.125C_2^n + 0.375$$
$$C_2^{n+1} = 0.375C_1^n + 0.75C_2^n - 0.125C_3^n$$
$$C_3^{n+1} = 0.375C_2^n + 0.75C_3^n - 0.125C_4^n$$
$$C_4^{n+1} = 0.375C_3^n + 0.75C_4^n - 0.125C_5^n$$

In the last equation, C_5 may be eliminated by making a 2nd order backward difference approximation of the boundary condition at node 5.

$$\frac{C_3 - 4C_4 + 3C_5}{2\Delta x} = 0 \text{ or } C_5 = \tfrac{4}{3}C_4 - \tfrac{1}{3}C_3$$

Using this relation, the last equation becomes:

$$C_4^{n+1} = 0.4167C_3^n + 0.5833C_4^n$$

Now we can compute C_1-C_4 at all the time steps using the above equation using the initial conditions,

$$C_1^0 = C_2^0 = C_3^0 = C_4^0 = 0$$

The modified diffusion coefficient computed using (7.67) is $a' = 0.02$. This gives
$$\frac{a'\Delta t}{\Delta x^2} = 0.25$$

The new equation for the node (i) is,

$$C_i^{n+1} = 0.5C_i^n + 0.5C_{i-1}^n$$

Notice, that the value C_{i+1} is absent from the above equation. This essentially reduced the formulation to upwind scheme. So, use of modified diffusion coefficient and the upwind scheme are equivalent in this case. Because of this, we do not even need the boundary condition to the right. We can write the set of equations as follows:

$$C_1^{n+1} = 0.5C_1^n + 0.5$$
$$C_2^{n+1} = 0.5C_2^n + 0.5C_1^n$$
$$C_3^{n+1} = 0.5C_3^n + 0.5C_2^n$$
$$C_4^{n+1} = 0.5C_4^n + 0.5C_3^n$$

Lastly, the equation for node (i) for the *Crank-Nicholson* scheme is:

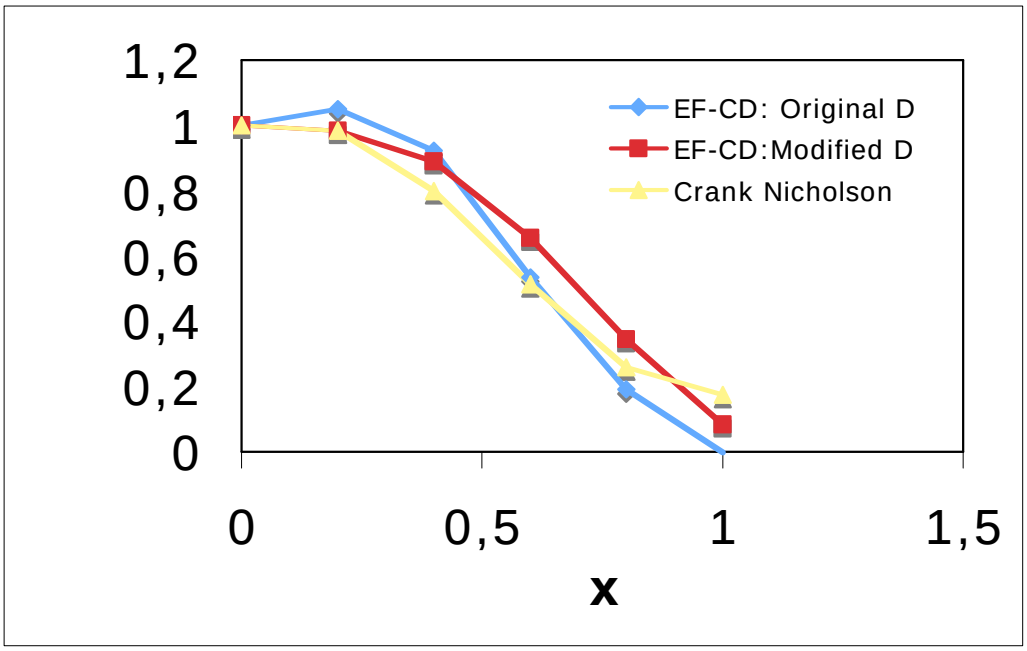
$$-0.1875C_{i-1}^{n+1} + 1.125C_i^{n+1} + 0.0625C_{i+1}^{n+1} = 0.1875C_{i-1}^n + 0.875C_i^n - 0.0625C_{i+1}^n$$

Using the left hand boundary condition and 2nd order backward difference approximation of the right hand boundary condition, we obtain the following system of equation for values at the interior nodes:

$$\begin{bmatrix} 1.125 & 0.0625 & 0 & 0 \\ -0.1875 & 1.125 & 0.0625 & 0 \\ 0 & -0.1875 & 1.125 & 0.0625 \\ 0 & 0 & -0.2083 & 1.2083 \end{bmatrix} \begin{bmatrix} C_1^{n+1} \\ C_2^{n+1} \\ C_3^{n+1} \\ C_4^{n+1} \end{bmatrix} =$$

$$\begin{bmatrix} 0.875 & -0.0625 & 0 & 0 \\ 0.1875 & 0.875 & -0.0625 & 0 \\ 0 & 0.1875 & 0.875 & -0.0625 \\ 0 & 0 & 0.2083 & 0.7917 \end{bmatrix} \begin{bmatrix} C_1^{n} \\ C_2^{n} \\ C_3^{n} \\ C_4^{n} \end{bmatrix} + \begin{bmatrix} 0.375 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

All three methods can now be solved using any spreadsheet or Matlab. We show the result in the figure below:



In the solution using EF-CD scheme with original diffusion coefficient, the concentration goes above 1 at node 1 and negative at node 5. The solution is better behaved with the modified diffusion coefficient. The solution with Crank-Nicholson scheme is good as expected. The value at Node 5 is a function of the difference scheme used for discretization of the boundary condition. The 2nd order backward difference scheme has put weights on the previous two nodes. This would be good for an advection dominated problem but for diffusion problems, it is affecting the value at node 4 as well because of the replacement. A first order backward difference approximation in this case may work better. Lastly, we limited the number of unknown nodes to 4 in order to illustrate the solution method manually. In reality, the problem requires finer grids in the spatial dimension to adequately resolve the effects of advection. We will see the pure advection part of this problem with finer grid in section 7.2.3.

We outlined the solution method for parabolic equations using the *m-CD* scheme in time and central difference in space. Both the schemes use the same concept as was described in the previous chapter for ODEs. We have also shown that the PDE is reduced to a system of IVP once the spatial discretization is done. Then one can use any other method for the solution of system of IVP such that, Runge-Kutta, Gears method etc. Especially, the reader may try to apply 4^{th} order Runge-Kutta method for the example problem 7.2 and compare the solution. The 4^{th} order Runge-Kutta method combines high order of accuracy with decent stability property. Its stability region also encompasses some part of purely imaginary l (Figure 6.14) which is important for advection dominated problems (section 7.2.3). As a result, the method has found wide spread application in the solution of ODEs as well as PDEs.

7.2.2 Laplace Equation

Most commonly encountered PDE in engineering problems is the laplace equation. It is encountered in stream function, flow potential, temperature distribution, steady state concentration, stress distribution, etc., to name a few. We will illustrate the solution methods with a 2-D problem which is given by equation 7.6. We rewrite it in terms of our state variable f:

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = 0 \tag{7.68}$$

Both the independent variables are in spatial dimension. Therefore, we need to have finite boundaries for both the variables to obtain a numerical solution although for analytical solution, infinite boundaries may be allowed. The boundaries are governed by the physics of the problem. In most of the practical problem, these are physical boundaries and therefore finite. However, in a very large domain, if we are concerned about variation or perturbation at a particular point, we need to put the finite boundary far enough from this point so that the effect of perturbation is negligible and can be treated as infinity of the analytical problem. We shall also use 1^{st} Type boundary conditions at all the boundaries. This however will not affect our description of the solution of the PDE and we will show how to incorporate other types of BCs later on. Once the boundary is defined, the first job is to lay a grid on *x* and *y* directions. In order to illustrate the solution method, let us consider a finite boundary in Cartesian geometry with first type BC (Figure 7.3). The problem shown in Figure 7.3 may be mathematically stated as follows:

Solve Laplace equation in the region $x \in (0, L_x)$ and $y \in (0, L_y)$ with the boundary conditions $f(x,0) = c$, $f(x, L_y) = a$, $f(0, y) = d$ and $f(L_x, y) = b$.

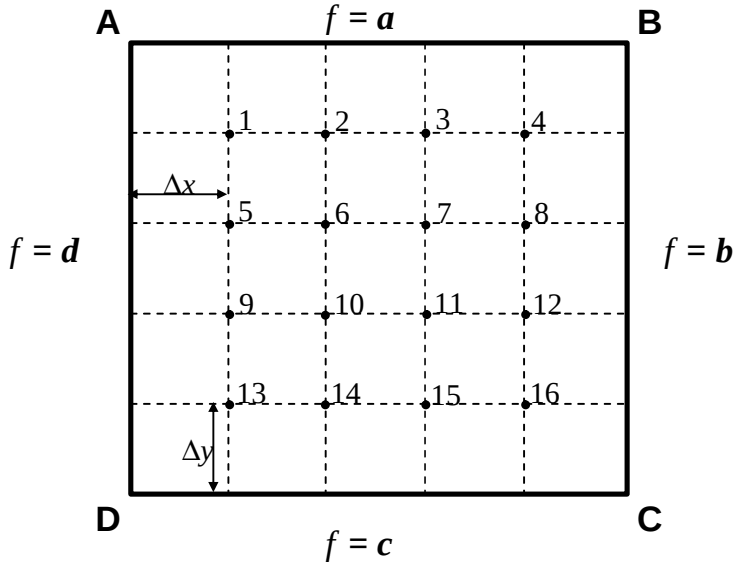


Figure 7.3: A finite boundary in Cartesian geometry with uniform grid sizes.

We have laid uniform grids in both x and y directions with grid sizes Δx and Δy, respectively. However, Δx may not be equal to Δy. In Figure 7.3, we have laid out five equally spaced intervals in each directions. This essentially resulted in a total of 36 intersection points. However, 20 of these 36 points are on the boundaries. The values of f are known at these 20 points from the boundary conditions. So, we only need to determine the unknown values of f at 16 interior points. These points are numbered as shown in Figure 7.3. Let us denote the unknown values of the state variable as $f_1 - f_{16}$. We will now write discrete approximation of the Laplace equation at each of these 16 points and obtain 16 algebraic equations. It will then be easy to compute the required values. Let us choose central difference for approximating the partial derivatives in the Laplace equation. We will use i and j as the space variable indices for x and y, respectively. Following our convention, we will put both the indices in the subscript, since both are space variables. Therefore, we can write the following general expressions for the central difference approximation of the double derivatives in a regular grid:

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{i,j} = \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{\Delta x^2} \quad \text{and} \quad \left.\frac{\partial^2 f}{\partial y^2}\right|_{i,j} = \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{\Delta y^2} \tag{7.69}$$

A visual representation of the values of different nodes used in these expressions is shown in the Figure 7.4. Approximation of the partial derivative at a point requires values from two adjacent nodes in each direction.

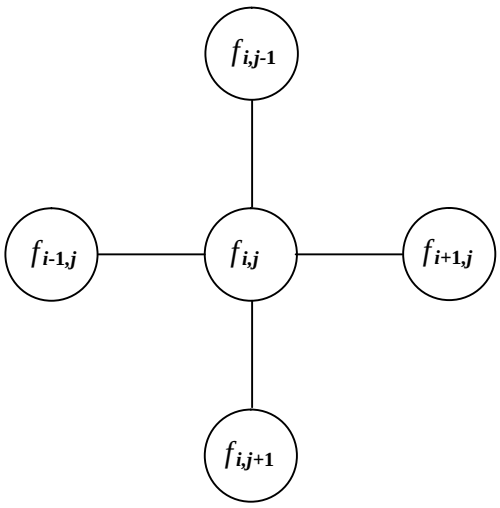


Figure 7.4: Approximation molecule for the CD scheme applied to the Laplace equation.

