AA 372

Homework II

Please submit your codes together with your write-ups. Please email/meet me if something is unclear.

1. **Lax-Wendroff Method:** In class we showed that the forward in time centered in space (FTCS) scheme for advection equation,

$$\frac{\partial f}{\partial t} + u\frac{\partial f}{\partial x} = 0, \tag{1}$$

is unconditionally unstable. FTCS is first order accurate in time and second order accurate in space. We can construct a method which is second order accurate in both space and time as follows. Recall that the the second order accurate approximation in time for $\partial f/\partial t$, centered at (i, n), is

$$\frac{\partial f}{\partial t} = \frac{f_i^{n+1} - f_i^n}{\Delta t} - \frac{\Delta t}{2}\frac{\partial^2 f}{\partial t^2} + \mathcal{O}(\Delta t^2).$$

Using Eq. 1 we can express $\partial^2 f/\partial t^2$ as $u^2(\partial^2 f/\partial x^2)$. Thus, the second order accurate method (both in space and time) is

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} - \left(\frac{u^2\Delta t}{2}\right)\frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{\Delta x^2} + u\frac{f_{i+1}^n - f_{i-1}^n}{2\Delta x} = 0, \tag{2}$$

where we have used the centered in space (and hence second order accurate) expressions for $\partial f/\partial x$ and $\partial^2 f/\partial x^2$. Eq. 2 is known as the *Lax-Wendroff* scheme for advection equation.

Perform the **von Neumann stability analysis** (*VNSA*) on Eq. 2. What is the limit on $u\Delta t/\Delta x$ such that the amplification factor is ≤ 1? This limit is known as the Courant-Friedrichs-Lewy condition and is generally applicable for all stable methods solving the advection equation (and to hyperbolic/wave equations in general where u is replaced by the fastest signal speed).

Is Eq. 2 consistent with the advection equation (Eq. 1)? Will the solution converge to the correct result as Δx, $\Delta t \to 0$? Write the **modified equation** for the Lax-Wendroff scheme. Is the leading order error term dispersive or diffusive? How does it connect to VNSA?

We will come back to this once we start with PDEs.

2. **Thomas Algorithm for tridiagonal systems:** Write a program, in the language of your choice, to solve a tridiagonal system of equations $Ax = d$, where A is a tridiagonal matrix,

$$A = \begin{bmatrix} b_1 & c_1 & & & & & \\ a_2 & b_2 & c_2 & & & & \\ & a_3 & b_3 & c_3 & & & \\ & & & \ldots & & & \\ & & & a_{n-2} & b_{n-2} & c_{n-2} & \\ & & & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & & & a_n & b_n \end{bmatrix}.$$

Use Thomas algorithm with forward elimination and back-substitution as discussed in class. As discussed in the class, a tridiagonal system is obtained when performing cubic spline interpolation.

Another situation where an almost-tridiagonal system results is when writing the implicit finite-difference formula (first order accurate in time and second ordered accurate in space) for the heat diffusion equation,

$$\frac{\partial f}{\partial t} = D \frac{\partial^2 f}{\partial x^2}, \tag{3}$$

given by

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = D \frac{f_{i+1}^{n+1} - 2f_i^{n+1} + f_{i-1}^{n+1}}{\Delta x^2}, \text{ or} \tag{4}$$

$$A f^{n+1} = f^n, \tag{5}$$

where

$$A = \begin{bmatrix} b_1 & c_1 & & & \ldots & & a_1 \\ a_2 & b_2 & c_2 & & & & \\ & a_3 & b_3 & c_3 & & & \\ & & & \ldots & & & \\ & & & a_{n-2} & b_{n-2} & c_{n-2} & \\ & & & & a_{n-1} & b_{n-1} & c_{n-1} \\ c_n & & & \ldots & & a_n & b_n \end{bmatrix};$$

f^{n+1} and f^n are vectors at times t^{n+1} and t^n respectively; $b_i = 1 + 2D\Delta t/\Delta x^2$, $a_i = -D\Delta t/\Delta x^2$, and $c_i = -D\Delta t/\Delta x^2$. Notice that we have used periodic boundary conditions; this results in appearance of

a_1 and c_n in corners apart from the tridiagonal structure. Thomas method can be modified to solve Eq. 5 using the **Sherman-Morrison** formula. Sherman-Morrison formula states that the solution of the matrix equation

$$(A + uv^T)x = d, \tag{6}$$

where u and v are column vectors (v^T is the transpose of v), is given by solving

$$Ay = d, Aq = u,$$

and computing $x = y - q(v^T y)/(1 + v^T q)$. To solve Eq. 4 you will need to choose column vectors u and v appropriately and apply the tridiagonal method twice. Hence the solution is $\mathcal{O}(n)$, where n is the number of grid points.

Write a code to solve the diffusion equation (Eq. 3) using periodic boundary conditions with the method discussed above and the tridiagonal code that you wrote. Use $D = 1$ and a domain going from 0 to 1. The initial condition is $f = 1$ for $0.4 < x < 0.6$; outside this f vanishes. Find the solution (f) at time $t = 0.025$; choose the timestep $\Delta t = 4\Delta x^2/D$. Use 256 grid points such that $\Delta x = 1/256$. See how the solution at $t = 1$ converges with increasing resolution. Plot L1 Richardson error as a function of Δx on log-log scale; what is is order of convergence?

Bonus: Perform *VNSA* on Eq. 4 and show that the implicit method is unconditionally stable. Can you reach the same conclusion using the *modified equation*?