APPM/MATH 4660 Homework #2

DUE 02/15/08

February 12, 2008

Feel free to work in groups, but your final code and your final writeup must be your own work. Please also hand in a copy of both your code as well as the output.

1. Write software implementations of Gauss-Seidel, Successive Over-Relaxation, and Conjugate Gradient to solve $Ax = b$ for A, b known. For GS and SOR stopping criteria, test to see if the ∞-norm of the sum of the absolute and relative changes for each step is less than 10^{-2}, i.e.,

$$\left\| x^k - x^{k-1} \right\|_\infty + \frac{\left\| x^k - x^{k-1} \right\|_\infty}{\left\| x^k \right\|_\infty} < \varepsilon = 10^{-2}.$$

   Your code should have inputs of A, b, x^0, $\bar{k}$, and ε, where $\bar{k}$ is the maximum number of iterations and ε is the tolerance. For CG, just use the tolerance given in class.

2. Consider an example similar to the one from class

$$\begin{aligned} Ax &= b \\ \begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -2 \\ 0 & -2 & 4 \end{bmatrix} x &= \begin{bmatrix} 24 \\ 30 \\ -24 \end{bmatrix} \end{aligned}$$

   where $x = (6, 0, -6)^T$ is the solution.

   (a) For GS and SOR, create a plot showing (for each iteration) the 2-norm of the error and the upper bound in Theorem 7.27 (from the book). That is, the x-axis will be iteration count and there should be two values at each iteration.

(b) Rerun your 3 codes with the (2,3) element in A changed to -4. What is different about the convergence (or lack thereof) for the three methods? Justify your answer.

3. The motivating example for this section was the one-dimension version of this PDE:

$$\nabla^2 x(t) = f(t),$$

and for $t \in [0,1]$ with Dirichlet boundary conditions $x(0) = x(1) = 0$. For a uniform discretization of size $h = 1/n$, the discretized equation is

$$\frac{1}{h^2}\begin{bmatrix} -2 & 1 & 0 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \ddots & \vdots \\ 0 & 0 & 1 & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & -2 & 1 \\ 0 & 0 & \cdots & 0 & 1 & -2 \end{bmatrix} x = \begin{bmatrix} f(t_1) \\ f(t_2) \\ f(t_3) \\ \vdots \\ f(t_{n-2}) \\ f(t_{n-1}) \end{bmatrix}.$$

Let $f(t) = 10 * \sin(2\pi t)$. For those of you playing along in matlab, the code to generate this matrix would look like:

```
>> A = (-2*diag(ones(n-1,1))+diag(ones(n-2,1),-1)+diag(ones(n-2,1),1))/h∧2;
```

(a) For $\bar{k} = 10$, plot the 3 approximation solutions for $n = 10$ and $n = 100$ in comparison to the analytical solution (remember this is just a 1D equation...integrate twice for enlightenment).

(b) Do your computational results match with the *a priori* predictions of errors at each step? Justify your answer (again, refer to Corollary 7.20 here).

(c) The eigenvalues of A are all negative. Why does Conjugate Gradient still work? (HINT: Think about the optimization condition at each step.)