Combining two independent expressions of the partial derivatives (7.69), we obtain a discrete approximation of the Laplace equation for the node (*i, j*):

$$\left(\frac{\partial^2 f}{\partial x^2}+\frac{\partial^2 f}{\partial y^2}\right)_{i,j}=\frac{f_{i+1,j}-2f_{i,j}+f_{i-1,j}}{\Delta x^2}+\frac{f_{i,j+1}-2f_{i,j}+f_{i,j-1}}{\Delta y^2}=0 \tag{7.70}$$

$$\left(\frac{1}{\Delta y^2}\right)f_{i,j-1}+\left(\frac{1}{\Delta x^2}\right)f_{i-1,j}+\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right)f_{i,j}+\left(\frac{1}{\Delta x^2}\right)f_{i+1,j}+\left(\frac{1}{\Delta y^2}\right)f_{i,j+1}=0 \tag{7.71}$$

Therefore, an approximation at note (*i, j*) requires values from 4 neighbouring nodes. This we will term as the approximation molecule (Figure 7.4) for the node (*i, j*) using central difference scheme. The molecule will be different for the same equation if we change the discretization scheme in any direction, e.g. forward or backward difference in place of central difference. This will be required for the boundary nodes as we will be missing nodes on one of the sides depending on which side the boundary is. Now let us apply this scheme for the problem shown in Figure 7.3. For node 1, two of the neighbouring nodes are on the boundary and therefore the values at these nodes are known from the boundary condition. Nodes 4, 13 and 16 are similar to node 1. On some boundary points such as 2, 3, 5, 8, 9, 12, 14 and 15, only one of the neighbouring node is on the boundary. For the interior points 6, 7, 10 and 11, all the nodes in the molecule are interior nodes with unknown values. Two examples of approximation molecules for nodes 1 (with neighbouring boundary nodes) and 6 (neighbouring interior nodes) are shown in Figure 7.5. The approximate equations for these nodes are:

$$\text{Node 1: } \left(\frac{1}{\Delta y^2}\right)a+\left(\frac{1}{\Delta x^2}\right)d+\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right)f_1+\left(\frac{1}{\Delta x^2}\right)f_2+\left(\frac{1}{\Delta y^2}\right)f_5=0 \tag{7.72}$$

or $$\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right)f_1+\left(\frac{1}{\Delta x^2}\right)f_2+\left(\frac{1}{\Delta y^2}\right)f_5=-\left(\frac{1}{\Delta y^2}\right)a-\left(\frac{1}{\Delta x^2}\right)d \qquad (7.73)$$

Node 6: $$\left(\frac{1}{\Delta y^2}\right)f_2+\left(\frac{1}{\Delta x^2}\right)f_5+\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right)f_6+\left(\frac{1}{\Delta x^2}\right)f_7+\left(\frac{1}{\Delta y^2}\right)f_{10}=0 \qquad (7.74)$$

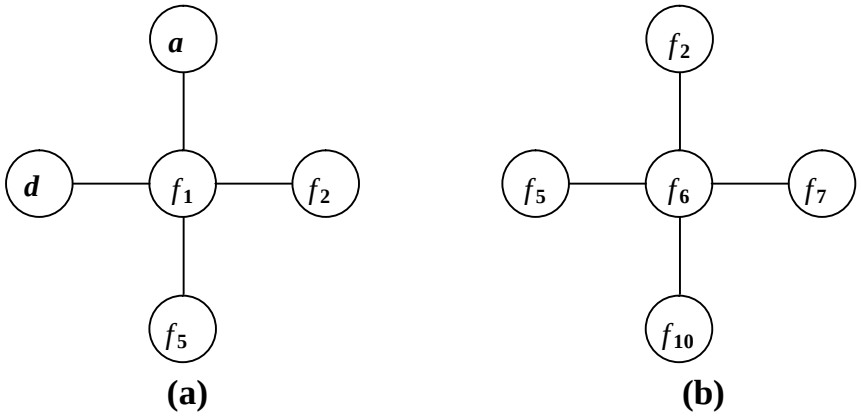


Figure 7.5: Approximation molecules for nodes 1 (a) and 6 (b)

Writing similar equations for all 16 nodes, one obtains 16 linear algebraic equation with in the form of **A**F = **b**. The matrix **A** contains the coefficients of the equations similar to 7.73-74, F is the vector containing unknowns $f_1 - f_{16}$, and **b** is a vector containing the known constants originating from the boundary conditions contained in the equations for the nodes adjacent to the boundaries (eq 7.74). Easiest way to generate matrix **A** and vector **b** would be to initiate a square matrix and a vector of size 16 with zero entries. Then for each node, change the entries corresponding to the nodes contained in that equation with appropriate values. Let us denote the values of matrix **A** as a_{ij} and those of vector **b** as b_i. Then for node 1 (eq 7.74) we will set the following:

$$a_{11}=\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right),\ a_{12}=\left(\frac{1}{\Delta x^2}\right),\ a_{15}=\left(\frac{1}{\Delta y^2}\right) \text{ and } b_1=-\left(\frac{1}{\Delta y^2}\right)a-\left(\frac{1}{\Delta x^2}\right)d \qquad (7.75)$$

Once, the Δx and Δy are chosen, the coefficients of the unknown f values are constant (equations 7.73-74) which are the entries in the matrix **A**. One can generate these coefficients from the Figure 7.3 without having to write equations 7.73-74 explicitly. For any approximation molecule, the coefficient for the node at the center is $-\left(\frac{2}{\Delta x^2}+\frac{2}{\Delta y^2}\right)$, coefficient of two neighbouring nodes in x direction is $\left(\frac{1}{\Delta x^2}\right)$ irrespective of whether they are on the left or right of the center and coefficient of the neighbouring nodes in y direction is $\left(\frac{1}{\Delta y^2}\right)$. If the approximation molecule contains one of the nodes at the boundary, the term corresponding to that node contributes an entry in the vector **b**. For example, node 9 contributes the following entries:

$$a_{95}=\left(\frac{1}{\Delta y^2}\right), a_{99}=\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right),\ a_{9\,10}=\left(\frac{1}{\Delta x^2}\right),\ a_{9\,13}=\left(\frac{1}{\Delta y^2}\right) \text{ and } b_9=-\left(\frac{1}{\Delta x^2}\right)d \tag{7.76}$$

For the node 7, the entries are:

$$a_{73}=\left(\frac{1}{\Delta y^2}\right), a_{76}=\left(\frac{1}{\Delta x^2}\right) a_{77}=\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right),\ a_{78}=\left(\frac{1}{\Delta x^2}\right) \text{and } a_{7\,11}=\left(\frac{1}{\Delta y^2}\right) \tag{7.77}$$

The entries are made on the k^{th} row for the equation (or approximation molecule) corresponding to node k. The node numbers in the k^{th} approximation molecules are the columns where the entries are made. The coefficient of the node at the center is therefore, always a diagonal element of the matrix. Since, the approximation molecule for central difference contains 5 nodes, a maximum of 5 entries are non-zero in any row. The rest are all zeros. This essentially leads to a sparse matrix A. Therefore, the iterative methods described in section (2.2.2) are most appropriate for solution of the system of equation. Since, a_{kk} entries are the largest in magnitude (7.75-7.77), the coefficient matrix is diagonal dominant and the convergence is assured. An oft-used method to accelerate the convergence of the solution is known as *multi-grid* method. This will be briefly described in the next chapter. We shall now describe how to incorporate a derivative boundary condition. If in the problem of Figure 7.3, the gradient of f is specified along boundaries BC and CD, the problem will be defined as:

Solve Laplace equation in the region $x \in (0, L_x)$ and $y \in (0, L_y)$ with the boundary conditions $\left.\frac{\partial f}{\partial y}\right|_{(x,0)} = c,\ f(x, L_y) = a,\ f(0, y) = d$ and $\left.\frac{\partial f}{\partial x}\right|_{(L_x, y)} = b$.

The values of f at the nodes along boundaries BC and CD are also unknown. So, the number of unknowns now increases to 25 (Figure 7.6). The obvious solution is to write the discrete approximation of Laplace equation for all the interior nodes and discrete approximations of boundary condtions for the boundary nodes. Since, we will be able to write one equation for each node, this will yield the same number of equations as the number of unknowns.

At nodes 5, 10, 15, 20, 21, 22, 23 and 24 one can write a 2nd order backward difference approximation of the boundary condition similar to the boundary value problems (Section 6.5). For example, equations for node 5 and 21 are shown below:

Node 5: $$\frac{f_3 - 4f_4 + 3f_5}{2\Delta x} = b \text{ or } \left(\frac{1}{2\Delta x}\right)f_3 + \left(-\frac{2}{\Delta x}\right)f_4 + \left(\frac{3}{2\Delta x}\right)f_5 = b \tag{7.78}$$

Node 21: $$\frac{f_{11} - 4f_{16} + 3f_{21}}{2\Delta y} = c \text{ or } \left(\frac{1}{2\Delta y}\right)f_{11} + \left(-\frac{2}{\Delta y}\right)f_{16} + \left(\frac{3}{2\Delta y}\right)f_{21} = c \tag{7.79}$$

Entries corresponding to these nodes are:

Node 5: $a_{53} = \left(\frac{1}{2\Delta x}\right)$, $a_{54} = \left(-\frac{2}{\Delta x}\right)$, $a_{55} = \left(\frac{3}{2\Delta x}\right)$, and $b_5 = b$ (7.80)

Node 21: $a_{21\,11} = \left(\frac{1}{2\Delta y}\right)$, $a_{21\,16} = \left(-\frac{2}{\Delta y}\right)$, $a_{21\,21} = \left(\frac{3}{2\Delta x}\right)$, and $b_{21} = c$ (7.81)

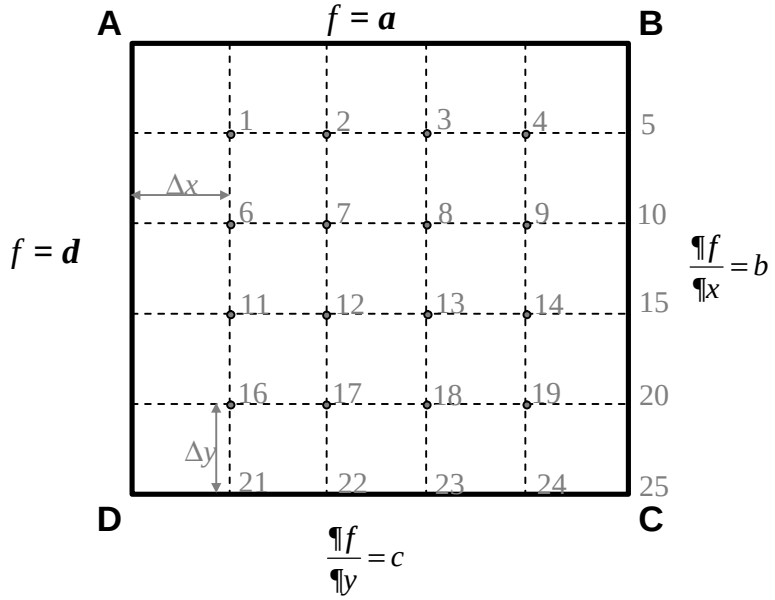


Figure 7.6: New node numbers with gradient boundary conditions for the problems in Figure 7.3.

For node 25, both the boundary conditions are valid. So, one may wonder which equation to write. However, it is easy to see that node 25 does not belong to the approximation molecule of any other node. Therefore, the entry in the coefficient matrix corresponding to this node is zero in every other row except row 25. Therefore, this node can be eliminated from the unknown, which essentially reduces the unknown vector size to 24. The flux in both x and y directions at node 25 can be computed from the values at nodes 15, 20, 23 and 24 using 2nd order backward difference.

You must have realized by now that changing the boundary conditions has increased the size of the matrix equation to be solved to 24 from 16. In addition node numbers of the interior nodes have changed. Therefore, location of the entries corresponding to these nodes in the coefficient matrix also changed. This essentially means, the complete matrix have to be regenerated. It would be nice, if we could keep the existing matrix of the original boundary conditions and only change the entries in the rows corresponding to the nodes adjacent to the boundary. This will also keep the size of the equation system to 16. This can be accomplished by keeping the original node numbers and give some temporary numbers to the boundary nodes where the f values are unknown (Figure 7.7).

