AMS 209, Fall 2016
Final Project Type B – Numerical PDE: Linear Advection and Diffusion Equation

1. Overview

We are interested in solving a linear advection-diffusion PDE given as

$$u_t + a u_x = \kappa u_{xx}, \tag{1}$$

where a is a constant advection velocity and $\kappa \geq 0$ is a constant diffusion coefficient. Note if $\kappa < 0$ then Eq. (1) would be a "backward heat equation", which is an ill-posed problem.

2. Initial and boundary conditions

We impose an initial condition at $t = 0$,

$$u(x, 0) = u_0(x) \tag{2}$$

and a boundary condition on a bounded domain $x_a \leq x \leq x_b$

$$u(x_a, t) = g_a(t) \text{ and } u(x_b, t) = g_b(t), \text{ for } t > 0. \tag{3}$$

3. Discretization in space and time

Let us take the discretization technique with which we have a spatial resolution of N and a temporal resolution of M:

$$x_i = x_a + (i - \frac{1}{2})\Delta x, \quad i = 1, ..., N, \tag{4}$$

$$t^n = n\Delta t, \quad n = 0, ...M. \tag{5}$$

Notice that the cell interface-centered grid points are written using the 'half-integer' indices:

$$x_{i+\frac{1}{2}} = x_i + \frac{\Delta x}{2}. \tag{6}$$

4. Imposing Boundary Conditions via Guard-cell (or ghost-cell)

We can introduce the so-called 'guard-cell' or 'ghost-cell' (simply GC) on each end, having extra two GC points,

$$x_0 = x_a - \Delta x/2 \tag{7}$$

$$x_{N+1} = x_b + \Delta x/2. \tag{8}$$

With these two extra GC points (one GC on each end) over the spatial domain the difference equation are evolved only over the interior points, whereas the boundary conditions are explicitly imposed at the two GC points,

$$u_0^n = g_a(t^n), \quad u_{N+1}^n = g_b(t^n). \tag{9}$$

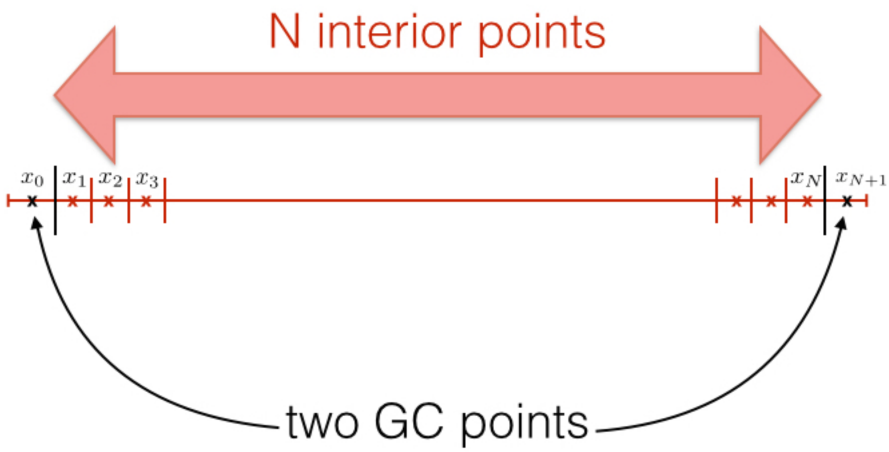

Figure 1.

5. Finite discretizations of advection and diffusion equations

5.1. Finite difference scheme for 1D advection

First consider a simple advection equation with constant speed $a > 0$:

$$u_t + au_x = 0, \text{ with } u(x, 0) = u_0(x). \tag{10}$$

Let us denote our discrete data at each (x_i, t^n) as:

$$u_i^n = u(x_i, t^n) \tag{11}$$

The forward difference approximation scheme for first-order spatial and temporal derivatives writes, respectively:

$$u_x(x,t) = \frac{u(x+\Delta x, t) - u(x,t)}{\Delta x} + O(\Delta x), \tag{12}$$

$$u_t(x,t) = \frac{u(x, t+\Delta t) - u(x,t)}{\Delta t} + O(\Delta t). \tag{13}$$

