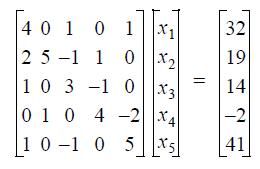
EM425 Assignment #4

Problem Statements

1. (Based on 4.10) Determine the LU decomposition of the matrix  by hand.

2. (Based on 4.15) Carry out (by hand) the first three iterations of the solution of the following system of equations using the Gauss-Seidel iterative method. For the first guess of the solution, take the value of all the unknowns to be zero.



3. Consider the linear system given below.

1. Find the solution using the built-in MATLAB function [L,U,P] = lu(A) – be sure to get and use the permutation matrix P.
2. Find the relative residual for your solution where the  where  is the solution you found in part a) and  is the vector 2-norm.
3. Find the inverse of the matrix, A-1, using the built-in MATLAB function inv(A), and compute the 2-norm of A and A-1 using the built-in MATLAB function norm(A,p), where p should be set to 2 for the 2-norm.
4. Calculate the size of the error bound presented in Lecture 10, shown in the equation below:

where  is the condition number.

1. Repeat steps a)-d) for the test matrix nos5.mtx provided. Once you have read the matrix into MATLAB, convert the matrix to a “full” (non-sparse) format by using the MATLAB built-in function A\_full = full(A\_sparse). For the “right hand side” vector b, use a vector of all ones. Note how different the error bound is in this case.

4. Using Jacobi, Gauss-Seidel, and SOR  iterative methods, write and run code to solve the following linear system of equations:



The stopping criteria should be the relative change of the estimated solution in the 2-norm:



with a tolerance set to 10-9. Compare the number of iterations required in each case.

5. For this problem we will compare two iterative methods to solve nos5—a test matrix derived from a structural analysis of a three-story building. For the “right hand side” vector b, use a vector of all ones.

a) Use the SOR method with an estimated relative error tolerance of 10-4. Vary the relaxation parameter over the range . Make a plot of the number of iterations required as a function of  and approximate the best value of  for this system.

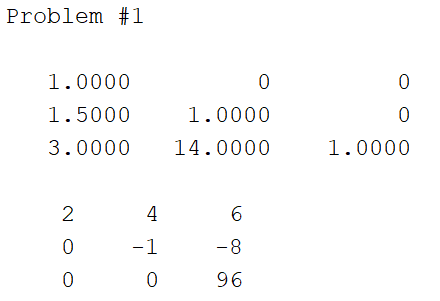
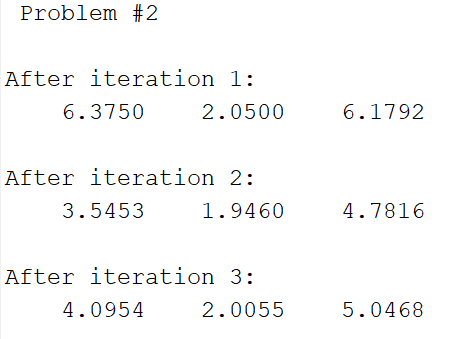
b) Use the MATLAB built-in function gmres without preconditioner to solve this problem.

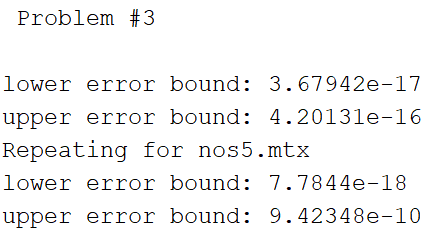
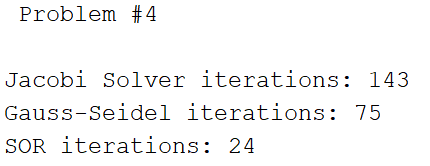
c) Use gmres again but with an incomplete LU preconditioner. Use opts\_ilu.type=’ilutp’ and opts\_ilu.droptol=dtol and vary the value of dtol over the range . What happens as dtol gets larger? ***Hint:*** *Use the built-in function spy(L) to show the sparsity pattern of L and note the number of non-zeros (nnz) of L. How does nnz(L) change as dtol is changed?*

d) This is a relatively small sparse system and can be solved using direct methods. Use the MATLAB built-in function timeit to estimate the time required to solve the system of equations using the built-in mldivide (“backslash”) function.

Organize your findings from part a) - d) into a short document. A couple of paragraphs should be enough. Include the plot you created from part a) and any other graphics/tables that you find helpful to communicate what you observed in parts b) through d). Upload the document in PDF format for your homework submission.

***Numeric Answers***





Your answers for Problem #3 may vary somewhat.

Problem #5:

I leave most of this to you as I want you to spend some time experimenting with the different SOR relaxation parameters and MATLAB built-in pre-conditioner settings.