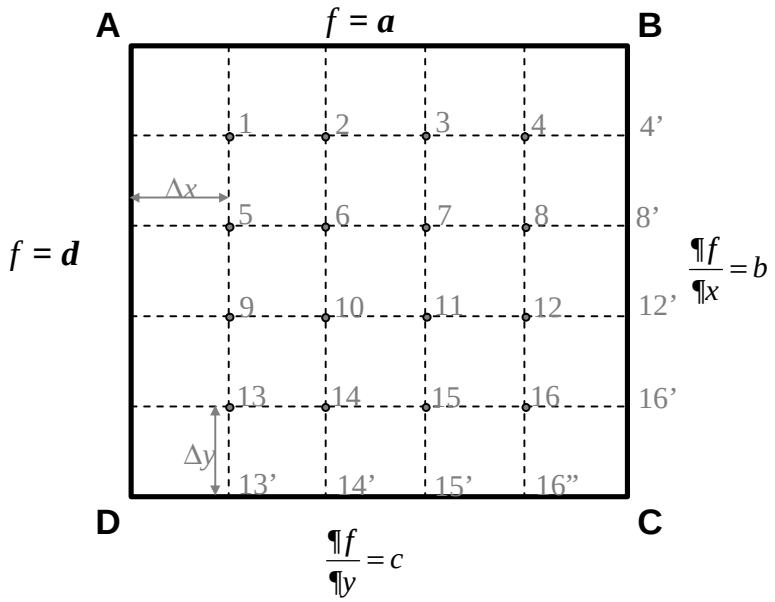


Figure 7.7: Temporary node numbers along the boundary for derivative boundary conditions for the problem of Figure 7.3.

The discrete approximation of the Laplace equation at the interior nodes adjacent to the boundary would contain these temporary node variables. These can be eliminated by using the boundary condition approximations (7.78-79) at the boundary nodes. We will illustrate this using the example of node 16 (Figures 7.6 and 7.7). With the 1st Type boundary conditions (Figure 7.3), the equation for node 16 was:

$$\left(\frac{1}{\Delta y^2}\right)f_{12}+\left(\frac{1}{\Delta x^2}\right)f_{15}+\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right)f_{16}=-\frac{b}{\Delta x^2}-\frac{c}{\Delta y^2} \tag{7.82}$$

With the 2nd Type boundary conditions, the approximation molecule of node 16 has two temporary unknown nodes at the boundary. One can write the following discrete approximations for Laplace equation as node 16 and boundary conditions at nodes 16’ and 16”:

Node 16: $$\left(\frac{1}{\Delta y^2}\right)f_{12}+\left(\frac{1}{\Delta x^2}\right)f_{15}+\left(-\frac{2}{\Delta x^2}-\frac{2}{\Delta y^2}\right)f_{16}+\left(\frac{1}{\Delta x^2}\right)f_{16'}+\left(\frac{1}{\Delta y^2}\right)f_{16''}=0 \tag{7.83}$$

Node 16’: $$\frac{f_{15}-4f_{16}+3f_{16'}}{2\Delta x}=b \text{ or } \left(\frac{1}{2\Delta x}\right)f_{15}+\left(-\frac{2}{\Delta x}\right)f_{16}+\left(\frac{3}{2\Delta x}\right)f_{16'}=b \tag{7.84}$$

Node 16”: $$\frac{f_{12}-4f_{16}+3f_{16''}}{2\Delta y}=c \text{ or } \left(\frac{1}{2\Delta y}\right)f_{12}+\left(-\frac{2}{\Delta y}\right)f_{16}+\left(\frac{3}{2\Delta y}\right)f_{16''}=c \tag{7.85}$$

Using (7.84 and 7.85) in (7.83) we obtain the following equation for node 16:

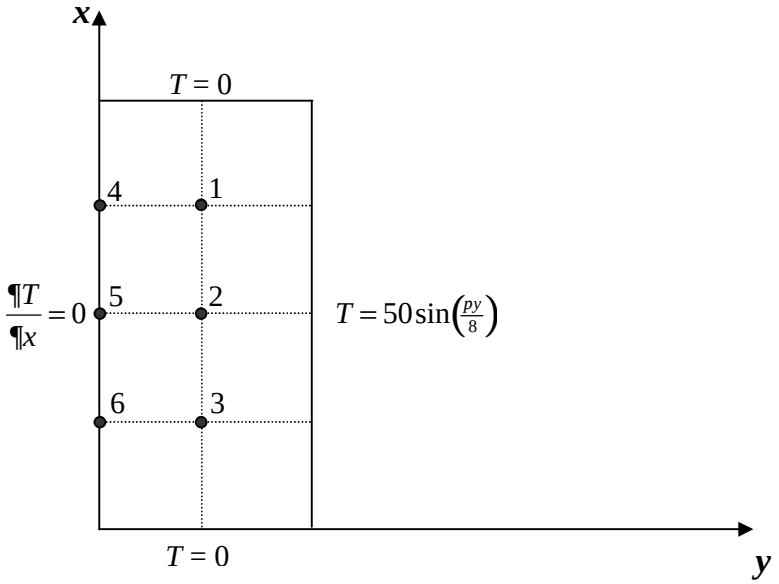
$$\left(\frac{2}{3\Delta y^2}\right)f_{12}+\left(\frac{2}{3\Delta x^2}\right)f_{15}+\left(-\frac{2}{3\Delta x^2}-\frac{2}{3\Delta y^2}\right)f_{16}=-\frac{2b}{3\Delta x}-\frac{2c}{3\Delta y} \tag{7.86}$$

By changing boundary condition, the original equation for node 16 (7.82) will be replaced by (7.86). Similarly, equations corresponding to all the nodes adjacent to the boundary (ies) where the condition has (have) changed will have to be replaced with new ones. Entries corresponding to all other entries will remain unaltered. For the original example problem (Figure 7.3), the entries will be changed in the rows 4, 8 and 12-16 as result of changing boundary conditions along two of the boundaries (Figure 7.6). Only 7 out of 16 (43.75%) rows will change while the majority of the matrix will remain unaltered. The fraction of rows that actually changes gets smaller as the total number of nodes increase. For example, for the same problem, if we divide each direction in to 10 intervals instead of 5, we will have a total of 81 interior nodes. Similar change of boundary conditions will lead to change of 18 out of 81 rows (22.22%). For 100 intervals in each direction, the change is in 198 rows out of total 9801 rows (nodes), which is only 2.02%.

Once f_1-f_{16} are computed from the matrix equation, the values at the temporary nodes can be computed from the boundary conditions approximations used to replace these variables. For example, values at 16' and 16" can be computed from 7.84 and 7.85, respectively. Solution of Laplace equation with different types of boundary conditions is shown in the example below.

Example 7.3: Compute the steady state temperature (T) distribution on plate of size L_x=2 m and L_y=8 m, with the boundary conditions $\left.\frac{\partial T}{\partial x}\right|_{(0,y)} = 0$, $T(8,y) = 50\sin\left(\frac{py}{8}\right)$, $T(x,0)$=0 and $T(x,8)$=0. Use 2nd order finite difference approximations with Δx=1 m and Δy=2 m.

Solution: Steady state temperature distribution in a plate is governed by the Laplace equation. The plate with grid is shown below:



We have not numbered any node on the boundary where the values of temperature (T) are known or in other words, the boundary condition is of the 1st Type. Nodes 1, 2 and 3 are interior nodes. Therefore, *Laplace equation* needs to be solved. Using (7.70), we can write the following approximation for the node 1 with Δx=1 and Δy=2:

$$\frac{T_4 - 2T_1 + 50\sin\left(\frac{6p}{8}\right)}{1^2} + \frac{0 - 2T_1 + T_2}{2^2} = 0$$

or $-\frac{5}{2}T_1 + \frac{1}{4}T_2 + T_4 = -35.3553$

Similarly, we can write the following two equations for nodes 2 and 3, respectively:

$\frac{1}{4}T_1 - \frac{5}{2}T_2 + \frac{1}{4}T_3 + T_5 = -50$

$\frac{1}{4}T_2 - \frac{5}{2}T_3 + T_6 = -35.3553$

Now, for the boundary nodes 4, 5 and 6, we can write the 2nd order forward difference approximation for the derivative boundary condition similar to equation (7.84). In equation (7.84), we wrote a backward difference because the boundary was on the right and the unknown interior nodes were on the left of the boundary. In this case, since the boundary is on the left and the interior nodes are to the right of the boundary, we have to write forward difference approximation. The approximations for node 4 is:

$$\frac{-3T_4 + 4T_1 - 50\sin\left(\frac{6p}{8}\right)}{2(1)} = 0$$

or $T_4 = \frac{4}{3}T_1 - 11.7851$

Similarly, for nodes 5 and 6, we obtain:

$T_5 = \frac{4}{3}T_2 - 16.6667$

$T_6 = \frac{4}{3}T_3 - 11.7851$

Replacing the values of T_4, T_5 and T_6, in the equations for nodes 1, 2 and 3, we obtain the following set of equation:

$-\frac{7}{6}T_1 + \frac{1}{4}T_2 = -23.5702$

$\frac{1}{4}T_1 - \frac{7}{6}T_2 + \frac{1}{4}T_3 = -33.3333$

$\frac{1}{4}T_2 - \frac{7}{6}T_3 = -23.5702$

This is a tri-diagonal system of equation. This was solved by Thomas algorithme and the solution is:

$T_1 = T_3 = 28.9876$ and $T_2 = 40.9947$

Using these, the temperatures at node 4, 5 and 6 were computed from the equations for these nodes as:

$T_4 = T_6 = 26.865$ and $T_5 = 37.9929$

This problem could actually be solved by much less number of nodes by utilizing the symmetry of the problem and the boundary condition. For example, if we draw a horizontal line through node 2 and 5, this constitutes an axis of symmetry. All the boundary conditions are also symmetric with respect to this axis. This is important. It is not just the geometry of the plate that matters. The line joining nodes 1-2-3 is not an axis of symmetry. At the true axis of symmetry, one can always assume a no flux condtion. Therefore, we can put a pseudo boundary through nodes 2-5 and assume a no-flux condition, i.e., $\frac{\partial T}{\partial y} = 0$. Then, only interior node is the node 1. So, we can solve the whole problem by solving only one equation since in that case, T_2 can also be replaced in the equation for node 1 by making a backward difference approximation of this new boundary condition.

7.2.3 1st Order Wave Equation

Simplest form of first order wave equation is obtained if we put $a = 0$ in the advective-diffusion equation (7.34). It is written as:

$$\frac{\partial f}{\partial t} + u\frac{\partial f}{\partial x} = 0 \tag{7.87}$$

Application of our *m-CD* scheme to this equation leads to the following discrete equation:

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = -mu_i^n \frac{f_{i+1}^n - f_{i-1}^n}{2\Delta x} - (1-m)u_i^{n+1} \frac{f_{i+1}^{n+1} - f_{i-1}^{n+1}}{2\Delta x} \tag{7.88}$$

$$\text{or } (1-m)u_i^{n+1}\frac{\Delta t}{2\Delta x} f_{i+1}^{n+1} + f_i^{n+1} - (1-m)u_i^{n+1}\frac{\Delta t}{2\Delta x} f_{i-1}^{n+1} = -mu_i^n \frac{\Delta t}{2\Delta x} f_{i+1}^n + f_i^n + mu_i^n \frac{\Delta t}{2\Delta x} f_{i-1}^n \tag{7.89}$$

For constant u, the above translates to:

$$(1-m)\frac{C}{2} f_{i+1}^{n+1} + f_i^{n+1} - (1-m)\frac{C}{2} f_{i-1}^{n+1} = -m\frac{C}{2} f_{i+1}^n + f_i^n + m\frac{C}{2} f_{i-1}^n \tag{7.90}$$

We will see in Section 7.3 that for $m = 1$ (Euler Forward), the method is unconditionally unstable for this problem with central difference scheme for spatial discretization. Let us now analyze the scheme for accuracy and consistency. Using the expansions 7.52-54 and identities derived from the original PDE (7.87) relating the time derivative with space derivative (similar to 7.58 and 7.60), we obtain the following:

$$\left.\frac{\partial f}{\partial t}\right|_i^n + u\left.\frac{\partial f}{\partial x}\right|_i^n = -\left(m-\tfrac{1}{2}\right)u^2\Delta t\left.\frac{\partial^2 f}{\partial x^2}\right|_i^n + \left(\tfrac{m}{2}-\tfrac{1}{3}\right)u^3\Delta t^2\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n - \frac{\Delta x^2}{6}\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n + HOT \qquad (7.91)$$