Dropping the truncation error terms $O(\Delta x)$ and $O(\Delta t)$ yields a simple first-order difference scheme that approximates the advection PDE. As a result, we arrive at a first-order accurate discrete difference equation from an analytic differential equation:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + a\frac{u_{i+1}^n - u_i^n}{\Delta x} = 0, \tag{14}$$

which gives a temporal update scheme of u_i^{n+1} in terms of the known data at $t = t^n$:

$$u_i^{n+1} = u_i^n - a\frac{\Delta t}{\Delta x}\Big(u_{i+1}^n - u_i^n\Big) \tag{15}$$

On the other hand, if we use a backward difference scheme for u_x

$$u_x(x,t) = \frac{u(x,t) - u(x-\Delta x,t)}{\Delta x} + O(\Delta x), \tag{16}$$

we arrive at another first-order difference equation

$$u_i^{n+1} = u_i^n - a\frac{\Delta t}{\Delta x}\Big(u_i^n - u_{i-1}^n\Big). \tag{17}$$

Another approximation is available using the centered differencing scheme,

$$u_i^{n+1} = u_i^n - a\frac{\Delta t}{2\Delta x}\Big(u_{i+1}^n - u_{i-1}^n\Big). \tag{18}$$

The choice of Δt should be small enough, satisfying:

$$|a|\Delta t \le \Delta x. \tag{19}$$

This is called the Courant–Friedrichs–Lewy (CFL) condition (or simply the Courant condition). The CFL condition describes a necessary (but *not* sufficient) condition for convergence when solving discrete PDEs using finite difference approximations (e.g., finite difference, finite volume methods).

5.2. Finite difference scheme for 1D diffusion

Consider now a temporal evolution of solving the classical homogeneous heat equation (or diffusion equation) of the form

$$u_t = \kappa u_{xx} \tag{20}$$

with $\kappa > 0$.

We use a similar but different discretization technique from the previous example of the 1D advection finite difference scheme in order to discretize Eq. (20). For a spatial discretization, we adopt the standard second-order central difference difference scheme,

$$u_{xx}(x,t) = \frac{u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)}{\Delta x^2} + O(\Delta x^2), \tag{21}$$

which gives a final discrete form of our explicit finite difference scheme for the heat equation:

$$u_i^{n+1} = u_i^n + \kappa\frac{\Delta t}{\Delta x^2}\Big(u_{i+1}^n - 2u_i^n + u_{i-1}^n\Big) \tag{22}$$

Similar to the 1D advection case, we choose Δt satisfying

$$\kappa\Delta t \le \frac{\Delta x^2}{2}. \tag{23}$$

Notice that Eq. (21) is nothing but is naturally obtained by applying the forward and backward difference schemes consecutively:

$$u_{xx}(x,t) = \frac{\partial}{\partial x}\Big(\frac{\partial u}{\partial x}\Big) \tag{24}$$
$$\approx \frac{\partial}{\partial x}\frac{u(x+\Delta x,t)-u(x,t)}{\Delta x} \tag{25}$$
$$\approx \frac{u(x+\Delta x,t)-2u(x,t)+u(x-\Delta x,t)}{\Delta x^2}. \tag{26}$$

5.3. The CFL condition

As mentioned, the CFL condition provides a necessary condition for choosing the length of Δt depending on the PDE under consideration. The CFL condition amounts to say, if we let C_a to be the CFL number that satisfy $0 < C_a \le 1$, C_a becomes, for the advection case,

$$C_a = \max_p |a_p|\frac{\Delta t}{\Delta x}, \tag{27}$$

and for the diffusion case,

$$C_a = \max_p \kappa_p\frac{2\Delta t}{\Delta x^2}, \tag{28}$$

where p is the number of all available wave speeds a_p or the diffusion coefficients κ_p, respectively. Note that $p = 1$ for a linear 'scalar' equation, which is the current case.

It is important to note that the CFL condition is only a *necessary* condition for stability (and hence convergence). It is not always *sufficient* to guarantee stability, and a numerical method satisfying the CFL condition can become unstable.

Note that the above CFL conditions in Eqs. (19) and (23) for choosing Δt_{advect} and Δt_{diff}, respectively, need to be combined together for a linear advection-diffusion equation:

$$\Delta t = C_a \min\Big(\frac{\Delta x}{|a|}, \frac{\Delta x^2}{2\kappa}\Big), \tag{29}$$

for $0 < C_a \le 1$.

