1 Finite-Difference Method for the 1D Heat Equation

Consider the one-dimensional heat equation,

$$\begin{array}{rcll} u_t &=& \alpha^2 u_{xx} & 0 < x < L,\ 0 < t < \infty \\ u(0,t) &=& 0 & 0 < t < \infty \\ u(1,t) &=& 0 & 0 < t < \infty \\ u(x,0) &=& \phi(x) & 0 \le x \le L \end{array} \tag{1}$$

We will employ the finite-difference technique to obtain the numerical solution to (1). In this technique, the approximations require that the model domain (space) and time be discretized. The space domain is represented by a network of grid cells or elements and the time of the simulation is represented by time steps. The accuracy of the numerical method will depend upon the accuracy of the model input data, the size of the space and time discretization, and the scheme used to solve the model equations.

1.1 Domain Discretization

We first partition the intervals $[0, L]$ and $[0, T]$ into respective finite grids as follows. We let,

$$x_i = i\Delta x, \quad i = 0, 1, \ldots, M \quad \text{where} \quad \Delta x = \frac{L}{M}$$

Similarly we partition $[0, T]$ as,

$$t_n = n\Delta t, \quad n = 0, 1, 2, \ldots, N \quad \text{where} \quad \Delta t = \frac{T}{N}$$

1.2 Finite-Difference FTCS Discretization

We consider the Forward in Time Central in Space Scheme (FTCS) where we replace the time derivative in (1) by the forward differencing scheme and the space derivative in (1) by the central differencing scheme. This yields,

$$\frac{u_{i,n+1} - u_{i,n}}{\Delta t} - \alpha^2 \frac{u_{i+1,n} - 2u_{i,n} + u_{i-1,n}}{(\Delta x)^2} \approx 0$$

where $u_{i,n} \approx u(x_i, t_n)$. This can be simplified to yield,

$$u_{i,n+1} = (1 - 2\lambda)u_{i,n} + \lambda(u_{i+1,n} + u_{i-1,n}) \tag{2}$$

where the parameter

$$\lambda = \frac{\alpha^2 \Delta t}{(\Delta x)^2}$$

We would impose (2) only on the interior nodes, namely for $i = 1, 2, \ldots, M - 1$ and for $n = 1, 2, \ldots, N$. Since the initial condition is given in (1), numerically we have,

$$u_{i,0} = \phi(x_i), \quad i = 0, 1, 2, \ldots, M$$

Moreover, the boundary conditions in (1) become,

$$u_{0,j} = u_{M,j} = 0, \quad j = 1, 2, \ldots, N$$

The *explicit* nature of the difference method can then be reexpressed in matrix form as,

$$\begin{bmatrix} u_{1,n+1} \\ u_{2,n+1} \\ \vdots \\ u_{M-1,n+1} \end{bmatrix} = \begin{bmatrix} a & b & 0 & \ldots & 0 \\ b & a & b & \ldots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & \ldots & b & a \end{bmatrix} \begin{bmatrix} u_{1,n} \\ u_{2,n} \\ \vdots \\ u_{M-1,n} \end{bmatrix}$$

where $a = 1 - 2\lambda, b = \lambda$ and $n = 0, 1, \ldots, N$.

It must be pointed out that eventhough the implementation of the FTCS method looks simple and attractive, one has to take caution when making appropriate choices of Δx and Δt. This is illustrated in the following example.

Consider the initial data:

$$\phi(x) = \epsilon \cos\left(\frac{\pi x}{\Delta x}\right) \tag{3}$$

Then it is clear that this data oscillates exactly with spatial frequency of the grid because,

$$\phi(x_i) = \phi(i\Delta x) = \epsilon(-1)^i$$

One can show that the exact solution to the heat equation (1) for this initial data satisfies,

$$|u(x,t)| \leq \epsilon$$

for all x and t. So, it is reasonable to expect the numerical solution to behave similarly. Unfortunately, this is not true if one employs the FTCS scheme (2). For this scheme, with the initial data given in (3), it is possible to prove that the numerical solution satisfies,

$$u_{i,n} = \epsilon(-1)^i \, (1 - 4\lambda)^n \tag{4}$$

Therefore, inorder for the computed discrete approximation to be bounded as $n \to \infty$, we require that,

$$|1 - 4\lambda| < 1$$

which yields the restriction,

$$\Delta t \leq \frac{(\Delta x)^2}{2\alpha}$$

This for a choice of $\Delta x = 0.1$, the FTCS scheme (for $\alpha = 1$, say) will become unstable if the initial data oscillates on the order of Δx, unless, we choose Δt to be,

$$\Delta t \leq 0.005$$

This is a very small step size and so to make the FTCS scheme stable is not always practical.

1.3 Stability of the FTCS Scheme

Let us suppose that the solution to the difference equations is of the form,

$$u_{j,n} = e^{ij\beta\Delta x}\ e^{n\lambda\Delta t} \tag{5}$$

where $j = \sqrt{-1}$. Now we examine the behaviour of this solution as $t \to \infty$ or $n \to \infty$ for a suitable choice of λ.