The accuracy of the scheme is 1st order in time and 2nd order in space as expected except for m = ½, when it is 2nd order in both time and space. Also, for all values of m other than ½, there is numerical diffusion. We will see in section 7.6 that for m = 1 (Euler Forward), the method is unconditionally unstable in addition to having numerical diffusion. A possible way out of numerical diffusion while preserving the explicit structure is to simulate the first term on the right in the discrete equation. If we add a central difference approximation of the first residual term on the right with the existing method and set m = 1 (explicit), the numerical diffusion term is also simulated. This reduces the numerical diffusion as well as provides the advantage of explicit computation. The resulting scheme is:

$$\frac{f_i^{n+1}-f_i^n}{\Delta t} = -u\frac{f_{i+1}^n - f_{i-1}^n}{2\Delta x} + \frac{u^2\Delta t}{2}\frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{\Delta x^2} \qquad (7.92)$$

$$\text{or } f_i^{n+1} = f_i^n - \frac{C}{2}\left(f_{i+1}^n - f_{i-1}^n\right) + \frac{C^2}{2}\left(f_{i+1}^n - f_i^n + f_{i-1}^n\right) \qquad (7.93)$$

This is known as *Lax-Wendorf* scheme. It is conditionally stable (Exercise 7.3). Another commonly used explicit method to counter numerical diffusion is the upwind scheme. In this, the advection term is simulated by a Backward difference approximation in place of central difference. The scheme is:

$$\frac{f_i^{n+1}-f_i^n}{\Delta t} + u_i^n\frac{f_i^n - f_{i-1}^n}{\Delta x} = 0 \qquad (7.94)$$

For a constant u, the above yields:

$$f_i^{n+1} = f_i^n - C\left(f_i^n - f_{i-1}^n\right) \qquad (7.95)$$

If we perform a truncation error analysis, the upwind scheme results in,

$$\left.\frac{\partial f}{\partial t}\right|_i^n + u\left.\frac{\partial f}{\partial x}\right|_i^n = \left(\frac{u\Delta x}{2} - \frac{u^2\Delta t}{2}\right)\left.\frac{\partial^2 f}{\partial x^2}\right|_i^n + \left(\frac{u\Delta x^2}{6} + \frac{u^3\Delta t^2}{6}\right)\left.\frac{\partial^3 f}{\partial x^3}\right|_i^n + HOT \qquad (7.96)$$

The numerical diffusion is eliminated if we choose C =1. However, exact choice depends on the stability condition of the scheme, which we shall learn in Section 7.3. If the velocity in the equation $u < 0$, the equation becomes:

$$\frac{\partial f}{\partial t} - u\frac{\partial f}{\partial x} = 0 \qquad (7.97)$$

The upwind scheme for this case is,

$$\frac{f_i^{n+1}-f_i^n}{\Delta t} - u_i^n\frac{f_{i+1}^n - f_i^n}{\Delta x} = 0 \qquad (7.98)$$

Notice that the forward difference is replaced by the backward difference. In fact, for any moving front, approximation involves a point behind the front. For a forward moving wave ($u > 0$), the point (i-1) is taken and for the receding wave ($u < 0$), the point (i+1) is taken. We leave it to the readers to show the later also yields a similar truncation error as in the case of $u > 0$. We have shown the upwind scheme with first order Forward/Backward differences. For increasing the accuracy or order of the method in space, one may incorporate higher order Forward/Backward difference approximations for the space derivatives involving more points.

The solution of 1st order wave equation (7.87) requires one initial condition and one boundary condition. However, for numerical solution using central difference scheme in space (*m-CD* and *Lax-Wendorf*), it requires a 2nd boundary condition. This is because, adjacent to both the boundaries, the difference approximation in spatial dimension involves a point on the boundary. This can be avoided by incorporating a backward or forward difference approximation adjacent to the open boundary. The later is automatically achieved in the upwind schemes. Example application of various schemes are shown (Example 7.4) demonstrating numerical diffusion.

Example 7.4. Consider the 1-D advection diffusion equation of Example 7.2. The equation is typically used for contaminant transport in wide rivers and estuaries. For narrow rivers, the dispersion can be neglected reducing the advection-diffusion equation to 1-D wave equation. Solve it using the upwind scheme with $\Delta x = 0.1$, $\Delta t = 0.1$ and 0.5. Compare the results graphically. The equation is given by:

$$\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} = 0 \qquad 0 \le x < \infty$$

where,

$$u = 0.2;\ \ C(0,t) = 1;\ \ C(x,0) = 0$$

Solution:

The upwind scheme is given by equation (7.94), which for the present case of constant velocity, we write as follows:

$$C_i^{n+1} = C_i^n - \tfrac{u\Delta t}{\Delta x}\left(C_i^n - C_{i-1}^n\right)$$

For $\Delta t = 0.1$, $\dfrac{u\Delta t}{\Delta x} = 0.2$

This can be explicitly solved for all i given the initial conditions of $C_i^n = 0$ for $i = 1,2,3\cdots\infty$ and the boundary condition $C_0^0 = 0$. For example, at the first time step, $t = 0.1$, the concentrations are:

$$C_1^{0.1} = 0 - 0.2(0-1) = 0.2$$

$$C_2^{0.1} = C_3^{0.1} = \cdots = C_\infty^{0.1} = 0$$

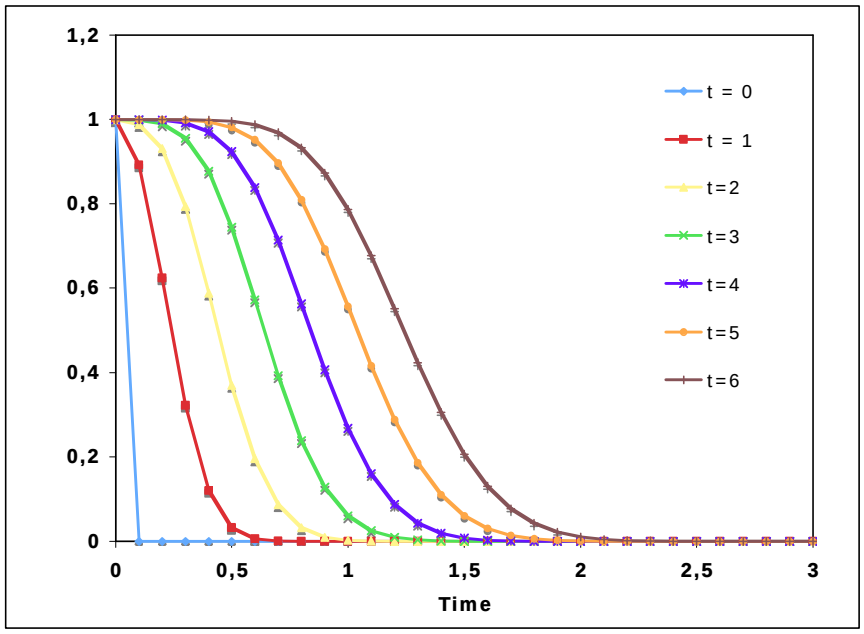
At the second time step, $t = 0.2$, the concentrations are:

$$C_1^{0.2} = 0.2 - 0.2(0.2-1) = 0.36$$

$$C_2^{0.2} = 0 - 0.2(0 - 0.2) = 0.04$$

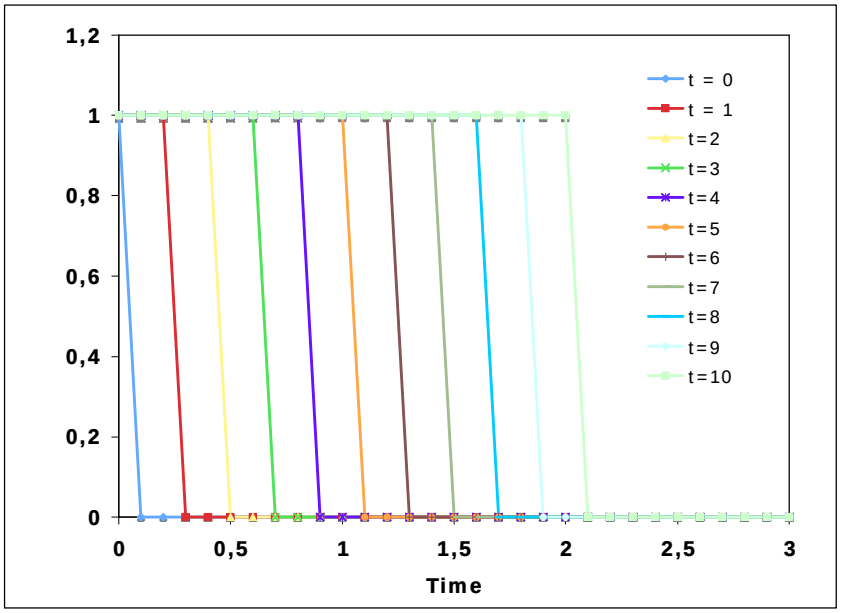
$$C_3^{0.1} = \cdots = C_\infty^{0.1} = 0$$

We can compute the solutions for $\Delta t = 0.5$ similarly. The progress of solution with time is shown in the figures below. First we show the solution with $\Delta t = 0.1$.



We see that a significant numerical diffusion is added and as a result, the expected to progression of sharp front is diffused. This is expected with upwind scheme for CFL number less than unity (eq 7.96). The line at t = 0 is supposed to drop vertically from 1 to 0 but for the plotting program, which joins the point at $x = 0$ and Δx with a straight line. This is a common artifact of nearly all computer plotting programs and one needs to be careful while interpreting the plotted result.

For $\Delta t = 0.5$, the CFL number is unity and we expect the numerical diffusion to disappear. This indeed happens as we can see in the following figure where the sharp front progresses with time.



7.2.4 2nd Order Wave Equation

We come across the 2nd order wave propagation equation in many engineering problems involving oscillations and waves. Areas of earthquake engineering and hydraulics use it extensively. In most general form, the equation appears as:

$$\frac{\partial^2 f}{\partial t^2} = u(x)^2 \frac{\partial^2 f}{\partial x^2} \tag{7.99}$$

The wave celerity u may or may not be a function of x. For solution, we require two initial conditions and two boundary conditions. The initial conditions generally looks like,

$$f(x,0) = a \quad \text{and} \quad \left.\frac{\partial f}{\partial t}\right|_{(x,0)} = b \tag{7.100}$$

The boundary conditions in physical problem are often specified asymptotically at infinity making $-\infty < x < \infty$. However, for the solution using numerical method, we have to restrict the boundaries to finite space and specify meaningful conditions at the boundaries. This is generally governed by the physics of the problem. Let us denote these finite boundaries as ($-L$, L)

In order to understand the behaviour of the equation (7.99), let us factorize it as follows:

$$\left(\frac{\partial}{\partial t} + u\frac{\partial}{\partial x}\right)\left(\frac{\partial}{\partial t} - u\frac{\partial}{\partial x}\right) f = 0 \tag{7.101}$$

It is easy to see that the above describes two first order wave equations, one right moving wave and one left moving wave. Therefore, for (0, L), the solution will follow the right moving wave,

$$\frac{\partial f}{\partial t} + u\frac{\partial f}{\partial x} = 0 \tag{7.102}$$

and, for (-L, 0), the solution will involve the left moving wave,

$$\frac{\partial f}{\partial t} - u\frac{\partial f}{\partial x} = 0 \tag{7.103}$$

This provides a rational way to specify boundary conditions at the finite boundaries as follows:

$$\left.\frac{\partial f}{\partial t}\right|_{(L,t)} = -u(L)\left.\frac{\partial f}{\partial x}\right|_{(L,t)} \quad \text{and} \quad \left.\frac{\partial f}{\partial t}\right|_{(-L,t)} = u(-L)\left.\frac{\partial f}{\partial x}\right|_{(-L,t)} \tag{7.104}$$

For the ease of numerical solution, let us break down PDE into two first order PDEs in time. This will enable us to apply the methods already learned for IVPs in Chapter 6 and in the earlier sections for diffusion equation. This can be accomplished as follows:

$$\begin{aligned} \frac{\partial f}{\partial t} &= y \\ \frac{\partial y}{\partial t} &= u(x)^2\frac{\partial^2 f}{\partial x^2} \end{aligned} \tag{7.105}$$

The second PDE is a diffusion equation which we already solved in section 7.2.1. Using central difference for the spatial discretization, this PDE can be converted to a system of IVPs. For spatial discretization, the boundary conditions are either given by the physics of the problem or obtained from (7.104). The first PDE is already in the form of a system of IVP. Both the system of IVPs can then be solved using any scheme described in Chapter 6. Notice that the initial conditions for the both the ODEs are known from 7.100. Once again, popular scheme for this is 4th order Runge Kutta method which combines the computational advantage of an explicit method with high order of accuracy and decent stability property.

