Math 126 – Homework 9 (Due Friday Nov 6)

This assignment requires the use of a mathematical software package like Matlab. Matlab is available for student use in Wheeler 211, VLSB 2180, and Moffitt 1st floor. For more information: `https://ets-computing.berkeley.edu/`. If you are using Matlab, use the files 'heat_equation.m', 'wave_equation.m', and 'poisson_equation.m'.

There is also a free open source Matlab alternative called Octave: `https://www.gnu.org/software/octave/`. The provided Matlab code runs in Octave as well, provided the interactive plotting features in the code are disabled. If you are using Octave, use the files 'heat_equation2.m', 'wave_equation2.m', and 'poisson_equation2.m'. After the code executes, run 'plot(x,u)' to plot the result ('surf(x,y,u)' for Poisson's equation).

1. **Numerically solving the heat equation**

   (a) Consider the Dirichlet problem for the heat equation

   $$\left.\begin{aligned} u_t &= u_{xx} && \text{if } 0 < x < 1,\ t > 0 \\ u(0,t) = u(1,t) &= 0 && \text{if } t > 0 \\ u(x,0) &= \varphi(x) && \text{if } 0 < x < 1, \end{aligned}\right\}.$$

   and the finite difference approximation

   $$\left.\begin{aligned} &u_j^{n+1} = u_j^n + \tfrac{\Delta t}{\Delta x^2}\left(u_{j-1}^n - 2u_n^j + u_{j+1}^n\right) && \text{if } n \geq 1 \text{ and } 1 \leq j \leq J-1 \\ &u_0^n = u_J^n = 0 && \text{for } n \geq 1 \\ &u_j^0 = \varphi(j\Delta x) && \text{for } 1 \leq j \leq J-1, \end{aligned}\right\}.$$

   where $J = 1/\Delta x$. Set $s = \Delta t/\Delta x^2$. Compute the solution u_j^n of the finite difference scheme for various choices of $\varphi(x)$ and $s = 0.45, s = 0.49, s = 0.5$, and $s = 0.51$. Print out plots showing both stable and unstable solutions. Find a smooth initial condition $\varphi(x)$ that becomes oscillatory when $s = 0.5$. [Hint: Use the provided Matlab code.]

   (b) Modify the provided code to work for homogeneous Neumann boundary conditions $u_x(0,t) = u_x(\pi,t) = 0$ and repeat part (a).

   (c) (Optional) Modify the provided code to implement the Crank-Nicolson scheme described in section 8.2.

2. **Numerically solving the wave equation**

   (a) Consider the Dirichlet problem for the wave equation

   $$\left.\begin{aligned} u_{tt} &= u_{xx} && \text{if } 0 < x < 1,\ t > 0 \\ u(0,t) = u(1,t) &= 0 && \text{if } t > 0 \\ u(x,0) &= \varphi(x) && \text{if } 0 < x < 1 \\ u_t(x,0) &= \psi(x) && \text{if } 0 < x < 1, \end{aligned}\right\}.$$

and the finite difference approximation

$$\left.\begin{array}{ll} u_j^{n+1} = 2u_j^n - u_j^{n-1} + s\left(u_{j-1}^n - 2u_n^j + u_{j+1}^n\right) & \text{if } n \geq 2 \text{ and } 1 \leq j \leq J-1 \\ u_0^n = u_J^n = 0 & \text{for } n \geq 2 \\ u_j^0 = \varphi_j & \text{for } 1 \leq j \leq J-1 \\ u_j^1 = \frac{s}{2}\left(\varphi_{j-1} + \varphi_{j+1}\right) + (1-s)\,\varphi_j + \psi_j \Delta t & \text{for } 1 \leq j \leq J-1, \end{array}\right\}.$$

where $J = 1/\Delta x$, $s = \Delta t^2/\Delta x^2$, $\varphi_j = \varphi(j\Delta x)$ and $\psi_j = \psi(j\Delta x)$. Compute the solution u_j^n of the finite difference scheme for $\psi \equiv 0$, various choices of $\varphi(x)$, and $s = 0.99, s = 1.00$, and 1.1. Print out plots showing both stable and unstable solutions. [Hint: Use the provided Matlab code.]

