LESSON 17 GAUSS SEIDEL and GUASS JACOBI

Exercise 1 Use Gauss Seidel and Gauss Jacobi method to obtain the solution for

$$3x_1 - 0.1x_2 - 0.2x_3 = 7.85$$
$$0.1x_1 + 7x_2 - 0.3x_3 = -19.3$$
$$0.3x_1 - 0.2x_2 + 10x_3 = 71.4$$

Solution: First, solve each of the equations for its known on the diagonal:

$$x_1 = \frac{7.85 + 0.1x_2 + 0.2x_3}{3} \quad (1)$$
$$x_2 = \frac{-19.3 - 0.1x_1 + 0.3x_3}{7} \quad (2)$$
$$x_3 = \frac{71.4 - 0.3x_1 + 0.2x_2}{10} \quad (3)$$

By assuming that x2 and x3 are zero, eq. (1) can be used to compute

$$x_1 = \frac{7.85 + 0.1(0) + 0.2(0)}{3} = 2.616667\,.$$

This value along with the assumed value of x3 = 0, can be substituted into eq. 2 to calculate

$$x_2 = \frac{-19.3 - 0.1(2.616667) + 0.3(0)}{7} = -2.794524$$

the first iteration is completed by substituting the calculated values for x1 and x2 into eq. 3 to yield

$$x_3 = \frac{71.4 - 0.3(2.616667) + 0.2(-2.794524)}{10} = 7.005610$$

For the second iteration the same process is repeated to compute

$$x_1 = \frac{7.85 + 0.1(-2.794524) + 0.2(7.005610)}{3} = 2.990557$$
$$x_2 = \frac{-19.3 - 0.1(2.990557) + 0.3(7.005610)}{7} = -2.499625$$
$$x_3 = \frac{71.4 - 0.3(2.990557) + 0.2(-2.499625)}{10} = 7.000291$$

this method is, therefore, converging on the true solution.

MATLAB Code

```
function x = GaussSeidel(A,b,es,maxit)
%GaussSeidel: GaussSeidel Method
%x=GaussSeidel(A,b)
%input:
%     A = Coefficient matrix
%     b = right handside vector
%    es = stop criterion (default = 0.00001%)
% maxit = max iterations (default = 50)
%output:
% x = solution vector
ifnargin<2, error('at least 2 input arguments required'),end
ifnargin<4|isempty(maxit),maxit=50;end
ifnargin<3|isempty(es), es=0.00001;end
[m,n]=size(A);
if m~=n, error('Matrix A must be square');end
C = A;
fori = 1:n
C(i,i) = 0;
x(i) = 0;
end
x=x';
fori=1:n
C(i,1:n)=C(i,1:n)/A(i,i);
end
fori = 1:n
d(i) = b(i)/A(i,i);
end
iter = 0;
while(1)
xold=x;
fori = 1:n
x(i)=d(i)-C(i,:)*x;
if x(i)~=0
ea(i) = abs((x(i) - xold(i))/x(i)) * 100;
end
end
iter=iter+1;
if max(ea)<=es | iter>=maxit, break, end
end
```

Solution

```
>> A=[3 -0.1 -0.2; 0.1 7 -0.3; 0.3 -0.2 10]

A =

    3.0000   -0.1000   -0.2000

    0.1000    7.0000   -0.3000

    0.3000   -0.2000   10.0000

>> b=[7.85 -19.5 71.4]'
```

```
b =

    7.8500

  -19.5000

   71.4000

>> GaussSeidel(A,b)

ans =

    2.9990

   -2.5286

    6.9995
```

Exercise 2: Gauss Jacobi Method

```
function x = JACOBI(A,b,x,M,tau,lambda)
% Jacobi solves Ax=b using Jacobi iteration
% x = JACOBI(A,b,x,M,tau,lambda);
%
% Inputs:
% A = nxn matrix
% b = nxm matrix
%[x]= nxm for starting values. Default:b
%[M]= maximum number of iterations
%[tau]= Convergence tolerance
%Output
% x = nxm matrix of approximate solution to Ax=b
if nargin<3 |isempty(x), x=b; end
if nargin<4 |isempty(M), M=1000; end
if nargin<5 |isempty(tau), tau = sqrt(eps);end
d=diag(A);
r0=max(abs(b-A*x));
for i = 1:M
    r=b-A*x; %residual
    dx=r./d; %new diresction
    x=x+dx; %new value
    %Convergance check
    if all (abs(r)<tau) & all(abs(dx)<tau),
      return
    end
    %Divergence check
    if max (abs(r))>100*r0;
        disp('WAENING: Iterations Diverging');
        return
    end
end
if i==M, disp('WARNING: Maximum iterations exceeded');
end

>> A=[3 -0.1 -0.2; 0.1 7 -0.3; 0.3 -0.2 10]

A =
```

```
    3.0000   -0.1000   -0.2000

    0.1000    7.0000   -0.3000

    0.3000   -0.2000   10.0000

>> b=[7.85 -19.5 71.4]'

b =

    7.8500

  -19.5000

   71.4000

JACOBI(A,b)

ans =

    2.9990

   -2.5286

    6.9995
```