Homework #4, due April 30

(1) Explore the von Neumann stability analysis for wave equation.

We want to explore the stability of the leapfrog method for the one-dimensional linear wave equation

$$u_t + au_x = 0\,.$$

The Leap-frog method can be written as

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\kappa} + a\frac{u_{j+1}^n - u_{j-1}^n}{2h} = 0$$

on the grid, $x_j = jh$.

The exact solution to the wave equation is given by a sum of harmonics of the form $e^{i(\omega t + kx)}$. For stability analysis we consider a single harmonic, and therefore, we write

$$u_j^n = \sigma^n e^{ikjh}$$

(a) Substituting the above normal mode into the difference equation to show that the method is stable if the CFL condition

$$|a|\kappa/h < 1,$$

is satisfied and show that then the amplification factor σ satisfies $|\sigma| = 1$.

(b) Derive the modified equation for the leapfrog as an infinite series and show that all terms with even order derivatives drop out. Hence there are no dissipative terms in the modified equation. (This is consistent with the results in **1(a)**.)

(c) Derive the truncated modified equation of the form

$$v_t + av_x = \mu v_{xxx}$$

what is μ? The computed u_j^n satisfy this equation to what order?

(d) Follow a similar analysis for the Leap-frog method applied to the second order wave equation

$$u_{tt} - a^2 u_{xx} = 0$$

For a grid $x_j = jh$,

$$\frac{u_j^{n+1} - 2u_j^n + u_j^{n-1}}{\kappa^2} - a^2\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} = 0$$

(e) The second order wave equation can be split into two first order linear wave equation. Set $v = u_t - au_x$ and $w = u_t + au_x$. Show that they satisfy

$$v_t + av_x = 0 \qquad \text{and} \qquad w_t - aw_x = 0\,.$$

(2) Explore implicit and explicit schemes for heat equation:

$$u_t = ku_{xx}, \quad u(0,t) = u(1,t)$$

(a) Write a code with explicit scheme (forward Euler and mid-point rule second order scheme) and find out (using maximum principle or normal mode analysis or energy estimate) the the stability condition and verify it numerically.

(b) Write a code with implicit scheme (backward Euler and Crank-Nicholson) and find out the the stability condition and verify it numerically. You can use LAPACK or FFT to solve the resulted linear system.

(c) Check accuracy for both schemes. Compare the efficiency between the implicit and explicit schemes for both small k and large k.

(3) Explore the *Péclet number* and *cell Péclet number constrain.*

Consider the following convection–diffusion problem

$$au_x = \epsilon u_{xx}, \quad \text{for } 0 \le x \le L \tag{1}$$

with boundary condition

$$u(0) = 0, \quad u(L) = 1 \tag{2}$$

(a) The exact solution,

$$u(x) = (1 - e^{Rx})/(1 - e^{R}) \tag{3}$$

for the above equation can be obtained by integrating twice and by using the boundary conditions. Here $R = aL/\epsilon$ is called the *Péclet number.*

Plot the exact solution in matlab for a small ϵ and show that there is a boundary layer at the right-end $x = L$ if $a > 0$ and there is a boundary layer at the left-end $x = 0$ if $a < 0$. Find out numerically that the width of the boundary layer is proportional to R.

(b) The standard central difference approximation

$$a\frac{u_{j+1} - u_{j-1}}{2h} = \epsilon\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} \tag{4}$$

can be written as

$$u_j = \alpha u_{j-1} + (1 - \alpha)u_{j+1} \tag{5}$$

where $\alpha = (2 - R_h)/4$ and $R_h = ah/\epsilon$ is called *cell Péclet number.* Show that

$$|R_h| \le 2 \tag{6}$$

is the necessary and sufficient condition for u_j being a convex combination of u_{j-1} and u_{j+1}, i.e., $0 < \alpha < 1$. The condition (6) is known as *cell Péclet number constrain.*

(c) Show that when u_j is a convex combination of u_{j-1} and u_{j+1}, there is no interior extrema points for the solution (this is known as maximum principle). Hence the solution is monotone.

(d) Show numerically that when the *cell Péclet number* close to 2, the solution at the point next to the boundary is about 85% of the value across the boundary layer, i.e., there is only one point in the boundary layer. Show also that when the *cell Péclet number* is bigger than to 2, the solution behaves oscillatory. (see H. B. Keller, 1976, "Numerical Solution of Two-Point Boundary-Value Problems").

(4) Explore upwind scheme and numerical viscosity.

