**Experiment No. 05**

* 1. **Experiment Name**

Generate an algorithm and write a program on load flow study of a given power system using Gauss-Seidel method

* 1. **Objectives**
* To become acquainted with the load flow study of a given power system
* To learn how to generate a MATLAB code for numerical analysis using Gauss-Seidel method
* To get familiar with the procedure of designing and analyzing a power system in MATLAB
  1. **Theory**

A load flow study is a numerical analysis of the flow of electric power in any electrical system. Its goal is to determine the flow, current, voltage, real power, and reactive power in a system under any load conditions.

A single-phase model is used to solve a load flow problem because the system is assumed to be operating under balanced conditions. Each bus is associated with four quantities. These are voltage magnitudes phase angle , real power ***P*** and reactive power ***Q***.

For a power system with n nodes the network equation can be given by the matrix equation

The current injected into the ***ith*** node can be obtained as

The power injected into the ***ith*** node can be written as

**= + =**

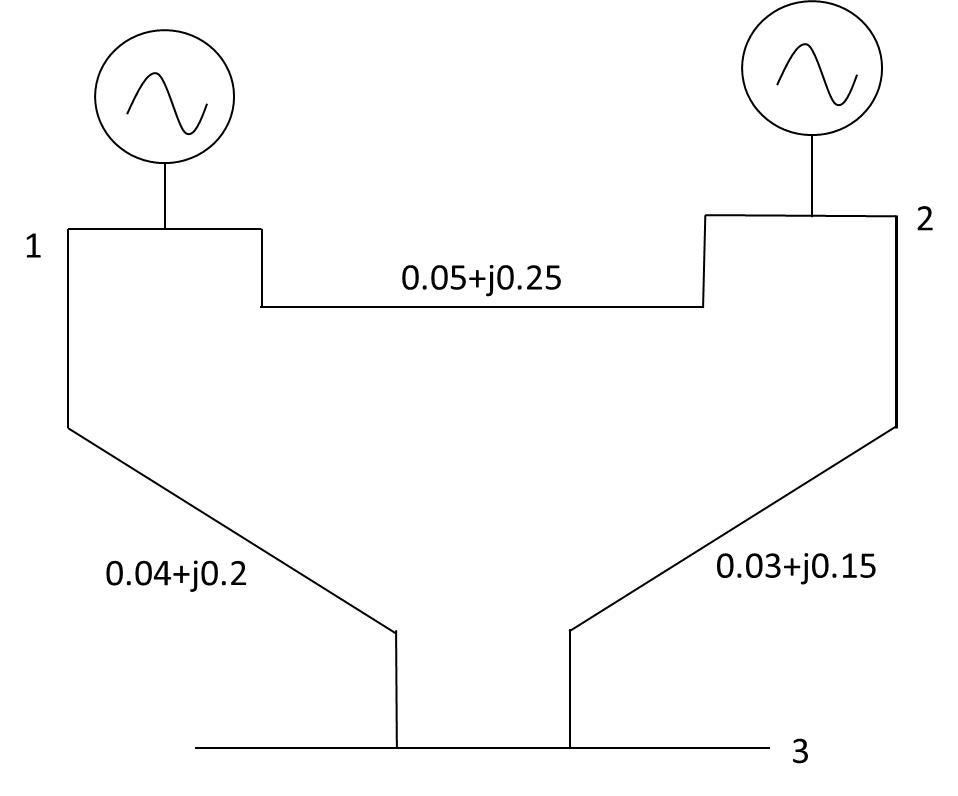
Hence, the real power,  ***=* Re()=**

The reactive power, = **Im()** =

The node voltage of a system can be found using the following equation

**= [ - ]** [for ***i***= 1,2,3…...***n***]

* 1. **Required apparatus**
* MATLAB
  1. **Block diagram**

****

*Fig. 5.1: Diagram of a three- bus system*

* 1. **Data table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Bus** | **Bus** | **R** | **X** |
| 1 | 1 | 0 | 1 |
| 1 | 2 | 0.05 | 0.25 |
| 1 | 3 | 0.04 | 0.02 |
| 2 | 2 | 0 | 1 |
| 2 | 3 | 0.03 | 0.15 |

