AM213B Assignment #6

Problem 1 (Theoretical)

Consider the Lax-Friedrichs method for the general case ($a > 0$ or $a < 0$)

$$u_i^{n+1} = \frac{u_{i+1}^n + u_{i-1}^n}{2} - \frac{ar}{2}\left(u_{i+1}^n - u_{i-1}^n\right), \qquad r = \frac{\Delta t}{\Delta x}$$

Part 1: Carry out von Neumann stability analysis.

Part 2: Use Taylor expansion to find the local truncation error e_i^n.

Convert r back to $\Delta t/(\Delta x)$. Find coefficients of $(\Delta t)^2$, $(\Delta t)(\Delta x)$ and $(\Delta x)^2$ in e_i^n.

Part 3: Answer the two questions below.

For fixed $\frac{\Delta t}{\Delta x} = r$, write $\frac{e_i^n}{\Delta t}$ in terms of Δx only. Do we have $\lim_{\Delta x \to 0} \frac{e_i^n}{\Delta t} = 0$?

For fixed $\frac{\Delta t}{(\Delta x)^2} = c$, write $\frac{e_i^n}{\Delta t}$ in terms of Δx only. Do we have $\lim_{\Delta x \to 0} \frac{e_i^n}{\Delta t} = 0$?

Problem 2 (Theoretical)

Carry out von Neumann stability analysis on each of the methods below

i) the BTCS method for the general case ($a > 0$ or $a < 0$)

$$u_i^{n+1} = u_i^n - \frac{ar}{2}\left(u_{i+1}^{n+1} - u_{i-1}^{n+1}\right), \qquad r = \frac{\Delta t}{\Delta x}$$

ii) the implicit upwind method for the case of $a > 0$

$$u_i^{n+1} = u_i^n - ar\left(u_i^{n+1} - u_{i-1}^{n+1}\right), \qquad r = \frac{\Delta t}{\Delta x}$$

Hint: Examine $|1/\rho|^2$.

Problem 3 (Computational)

Use respectively, the first order and the second order numerical differentiations to approximate $\frac{d}{dx}\sin(x)$ at $x = 1.45$

$$q_1(h) = \frac{\sin(x+h) - \sin(x)}{h}$$

$$q_2(h)=\frac{\sin(x+h)-\sin(x-h)}{2h}$$

Use the exact derivative to calculate the total error of each method

$$E_{T,1}(h)=\left| q_1(h)-\cos(x)\right|$$
$$E_{T,2}(h)=\left| q_2(h)-\cos(x)\right|$$

Use loglog to plot $E_{T,1}(h)$ vs h and $E_{T,2}(h)$ vs h for h = 10.^(−[1:0.2:14]).

Plot the two curves in one figure for comparison.

Remark: In the figure, you should see

For each method, the total error attains a minimum at a certain value of h.

A higher order method achieves a smaller minimum total error.

Problem 4 (Computational)

Continue with the IBVP in problem 6 of Assignment 5.

$$\begin{cases} u_t=u_{xx}\,, & x\in(0,\ L)\,,\quad t>0 \\ u(x,0)=p(x)\,, & x\in(0,\ L) \\ u_x(0,t)-\alpha u(0,t)=0\,, & u(L,t)=q(t) \end{cases}$$

where $L = 2$, $\alpha = 0.4$, $p(x)=(1-0.5x)^2$, $q(t)=2\sin^2(t)$.

Use the FTCS method to solve the IBVP to $T = 3$ using $\Delta t = 10^{-5}$ and respectively

$N = 100$, $N = 200$, and $N = 400$

Let $u_{\{N=100\}}$ denote the numerical solution of $N = 100$.

$u_{\{N=100\}}$, $u_{\{N=200\}}$, and $u_{\{N=400\}}$ are represented on *different spatial grids*.

Use spline in Matlab to map $u_{\{N=100\}}$, $u_{\{N=200\}}$, and $u_{\{N=400\}}$ to grid x=[0:0.002:1]*L.

Plot ($u_{\{N=100\}} - u_{\{N=200\}}$) vs x and ($u_{\{N=200\}} - u_{\{N=400\}}$) vs x at $T = 3$ in one figure.

Remark: We can derive that the error in $u_{\{N\}}$, associated with the spatial discretization, is approximately

$$E(h_1)\approx\frac{u(h_1)-u(h_2)}{1-\left(\frac{h_2}{h_1}\right)^2}=\frac{u_{\{N\}}-u_{\{2N\}}}{1-\left(\frac{N-0.5}{2N-0.5}\right)^2}$$

Problem 5 (Computational)

Consider the 2D IBVP of the heat equation

$$\begin{cases} u_t = u_{xx} + u_{yy}\,, \quad (x,y)\in(0,8)\times(0,8)\,, \quad t>0 \\ u(x,y,0) = f(x,y) \\ u(0,y,t) = g_L(y,t)\,, \quad u(8,y,t) = g_R(y,t) \\ u(x,0,t) = g_B(x,t)\,, \quad u(x,8,t) = g_T(x,t) \end{cases}$$

where

$f(x,y) = 0$

$g_L(y,t) = -\sin(\pi y/8)\cdot\tanh(2t)\,, \quad g_R(y,t) = -3\sin(\pi y/8)\cdot\tanh(2t)$

$g_B(x,t) = 0\,, \quad g_T(x,t) = x(1-x/8)\cdot\tanh(2t)$

Use the 2D FTCS method to solve the IBVP to $T = 2$.

Use $\Delta x = \Delta y = 0.08$ and $\Delta t = 1.25\times10^{-3}$.

Part 1: Plot $u(x, 6)$ vs x at $t = 0.1$, 0.5, 1, and 2 in one figure.

Part 2: Plot u vs (x, y) at $T = 2$ as a surface.

Hint: See the sample code on implementing FTCS to solve the 2D heat equation.

Problem 6 (Computational)

Continue with the numerical solution of IBVP in Problem 5.

Use $\Delta x = \Delta y = 0.08$ and $\Delta t = 1.25\times10^{-3}$ to solve the IBVP to $T = 20$.

At each t_n, calculate the maximum of $|u_t|$ over all internal points.

$$E(t_n) = \max_{\substack{1\le i\le N_x-1 \\ 1\le j\le N_y-1}} \frac{|u_{i,j}^{n+1} - u_{i,j}^{n}|}{\Delta t}$$

Part 1: Plot $E(t)$ vs t. Use logarithmic scale for $E(t)$ and linear scale for t.

Hint: In Matlab, to find the largest (absolute value) element of matrix B, use max(abs(B), [ ], 'all');

Part 2: Repeat the simulation using $\Delta x = \Delta y = 0.16$ and $\Delta t = 1.25\times10^{-3}$.

Calculate $u_{\{\Delta=0.16\}} - u_{\{\Delta=0.08\}}$ at $T = 20$ at their common grid points.

Plot $(u_{\{\Delta=0.16\}} - u_{\{\Delta=0.08\}})$ vs (x, y) at $T = 20$ as a surface.