The equation (1) can be approximated by an upwind scheme ($a > 0$)

$$a\frac{u_j - u_{j-1}}{h} = \epsilon\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} \tag{7}$$

(a) Show that the above upwind scheme can always be written in a convex combination form

$$u_j = \alpha u_{j-1} + (1-\alpha) u_{j+1}, \quad 0 < \alpha < 1 \tag{8}$$

That is the solution is always monotone and the scheme is always stable.

(b) Derive the modified equation for the upwind scheme and find out the effective numerical viscosity and the effective *cell Péclet number*. Explain why the upwind scheme is always stable.

(c) When ϵ is small, find out how many points in the boundary layer and explain why.

(d) The solution to the upwind scheme can be given by a closed-form formula

$$u_j = A + B\kappa^j \,. \tag{9}$$

Find κ from (8) and then determine A and B from the boundary condition (2). Show that as $h \to 0$, $x_j \to x$, then $u_j \to u(x)$, where $u(x)$ is given by (3).

(5) Explore the steady state solution of a dynamic equation.

The solution to the convection-diffusion equation,

$$a u_x = \epsilon u_{xx}, \quad u(0) = 0, \quad u(1) = 1$$

can be approximated by the following dynamic equation,

$$u_t + a u_x = \epsilon u_{xx}, \quad u(0) = 0, \quad u(1) = 1. \tag{10}$$

as t large. (Indeed, few convection time scale, say $t = 4$, will be sufficient)

(a) Use the standard upwind differencing (for $a > 0$)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\frac{u_j^n - u_{j-1}^n}{h} = \epsilon\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} \tag{11}$$

for the dynamic equation and show that the scheme is stable if

$$a\frac{\Delta t}{\Delta x} < 1, \qquad \epsilon\frac{\Delta t}{\Delta x^2} < \frac{1}{2}. \tag{12}$$

(b) For how large n the solution reach steady state? Plot the solution to see if there is any oscillation.

(c) A simple method of improving the convergence characteristics is to introduce some artificial implicitness into the problem without actually using an implicit time marching algorithm. This is done by using the solution at the new time-step at points where it is already calculated, i.e., the discreteized equation are now

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\frac{u_j^n - u_{j-1}^{n+1}}{h} = \epsilon\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^{n+1}}{h^2}\,. \tag{13}$$

This is similar to Gauss-Seidel iteration, which improves the convergence of Jacobi iteration method in a similar manner. During each time step we "sweep" along the x-direction, and therefore, the solution u_{j-1}^{n+1} is already calculated when u_j^{n+1} is to be calculated. Although this is a rather simple change to the algorithm, it improves the convergence considerable. An even better scheme can be formated as

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\frac{u_j^{n+1} - u_{j-1}^{n+1}}{h} = \epsilon\frac{u_{j+1}^n - 2u_j^{n+1} + u_{j-1}^{n+1}}{h^2}\,. \tag{14}$$

Perform and compare with these three schemes.

(6) Explore the high order upwind scheme and TVD property.

A second order upwind scheme for (10) can be constructed as follows:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + a\frac{u_{j+1/2}^n - u_{j-1/2}^n}{h} = \epsilon\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} \tag{15}$$

and

$$u_{j+1/2}^n = u_j^n + s_j h/2, \quad s_j = \text{minmod}\left(\frac{u_{j+1}^n - u_j^n}{h}, \frac{u_j^n - u_{j-1}^n}{h}\right) \tag{16}$$

where the

$$\text{minmod}(\alpha,\beta) = \begin{cases} 0, & \text{if } \alpha\beta < 0 \\ \alpha, & \text{if } \alpha\beta > 0 \text{ and } |\alpha| \le |\beta| \\ \beta, & \text{if } \alpha\beta > 0 \text{ and } |\alpha| \ge |\beta| \end{cases} \tag{17}$$

In the case $a < 0$, (14) becomes

$$u_{j+1/2}^n = u_{j+1}^n - s_{j+1}h/2, \tag{18}$$

(a) Compare the performance between the first order upwind scheme and second order upwind scheme in terms of shapeness of the boundary layer and number of iterations need to reach the steady state.

(b) Show that

$$\text{minmod}(\alpha,\beta) = \text{minmod}(1,\beta/\alpha)\alpha \qquad \text{minmod}(\alpha,\beta) = \text{minmod}(1,\alpha/\beta)\beta \tag{19}$$

and

$$0 \le \text{minmod}(1,r) \le 1 \tag{20}$$

Use this property to reformat the above second order scheme in a TVD (total variation diminishing) form:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = c_j^n(u_{j+1}^n - u_j^n) - d_j^n(u_j^n - u_{j-1}^n) \qquad c_j^n, d_j^n \ge 0 \tag{21}$$

where

$$c_j^n = \frac{\epsilon}{h^2} \qquad d_j^n = \frac{\epsilon}{h^2} + \frac{a}{2h}\left(2 + \text{minmod}\left(1, \frac{u_{j+1}^2 - u_j^2}{u_j^2 - u_{j-1}^2}\right) - \text{minmod}\left(1, \frac{u_{j-1}^2 - u_{j-2}^2}{u_j^2 - u_{j-1}^2}\right)\right) \tag{22}$$