(b) Modify the provided code to work for homogeneous Neumann boundary conditions $u_x(0,t) = u_x(\pi,t) = 0$ and a nonzero initial velocity ψ, and repeat part (a).

(c) (Optional) Modify the provided code to implement a mixed boundary condition $u(1,t) = u_x(1,t) = 0$, and/or a Robin-type boundary condition. What do these boundary conditions correspond to physically?

3. **Numerically solving Poisson's equation**

(a) Consider the Dirichlet problem for Poisson's equation in the box

$$\left.\begin{array}{rl} -\Delta u(x,y) = f(x,y) & \text{if } 0 < x < 1, \text{ and } 0 < y < 1 \\ u(x,y) = g(x,y) & \text{if } x = 0, x = 1, y = 0 \text{ or } y = 1 \end{array}\right\}.$$

and the finite difference approximation

$$\left.\begin{array}{ll} u_{i,j} = \dfrac{1}{4}\left(u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i-1,j} + \Delta x^2 f_{i,j}\right) & \text{if } 1 \leq i,j \leq J-1 \\ u_{i,j} = g_{i,j} & \text{if } i = 0, i = J, j = 0 \text{ or } j = J, \end{array}\right\}.$$

where $J = 1/\Delta x$, $g_{i,j} = g(i\Delta x, j\Delta x)$ and $f_{i,j} = f(i\Delta x, j\Delta x)$. Compute the solution $u_{i,j}$ of the finite difference scheme using Jacobi iterations for various choices of g and f. Print out plots of some of your solutions. How many Jacobi iterations does it typically take to converge? How does the number of iterations depend on the grid size J. [Hint: Use the provided Matlab code.]

(b) (Optional) Modify the provided code to implement homogeneous Neumann boundary conditions $\partial u/\partial \mathbf{n} = 0$, and repeat part (a).

(c) (Optional) Modify the provided code to implement Gauss-Seidel iterations and successive overrelaxation, as described in Section 8.4 of the book. Compare the number of iterations required with the Jacobi method.

4. Let $u(x)$ be a smooth function and set $u_j = u(j\Delta x)$ where $\Delta x > 0$.

(a) Find real numbers a, b and c so that

$$\frac{au_j + bu_{j-1} + cu_{j-2}}{\Delta x} = u'(j\Delta x) + O(\Delta x^2).$$

(b) Find real numbers d, e, f, g and h so that

$$\frac{du_j + eu_{j-1} + fu_{j-2} + gu_{j-3} + hu_{j-4}}{\Delta x^2} = u''(j\Delta x) + O(\Delta x).$$

[Hint: Apply part (a) twice.]

(c) (Optional) Is the accuracy in part (b) better than $O(\Delta x)$? If so, what is it? [Hint: Use a Taylor series to write out the expression in part (a) to higher accuracy.]

5. For the diffusion equation $u_t = u_{xx}$, use centered differences for both u_t and u_{xx}. Write down the scheme and show that it is unstable no matter what Δx and Δt are.

6. Consider the Crank-Nicolson Scheme for the heat equation $u_t = u_{xx}$:

$$u_j^{n+1} = u_j^n + \frac{s}{2}\left(u_{j+1}^n - 2u_j^n + u_{j-1}^n\right) + \frac{s}{2}\left(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}\right),$$

where $s = \Delta t/\Delta x^2$. The scheme is *implicit*, since u^{n+1} appears on both sides of the equation, so one has to solve a linear system to find u^{n+1} at each iteration. Show that the Crank-Nicolson scheme is *unconditionally stable*, which means it is stable for all choices of $s > 0$. [Hint: Look for a solution of the form $u_j^n = \lambda_k^n e^{ij\Delta xk}$ and show that

$$\lambda_k = \frac{1 - s + s\cos(\Delta xk)}{1 + s - s\cos(\Delta xk)}.$$

Then show that $|\lambda_k| \leq 1$ for any choice of s.]