MA/CS 615 Spring 2019 Homework #4

Due before class starts on Mar 28th. Late homework will not be given any credit. Collaboration is OK but not encouraged. Indicate on your report whether you have collaborated with others and whom you have collaborated with.

1. (20 pts) Consider solving the wave equation $u_{tt} = c^2 u_{xx}$ (the wave speed $c > 0$ is a constant) on the interval $x \in [0, 2\pi]$ with periodic boundary conditions. A semi-discrete scheme with a fourth order approximation to the second order derivative is:

$$U_j''(t) = c^2 \frac{-U_{j+2}^n + 16U_{j+1}^n - 30U_j^n + 16U_{j-1}^n - U_{j-2}^n}{12\Delta x^2}. \tag{0.1}$$

With the leapfrog time discretization, we get

$$\frac{U_j^{n+1} - 2U_j^n + U_j^{n-1}}{\Delta t^2} = c^2 \frac{-U_{j+2}^n + 16U_{j+1}^n - 30U_j^n + 16U_{j-1}^n - U_{j-2}^n}{12\Delta x^2}, \tag{0.2}$$

which is second order accurate in time and fourth order accurate in space. Analyze the stability of the scheme (0.2) and discuss for what kind of initial conditions the convergence will be ensured. It will consist of three cases depending on whether $\lambda = c\frac{\Delta t}{\Delta x}$ is less than (6 points), equal to (7 points) or greater than (7 points) some number.

2. (20 pts) Consider solving the following problem with periodic boundary conditions:

$$u_t + a u_{xxxx} = 0, \quad x \in [0, 2\pi],$$

$$u(x, 0) = u_0(x),$$

where $a > 0$ is a constant. Let D_x^4 denote the approximation to the fourth order derivative by the centered difference, i.e,

$$D_x^4 U_j = \frac{U_{j-2} - 4U_{j-1} + 6U_j - 4U_{j+1} + U_{j+2}}{\Delta x^4}.$$

Analyze the accuracy and stability of the following scheme (such a time discretization is called Crank-Nicolson method):

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} + a\frac{D_x^4 U_j^{n+1} + D_x^4 U_j^n}{2} = 0.$$

3. (60 points) The vorticity stream-function formulation of the *2D incompressible Navier-Stokes* is given by:

$$\omega_t + u\omega_x + v\omega_x = \frac{1}{Re}\Delta\omega, \tag{0.3}$$

$$\Delta\psi = \omega, \qquad \langle u, v\rangle = \langle -\psi_y, \psi_x\rangle, \tag{0.4}$$

$$\omega(x, y, 0) = \omega_0(x, y)(\text{initial condition}), \qquad \langle u, v\rangle \cdot \mathbf{n} = \text{given on } \partial\Omega(\text{boundary condition}).$$

Here ψ is the stream function and ω is the vorticity, which is the curl of the velocity field $\mathbf{u} = \langle u, v\rangle$. Given ω, to find the velocity, first find ψ by solving $\Delta\psi = \omega$, then we have the velocity by computing $u = -\psi_y, v = \psi_x$. For simplicity, we only consider periodic cases to bypass the boundary condition.

(a) (**10 points**) *2D Poisson Solver*. We need to solve a Poisson equation every time step/stage. Implement the eigenvector method for solving $u_{xx} + u_{yy} = f$ on $[0, 2\pi] \times [0, 2\pi]$ with periodic assumptions. Test your code with the following solution: $f = 2\cos(2x) + 2\cos(2y)$ and $u = \sin^2 x + \sin^2 y - 1$. Recall that the matrix is singular thus we usually set the entry correponding to the zero eigenvalue to be zero, then we can compare it with an exact solution which sums to zero. Use $fft2$ and $ifft2$ functions for the eigenvector multiplication. Use uniform $N \times N$ meshes. Show loglog plot of the errors in max norm and compare it the second order slope line for $N = 20, 40, 80, 160, 320$. Show your CPU time (use tic, toc functions in MATLAB to track CPU time) for $N = 1280, 2560, 5120$.

