ASE 211 Homework 5 Solution

Due: 12:00 noon, Friday, February 25.

1. Work problem 4.4 in the text.

Jacobi iteration:

$$x_1^{(1)} = \frac{1}{8}[8] = 1$$
$$x_2^{(1)} = \frac{1}{7}[4] = .57142$$
$$x_3^{(1)} = \frac{1}{9}[12] = 1.3333$$

$$x_1^{(2)} = \frac{1}{8}[8 - .57142 + 1.3333] = 1.0952$$
$$x_2^{(2)} = \frac{1}{7}[4 + 1 + 2(1.3333)] = 1.0952$$
$$x_3^{(2)} = \frac{1}{9}[12 - 2 * 1 - .57142] = 1.0476$$

$$x_1^{(3)} = \frac{1}{8}[8 - 1.0952 + 1.0476] = .9940$$
$$x_2^{(3)} = \frac{1}{7}[4 + 1.0952 + 2(1.0476)] = 1.0272$$
$$x_3^{(3)} = \frac{1}{9}[12 - 2 * 1.0952 - 1.0952] = .9683$$

Gauss-Seidel iteration:

$$x_1^{(1)} = \frac{1}{8}[8] = 1$$
$$x_2^{(1)} = \frac{1}{7}[4 + 1] = .7143$$
$$x_3^{(1)} = \frac{1}{9}[12 - 2 * 1 - .7143] = 1.0317$$

$$x_1^{(2)} = \frac{1}{8}[8 - .7143 + 1.0317] = 1.0397$$
$$x_2^{(2)} = \frac{1}{7}[4 + 1.0397 + 2(1.0317)] = 1.0147$$
$$x_3^{(2)} = \frac{1}{9}[12 - 2 * 1.0397 - 1.0147] = .9895$$

$$x_1^{(3)} = \frac{1}{8}[8 - 1.0147 + .9895] = .9969$$

$$x_2^{(3)} = \frac{1}{7}[4 + .9969 + 2(.9895)] = .9966$$

$$x_3^{(3)} = \frac{1}{9}[12 - 2 * .9969 - .9966] = 1.0011$$

```
function x=jacobi(x,a,b,epsilon,itermax,n)
%
%  This function performs Jacobi iteration
%
r=b-a*x;
%
%  compute norm of r
%  use built-in matlab function norm
%
rnorm=norm(r);
k=0;
while (rnorm > epsilon & k < itermax)
   for i=1:n
     z(i)=b(i);
     for j=1:i-1
       z(i)=z(i)-a(i,j)*x(j);
     end
     for j=i+1:n
       z(i)=z(i)-a(i,j)*x(j);
     end
     z(i)=z(i)/a(i,i);
   end
   x=z';
   r=b-a*x;
   k=k+1;
   rnorm=norm(r);
end
k
rnorm
```

```
>> x=[0;0;0];
>> n=3;
```

```
>> itermax=10;
>> epsilon=.0001;
>> A=[8 1 -1;-1 7 -2; 2 1 9];
>> b=[8; 4; 12];
>> x=jacobi(x,A,b,epsilon,itermax,n)

x =

    1.0000
    1.0000
    1.0000


k =

    10


rnorm =

   5.8545e-05

function x=gauss_seidel(x,a,b,epsilon,itermax,n)
%
%  This function performs Gauss-Seidel iteration
%
%
r=b-a*x;
%
%  compute norm of r
%  use built-in matlab function norm
%
rnorm=norm(r);
k=0;
while (rnorm > epsilon & k < itermax)
   for i=1:n
     x(i)=b(i);
     for j=1:i-1
       x(i)=x(i)-a(i,j)*x(j);
```

```
      end
      for j=i+1:n
        x(i)=x(i)-a(i,j)*x(j);
      end
      x(i)=x(i)/a(i,i);
    end
    r=b-a*x;
    k=k+1;
    rnorm=norm(r);
end
k
rnorm

>> x=[0;0;0]

x =

     0
     0
     0

>> x=gauss_seidel(x,A,b,epsilon,itermax,n)

k =

     7


rnorm =

   1.4150e-05


x =

    1.0000
    1.0000
    1.0000
```

2. Consider the following generalization of problem A3.6. You are given a system of $m+1$ springs and m blocks. Each spring has a spring constant k_i and unstretched length L_i. The length between the walls is L_w. Write a matlab code which does the following:

a) allows the user to input data k, L and L_w.

b) assembles the matrix A and right hand side $\mathbf{b}$

c) solves for the distances $x_1, \ldots, x_m$ of the blocks using Gauss-Seidel iteration

d) plots the solution $\mathbf{x}$

Test your code on the following data:

k = (1, 1.5, 3, 4, 2, 2, 4, 6, 8, 1, 9, 1.3, 7, 2, 1, 8, 7)

L = (2, 1.5, 5, 3.2, 4, .5, 3, 4, 2, 1.4, 5, 1.1, 2, 2, 2.3, 4.3, 1)

and take different values of $L_w = 41$ and $L_w = 47$.