An alternative to decomposition of the original PDE is to use direct discretization in time and space. There are a number of such methods in the numerical modeling literature. One commonly used scheme with 2nd order accuracy in both space and time is given below:

$$\frac{f_j^{n+1} - 2f_j^n + f_j^{n-1}}{\Delta t^2} = u_j\left[\frac{1}{4}\frac{f_{j+1}^{n+1} - 2f_j^{n+1} + f_{j-1}^{n+1}}{\Delta x^2} + \frac{1}{2}\frac{f_{j+1}^{n} - 2f_j^{n} + f_{j-1}^{n}}{\Delta x^2} + \frac{1}{4}\frac{f_{j+1}^{n-1} - 2f_j^{n-1} + f_{j-1}^{n-1}}{\Delta x^2}\right] \tag{7.106}$$

The above is the approximation of the PDE at spatial node j. Therefore, this approximation can be written at all the spatial nodes and the resulting set of equation can be solved to obtain the values of f at various time steps. Incorporation of the

boundary and initial conditions are similar to those described in the earlier sections. It is easy to see that the left hand side is a central difference approximation of the double derivative in time and the right hand side is the weighted average of the central difference approximation of the double derivative in space at three difference time steps. So, far we have used one spatial dimension in all the PDEs described so far. We will describe one equation in the next section for multiple spatial dimensions.

Example 7.5: Transverse displacement (y) of a vibrating string is governed by the following equation:

$$\frac{\partial^2 y}{\partial t^2} = 4\frac{\partial^2 y}{\partial x^2}, \quad 0 < x < 6, \;\; t \geq 0$$

The string is fixed at two ends, *i.e.,* $y(0,t) = y(6,t) = 0$
The string is initially at rest, *i.e.,* $y(x,0) = 0$
In order to start the vibration, the following velocity was applied to the string by striking appropriately: $\left.\frac{\partial y}{\partial t}\right|_{(x,0)} = \sin\left(\frac{\pi x}{2}\right)$
Decompose into two first order PDEs in time (second order in space), similar to eq (7.105). Using central difference scheme for semi-discretization with $\Delta x = 1$, formulate the system of ODEs to be solved for the solution.

Solution:

Let us define a new variable $z(x, t)$ as,

$$\frac{\partial y}{\partial t} = z$$

The PDE then transforms to:

$$\frac{\partial z}{\partial t} = 4\frac{\partial^2 y}{\partial x^2}$$

Using $\Delta x = 1$, we have 7 nodes in the spatial domain. Let us number them as 0-6 with 0 and 6 on the boundary. So, from the boundary conditions, we have $y_0 = y_6 = 0$. Therefore, the goal is to obtain $y_1 - y_5$. Semi-discretization of the above set of PDEs lead to the following equation for any arbitrary interior node (i):

$$\frac{dy_i}{dt} = z_i$$

$$\frac{dz_i}{dt} = 4\frac{y_{i+1} - 2y_i + y_{i-1}}{\Delta x^2} = 4y_{i-1} - 8y_i + 4y_{i+1}$$

Writing the above set of equations for all the interior nodes 1-5, we obtain the following set of ODEs:

$$\frac{d}{dt}\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix} \text{ or } \frac{d\mathbf{y}}{dt} = \mathbf{z}$$

$$\frac{d}{dt}\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix} = \begin{bmatrix} -8 & 4 & 0 & 0 & 0 \\ 4 & -8 & 4 & 0 & 0 \\ 0 & 4 & -8 & 4 & 0 \\ 0 & 0 & 4 & -8 & 4 \\ 0 & 0 & 0 & 4 & -8 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} \text{ or } \frac{d\mathbf{z}}{dt} = \mathbf{A}\text{y}$$

The initial conditions for the two systems of ODEs are provided by two initial conditions given in the problems, as follows:

First initial condition $y(x,0) = 0$ leads to $\mathbf{y^0} = \begin{bmatrix} y_1^0 \\ y_2^0 \\ y_3^0 \\ y_4^0 \\ y_5^0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

Second initial condition $\left.\frac{\partial y}{\partial t}\right|_{(x,0)} = \sin\left(\frac{px}{2}\right)$ leads to $\mathbf{z^0} = \begin{bmatrix} z_1^0 \\ z_2^0 \\ z_3^0 \\ z_4^0 \\ z_5^0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \\ 0 \\ 1 \end{bmatrix}$

These systems of equations are now straight forward and any ODE method can be used to solve them. Once again, 4th order Runge Kutta is frequently used. We leave the reader to practice the solution since, we have already shown such solutions at many other places (Examples 6.10, 6.11, 6.12, 7.1 etc.)

7.2.5 Diffusion Equation in 2-D

Almost all the physical problems of nature contain 3 spatial dimensions (eq 7.1). However, in many cases, some reasonable practical assumption leads to elimination of the one of the spatial dimension leading to 2-D representation of physical world (eq. 7.3). In Laplace equation, we have seen such a process and how to solve the equation using numerical method. However, Laplace equation is time invariant process or *Steady State* of a physical process. In the transition state, the same process will be represented by the following equation:

$$\frac{\partial f}{\partial t} = a\left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}\right) \tag{7.107}$$

Solution of this equation requires an initial condition in addition to the boundary conditions similar to Laplace equation. Comparing with eq (7.33), it is easy to see that it is a 2-D diffusion equation. Therefore, the methods described in section 7.2.1 can be applied for the solution of this equation as well. If we perform spatial discretization, we can write the following IVP for node (*i, j*):

$$\frac{df_{i,j}}{dt} = a\left(\frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{\Delta x^2} + \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{\Delta y^2}\right) \tag{7.108}$$

After incorporating the boundary conditions, we have seen for the Laplace equation that the right hand side can be expressed as a matrix equation in terms of values of f at the nodes (eq 7.71). Following similar notations, the above can be expressed as,

$$\frac{d\bar{f}}{dt} = \mathbf{A}\bar{f} + \mathbf{b} \tag{7.109}$$

In fact, for a transition state temperature distribution problem for the plate shown in Figure (7.3) with same boundary conditions, the matrix **A** and vector **b** remains the same as in the Laplace equation. We only need to specify the initial condition. This is a system of linear IVP similar to eq (7.39) that can be solved using any of the method described in Chapter 6 given the initial condition. The simplest explicit method is the Euler Forward which in matrix representation, translates to:

$$\bar{f}^{n+1} = \bar{f}^{n} + \Delta t\left[\mathbf{A}\bar{f}^{n} + \mathbf{b}\right] \tag{7.110}$$

Each nodal equation in the above set of equation is a discrete form of the original PDE (7.107):

$$\frac{f_{i,j}^{n+1} - f_{i,j}^{n}}{\Delta t} = a\left(\frac{f_{i+1,j}^{n} - 2f_{i,j}^{n} + f_{i-1,j}^{n}}{\Delta x^2} + \frac{f_{i,j+1}^{n} - 2f_{i,j}^{n} + f_{i,j-1}^{n}}{\Delta y^2}\right) \tag{7.111}$$

$$\text{or } f_{i,j}^{n+1} = f_{i,j}^{n} + a\frac{\Delta t}{\Delta y^2} f_{i,j-1}^{n} + a\frac{\Delta t}{\Delta x^2} f_{i-1,j}^{n} - 2\left(a\frac{\Delta t}{\Delta x^2} + a\frac{\Delta t}{\Delta y^2}\right) f_{i,j}^{n} + a\frac{\Delta t}{\Delta x^2} f_{i+1,j}^{n} + a\frac{\Delta t}{\Delta y^2} f_{i,j+1}^{n} \tag{7.112}$$

Computational effort required in solving this is not much more than its 1-D form. However, as we know, use of the explicit method such as Euler Forward is severely restricted by its stability property. One can also use 4th order Runge Kutta for increasing accuracy. However, one is often interested in the implicit methods for better stability properties. Let us see application of Euler Backward method to eq (7.107):

$$\frac{f_{i,j}^{n+1} - f_{i,j}^{n}}{\Delta t} = a\left(\frac{f_{i+1,j}^{n+1} - 2f_{i,j}^{n+1} + f_{i-1,j}^{n+1}}{\Delta x^2} + \frac{f_{i,j+1}^{n+1} - 2f_{i,j}^{n+1} + f_{i,j-1}^{n+1}}{\Delta y^2}\right) \tag{7.113}$$

$$\text{or } -a\frac{\Delta t}{\Delta y^2}f_{i,j-1}^{n+1} - a\frac{\Delta t}{\Delta x^2}f_{i-1,j}^{n+1} + \left[1+2\left(a\frac{\Delta t}{\Delta x^2}+a\frac{\Delta t}{\Delta y^2}\right)\right]f_{i,j}^{n+1} - a\frac{\Delta t}{\Delta x^2}f_{i+1,j}^{n+1} - a\frac{\Delta t}{\Delta y^2}f_{i,j+1}^{n+1} = f_{i,j}^{n} \tag{7.114}$$

In the matrix notation, this is essentially a discrete form of the system of equation (7.109):

$$[\mathbf{I} - \Delta t\mathbf{A}]\overline{f}^{n+1} = \overline{f}^{n} + \Delta t\mathbf{b} \tag{7.115}$$

Recall from Section 7.2.2 that the vector **b** consists of the terms representing the known values at the boundary node. In this case also, the contributions will come from the known nodal values in equation (7.114). It is now easy to see that we need to solve this system of equation using an iterative method as described for Laplace equation. The difference is that, in Laplace equation we have to solve it once but here, we have to solve it at each time step. This will always be the case for any implicit method irrespective of the order of method. One can combine equations 7.111 and 7.111 to form a *m-CD* scheme.