(b) (**10 points**) *Linear Convection Diffusion*. To solve the nonlinear convection diffusion, test the discretization on the linear one first. Consider the following equation with periodic b.c. on $[0, 2\pi] \times [0, 2\pi]$:

$$u_t + u_x + u_y = d(u_{xx} + u_{yy}), \quad d > 0.$$

We can use centered difference for all spatial derivatives to achieve second order accuracy in space. Let $h = \Delta x = \Delta y$ denote mesh size of a uniform mesh and Δ_h denote the discrete Laplacian:

$$\frac{d}{dt}u_{i,j} = -\frac{u_{i+1,j} - u_{i-1,j}}{2h} - \frac{u_{i,j+1} - u_{i,j-1}}{2h} + d\Delta_h u_{i,j}. \tag{0.5}$$

If we use explicit methods, the time step constaint for the Laplacian part will be $\Delta t \sim \frac{h^2}{d}$ which is acceptable if d is very small (for instance $d \sim h$). However, using forward Euler or RK2 for (0.5) with $d = 0$ is never stable because their stability regions do not contain any imaginary axis (or we can look at (**??**)). The stability regions of RK3 and RK4 contain part of the imaginary axis thus we could have a reasonable stable time step for small d.

Use the following Strong Stability Preserving (SSP) RK3 to solve (0.5) with the time step $\Delta t = \min\{0.3\frac{h^2}{d}, 0.3h\}$ ($h = \Delta x = \Delta y$):

```
U1=U+dt*RHS(U);
U2=0.75*U+0.25*(U1+dt*RHS(U1));
U=U/3+2/3*(U2+dt*RHS(U2));
```

Test your code using the exact solution $u(x, t) = exp(-2dt)\sin(x + y - 2t)$ with $d = 0.01$ on $[0, 2\pi] \times [0, 2\pi]$. Run it till $T = 0.2$ with $N = 16, 32, 64, 128, 256, 512$. Show the table of error and order.

(c) (**10 points**) *2D Incompressible Flow*. Plugging the second equation of (0.4) into (0.3), we get

$$\omega_t - \psi_y \omega_x + \psi_x \omega_y = \frac{1}{Re}\Delta\omega.$$

Let D_x and D_y denote the central difference for the first order partial derivatives, we get

$$\omega_t = D_y\psi D_x\omega - D_x\psi D_y\omega + \frac{1}{Re}\Delta_h\omega.$$

Use the RK3 above for the time derivative. At each time step t^n, first compute the maximum of velocity by $U^n = \max\{|u^n|, |v^n|\}$, then time step can be taken as $\Delta t = \min\{0.3Reh^2, 0.3\frac{1}{U^n}h\}$ ($h = \Delta x = \Delta y$). Test the accuracy with the exact solution: $\omega(x, y, t) = -2\exp(-2t/Re)\sin x \sin y$ on domain $[0, 2\pi] \times [0, 2\pi]$ with $Re = 100$. Notice that the 2D Poisson equation needs to be solved for each time stage in one RK step. Run your code till $T = 0.2$ with $N = 16, 32, 64, 128, 256$. List the error and order as a table.

(d) (10 points) *Double Shear Layer.* Take $Re = 1000$. The initial condition is

$$\omega(x, y, 0) = \begin{cases} \delta\cos(x) - \frac{1}{\rho}\text{sech}^2((y - \pi/2)/\rho) & y \le \pi \\ \delta\cos(x) + \frac{1}{\rho}\text{sech}^2((3\pi/2 - y)/\rho) & y > \pi \end{cases}$$

on domain $[0, 2\pi] \times [0, 2\pi]$, where we take $\rho = \pi/15$ and $\delta = 0.05$. Show your vorticity at $T = 6$ and $T = 8$ with $N = 256$ using 30 contour lines. Make sure your x-axis and y-axis are correctly shown.

