Homework 6
MA/CS 375, Fall 2005
Due December 15

This homework will count as part of your grade so you must work independently. It is permissible to discuss it with your instructor, the TA, fellow students, and friends. However, the programs/scripts and report must be done only by the student doing the project. Please follow the guidelines in the syllabus when preparing your solutions.

1. (30 pts.) In this problem we examine the dynamics of repeated matrix multiplication and use spectral analysis to understand it.

   i Let:

$$W = \begin{pmatrix} -0.0325 & -0.5325 & 0.1400 & 0.1725 & -0.3125 \\ -0.5325 & -0.0325 & 0.1400 & 0.1725 & -0.3125 \\ 0.1400 & 0.1400 & 0.1033 & 0.4633 & 0.1833 \\ 0.1725 & 0.1725 & 0.4633 & -0.2092 & 0.4958 \\ -0.3125 & -0.3125 & 0.1833 & 0.4958 & 0.0708 \end{pmatrix}.$$

   Generate a random 5-vector x. Compute $W^n x$, $n = 1:100$. Plot $\frac{\|W^n x\|}{\|W^{n-1}x\|}$ versus n. What do you observe? For $n = 20:20:100$ print $\frac{W^n x}{\|W^n x\|}$. What do you observe?

   ii Use the Matlab command **eig** to compute the eigenvalues and eigenvectors of W. Use the results to explain your observations from part (i).

   iii Repeat (i) for the matrix:

$$W = \begin{pmatrix} -\frac{5}{9} & -\frac{5}{9} & \frac{10}{9} & \frac{1}{9} \\ \frac{5}{9} & \frac{5}{9} & -\frac{4}{9} & \frac{5}{9} \\ -\frac{1}{9} & \frac{8}{9} & -\frac{1}{9} & -\frac{1}{9} \\ \frac{7}{9} & -\frac{2}{9} & \frac{1}{9} & \frac{1}{9} \end{pmatrix}.$$

   iv Repeat (ii) for the results of part (iii).

2. (30 pts.) The method of successive overrelaxation or SOR is a way to accelerate the convergence of the Gauss-Seidel method for solving $Ax = b$. Let ω be a parameter and $x^{(k)}$ be the kth iterate. Then $x^{(k+1)}$ is computed via the formula:

$$\bar{x}_i^{(k+1)} = \frac{(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)})}{a_{ii}},$$

$$x_i^{(k+1)} = (1-\omega) x_i^{(k)} + \omega \bar{x}_i^{(k+1)}.$$

Note that if $\omega = 1$ then SOR is identical to Gauss-Seidel. Modify the book's program itermeth.m to accept a new parameter ω as an input and replace the Gauss-Seidel option by SOR with parameter ω. Use your method to solve $Ax = b$ where A is the 100×100 tridiagonal matrix whose diagonal entries are 2.01 and whose offdiagonal entries are -1. Considering $\omega = .2 : .2 : 1.8$ record the number of iterations required to converge to a tolerance of 10^{-4}. What is the best choice? How much of a difference does it make? Change the diagonal element to 2.001 and repeat the experiment. How do your results change?

3. (40 pts.) In this problem we compare three different schemes for solving the first order wave equation with periodic boundary conditions:

$$\frac{\partial u}{\partial t} = c\frac{\partial u}{\partial x} \ , \ a \le x \le b \ , \ u(a,t) = u(b,t) \ , \ u(x,0) = f(x) \tag{1}$$

We introduce space and time discretizations:

$$x_m = a + m * h \ , \ h = \frac{b-a}{M} \ , \ m = 0, \ldots, M$$

$$t_n = n * k \ , \ h = \frac{T}{N} \ , \ n = 0, \ldots, N \ .$$

(a) The forward-time, forward-space scheme (**FF** scheme):

$$\frac{u_m^{n+1} - u_m^n}{k} = c\frac{u_{m+1}^n - u_m^n}{h} \ . \tag{2}$$

$$u_m^{n+1} = u_m^n + \alpha\left(u_{m+1}^n - u_m^n\right) \ , \ \alpha = \frac{ck}{h} \ , m = 1, \ldots, M \ , \ , n = 0, \ldots, N-1 \ . \tag{3}$$

(b) The forward-time, centered-space scheme (**CF** scheme):

$$\frac{u_m^{n+1} - u_m^n}{k} = c\frac{u_{m+1}^n - u_{m-1}^n}{2h} \ . \tag{4}$$

$$u_m^{n+1} = u_m^n + \frac{\alpha}{2}\left(u_{m+1}^n - u_{m-1}^n\right) \ , m = 1, \ldots, M \ , \ , n = 0, \ldots, N-1 \ . \tag{5}$$

(c) The leap-frog (or centered-time, centered-space) scheme (**LF** scheme):

$$\frac{u_m^{n+1} - u_m^{n-1}}{2k} = c\frac{u_{m+1}^n - u_{m-1}^n}{2h} \ . \tag{6}$$

$$u_m^{n+1} = u_m^{n-1} + \alpha\left(u_{m+1}^n - u_{m-1}^n\right) \ , \ m = 1, \ldots, M \ , \ , n = 0, \ldots, N-1 \ . \tag{7}$$

Since the leap-frog scheme is two-levels deep in time, it is called a multi-step scheme. To get it started we need to specify data for the first two time steps. Here we will do it using the scheme (5).

Because of the periodic boundary conditions, the solution can be assumed to extend periodically in x outside the computational domain. Thus, we can identify the values $u^n_{M+k} = u^n_k$, for $k = -1, 0, 1$ needed in the above formulas.

If we write

$$\mathbf{U}^n = (u^n_1, u^n_2, \ldots, u^n_M)^T \ ,$$

then the above 3 schemes can be written in matrix form:

(a) The FF scheme (3)

$$\mathbf{U}^{n+1} = A_F \mathbf{U}^n \tag{8}$$

(b) The CF scheme (5)

$$\mathbf{U}^{n+1} = A_C \mathbf{U}^n \tag{9}$$

(c) The LF scheme (7)

$$\mathbf{U}^{n+1} = A_L \mathbf{U}^n + \mathbf{U}^{n-1} \tag{10}$$

You must implement these schemes to solve equation (1) with

$$a = -1 \ , \ b = 1 \ , \ T = 2 \ ,$$

with initial condition

$$u(x, 0) = f(x) = \text{sech}\,(5 * \sin(\pi x)) \ .$$

(a) Use $M = 200, 400$ and $N = 300$, and $c = .9$. Compute the numerical solutions at $t = T$ and plot each against the exact solution

$$u_{exact}(x, t) = \text{sech}\,(5 \sin(\pi\,(x + ct)))$$

How do the computations with the two different values of M differ?

(b) Now use $M = 200, N = 300$ and $M = 400, N = 600$. Compare the performance of the schemes (1) and (3).