The coefficient matrix in the application of an implicit method for this problem is sparse. As we have seen for the Laplace equation, each row can contain up to maximum of five elements. There is a way to formulate it in the form of a *Block Tridiagonal* matrix where each element is a small matrix. One can then apply a modified form of *Thomas Algorithme* to solve it. This reduces the computational effort at each time step and the solution can be found using a finite number of computations as opposed to the iterative methods. A second option is to modify the time stepping scheme such that the coefficient matrix is a tri-diagonal. Then one can solve it at each time step using vector operations of *Thomas Algorithme* described in Chapter 2. Such a method is *Alternating Direction Implicit* (ADI) scheme. In this method, one takes one time step in two ½ steps. In one half step, the x-direction space derivative is implicit and in the next half step, the y-direction space derivative is implicit. The scheme is shown as follows:

1st Half Step:

$$f_{i,j}^{n+\frac{1}{2}} = f_{i,j}^{n} + a\frac{\Delta t}{2\Delta x^2}\left(f_{i+1,j}^{n+\frac{1}{2}} - 2f_{i,j}^{n+\frac{1}{2}} + f_{i-1,j}^{n+\frac{1}{2}}\right) + a\frac{\Delta t}{2\Delta y^2}\left(f_{i,j+1}^{n} - 2f_{i,j}^{n} + f_{i,j-1}^{n}\right) \tag{7.116}$$

$$\text{or } -a\frac{\Delta t}{2\Delta x^2}f_{i+1,j}^{n+\frac{1}{2}} + \left(1+a\frac{\Delta t}{\Delta x^2}\right)f_{i,j}^{n+\frac{1}{2}} - a\frac{\Delta t}{2\Delta x^2}f_{i-1,j}^{n+\frac{1}{2}} = a\frac{\Delta t}{2\Delta y^2}f_{i,j+1}^{n} + \left(1-a\frac{\Delta t}{\Delta y^2}\right)f_{i,j}^{n} + a\frac{\Delta t}{2\Delta y^2}f_{i,j-1}^{n} \tag{7.117}$$

2nd Half Step:

$$f_{i,j}^{n+1} = f_{i,j}^{n+\frac{1}{2}} + a\frac{\Delta t}{2\Delta x^2}\left(f_{i+1,j}^{n+\frac{1}{2}} - 2f_{i,j}^{n+\frac{1}{2}} + f_{i-1,j}^{n+\frac{1}{2}}\right) + a\frac{\Delta t}{2\Delta y^2}\left(f_{i,j+1}^{n+1} - 2f_{i,j}^{n+1} + f_{i,j-1}^{n+1}\right) \tag{7.118}$$

$$\text{or } -a\frac{\Delta t}{2\Delta y^2}f_{i,j+1}^{n+1} + \left(1+a\frac{\Delta t}{\Delta y^2}\right)f_{i,j}^{n+\frac{1}{2}} - a\frac{\Delta t}{2\Delta y^2}f_{i,j-1}^{n+\frac{1}{2}} = a\frac{\Delta t}{2\Delta x^2}f_{i+1,j}^{n+\frac{1}{2}} + \left(1-a\frac{\Delta t}{\Delta x^2}\right)f_{i,j}^{n+\frac{1}{2}} + a\frac{\Delta t}{2\Delta x^2}f_{i-1,j}^{n+\frac{1}{2}} \tag{7.119}$$

In order to avoid accumulation of errors in one direction, it is recommended to change the order of the half steps. For example, in equations (7.116-119), *x*-direction is

implicit for the first half step and *y*-direction is implicit for the next half step. This will be reversed in the next time step. It is easy to see that in both the half steps, the coefficient matrix is tri-diagonal. This makes the solution of the equation simpler.

So far, we have presented a lot of numerical methods for the solution of a number of important engineering PDEs. During presentation of the methods, we have made sure that all the methods are consistent, i.e., in the asymptotic limit of $\Delta t \to 0$ and $\Delta x \to 0$, the discrete approximations converge to the original PDEs. This was ensured by the fact that the minimum power of Δt and Δx in the truncation error was 1. This is not true for all the methods one can design. We shall she in Exercise 7.2 a method called *Dufort-Frankel scheme*, which is not consistent because a term ($\Delta t^2/\Delta x^2$) appears in the truncation error. This term will go to zero conditionally depending on which of Δt and Δx goes to zero faster. If they go to zero at the same rate, the ratio becomes a constant. The term is zero only if Δt approach zero faster than Δx.

Recall that for convergence (section 6.3.2.8), the method has to be *consistent* as well as *stable*. In the next section, we will demonstrate the methods for stability analysis of the numerical schemes for PDEs.

Exercise 7.2

1. Temperature distribution in a plate is governed by the following equation: $\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$ in $x \in (0,1), y \in (0,1)$, subject to the boundary conditions $T(0,y) = T(1,y) = T(x,0) = 0$ and $T(x,1) = \sin px$. The exact solution of the problem is given by $T(x, y) = \sin px \frac{\sinh py}{\sinh p}$. Develop a computer code for the numerical solution of the problem using central difference approximations and graphically compare the numerical solution with the exact solution at x = 0.5 for $Dx = Dy = 0.1$.

2. Consider the following inhomogeneous heat equation:

$$\frac{\partial T}{\partial t} = a \frac{\partial^2 T}{\partial x^2} + \left(p^2 - 1\right) e^{-t} \sin px \qquad 0 \le x \le 1;\ t \ge 0$$

with initial and boundary conditions $T(0,t) = T(1,t) = 0$ and $T(x,0) = \sin px$

a) Write a computer program to solve the equation using Euler explicit-Central difference approximations, for $a = 1$, $Dx = 0.05$ and $Dt = 0.001$. Plot $T(x)$ vs. x at t = 0.0, 0.5, 1.0, 1.5 and 2.0 in one plot.

b) Take new $Dt = 0.0015$ and solve the equation for the same a and Dx. Plot $T(x)$ vs. x in the 2nd plot at t = (0.0, 0.075, 0.15, 0.153, 0.1545, 0.156).

c) Explain the results obtained in (a) and (b).

3. Toxic pollutant transport in a river is governed by the following equation:

$$\frac{\partial c}{\partial t} + v\frac{\partial c}{\partial x} - D\frac{\partial^2 c}{\partial x^2} + kc = 0;\ 0 \le x \le 1;\ \ c(0,t) = c_0;\ \ \left.\frac{\partial c}{\partial x}\right|_{(1,t)} = 0;\quad c(x,0) = x^2 e^{-x}$$

Discretize the above equation using *m*-CD scheme and express in terms of CFL number and Grid Peclet Number. The velocity v = 0.5 m/sec and the dispersion coefficient D = 0.1 m2/s. Normalized concentration at the inlet c0 = 1 mg/L. Solve the equation with m = 0, ½ and 1 using Δx = 0.1 and Δt = 0.1, 0.2 and 0.3. Write a program to solve the equation for all these cases. For each m and Δt combination (9 plots), plot the concentration vs. x at times t = 0, 0.6 and 1.2. Comment on the effect of m and Δt on the numerically observed transport behaviour of the pollutant. Can you explain some of the behaviour in the light of analysis done in this chapter (numerical diffusion, stability, etc.)

4. Consider the following hyperbolic dimensionless wave equation:

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} \text{ in } x \in [0,1] \text{ for } t = 0$$

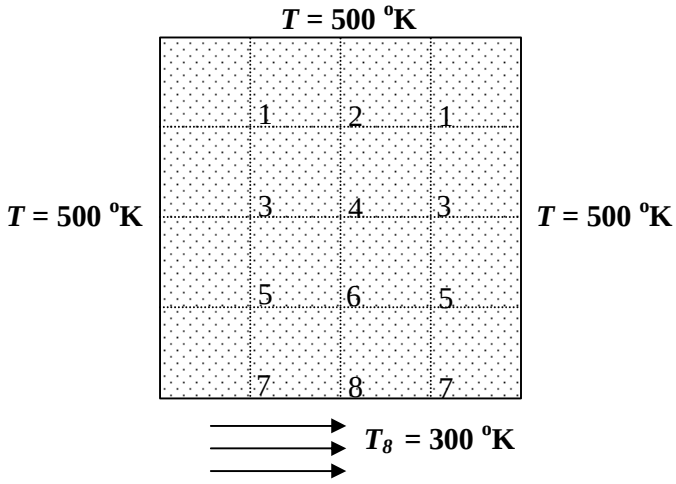
subject to $u(0,t) = \sin 2pt$, $u_x(1,t) = 0$, $u(x,0) = 0$, $u_t(x,0) = 2p \cos 2px$.

a) Discretize the equation express in the matrix form using the $O(\Delta x^2, \Delta t^2)$ accurate implicit scheme shown in equation (7.106). Incorporate the time derivative initial condition and the space derivative boundary condition in such a manner, which preserves the accuracy of the scheme. Find the solution using Δx = 0.25 and Δt = 0.1 at t = 0.5.

b) Decompose the PDE into two first order PDEs in time, similar to (7.105) and example 7.5. Solve it using 4th order Runge Kutta method in time and Central Difference in space with the same Δx and Δt as (a).

c) Compare two solution methods at t = 0.5.

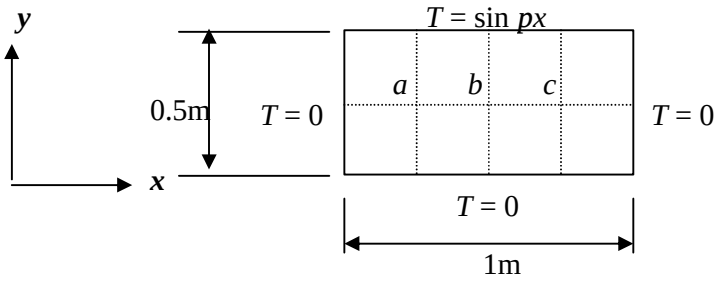
5. Consider the fire clay slab of dimension 1m x 1m shown in the figure below:



Three surfaces of the slab are maintained at 500 ^{o}K while the remaining surface is exposed to an air stream of temperature 300 ^{o}K and heat transfer coefficient (h) of 10 W/m^2- ^{o}K. Thermal conductivity (k) of the slab is 1 W/m- ^{o}K. The steady state temperature (T) distribution in the slab is governed by 2-D Laplace equation. Use a grid size of $\Delta x = \Delta y = 0.25$m. Formulate the matrix equation and obtain temperature at eight nodal points shown in the above figure.

Formulation of the BC's at the bottom boundary: The algebraic sum of the conduction heat flux and convective heat flux along the boundary (say node 7) is zero. In order to calculate the conduction flux at node 7, one has to calculate the energy flux from node 5 to node 7 by multiplying the thermal conductivity (k) of the solid with the temperature gradient. This quantity will be positive at node 7. For the convective flux, one has to consider, Newton's law of cooling. The film heat transfer coefficient, h has to be multiplied by the temperature difference between node 7 and the ambient temperature (fluid temperature). This energy will move out from the node.

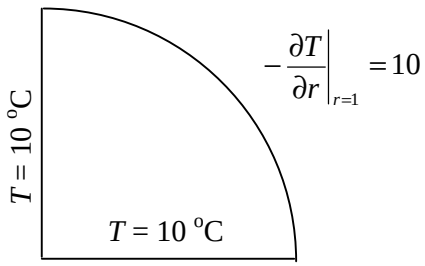
6. Steady state temperature (T) distribution in a plate is governed by the Laplace equation. The boundary conditions are as shown in the figure. Calculate the temperatures at points a, b and c. Given $\Delta x = \Delta y = 0.25$ m. 10



7. Temperature distribution in the quarter circular disc shown in the figure is governed by,

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right)+\frac{1}{r^2}\frac{\partial^2 T}{\partial q^2}=0$$

T(r, 0) = 10, T(r, p/2)=10, T(0, q)=10 and $-\dfrac{\partial T(1,q)}{\partial r}=10$. Use Δr = 1/2, Δq = p/6.



8. The 1-D Heat equation with a source term is

$$\frac{\partial T}{\partial t} = a\,\frac{\partial^2 T}{\partial x^2} + S(x)$$

where,

$$T(x,0) = 0;\ T(0,t) = 0;\ T(1,t) = T_{Steady}(1);\ S(x) = -(x^2 - 4x + 2)e^{-x};\ T_{Steady}(x) = x^2 e^{-x}$$

Discretize the above equation using Crank-Nicholson scheme (*q*-method in time with $q = ½$ and central difference in space). Formulate the matrix equation using $a = 1$, $Dx = 0.25$ and $Dt = 0.1$. Solve it for one time step.

9. Consider the 1-D Convection Diffusion equation:

$$\frac{\partial T}{\partial t} + u\,\frac{\partial T}{\partial x} = a\,\frac{\partial^2 T}{\partial x^2} \qquad 0 \le x \le 1$$

where,

$$u = 0.08;\ a = 0.001;\ T(0,t) = 0;\ \ T(1,t) = 0;\ \ T(x,0) = \begin{cases} 1-(10x-1)^2 & \text{for } 0 \le x \le 0.25 \\ 0 & \text{for } 0.25 \le x \le 1 \end{cases}$$

Using 2nd order R-K method (given) in time and Central Difference approximation in space, $Dx = 0.25$ and $Dt = 0.1$, solve for one time step. 20

10. A square plate 3 cm x 3 cm is initially at a temperature of 0^0. Suddenly, the temperature at x=3 cm and y=3 cm is raised to 100^0. Write the equations (don't solve them!) for the solution of the heat-conduction equation $\frac{\partial T}{\partial t} = k\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$ at half-time-step ($\Delta t/2$) and full step (Δt) using the ADI scheme for $\Delta x = \Delta y = 1$ cm, k=1 cm^2/s and $\Delta t = 1$ s.