6. A List of Finite Difference Methods for the Linear Problem

There are a couple of finite difference (FD) methods for solving the advection part of PDE, $u_t + au_x = 0$. We assume $a > 0$ for Beam-Warming and Fromm's methods. One can easily get appropriate forms for these two methods for $a < 0$.

- Upwind for $a > 0$ (FTBS – Forward Time Backward Space)

$$u_i^{n+1} = u_i^n - \frac{a\Delta t}{\Delta x}\Big(u_i^n - u_{i-1}^n\Big) \tag{30}$$

- Downwind for $a > 0$ (FTFS – Forward Time Forward Space)

$$u_i^{n+1} = u_i^n - \frac{a\Delta t}{\Delta x}\Big(u_{i+1}^n - u_i^n\Big) \tag{31}$$

- Centered for any a (FTCS – Forward Time Centered Space)

$$u_i^{n+1} = u_i^n - \frac{a\Delta t}{2\Delta x}\Big(u_{i+1}^n - u_{i-1}^n\Big) \tag{32}$$

- Leapfrog for any a

$$u_i^{n+1} = u_i^{n-1} - \frac{a\Delta t}{2\Delta x}\Big(u_{i+1}^n - u_{i-1}^n\Big) \tag{33}$$

- Lax-Friedrichs (LF) for any a

$$u_i^{n+1} = \frac{1}{2}\Big(u_{i+1}^n + u_{i-1}^n\Big) - \frac{a\Delta t}{2\Delta x}\Big(u_{i+1}^n - u_{i-1}^n\Big) \tag{34}$$

- Lax-Wendroff (LW) for any a

$$u_i^{n+1} = u_i^n - \frac{a\Delta t}{2\Delta x}\Big(u_{i+1}^n - u_{i-1}^n\Big) + \frac{1}{2}\Big(\frac{a\Delta t}{\Delta x}\Big)^2\Big(u_{i+1}^n - 2u_i^n + u_{i-1}^n\Big) \tag{35}$$

- Beam-Warming (BW) for $a > 0$

$$u_i^{n+1} = u_i^n - \frac{a\Delta t}{2\Delta x}\Big(3u_i^n - 4u_{i-1}^n + u_{i-2}^n\Big) + \frac{1}{2}\Big(\frac{a\Delta t}{\Delta x}\Big)^2\Big(u_i^n - 2u_{i-1}^n + u_{i-2}^n\Big) \tag{36}$$

- Fromm's method for $a > 0$

$$\begin{aligned} u_i^{n+1} = \quad & u_i^n - \frac{a\Delta t}{\Delta x}\Big(u_i^n - u_{i-1}^n\Big) - \frac{1}{4}\frac{a\Delta t}{\Delta x}\Big(1 - \frac{a\Delta t}{\Delta x}\Big)\Big(u_{i+1}^n - u_i^n\Big) \\ & + \frac{1}{4}\frac{a\Delta t}{\Delta x}\Big(1 - \frac{a\Delta t}{\Delta x}\Big)\Big(u_{i-1}^n - u_{i-2}^n\Big) \end{aligned} \tag{37}$$

Note: On the contrary, there is not so much to do with discretizing the diffusion part of PDE. This is because the physical process described by parabolic PDEs is diffusive and smooth, thereby it does not require those numerical attentions that are needed in resolving more complicated advective processes governed by tracing the wave information in advection hyperbolic PDEs.

7. Examples of advection: continuous and discontinuous

In Fig. 2 we display five different numerical solutions to two different types of initial conditions. The panels on the left column shows the smooth $\sin(2\pi x)$ wave initialized on $x \in [0, 1]$. The sine wave is solved numerically with – from top to