NOTE: The matrix has entries $a_{i,i} = -k_i - k_{i+1}$ on the main diagonal and $a_{i,i+1} = k_{i+1} = a_{i+1,i}$ on the diagonals above and below the main diagonal. All other entries are zero. Be careful with the first row and last row of the matrix. The right hand side has entries $b_i = -k_i L_i + k_{i+1} L_{i+1}$ except for the last entry which is $b_m = -k_m * L_m + k_{m+1}(L_{m+1} - L_w)$.

Matlab script for input:

```
% hwk5script
k=[1 1.5 3 4 2 2 4 6 8 1 9 1.3 7 2 1 8 7];
L=[2 1.5 5 3.2 4 .5 3 4 2 1.4 5 1.1 2 2 2.3 4.3 1];
n=size(k,2)-1;
Lw=41;
[A,b]=assemble(k,L,Lw,n);
epsilon=.0001;
itermax=1000;
for i=1:n
   x(i,1)=0;
end
x=gauss_seidel(x,A,b,epsilon,itermax,n);
plot(x)
```

```
function [A,b]=assemble(k,L,Lw,n)
for i=1:n
   A(i,i)=-k(i)-k(i+1);
   b(i,1)=-k(i)*L(i)+k(i+1)*L(i+1);
end
for i=1:n-1
   A(i,i+1)=k(i+1);
   A(i+1,i)=A(i,i+1);
end
b(n,1)=b(n,1)- k(n+1)*Lw;
```

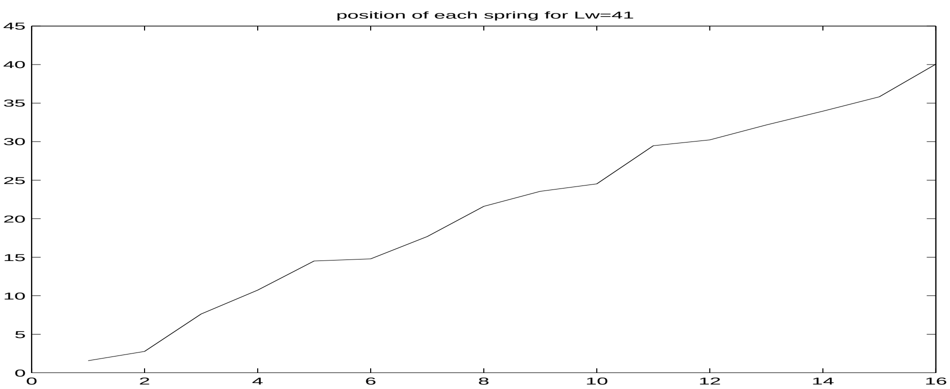

Figure 1: Plot for $L_w = 41$

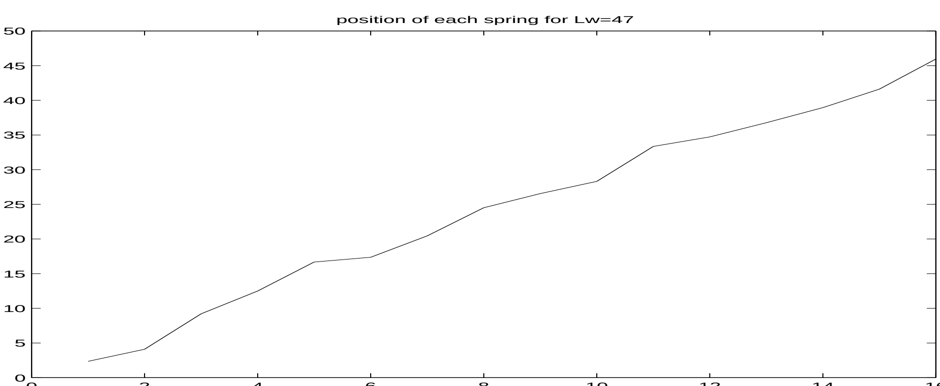

Figure 2: Plot for $L_w = 47$