Note that if $|e^{n\lambda\Delta t}| > 1$, then this solutoin becomes unbounded. Hence we want to study solutions with,

$$|e^{n\lambda\Delta t}| \leq 1$$

Consider the difference equation (2). Substituting (5) in (2) and rearranging terms yields,

$$e^{\lambda\Delta t} = 1 - 4\lambda\sin^2\left(\frac{\beta\Delta x}{2}\right)$$

Obviously $e^{\lambda\Delta t} \leq 1$. Hence we look for solutions that satisfy,

$$e^{n\lambda\Delta t} \geq -1$$

Since $\sin^2\left(\frac{\beta\Delta x}{2}\right)$ can be close to 1, one can then show,

$$\lambda \leq \frac{1}{2} \tag{6}$$

1.4 The BTCS Discretization

Instead of the FTCS, one could have alternatively considered the following Backward in Time Central in Space (BTCS) differencing scheme for (1):

$$\frac{u_{i,n+1} - u_{i,n}}{\Delta t} - \alpha^2\frac{u_{i+1,n+1} - 2u_{i,n+1} + u_{i-1,n+1}}{(\Delta x)^2} \approx 0$$

where $u_{i,n} \approx u(x_i, t_n)$. This can be simplified to yield,

$$u_{i,n} = (1 + 2\lambda)u_{i,n+1} - \lambda(u_{i+1,n+1} + u_{i-1,n+1}) \tag{7}$$

for $i = 1, 2, \ldots, M-1$ and for $n = 1, 2, \ldots, N$. Note that unlike the FTCS method, we are required to solve a system each time. The *implicit* nature of the difference method can then be reexpressed in matrix form as,

$$\begin{bmatrix} u_{1,n} \\ u_{2,n} \\ \vdots \\ u_{M-1,n} \end{bmatrix} = \begin{bmatrix} a & b & 0 & \ldots & 0 \\ b & a & b & \ldots & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & \ldots & b & a \end{bmatrix} \begin{bmatrix} u_{1,n+1} \\ u_{2,n+1} \\ \vdots \\ u_{M-1,n+1} \end{bmatrix}$$

where $a = 1 + 2\lambda, b = -\lambda$ and $n = 0, 1, \ldots, N$.

1.5 Stability of the BTCS scheme

Note that letting the solution to be of the form (5), then (7) simplifies to,

$$e^{\lambda \Delta t} = \frac{1}{1 + 4\lambda \sin^2\left(\frac{\beta \Delta x}{2}\right)}$$

This implies that any numerical solution obtained via the BTCS scheme is stable. Hence for any value of λ, the BTCS is *unconditionally stable*.

1.6 The weighted average or theta-method

The FTCS and BTCS schemes indicate that one can generate a whole range of schemes based on the following discretization:

$$u_{i,n+1} - u_{i,n} = \lambda[(1-\theta)(u_{i+1,n} - 2u_{i,n} + u_{i-1,n}) + \theta(u_{i+1,n+1} - 2u_{i,n+1} + u_{i-1,n+1})] \quad (8)$$

Note that for $\theta = 0$ and $\theta = 1$, (8) yields the Explicit FTCS and Implicit BTCS respectively. A more popular scheme for implementation is when $\theta = 0.5$ which yields the *Crank-Nicolson scheme* which is also unconditionally stable.

2 Homework

1. Analyze the numerical stability of the weighted average or theta-method. In particular, show that

   (a) If $0 \leq \theta < 0.5$, then the method is stable if and only if $\lambda \leq 0.5$.

   (b) If $0.5 \leq \theta \leq 1$, then the method is unconditionally stable, i.e., stable for all λ (or all Δt and Δx.)

2. Consider the problem,

$$
\begin{aligned}
u_t &= u_{xx} \quad 0 < x < 1, \quad 0 < t < \infty \\
u(0,t) &= 0 \qquad 0 < t < \infty \\
u(1,t) &= 0 \qquad 0 < t < \infty \\
u(x,0) &= \sin(\pi x) \quad 0 \leq x \leq 1
\end{aligned}
$$

   (a) Determine the analytical solution to the problem.

   (b) Write a MATLAB Program to implement the problem via "Explicit Forward in Time Central in Space (FTCS)" finite difference algorithm. Let Δx and Δt be the stepsizes in space and time respectively (i.e. the grid points are defined according to $x_j = j\Delta x,\ j = 0, 1, \cdots, M$ and $t_n = n\Delta t,\ n = 0, 1, \ldots, N$) with final time $T = N\Delta t = 0.5$. Run your program and compare the finite difference solution $u_{j,n}$ with the exact solution $u(x_j, t_n)$ from part(a) for $\Delta x = 0.1,\ \Delta t = 0.01$ and $\Delta x = 0.1,\ \Delta t = 0.0005$. If $E^n := \max\{|u_{j,n} - u(x_j, t_n)|,\ j = 0, 1, \cdots, M\}$. Plot the $\log(E^n)$ with respect to t_n for $n = 0, 1, \cdots, N$.

   (c) Repeat part(b) for an "Implicit Backward in Time Central in Space (BTCS)" finite difference algorithm.

   (d) Repeat part(b) for an "Implicit Crank-Nicholson" finite difference algorithm.

3. Consider the one-dimensional viscous Burger's equation for a given velocity u and viscocity coefficient ν.

$$
\begin{aligned}
u_t + u\ u_x &= \nu u_{xx} \qquad x \in R,\ t > 0 \\
u(x,0) &= \phi(x) \qquad x \in R
\end{aligned}
\tag{9}
$$

   (a) Consider the transformation

$$
u(x,t) = -2\nu \frac{v_x(x,t)}{v(x,t)} \tag{10}
$$

   for a function $v(x,t)$. Using the transformation (10), show that if $v(x,t)$ satisfies the heat equation ($v_t = \nu v_{xx}$) then $u(x,t)$ is a solution to the PDE in (9).

   (b) Also, use the transformation (10) to prove that for some constant C,

$$
v(x,0) = Ce^{-\frac{1}{2\nu}\int_0^x \phi(s)\ ds}
$$