**Experiment No. 03**

* 1. **Experiment Name**

Generate an algorithm and write a program to calculate the Y-bus matrix of a given power system

* 1. **Objectives**
* To become acquainted with the Y-bus matrix of a given power system
* To learn how to generate a MATLAB code for a Y-bus matrix of a given power system
* To get familiar with the procedure of designing and analyzing a power system in MATLAB
  1. **Apparatus**
* MATLAB
  1. **Theory**

The Y-bus matrix is a N × N matrix used in power engineering to describe a power system with N buses. It reflects the buses' nodal admittance in a power system. The Y matrix is relatively sparse in realistic systems with thousands of buses.

In a practical power system, each bus is often connected to only a few other buses via transmission lines. The Y Matrix is also one of the data requirements needed to formulate a power flow study.

One of the strategies developed to study power systems is the Node voltage method. In terms of load currents, the equations in the nodal admittance form result in a simultaneous complex algebraic equation. The voltages and currents of the buses are obtained by solving these equations. The power and load flow on all buses in the system may then be determined. That is why the construction of the admittance bus matrix Y-bus is very important.

If *Ibus* is the bus currents in matrix form and *Ybus* is the admittance matrix and V bus is the bus voltages in matrix form then they can be related as 𝐼𝑏𝑢𝑠=𝑌𝑏𝑢𝑠𝑉𝑏𝑢𝑠

The general mathematical expressions are

𝑌𝑖𝑗=−𝑦𝑖𝑗 ; if i≠j

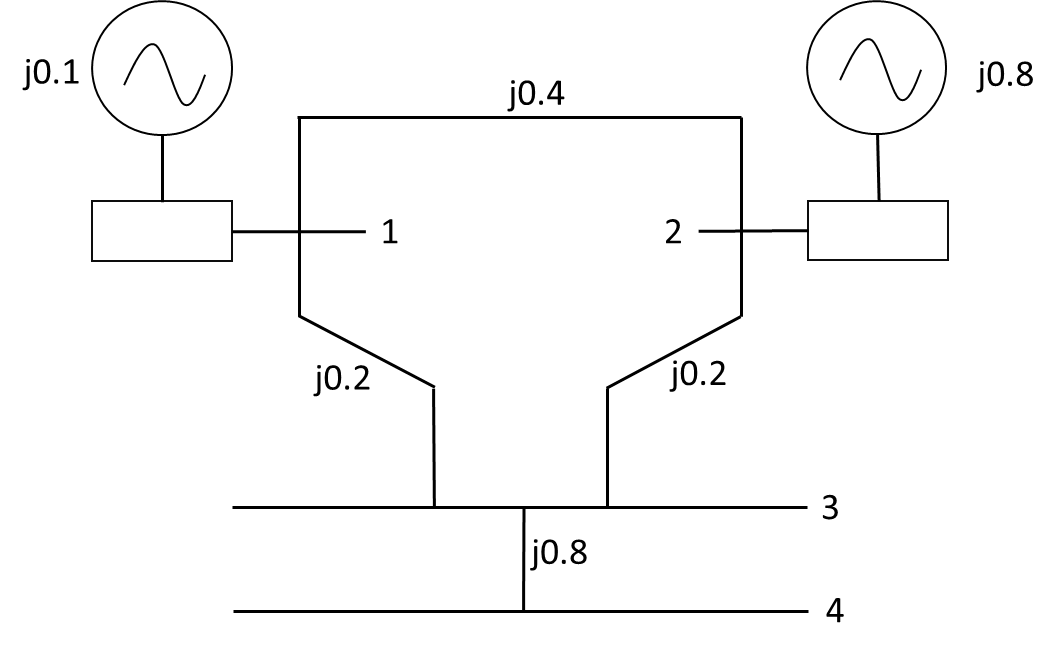
𝑌𝑖𝑗= ; if i=j

The nodal admittance matrix form

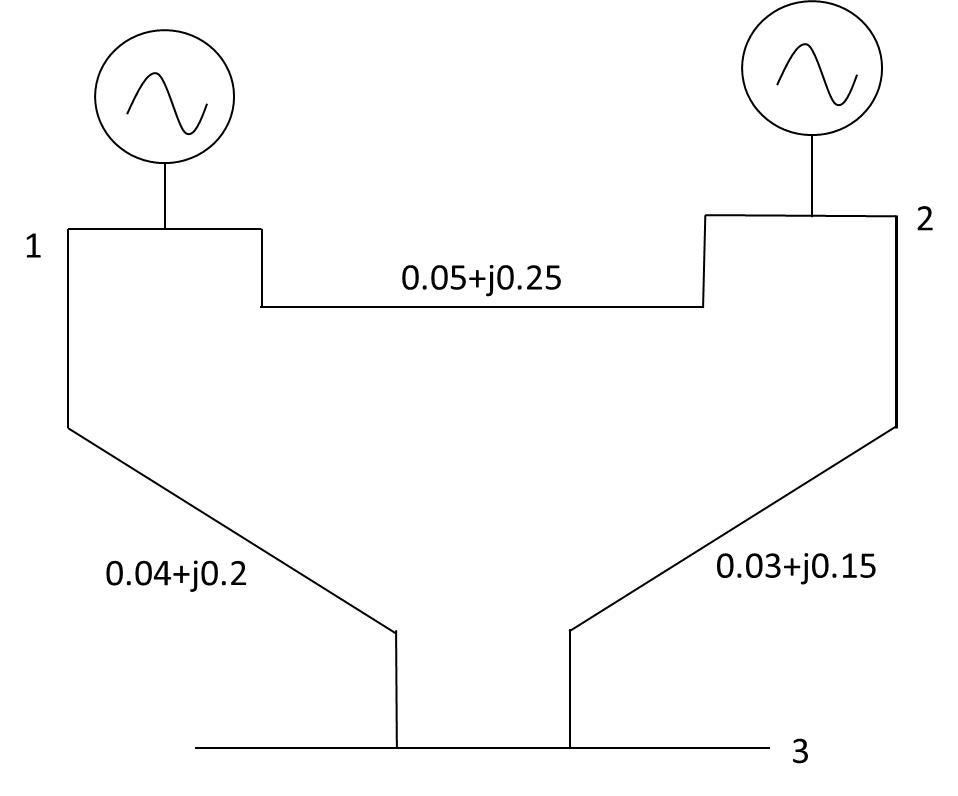
Before constructing the equations that comprise the Y Matrix, three major steps must be taken beginning with a single line diagram of a power system. The single line diagram is first transformed into an impedance diagram. Following that, all voltage sources are translated to their corresponding current sources. The impedance diagram is then transformed into an admittance diagram.