For example, the initial value can be visualize using 30 contour lines on a 256×256 mesh as follows

```
n=256;
L   = 2*pi;
x   = linspace(0,L,n+1)'; x = x(2:end);
y=x;
yy=y*ones(1,n);
xx=ones(n,1)*x';
rho=pi/15;
Delta=0.05;
omega1=Delta*cos(xx)-1/rho*sech((yy-pi/2)/rho).^2;
omega2=Delta*cos(xx)+1/rho*sech((3*pi/2-yy)/rho).^2;
indicator1=[ones(n/2,n); zeros(n/2,n)];
indicator2=[zeros(n/2,n); ones(n/2,n)];
omega=omega1.*indicator1+omega2.*indicator2;

contour(x,y,omega,30);colorbar
set(0,'DefaultTextFontSize',18,'DefaultAxesFontSize', 18)
xlabel('X');ylabel('Y')
```

(e) *Implicit diffusion treatment.* For small Re, the time step $\Delta t = 0.3Reh^2$ is too small to use. Instead, we can consider using the IMEX method, i.e., consider the following scheme (first order accurate in time):

$$\omega_{i,j}^{n+1} = \omega_{i,j}^n - u_{i,j}^n D_x \omega_{i,j}^n - v_{i,j}^n D_y \omega_{i,j}^n + \frac{1}{Re}\Delta_h \omega_{i,j}^{n+1},$$

$$u_{i,j}^n = D_y \psi_{i,j}^n, \quad v_{i,j}^n = D_x \psi_{i,j}^n$$

$$\Delta_h \psi_{i,j}^n = \omega_{i,j}^n.$$

(i) (10 points) Replace the velocity field by two constants $u_0 = \max_{i,j} |u^n_{i,j}|$ and $v_0 = \max_{i,j} |v^n_{i,j}|$, which is the same as considering the scheme for the linearized equation

$$\omega_t + u_0 \omega_x + v_0 \omega_x = \frac{1}{Re} \Delta \omega.$$

By plugging in the ansatz $\omega^n_{i,j} = \hat{\omega}^n_{k_1,k_2} e^{\mathbb{i}\, k_1 i \Delta x} e^{\mathbb{i}\, k_2 j \Delta y}$, find the amplification factor $g(\xi_1, \xi_2)$ (where $\xi_1 = k_1 \Delta x, \xi_2 = k_2 \Delta y$) for the linearized scheme

$$\omega^{n+1}_{i,j} = \omega^n_{i,j} - u_0 D_x \omega^n_{i,j} - v_0 D_y \omega^n_{i,j} + \frac{1}{Re} \Delta_h \omega^{n+1}_{i,j}.$$

Let $\lambda_1 = u_0 \Delta t / \Delta x$, $\lambda_2 = v_0 \Delta t / \Delta y$, $\mu_1 = \frac{1}{Re} \Delta t / \Delta x^2$ and $\mu_2 = \frac{1}{Re} \Delta t / \Delta y^2$. Show that the following time step is sufficient to ensure $|g(\xi_1, \xi_2)| \leq 1$:

$$\lambda_1^2 \leq \mu_1, \quad \lambda_2^2 \leq \mu_2,$$

which is

$$\Delta t \leq \frac{1}{\|u^n\|^2_\infty} \frac{1}{Re}, \quad \Delta t \leq \frac{1}{\|v^n\|^2_\infty} \frac{1}{Re}. \tag{0.6}$$

(ii) (10 points) Implement the scheme. Use the eigenvector method to invert the matrix and use FFT for the eigenvectors. Take $Re = 70$. The initial condition is

$$\omega(x, y, 0) = \begin{cases} \delta \cos(x) - \frac{1}{\rho} \operatorname{sech}^2((y - \pi/2)/\rho) & y \leq \pi \\ \delta \cos(x) + \frac{1}{\rho} \operatorname{sech}^2((3\pi/2 - y)/\rho) & y > \pi \end{cases}$$

on domain $[0, 2\pi] \times [0, 2\pi]$, where we take $\rho = \pi/15$ and $\delta = 0.05$. Show your vorticity at $T = 6$ and $T = 8$ with $N = 512$ using 30 contour lines. Set $\Delta t = \Delta x$. You can also try a larger Δt or larger Re so that (0.6) is violated on a coarse mesh, and see what happens.