7.3 Stability Analysis

The PDEs such as diffusion, advection-diffusion and 1st order wave equation were reduced to a system of IVPs by *semi-discretization*. Therefore, we should be able to analyze for stability using the matrix method described in section 6.5.1. Let us illustrate this using the diffusion equation with Euler Forward for time and Central Diffrence approximation for space derivatives (*EF-CD* scheme). After application of the central difference scheme for the space derivative, one can write an ODE for each node (*j*) as follows:

$$\frac{df_j}{dt} = a\,\frac{f_{j+1} - 2f_j + f_{j-1}}{\Delta x^2} \qquad (7.120)$$

Let us use two 1st Type zero boundary conditions for the ease of illustration. Then the above can be written in the matrix form as:

$$\frac{d\bar{f}}{dt} = \mathbf{A}\bar{f} \qquad (7.121)$$

where, **A** is a tridiagonal matrix whose diagonal elements are $-\frac{2a}{\Delta x^2}$ and the off-diagonal elements are $\frac{a}{\Delta x^2}$ and $\overline{f}$ is the vector containing the values of the state variable f at the interior nodes. This is in the same form as the model problem for the stability analysis of a system of IVP (section 6.5.1). If we used non-zero boundary conditions, a vector **b** containing the constants from the boundary conditions will appear on the right hand side. We have already seen that the constant additive vector **b** has no effect on the stability of the system (section 6.3.2). Therefore, the eigenvalues of the coefficient matrix is of interest. The ratio of maximum to minimum eigenvalue will determine the stiffness of the system and the magnitude of the maximum eigenvalue will determine the stability for the application of Euler Forward method (section 6.5.1). In order to see the importance of the eigenvalues, let us use Theorem 2.8 to diagonalize matrix **A** using a matrix **X** whose columns are the eigenvectors:

$$\mathbf{A} = X\Lambda X^{-1} \text{ or } \Lambda = X^{-1}\mathbf{A}X \tag{7.122}$$

where, Λ is a diagonal matrix containing the eigenvalues (l_k's). Using this relation, the equation 7.121 can be written as:

$$\frac{d\left(X^{-1}\overline{f}\right)}{dt} = \Lambda\left(X^{-1}\overline{f}\right) \tag{7.123}$$

Let us denote a new vector as: $\overline{y} = X^{-1}\overline{f}$. Then the above equation is:

$$\frac{d\overline{y}}{dt} = \Lambda\overline{y} \tag{7.124}$$

Each equation in the above system of equation looks as follows:

$$\frac{dy_k}{dt} = l_k y_k \tag{7.125}$$

where, y_k is related to f_k as $f_k = X^{(k)}y_k$ and $X^{(k)}$ is the k^{th} column of the matrix X. The equation (7.125) is the model equation for single IVP (equation 6.57). The solution ofcourse is $y_k = y_k(0)e^{l_k t}$ and the initial condition on y_k can be easily obtained from the original initial conditions specifying f_k's using their relation through matrix X. It is now clear that the stability of any time stepping scheme for the solution of the PDE depends on whether it is stable for the IVP (7.125) for all k. We already know from the Chapter 6 that for real and negative l_k's, most of the methods will have a finite stability limit while some are unconditionally stable. For purely imaginary l_k's, the method will be stable if its stability region contains some part of the imaginary axis. When l_k's have both negative real and imaginary components, the stability will depend on whether the chosen time step value is within the stability region. So, the problem now boils down to computing the l_k's of matrix **A** of equation (7.121) and compute the stability limit for various time stepping schemes in terms of these eigenvalues. Remember that the l_k's obtained from equation (7.121) is applicable only for the central difference scheme applied to the space derivative. If the

discretization scheme for the space derivative is changed, the matrix A will also change and so are the l_k's. Let us represent the tri-diagonal matrix **A** in equation (7.121) as:

$$\mathbf{A} = \frac{a}{\Delta x^2}\begin{bmatrix} -2 & 1 & 0 & 0 & \cdot\cdot & \cdot\cdot & 0 & 0 \\ 1 & -2 & 1 & 0 & \cdot\cdot & \cdot\cdot & 0 & 0 \\ 0 & 1 & -2 & 1 & \cdot\cdot & \cdot\cdot & 0 & 0 \\ 0 & 0 & 1 & -2 & \cdot\cdot & \cdot\cdot & \cdot\cdot & 0 \\ \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot & \cdot\cdot \\ 0 & 0 & \cdot\cdot & \cdot\cdot & \cdot\cdot & -2 & 1 & 0 \\ 0 & 0 & \cdot\cdot & \cdot\cdot & \cdot\cdot & 1 & -2 & 1 \\ 0 & 0 & \cdot\cdot & \cdot\cdot & \cdot\cdot & 0 & 1 & -2 \end{bmatrix} \tag{7.126}$$

The tri-diagonal matrix is of the form where all the diagonal elements are equal (*d*), all the upper diagonal elements are equal (*u*) and all the lower diagonal elements are also equal (*l*). Let us denote this matrix as **B**[*l, d, u*]. The eigenvalues of such a matrix can be computed in the closed form as (Yueh 2005, Moin 2001, Gregory and Karney 1969). For such a square matrix of size *n*, the eigenvalues are given by:

$$l_k = d + 2\sqrt{lu}\cos\frac{kp}{n+1} \quad k = 1,\ 2,\ \cdots\ n \tag{7.127}$$

For our problem, if we divide the spatial domain (0, *L*) in *m* equal segments such that $m\Delta x = L$, there are $m+1$ nodes including two boundary nodes where the values of f are known from the 1st Type boundary conditions. Therefore, the matrix **A** is of size (m-1). Using the values of *l, d* and *u*, we obtain the eigenvalues (l_k's) of matrix **A** as:

$$l_k = \frac{a}{\Delta x^2}\left(-2 + 2\cos\frac{kp}{m}\right) \quad k = 1,\ 2,\ \cdots\ m-1 \tag{7.128}$$

First things to notice that all the eigenvalues are negative and real since $\cos\frac{kp}{m} \le 1$ and $a > 0$. Therefore, any time stepping scheme that includes at least some part of the negative real axis in its stability region can be applied to solve this system as long as values chosen are within the stability limits. The largest and smallest absolute values of l_k would define the two limits of stability as well as the stiffness of the system. Theoretically, it is easy to see from eq (7.128) that the maximum and minimum absolute values of l_k would be achieved when $\cos\frac{kp}{m}$ approach -1 $\left(i.e.,\ \frac{kp}{m} = p\right)$ and 1 $\left(i.e.,\ \frac{kp}{m} = 0\right)$, respectively. For large *m* or small Δx, $\frac{m-1}{m} \approx 1$ and the eigenvalue with largest absolute magnitude is given by,

$$l_{m-1} \approx -\frac{4a}{\Delta x^2} \tag{7.129}$$

The other limit of zero can never be achieved but the closest to zero is given by $k = 1$. Therefore, the eigenvalue of smallest absolute magnitude is:

$$l_1 = \frac{2a}{\Delta x^2}\left(\cos\frac{p}{m} - 1\right) \tag{7.130}$$

Therefore, the ratio of maximum to minimum eigenvalue is given by:

$$\left|\frac{l_{m-1}}{l_1}\right| \approx \left|\frac{2}{\cos\frac{p}{m} - 1}\right| \tag{7.131}$$

If we approximate the quantity in the denominator using first two terms of the series representation of the cosine, the ratio is given by:

$$\left|\frac{l_{m-1}}{l_1}\right| \approx \frac{4m^2}{p^2} \tag{7.132}$$

It is now clear that larger the m is (smaller Δx), stiffer the system becomes. The stability criterion of the Euler Forward method to this system is given by eq (6.129) as:

$$\Delta t \le \frac{2}{|l_{m-1}|} \quad \text{or} \quad a\frac{\Delta t}{\Delta x^2} \le \frac{1}{2} \tag{7.133}$$

We leave it to the readers to compute this limit for other schemes such as AB, AM, RK, etc. For example, the limit for 2nd order RK-Central Difference scheme applied to the diffusion equation would be the same as (7.133) and that of 4th order RK-Central Difference scheme:

$$\Delta t \le \frac{2.785}{|l_{m-1}|} \tag{7.134}$$

Let us now apply similar analysis to the 1st order wave equation. Application of Central Difference scheme for the spatial derivative yields:

$$\frac{df_j}{dt} = -u\frac{f_{j+1} - f_{j-1}}{2\Delta x} \tag{7.135}$$

Once again, use of 1st Type zero boundary conditions lead to a system of equation similar to (7.121) with the coefficient matrix now defined as follows:

$$\mathbf{A} = -\frac{u}{2\Delta x}\mathbf{B}[-1,\ 0,\ 1] \tag{7.136}$$

Using (7.127), the eigenvalues of matrix **A** can be computed as:

$$l_k = -i\frac{u}{\Delta x}\cos\frac{kp}{m} \qquad k = 1,\ 2,\ \cdots\ m-1 \tag{7.137}$$

First thing to notice that all the eigenvalues are purely imaginary which essentially means that the coefficient of the model equation (7.125) are purely imaginary. Therefore, any time stepping scheme which does not include a part of the imaginary axis in its stability region will not be stable. Euler backward and Trapezoidal scheme can be applied. The 4th order Runge Kutta method is applicable because it includes some part of the imaginary axis. The eigenvalue of maximum absolute magnitude for large *m* is given by,

$$l_{m-1} \approx i\frac{u}{\Delta x} \tag{7.138}$$

The stability region of 4th order Runge-Kutta method (Figure 6.14) crosses the ordinate approximately at ± 2.83. Therefore, the stability limit is:

$$\left| l_{n-1}\Delta t \right| \le 2.83 \text{ or } C \le 2.83 \tag{7.139}$$

We leave to the readers to explore the stability of the upwind schemes using matrix analysis (Exercise 7.3).

In this section, we have seen that it is possible to analyze the methods applied to PDEs for stability using the procedures described in Chapter 6 for the system of IVPs. However, we had made some assumptions regarding the boundary conditions during the semi-discretization in spatial variable. If we relax this assumption and include the 2nd Type boundary condition, the tri-diagonal matrix may not have the properties such as equality of all diagonal and off-diagonal terms (section 7.2). If only first and last diagonal terms are different, a closed form solution for the eigenvalues can still be obtained easily (Yueh 2005). However, if some of the off-diagonal terms are also unequal (which may occur due to forward or backward difference approximation of the derivative boundary conditions), obtaining a closed form solution for the eigenvalue may be very difficult and it may be easier to compute the maximum and minimum eigenvalues using Power and Inverse Power methods, respectively. In the next section, we present a technique to analyze the numerical methods in full discretized form applied to PDEs for stability.

7.3.1 von Neumann Stability Analysis

The von Neumann stability analysis can be applied on the fully discrete form of any numerical method applied to a linear PDE. The basic analysis techniques rely on two basic assumptions:

- Separation of variable is possible, *i.e.*, the solution can be expressed as products of functions of single independent variables.
- The functions of space variables in the separated variables form can be represented in terms of orthogonal basis e^{ikx}, *i.e.*, Fourier series representation.

On the basis of these assumptions, one may write, the solution of the diffusion, advection-diffusion or wave equations $f(x, t)$ as:

$$f(x,t) = \sum_{k=-\infty}^{\infty} \Gamma_k(t) e^{ikx} \tag{7.140}$$

where, k is the wave number.

In order to see the Fourier series, let us take the example of 1-D wave equation (7.87). Putting the representation of (7.140) in (7.87), we obtain:

$$\sum_{k=-\infty}^{\infty} \left(\frac{d\Gamma_k(t)}{dt} + iku\Gamma_k(t) \right) e^{ikx} = 0 \tag{7.141}$$

We know from the Chapter 4 that the functions e^{ikx} are orthogonal. As a result, for the above to be true, coefficient for each k must go to zero. This essentially means:

$$\frac{d\Gamma_k(t)}{dt} + iku\Gamma_k(t) = 0 \quad \text{or} \quad \Gamma_k(t) = a_k e^{-ikut} \tag{7.142}$$

Using this in (7.140), we obtain the Fourier series solution of the 1-D advection equation as,

$$f(x,t) = \sum_{k=-\infty}^{\infty} a_k e^{ik(x-ut)} \tag{7.143}$$

where, a_k are the coefficients of the Fourier series which can be computed using the initial condition. Thus, the expression in (7.140) is indeed a solution of the PDE. Similar analysis can also be done with the other equations presented in this chapter. In fact, this is valid for all linear PDEs. This means that the $\Gamma_k(t_n)$ in (7.140) is the amplitude of the unit waves in the Fourier series. Therefore, one can define an amplification factor for the analytical problem as:

$$s_k^A = \frac{\Gamma_k(t_{n+1})}{\Gamma_k(t_n)} \tag{7.144}$$

The solution will remain bounded if and only if the amplitude is attenuated for each wave number k. In that case, their linear combination will also remain bounded. This means, that the analytical problem is stable if $s_k^A \leq 1$. In this section, we will only consider the problems which have stable analytical solution.