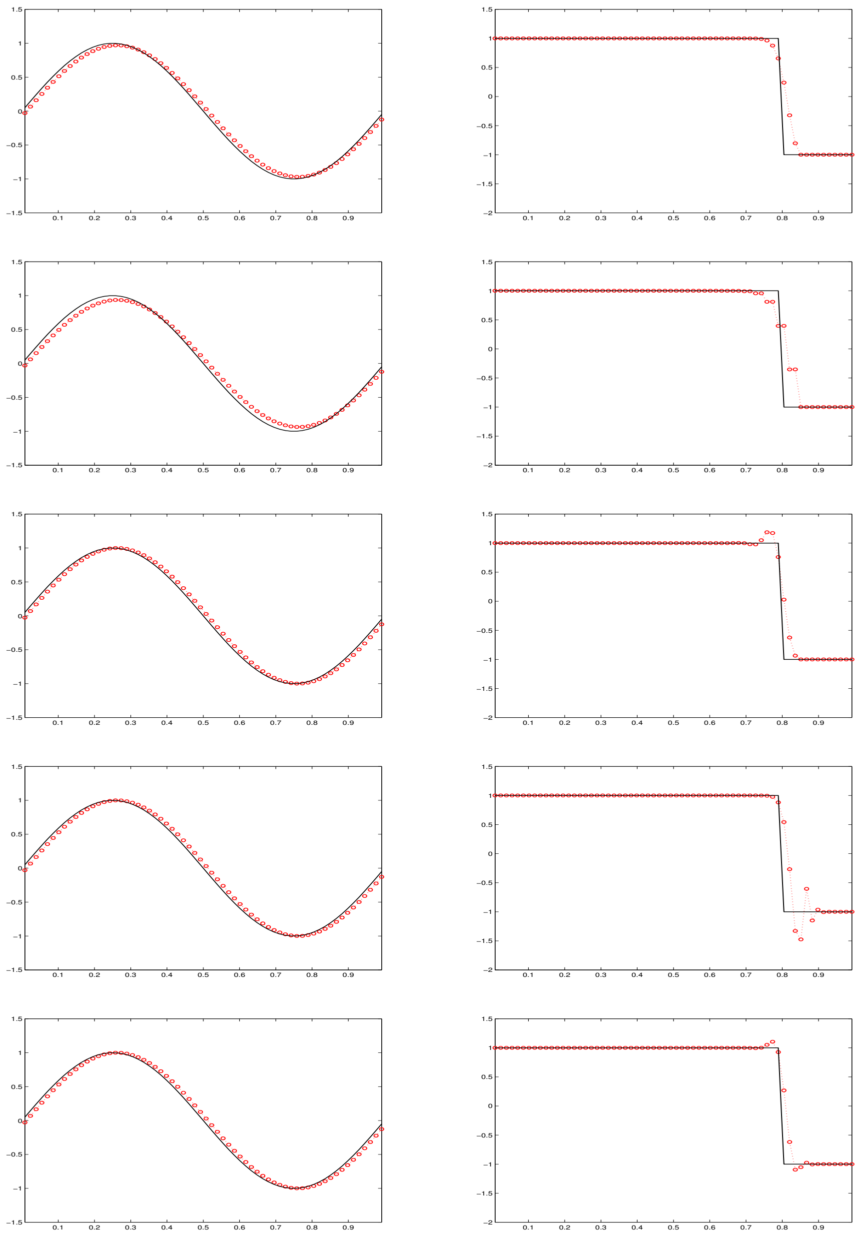

Figure 2. Numerical (red circles) and exact (black solid curves) solutions to the scalar advection equation $u_t + au_x = 0, a > 0$ with two different initial conditions: Left column: sinusoidal wave, Right column: discontinuous Riemann problem. Five different schemes are shown from top to bottom: (1) Upwind, (2) Lax-Friedrichs, (3) Lax-Wendroff, (4) Beam-Warming, (5) Fromm's method.

bottom – (1) Upwind method, (2) Lax-Friedrichs, (3) Lax-Wendroff, (4) Beam-

Warming, and (5) Fromm's method. On the right column, the same methods are applied – in the same order – to solve the initially discontinuous Riemann problem,

$$u_0(x) = \begin{cases} 1 \text{ for } x < 0.5 \\ -1 \text{ for } x > 0.5. \end{cases} \tag{38}$$

All numerical methods solve the sine wave until the wave completes the first cycle on a periodic domain which is resolved on 64 grid cells, $N = 64$. The same number of grid cells is used for the discontinuous case where the solutions have been integrated on a domain with outflow boundary condition until the location of the shock reaches to $x = 0.8$ which is 0.3 distance away from its initial location $x = 0.5$.

There are two first-order methods (upwind and Lax-Friedrichs) and three second-order methods (Lax-Wendroff, Beam-Warming, and Fromm's method). We note that all methods behave equally well on the smooth flow. On the contrary, there are two distinctive solution characteristics – dissipation and oscillations – on the discontinuous flow, particularly near the discontinuity: the first-order methods give very smeared solutions, while the second-order methods give oscillations. Understanding these types of behaviors is a key topic in modeling numerical methods in computational fluid dynamics.