Fig. 5.1 Excel file of the Impedance data

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Bus** | **|v|** | **Pg** | **Qg** | **Pl** | **Ql** | **Angle** |
| 1 | 1.03 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1.02 | 0.8 | 0 | 0.4 | 0.3 | 0 |
| 3 | 1 | 0 | 0 | 1 | 0.8 | 0 |

Fig. 5.2 Excel file of the Load data

* 1. **Algorithm**

1. Start
2. Read an excel file for taking input data and locate it
3. Generate a Y-bus matrix
4. Read node data. For P-V nodes take the starting values of the voltages as

***= ∠*** fork=1,2,3…...m***.***

1. For load nodes take the starting values as  **= 1.0∠** for k= m+1, m+2…...
2. Start iteration count ***i*** = 0 and node number ***k*** = 2
3. If ***k***>***m***, proceed to step 10, else move on to 9
4. As this is a P-V bus bar, compute the reactive power injection as follows

**= Im() = Im [+ ]**

1. Calculate **= +j**
2. Calculate the new value of voltage as follows

**= [ - - ]**

1. If ***k***>***m,*** proceed to step14, else move to step 13
2. For P-V node and has already calculated, determine the phase angle from the value of obtained in the above equation  **= ()** and compute the voltage at this node as  **= ∠**
3. If ***k***<***n*** go to step 15 otherwise go to step 16
4. Take ***k*** = ***K***+**1** and go to step 8
5. If for all ***k*** = 2,3,4…. ***n*** is within tolerance limit, proceed to step 18, else go to step 17.
6. Take ***i*** = ***i***+1 and go to step 7.
7. If break
8. Display output
9. End
   1. **Flow chart**

No

Yes

Compute reactive power

End

Read an excel file with impedance values

Generate a Y-bus matrix

Read an excel file with node data

Take initial voltage for load nodes

Enter bus data in excel file

Display ***Y*** as output

Start

Determine phase angle for that node voltage

If k>m

Take iteration count i=0 to n and node no. k=2 to m

Calculate **= +j**

Take new initial voltage for load nodes

* 1. **MATLAB Code & Output**

clc; %Clears previous data from command window

clear all; %Removes all variables from the current workspace

cd('F:\Study material\Lab\3-2\Power System I'); %change the file directory

A = xlsread('EXp02p02'); %Read the excel file

n = length(A); %Determine the length of the excel file

% Applying symmetric condition

for w=1:n

Z(A(w,1),A(w,2)) = A(w,3)+i\*A(w,4);

Z(A(w,2),A(w,1)) = A(w,3)+i\*A(w,4);

end

m = length(Z); %Determine the length of the new matrix

for j=1:m

for k=1:m

if Z(j,k) == 0

Z(j,k) = inf;

end

end

end

fprintf(' Z matrix is \n') %Display the text

disp(Z) %Display the output

y = 1./Z %Taking inverse impedance matrix

p = sum(y,2) %Taking symmetric summation

%Apply looping condition to determine value of the matrix element

for u=1:m

for x=1:m

if u~=x

Y(u,x)= -y(u,x); %For diagonal element

else

Y(u,x)= p(x); %For non-diagonal element

end

end

end

fprintf(' Y- bus matrix is \n') %Display the text

disp(Y) %Display the output

cd('F:\Study material\Lab\3-2\Power System I'); %change the file directory

B = xlsread('Exp05'); %Read the excel file

j = 3;

V = B(:,2);

V0=B(:,2);

%to get the value of real power

P=B(:,3)-B(:,5);

%to get the value of reactive power

Q=B(:,4)-B(:,6);

%to get the angle

ang=B(:,7);

V1=V;