For a numerical method, the value of f at the j^{th} node at n^{th} time step can be expressed using (7.140) as:

$$f_j^n = \sum_{k=-\infty}^{\infty} \Gamma_k(t_n) e^{ikx_j} \tag{7.145}$$

We will represent each term in a numerical approximation using the convention of (7.145). Let us illustrate von Neumann analysis through application of *m-CD* scheme to the advection diffusion equation. The fully discrete form is shown in eq (7.47).

Applying the Fourier representation of eq (7.145), to the fully discrete form for node j with constant u and a:

$$\sum_{k=-\infty}^{\infty}\left\{\begin{array}{l}\dfrac{\Gamma_k(t_{n+1})e^{ikx_j}-\Gamma_k(t_n)e^{ikx_j}}{\Delta t}+mu\dfrac{\Gamma_k(t_n)e^{ikx_{j+1}}-\Gamma_k(t_n)e^{ikx_{j-1}}}{2\Delta x}\\ -ma\dfrac{\Gamma_k(t_n)e^{ikx_{j+1}}-2\Gamma_k(t_n)e^{ikx_j}+\Gamma_k(t_n)e^{ikx_{j-1}}}{\Delta x^2}+(1-m)a\dfrac{\Gamma_k(t_{n+1})e^{ikx_{j+1}}-\Gamma_k(t_{n+1})e^{ikx_{j-1}}}{2\Delta x}\\ -(1-m)a\dfrac{\Gamma_k(t_{n+1})e^{ikx_{j+1}}-2\Gamma_k(t_{n+1})e^{ikx_j}+\Gamma_k(t_{n+1})e^{ikx_{j-1}}}{\Delta x^2}\end{array}\right\}=0 \tag{7.146}$$

By definition of uniform spatial grid length Δx, we can write:

$$x_{j+1}-x_j=x_j-x_{j-1}=\Delta x \tag{7.147}$$

Using this relation in (7.146), we obtain:

$$\sum_{k=-\infty}^{\infty}\left\{\begin{array}{l}\dfrac{\Gamma_k(t_{n+1})-\Gamma_k(t_n)}{\Delta t}+mu\dfrac{\Gamma_k(t_n)e^{ik\Delta x}-\Gamma_k(t_n)e^{-ik\Delta x}}{2\Delta x}\\ -ma\dfrac{\Gamma_k(t_n)e^{ik\Delta x}-2\Gamma_k(t_n)+\Gamma_k(t_n)e^{-ik\Delta x}}{\Delta x^2}+(1-m)a\dfrac{\Gamma_k(t_{n+1})e^{ik\Delta x}-\Gamma_k(t_{n+1})e^{-ik\Delta x}}{2\Delta x}\\ -(1-m)a\dfrac{\Gamma_k(t_{n+1})e^{ik\Delta x}-2\Gamma_k(t_{n+1})+\Gamma_k(t_{n+1})e^{-ik\Delta x}}{\Delta x^2}\end{array}\right\}e^{ikx_j}=0 \tag{7.148}$$

Since, e^{ikx} are orthogonal, the coefficient must go to zero for each k. Therefore, we can write:

$$\begin{array}{l}\dfrac{\Gamma_k(t_{n+1})-\Gamma_k(t_n)}{\Delta t}+mu\dfrac{\Gamma_k(t_n)e^{ik\Delta x}-\Gamma_k(t_n)e^{-ik\Delta x}}{2\Delta x}-ma\dfrac{\Gamma_k(t_n)e^{ik\Delta x}-2\Gamma_k(t_n)+\Gamma_k(t_n)e^{-ik\Delta x}}{\Delta x^2}\\ +(1-m)a\dfrac{\Gamma_k(t_{n+1})e^{ik\Delta x}-\Gamma_k(t_{n+1})e^{-ik\Delta x}}{2\Delta x}-(1-m)a\dfrac{\Gamma_k(t_{n+1})e^{ik\Delta x}-2\Gamma_k(t_{n+1})+\Gamma_k(t_{n+1})e^{-ik\Delta x}}{\Delta x^2}=0\end{array} \tag{7.149}$$

For the numerical solution to be stable, the amplitude of the unit waves for each wave number must attenuate with the progression of time. This will ensure that their linear combination also remain bounded. Therefore, we can define an amplification factor similar to the analytical problem as:

$$s_k=\frac{\Gamma_k(t_{n+1})}{\Gamma_k(t_n)} \tag{7.150}$$

Then for stability, one must satisfy $|s_k|\le 1$ for all k. The overall stability condition for the method will then be governed by wave number for which the absolute value of the amplification factor is maximum.

Dividing (7.149) by $\Gamma_k(t_n)$ and using the definition of (7.150), we obtain:

$$\frac{s_k - 1}{\Delta t} = -mu\frac{e^{ik\Delta x} - e^{-ik\Delta x}}{2\Delta x} + ma\frac{e^{ik\Delta x} - 2 + e^{-ik\Delta x}}{\Delta x^2}$$
$$-(1-m)u\frac{s_k e^{ik\Delta x} - s_k e^{-ik\Delta x}}{2\Delta x} + (1-m)a\frac{s_k e^{ik\Delta x} - 2s_k + s_k e^{-ik\Delta x}}{\Delta x^2} \quad (7.151)$$

Using Euler's formula, one can write the following identities:

$$\frac{e^{ik\Delta x} - e^{-ik\Delta x}}{2} = i\sin(k\Delta x) \text{ and } \frac{e^{ik\Delta x} + e^{-ik\Delta x}}{2} = \cos(k\Delta x) \quad (7.152)$$

Using these identities (7.152) and definition of C and P_g (equations 7.48 and 7.49) in (7.151), we obtain the following expression for the amplification factor:

$$s_k = \frac{1 + 2m\frac{C}{P_g}(\cos(k\Delta x) - 1) - imC\sin(k\Delta x)}{1 - 2(1-m)\frac{C}{P_g}(\cos(k\Delta x) - 1) + i(1-m)C\sin(k\Delta x)} \quad (7.153)$$

Let us define $x_k = 2\frac{C}{P_g}(\cos(k\Delta x) - 1) - iC\sin(k\Delta x)$ (7.154)

Therefore, the amplification factor s is,

$$s_k = \frac{1 + mx_k}{1 - (1-m)x_k} \quad (7.155)$$

Let us verify if this indeed is the stability condition with the known case of equation (7.133) for application of *EF-CD* scheme to purely diffusion problem. Set, $u = 0$ for purely diffusion problem and $m = 1$ for Euler Forward scheme. For this case,

$$s_k = 1 + x_k \text{ and } x_k = 2a\frac{\Delta t}{\Delta x^2}(\cos(k\Delta x) - 1) \quad (7.156)$$

Notice the similarity of x_k in (7.156) with l_k (eq 7.128) and that of s_k with the amplification factor of Euler Forward scheme for a system of ODE (6.128). Using the stability condition $|s_k| \le 1$, we obtain,

$$a\frac{\Delta t}{\Delta x^2} \le \frac{1}{1 - \cos(k\Delta x)} \quad (7.157)$$

The most critical condition arises for the wave number k when $\cos(k\Delta x) = -1$. Therefore, the condition is,

$$a\frac{\Delta t}{\Delta x^2} \le \frac{1}{2} \quad (7.158)$$

For Euler Backward method applied to the diffusion equation with central difference scheme, the x_k remains the same but $m = 0$ gives the following for the amplification factor:

$$s_k = \frac{1}{1 - x_k} \tag{7.159}$$

Since $x_k \le 0$ for all *k*, the denominator is always less than or equal to unity. So, the scheme is unconditionally stable. We leave the readers to derive the conditions for the advection diffusion equation (extercise 7.5).

In order to obtain a general condition for *m-CD* scheme, let us denote the real and imaginary parts of *x* as:

$$x_k = x_{Rk} + i x_{Ik} \tag{7.160}$$

Using this in (7.155) and applying the stability condition, one obtains the following (after some algebraic operations):

$$\left(x_{Rk} + \frac{1}{2m-1} \right)^2 + \left(x_{Ik} \right)^2 \le \left(\frac{1}{2m-1} \right)^2 \tag{7.161}$$

Let us now see an example application of the von-Neumann analysis.

Example 7.6: Analyze the upwind scheme applied to the 1st order wave equation (eq 7.98) for stability using von Neumann analysis.

Solution: Using the representation of (7.145) in (7.98) we obtain:

$$\Gamma_k(t_{n+1}) e^{ikx_j} = \Gamma_k(t_n) e^{ikx_j} - C\left[\Gamma_k(t_n) e^{ikx_j} - \Gamma_k(t_n) e^{ikx_{j-1}} \right]$$

The amplification factor is given by,

$$s_k = 1 + C\{\cos(k\Delta x) - 1\} - iC\sin(k\Delta x)$$

Using the stability condition leads to,

$$\left[1 + C\{\cos(k\Delta x) - 1\} \right]^2 + \left[C\sin(k\Delta x) \right]^2 \le 1$$

After some algebraic operations, one arrives at the condition of stability as $C \le 1$.

Reader, may have noticed by now that the application of von Neumann analysis can be made on fully discrete form of the numerical schemes in this chapter using the principle and example described in this section. We leave the readers to analyze other schemes as exercise.

Exercise 7.3

1. The following numerical method has been proposed to solve $\frac{\partial u}{\partial t} = c\frac{\partial u}{\partial x}$:

$$\frac{1}{\Delta t}\left[u_j^{n+1} - \frac{1}{2}\left(u_{j+1}^n + u_{j-1}^n\right)\right] = \frac{c}{2\Delta x}\left[u_{j+1}^n - u_{j-1}^n\right]$$

a) Find the range of CFL number for which the method is stable using Von Neuman Analysis.

b) Is the method consistent (i.e., does it reduce to the original PDE as $\Delta x, \Delta t \to 0$)?

2. Consider the Lax Wendorf scheme (equation 7.93) for the solution of 1-D convection-diffusion equation What is the order of accuracy of this scheme? Is the scheme consistent? Obtain the stability criteria for this scheme.

4. *Du Fort-Frankel* scheme for the diffusion equation $\frac{\partial T}{\partial t} = a\frac{\partial^2 T}{\partial x^2}; a > 0$ is given by,

$$\frac{T_j^{n+1} - T_j^{n-1}}{2\Delta t} = a\left[\frac{T_{j+1}^n - T_j^{n+1} - T_j^{n-1} + T_{j-1}^n}{\Delta x^2}\right]$$

(a) Obtain three leading order terms of the truncation error ?

(b) In the limit ($\Delta t, \Delta x$? 0), what is the modified form of the original equation one obtains from the discretized equation? Based on this, comment on the consistency of the scheme.

(c) Obtain the stability criteria for this scheme.

7.4 Summary

In this chapter, we presented numerical methods for solution of PDEs. The emphasis was on utilization of the concepts developed in the earlier chapters on finite difference approximation of derivatives (Chapter 5) and solution of initial and boundary value problems (Chapter 6). Very few new concepts were introduced. We have observed how an initial value problem and a boundary value problem combine to form a PDE. At the same time, we have observed how the time and space derivatives can interact and their residuals can cancel each other to increase the order of the method. The von-Neumann stability analysis introduced in this chapter is a powerful tool and finds many applications in the literature. While we have introduced the basic concepts for the finite difference approach of solution of PDE, the literature is too rich to do full justice within the scope of an introductory numerical analysis book. We hope that the readers have gained enough encouragement from this book to explore further for the more advanced methods for the solution of PDEs.