The bus impedances can be recorded in an excel file for better management. The excel file can then be put into MATLAB to create the admittance matrix.

* 1. **Block diagram**

****

*Fig 3.1: Diagram of Y- bus no.1*

****

*Fig 3.1: Diagram of Y- bus no.2*

* 1. **Algorithm**

1. Start
2. Read an excel file that has bus numbers ***i*** and ***j*** in the first two columns and resistance and admittance value in the third and fourth columns which represent impedance between the buses
3. Construct a matrix ***A*** whose element ***ai,j***denotes the impedance between ***i*** and ***j*** buses
4. Determine the length of matrix ***A***
5. Perform symmetric condition and construct a new matrix ***Z*** where 3rd and 4th column element of the excel file is equivalent to ***a+bj*** and it can be assigned to ***Zi,j***  later applying symmetric condition ***Zi,j*** = ***Zj,i***
6. Determine the length of new matrix ***Z***
7. Use looping condition for row j from 1 to length of Z and for column k from1 to length Z
8. If value of ***j*** row and ***k*** column is equal to zero, assign the element value as infinity
9. Determine inverse matrix of ***Z*** and assigned to ***Z***
10. Calculate the summation of ***Y*** matrix element row wise
11. Use looping condition for row u from1 to length of ***Z*** and for column x from1 to length ***Z***
12. Use formula if ***u≠x*** 𝑌***ux***= ***−𝑦ux*,** else 𝑌***ux*** =
13. Display ***Y*** as output
14. End
    1. **Flow chart**

**Yes**

***Y(u,x)= sum of entries element wise and they are the value of diagonal elements***

Read an excel file

Construct impedance matrix ***A***

Determining length of ***A*** matrix and n=length

Running loop for row ***w*** from 1 to ***n*** in symmetric condition

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

Z(A(w,2),A(w,1)) = Z(A(w,1),A(w,2))

Determining length of ***Z*** matrix and m= length

Running loop for row ***j*** from 1 to ***m*** and column ***k*** from 1 to ***m***

If ***Z(j,k)***==0

Assigning infinity to ***Z(j,k)***

Enter bus data in excel file

Display ***Z*** as output

Assigning inverse of ***Z(j,k)*** to ***y*** variable

If ***u=x***

Running loop for row ***u*** from 1 to ***m*** and column ***x*** from 1 to ***m***

***Y(u,x)= -y(u,x)***

**No**

* 1. **MATLAB Code & Output**
* **Generalized MATLAB Code for Y bus no.1**

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('Exp02'); %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(u); %For non-diagonal element

end

end

end

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

disp(Y) %Display the output

* **Output**

m =

4

Z matrix is

0.0000 + 0.1000i 0.0000 + 0.4000i 0.0000 + 0.2000i Inf + 0.0000i

0.0000 + 0.4000i 0.0000 + 0.8000i 0.0000 + 0.2000i Inf + 0.0000i

0.0000 + 0.2000i 0.0000 + 0.2000i Inf + 0.0000i 0.0000 + 0.0800i

Inf + 0.0000i Inf + 0.0000i 0.0000 + 0.0800i Inf + 0.0000i

y =

0.0000 -10.0000i 0.0000 - 2.5000i 0.0000 - 5.0000i 0.0000 + 0.0000i

0.0000 - 2.5000i 0.0000 - 1.2500i 0.0000 - 5.0000i 0.0000 + 0.0000i

0.0000 - 5.0000i 0.0000 - 5.0000i 0.0000 + 0.0000i 0.0000 -12.5000i

0.0000 + 0.0000i 0.0000 + 0.0000i 0.0000 -12.5000i 0.0000 + 0.0000i

p =

0.0000 -17.5000i

0.0000 - 8.7500i

0.0000 -22.5000i

0.0000 -12.5000i

Y- bus matrix is

0.0000 -17.5000i 0.0000 + 2.5000i 0.0000 + 5.0000i 0.0000 + 0.0000i

0.0000 + 2.5000i 0.0000 - 8.7500i 0.0000 + 5.0000i 0.0000 + 0.0000i

0.0000 + 5.0000i 0.0000 + 5.0000i 0.0000 -22.5000i 0.0000 +12.5000i

0.0000 + 0.0000i 0.0000 + 0.0000i 0.0000 +12.5000i 0.0000 -12.5000i

* **Generalized MATLAB Code for Y bus no.2**

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('Exp02'); %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(u); %For non-diagonal element

end

end

end

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

disp(Y) %Display the output

* **Output**

m =

3

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

* 1. **Discussion & Conclusion**

In this experiment, we designed an algorithm, flow chart, and programmed a generalized code for two different Y- bus system. In each case, we extracted the values from an excel file and formulated necessary condition to assign values to variables. Through this generalized coding format, we easily design and calculate Y bus matrix for any given power system.

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.