% to get the value of generator bus

Pg=B(:,3);

for w=1:100

z=V;

for k=2:j

yv1=0;

yv2=0;

for h=1:j

yv2=yv2+Y(k,h)\*V(h); %to find the product of Y bus and voltages

if h~=k

yv1=yv1+Y(k,h)\*V(h); %to find the product of Y bus and voltages

end

end

if Pg(k)~=0

g(k)=imag(V(k)\*(conj(yv2))); %to get the imaginary value

S(k)=P(k)+1i\*g(k); %to calculate the apparent power

else S(k)=P(k)+1i\*Q(k);

end

V(k)=(1/Y(k,k))\*((conj(S(k))/conj(V(k)))-yv1); %to get the value of node voltages

ang1(k)=angle(V(k)); %to get the angles

ang2(k)=rad2deg(ang1(k)); %to convert the radian values to degrees

if Pg(k)~=0

V(k)=V0(k)\*exp(1i\*ang1(k));

end

end

V1=abs(V);

ang2=rad2deg(ang1);

E=abs((V-z)/V);

if E<=10e-4

break; %to break the for loop

end

Vlt\_1(w)=V1(1);

Vlt\_2(w)=V1(2);

Vlt\_3(w)=V1(3);

ang\_1(w)=ang2(1);

ang\_2(w)=ang2(2);

ang\_3(w)=ang2(3);

end

% to show the value column wise

Vlt\_1=Vlt\_1';

ang\_1=ang\_1';

Vlt\_2=Vlt\_2';

ang\_2=ang\_2';

Vlt\_3=Vlt\_3';

ang\_3=ang\_3';

iteration=(1:w-1)';

% to show the values in a table

table(iteration,Vlt\_1,ang\_1,Vlt\_2,ang\_2,Vlt\_3,ang\_3)

**Output**

Z matrix is

0.0000 + 1.0000i 0.0500 + 0.2500i 0.0400 + 0.0200i

0.0500 + 0.2500i 0.0000 + 1.0000i 0.0300 + 0.1500i

0.0400 + 0.0200i 0.0300 + 0.1500i Inf + 0.0000i

y =

0.0000 - 1.0000i 0.7692 - 3.8462i 20.0000 -10.0000i

0.7692 - 3.8462i 0.0000 - 1.0000i 1.2821 - 6.4103i

20.0000 -10.0000i 1.2821 - 6.4103i 0.0000 + 0.0000i

p =

20.7692 -14.8462i

2.0513 -11.2564i

21.2821 -16.4103i

Y- bus matrix is

20.7692 -14.8462i -0.7692 + 3.8462i -20.0000 +10.0000i

-0.7692 + 3.8462i 2.0513 -11.2564i -1.2821 + 6.4103i

-20.0000 +10.0000i -1.2821 + 6.4103i 21.2821 -16.4103i

ans =

3×7 table

iteration Vlt\_1 ang\_1 Vlt\_2 ang\_2 Vlt\_3 ang\_3

\_\_\_\_\_\_\_\_\_ \_\_\_\_\_ \_\_\_\_\_ \_\_\_\_\_ \_\_\_\_\_\_ \_\_\_\_\_\_\_ \_\_\_\_\_\_\_

1 1.03 0 1.02 1.7965 0.98557 0.48806

2 1.03 0 1.02 2.2139 0.98602 0.54621

3 1.03 0 1.02 2.3017 0.98628 0.56041

* 1. **Discussion & Conclusion**

In this experiment, we designed an algorithm, flow chart, and programmed a generalized code for load flow study of a given power system. The sinusoidal steady state of the entire system is provided by the load flow.

It is critical for evaluating the best operating of the existing system and planning future system expansion. It aids in the design of a new power system network, the reduction of system loss, and the selection of transformer taps for efficient operation.

The Gauss-Seidel method is utilized for doing load flow analysis because of these consequences. Though this method takes longer to converge than others, its advantage is its simplicity and ease of performance.

The only adjustment to the code we may need is changing the directory of the file to work with and the given data saved inside the file. The bus numbers and the resistance and reactance values must also be given in the order defined for the code to work and give accurate result.