Show that

$$c_j^n, d_j^n \ge 0 \tag{23}$$

(c) Let $v_j^n = u_{j+1}^n - u_j^n$ and derive from (21) that

$$v_j^{n+1} = (1 - \Delta t(c_j^n + d_{j+1}^n))v_j^n + \Delta t c_{j+1}^n v_{j+1}^n + \Delta t d_j^n v_{j-1}^n \tag{24}$$

Show when (CFL condition)

$$1 - \Delta t(c_j^n + d_{j+1}^n) \ge 0 \tag{25}$$

$$|v_j^{n+1}| \le |v_j^n| + \Delta t(c_{j+1}^n|v_{j+1}^n| - c_j^n|v_j^n|) + \Delta t(d_j^n|v_{j-1}^n| - d_{j+1}^n|v_j^n|) \tag{26}$$

and hence TVD:

$$TV(v^{n+1}) \equiv \sum_j |v_j^{n+1}| \le \sum_j |v_j^n| = TV(v^n) \tag{27}$$

(d) Show that the steady state solution can be written as a convex combination form

$$u_j = \alpha_j u_{j-1} + (1-\alpha_j)u_{j+1}, \quad 0 < \alpha_j < 1 \,. \tag{28}$$

Hence the solution is monotone.

(e) Show that the steady state solution is indeed equivalent to the standard 2nd order upwind scheme: $(a > 0)$

$$\frac{a}{h}(u_j - u_{j-1} + \frac{1}{2}(u_j - 2u_{j-1} + u_{j-2})) = \frac{\epsilon}{h^2}(u_{j+1} - 2u_j + u_{j-1})$$

(7) The methodology discussed in the previous problems can be readily extended to two-dimensional flows.

Consider the two-dimensional convection–diffusion equation

$$au_x + bu_y = \epsilon(u_{xx} + u_{yy}), \tag{29}$$

with $a(x, y) = y$ and $b(x, y) = -x$ and the boundary conditions:

$$u(0, y) = 1, \text{ if } y > 1/2, \quad u(0, y) = 0, \text{ if } y < 1/2,$$

$$u(x, 1) = 1 \quad u(1, y) = 1, \quad u(x, 0) = 0$$

There is an internal layer at the curve $x^2 + y^2 = 1/4$ and a boundary layer at $y = 0, 1/2 \leq x \leq 1$. Resolving the internal layer is much more difficult than resolving the boundary layer.

(a) Show that the standard upwind scheme for the above equation can be written in a convex combination form

$$u_{i,j} = a_1 u_{i+1,j} + a_2 u_{i-1,j} + a_3 u_{i,j+1} + a_4 u_{i,j-1}, \qquad a_i \geq 0 \text{ and } \sum_i a_i = 1. \tag{30}$$

and that there is no interior extrema points. Hence there is no oscillation in the solution.

(b) In a manner similar to the problem 3, we seek the steady state solution of the time-dependent problem given by

$$u_t + au_x + bu_y = \epsilon(u_{xx} + u_{yy}). \tag{31}$$

Using upwind differencing for the convection term we have

$$\frac{u_{i,j}^{n+1} - u_{i,j}^n}{\Delta t} + y_j \frac{u_{i,j}^n - u_{i-1,j}^n}{h} - x_i \frac{u_{i,j+1}^n - u_{i,j}^n}{h} = \frac{\epsilon}{h^2}(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{i,j}^n) \tag{32}$$

The solution is time-stepped starting an initial guess solution using the above formula until a steady state is reached. Find how many steps needed for the solution reaching steady state? Use `mesh` or `contour` in matlab to view the solution and find out shapeness and oscillation, if there is any.

(c) Using the second order upwind scheme described in **4** to improve the shapeness of the internal layer. With an impulsively start ($u_{i,j}^0 = 0$) and the Gauss-Seidel iteration as described in **3(d)**, you shall able to reach the steady state with a residual of 10^{-14} at about 600 steps for a grid 128×128.

(d) Show that the steady state solution of the second order scheme can be written as a convex combination form (30). Hence there is no oscillation in the solution.

(e) Write a matlab movie code for your computation with 100 frames. Some useful matlab commands: `replay=100; for k = 1:replay, mesh(u(:,:,k)); M(:,j) = getframe; end; movie(M,replay)`