EE 522 Spring 2008
Homework 1: Iterative Methods

The due date for this assignment is Tuesday February 26.

1. Generate a positive definite, symmetric, $N \times N$ random matrix A with condition number 20, and a random vector y. Use Gauss elimination (`x=A\y`), CG, Bi-CG, Bi-CGSTAB, QMR, and GMRES algorithms of MATLAB to solve the system $Ax = y$ for $N = 200 : 50 : 1000$. For the iterative algoritms, use a tolerance of 10^{-4} and set the maximum number of iterations to N. For each value of N measure the elapsed time by using `tic` and `toc` functions of MATLAB, and also the number of iterations used for each algorithm (except for Gauss elimination). Plot the elapsed time vs. size N for each algorithm on the same figure. Plot the number of iterations vs. N on a separate figure. Using these results, compare these algorithms.

2. Generate a positive definite, symmetric, 500×500 random matrix with condition number 20. Make a plot of the residual error as a function of iteration count for CG, Bi-CG, and Bi-CGSTAB algorithms (use `semilogy` command of MATLAB for the plot to see small error values) and compare the algorithms.

3. Use CG on indefinite and/or non-symmetric matrices and check the convergence flag of the algorithm. Comment on the results.

4. Generate a 500×500 matrix A by the MATLAB command

```
A = toeplitz(1./(1.01.^[0:499]),1./(1.2.^[0:499]));
```

   This matrix is not symmetric and its condition number is relatively large. Generate a vector y by the MATLAB command

$$\mathtt{y} = \mathtt{ones}(500, 1);$$

   and solve the system $Ax = y$ by using the Bi-CG and Bi_CGSTAB algorithms. Make a plot of the residual error as a function of iteration count for both algorithms and comment on the results.

Note Generation of a random matrix with desired condition number in MATLAB.

Assume that we want to generate an $N \times N$ real, symmetric matrix with a given condition number. Assume that we have a set of eigenvalues $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_N]$ sorted in ascending order. Let $D = \operatorname{diag}(\lambda)$ be the matrix whose diagonal entries are λ_i, and T be a unitary matrix so that $T^{-1} = T^H$. Define $A = T^H DT$.
Claim: A is a symmetric matrix.
Proof:

$$A^H = \left(T^H DT\right)^H = T^H D^H T = T^H DT, \quad \text{since } D \text{ is a diagonal matrix } D^H = D$$

Claim: The eigenvalues of A are λ_i.

Since D is a diagonal matrix with diagonal elements being λ_i its eigenvalues are λ_i. Let e_i be an eigenvector of D associated with the eigenvalue λ_i, i.e., $De_i = \lambda_i e_i$. Consider the vector $T^H e_i$. We have

$$\begin{aligned} AT^H e_i &= T^H DTT^H e_i \quad (A = T^H DT) \\ &= T^H De_i \quad (T^H T = I) \\ &= \lambda_i T^H e_i \quad (De_i = \lambda_i e_i) \end{aligned}$$

hence $T^H e_i$ is an eigenvector of A associated with the eigenvalue λ_i.
To generate a unitary matrix, we can use the fact that all eigenvectors of a symmetric matrix are orthogonal. So let S be any matrix and determine the eigenvectors of the symmetric matrix $S^H S$. The matrix T formed by placing the eigenvectors of $S^H S$ in columns is a unitary matrix. Thus the following MATLAB function generates a matrix of given size with the given condition number. You can check the condition number of the generated matrix by using the MATLAB function `cond(A)`.

```
function A = genmat(N,cond_num,sym)
% Generates a real, random NxN matrix with the
% condition number cond_num.
% If the third argument is not specified or is non-zero
% A is symmetric.  If the third argument is zero, A is
% non-symmetric.
if nargin<3, sym = 1; end
% Generate Gaussian random numbers.
lambda = randn(1,N);
lambda_min = min(abs(lambda)); % The one with smallest modulus
lambda_max = max(abs(lambda)); % The one with largest modulus
% Map the modulus of random numbers into the interval [1,cond_num]
lambda = ((cond_num-1)/(lambda_max-lambda_min)*...
    (abs(lambda)-lambda_min)+1).*sign(lambda);
D = diag(lambda);
S = randn(N);
if sym
    [T,e] = eig(S'*S);
    A = T'*D*T;
else
    [U,s,V] = svd(S);
    A = U*D*V';
end
```