AMS 213B Homework 5 – due Monday, May 16, 2016

Note: Please submit your MATLAB codes to your git repo.

Simple Notation: $C_a \equiv \frac{\Delta t}{\Delta x^2}$

Problem 1

Use the von Neumann stability analysis to show that the fully implicit difference method given by

$$\frac{1}{\Delta t}\Big[U_i^{n+1} - U_i^n\Big] = \frac{\kappa}{\Delta x^2}\Big[U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1}\Big] \tag{1}$$

approximating the heat equation

$$\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2}, \quad \kappa > 0, \tag{2}$$

is unconditionally stable (i.e., stable for all C_a). (Hint: $\cos 2\theta = 1 - 2\sin^2\theta$).

Problem 2

Repeat the von Neumann stability analysis for the explicit differencing scheme given by

$$\frac{1}{\Delta t}\Big[U_i^{n+1} - U_i^n\Big] = \frac{\kappa}{\Delta x^2}\Big[U_{i-1}^{n} - 2U_i^{n} + U_{i+1}^{n}\Big] \tag{3}$$

for Eq. (2) and find the stability condition (or bound) for C_a.

Problem 3

Show that the Crank-Nicolson method is of order $\mathcal{O}(\Delta t^2 + \Delta x^2)$ by explicitly identifying the leading term in the local truncation error.

Problem 4

Find the local truncation error of the difference method given as

$$\frac{\theta}{2\Delta t}\Big[U_i^{n+1} - U_i^{n-1}\Big] + \frac{1-\theta}{\Delta t}\Big[U_i^{n} - U_i^{n-1}\Big] = \frac{1}{\Delta x^2}\Big[U_{i-1}^{n} - 2U_i^{n} + U_{i+1}^{n}\Big] \tag{4}$$

for Eq. (2) with $\kappa = 1$, and find the value of θ that will make the scheme to be of $\mathcal{O}(\Delta t^2 + \Delta x^4)$ (i.e., second-order in time and fourth-order in space).

Problem 5

Please implement the 1D explicit MATLAB code based on FTCS to solve Eq. (2) on a 1D domain $[x_a, x_b] = [-1, 1]$.

0.1. Initial condition

The initial condition is given as Gaussian:

$$u(x, y, 0) = \exp\Big(-\frac{(x - x_{ctr})^2}{\sigma^2}\Big), \tag{5}$$

with $x_{ctr} = 0$ and $\sigma = 0.1$.

0.2. Discretizations

The spatial and temporal discretizations are configured as, see Fig. 1:

$$x_i = x_a + (i - \frac{1}{2})\Delta x, \quad i = 1, ..., N, \tag{6}$$

$$t^n = n\Delta t, \quad n = 0, ...M, \tag{7}$$

together with one guardcell (GC) point on each side of the domain,

$$x_0 = x_a - \Delta x/2 = x_1 - \Delta x, \tag{8}$$

$$x_{N+1} = x_b + \Delta x/2 = x_N + \Delta x. \tag{9}$$

0.3. Boundary condition

With these two extra GC points 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. We use the outflow condition through the GCs:

$$u(x_0, t) = u(x_1, t), \quad u(x_{N+1}, t) = u(x_N, t), \quad \text{for } t > 0. \tag{10}$$

0.4. CFL stability condition

The CFL condition provides a necessary condition for choosing the length of Δt for FTCS to solve Eq. (2). The CFL condition amounts to say, if we let C_{cfl} to denote the CFL number satisfying $0 < C_{cfl} \leq 1$, C_{cfl} becomes, for the diffusion case

$$C_{cfl} = \kappa\frac{2\Delta t}{\Delta x^2}. \tag{11}$$

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

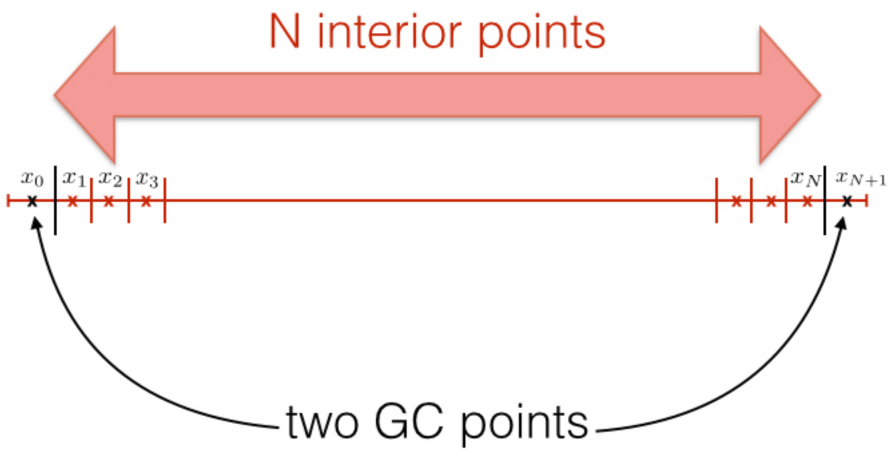

Figure 1. Two guardcells, x_0 and x_{N+1}, are the extra cells outside the computational domain, through which the boundary conditions are applied to reflect that the initial Gaussian temperature in the middle of the domain diffuses out through the end boundaries over time. The steady state solution with a flat temperature profile over the domain is to be reached later in time.

unstable. Note that the above CFL condition allows us to pick a proper time step size Δt,

$$\Delta t = C_{cfl} \frac{\Delta x^2}{2\kappa}, \tag{12}$$

for $0 < C_{cfl} \leq 1$.

(a) Please run your simulation until $t = t_{\max}$, where the final time step $t_{\max}$ is to be determined when the steady state is reached: i.e.,

$$||E^n||_1 = \Delta t \sum_{i=1}^{N} |U_i^n - U_i^{n-1}| \tag{13}$$

becomes smaller than ϵ, say, $||E||_1 < \epsilon$. Use $C_{cfl} = 0.9$ and $\kappa = 0.230$. Determine $t_{\max}$ when $\epsilon = 10^{-5}$ on the $N = 32$ grid resolution, and evolve your numerical solution up to $t_{\max}$. How many time steps does it take?

(b) Plot your numerical solutions at $t = ht_{\max}$, where $h = 0, 0.25, 0.5, 0.75$ and 1.

(c) Repeat (a) and (b) using $\kappa = 1.156$. What are the differences you observe in this case, in terms of $t_{\max}$ and the total number of steps it takes? What can you say about different values of κ in general?