**EE 523 | Spring 2023**

**Assignment 01**

**Submitted by**

**Sajjad Uddin Mahmud**

**WSU ID: 011789534**

**Logo, company name

Description automatically generated**

Code:

# Platform: MATLAB

## Main Code :

%% EE 523 Assignment 01 - Sajjad Uddin Mahmud - Spring 2023 - WSU

%% Basic Initialization

clc;

clear all;

close all;

%% Adding Folder Path

Data\_Path = [fileparts(pwd),'/Data'];

addpath(Data\_Path);

Function\_Path = [fileparts(pwd),'/Functions'];

addpath(Function\_Path);

%% Setting Up The Input Data As Per Assignment

Problem = 'A';

if Problem == 'A'

Excel\_Worksheet = 'Problem\_A';

elseif Problem == 'B'

Excel\_Worksheet = 'Problem\_B';

elseif Problem == 'C'

Excel\_Worksheet = 'Problem\_C';

end

%% Reading From Bus Data

%% Bus Number

All\_Bus\_Number = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'A4:A14'); % Reading All Bus ID Data

Total\_Bus = length(All\_Bus\_Number); % Calculating Total Bus Number

%% Bus Type

All\_Bus\_Type = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'G4:G14'); % Reading All Bus Type Data

PQ\_Bus\_Type = 0;

PQ\_Bus\_Type\_1 = 1;

PV\_Bus\_Type = 2;

Slack\_Bus\_Type = 3;

Bus = All\_Bus\_Number(find(All\_Bus\_Type ~= Slack\_Bus\_Type)); % Bus Type Data Except the Slack Bus

Bus\_Type = All\_Bus\_Type(Bus); % Bus Type Data Except the Slack Bus

%% Bus Information

% Slack\_Bus\_Number = 1

Base\_MVA = 100;

V\_Desired = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'O4:O14'); % Given Desired Voltage

Delta\_in\_Rad = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'I4:I14'); % Given Voltage Angle

P\_Load = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'J4:J14')/Base\_MVA; % Load MW pu

Q\_Load = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'K4:K14')/Base\_MVA; % Load MVAR pu

P\_Gen = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'L4:L14')/Base\_MVA; % Generator MW pu

Q\_Gen = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'M4:M14')/Base\_MVA; % Generator MVAR pu

%% Reading from Branch Data

%% Branch Number

From\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'A18:A27');

To\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'B18:B27');

%% Bus Shunt Conductance and Shunt Susceptance

G\_Shunt\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'R4:R14');

B\_Shunt\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'S4:S14');

%% Calculating Bus Shunt Admittance

Y\_Shunt\_Bus = G\_Shunt\_Bus + j.\*B\_Shunt\_Bus;

%% Branch Resistance Per Unit

R\_Branch = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'G18:G27');

%% Branch Reactance Per Unit

X\_Branch = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'H18:H27');

%% Line Charging B Per Unit

B\_Branch = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'I18:I27');

%% Transformer Turns Ratio

XFR\_TurnRatio = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'O18:O27');

%% Calculating Branch Impedence and Admittance

for i=1:length(From\_Bus)

Z\_Branch(i) = R\_Branch(i) + j \* X\_Branch(i); % Per Unit Impedance

Y\_Branch(i) = 1 / Z\_Branch(i); % Per Unit Admittance

end

%% Tap Consideration

Tap\_Consideration = 1; % 0 = Without Taps, 1 = With Taps

if (Tap\_Consideration == 0)

for i = 1:length(XFR\_TurnRatio)

XFR\_TurnRatio(i) = 0; % If We Do Not Consider Tap, All the Turn Ratio of Transformer are 0

end

end

%% Calculating Y Bus Matrix:

% Initialization

Y\_Bus = zeros(Total\_Bus,Total\_Bus);

% LOOP: Computing Off-Diagonal Elements

for i=1:length(Y\_Branch)

if (XFR\_TurnRatio(i)==0)

Y\_Bus(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i);

Y\_Bus(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i);

Y\_Bus\_Diag(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i);

Y\_Bus\_Diag(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i);

else

T = (1/(XFR\_TurnRatio(i)));

Y\_Bus(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i) \* (T);

Y\_Bus(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i) \* (T);

Y\_Bus\_Diag(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i);

Y\_Bus\_Diag(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i) \* (T^2);

end

end

% LOOP: Computing Diagonal Elements

Y\_Bus\_Sum = sum(Y\_Bus\_Diag);

for i=1:Total\_Bus

Y\_Bus(i,i) = -Y\_Bus\_Sum(i) + Y\_Shunt\_Bus(i); % Adding Shunt Capacitance

end

% LOOP: Adding Line Charaging Capacitance

for i=1:length(From\_Bus)

Y\_Bus(From\_Bus(i),From\_Bus(i)) = Y\_Bus(From\_Bus(i),From\_Bus(i)) + j \* (B\_Branch(i) / 2);

Y\_Bus(To\_Bus(i),To\_Bus(i)) = Y\_Bus(To\_Bus(i),To\_Bus(i)) + j \* (B\_Branch(i) / 2);

end

% Converting Y Bus Data into Polar Form

Rho = abs(Y\_Bus); % Magnitude of Y Bus Entries

Theta = angle(Y\_Bus); % Angle of Y Bus Entries in radian

B = imag(Y\_Bus); % Imaginary Part of Y Bus Entries

G = real(Y\_Bus); % Real Part of Y Bus Entries

% End of Y Bus Formation. Y Bus is Ready

%% Power Flow

%% Method

Task = 1; % Newton-Raphson = 1; Fast Decoupled = 2

%% Power Flow - Newton-Raphson Method

if Task == 1

[V, Delta\_in\_Rad, Iteration] = Newton\_Raphson\_Function(Y\_Bus, V\_Desired, Delta\_in\_Rad, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type);

%% Power Flow - Fast Decoupled

elseif Task == 2

[V\_FD, Delta\_in\_Rad\_FD, Iteration\_FD] = Newton\_Raphson\_Function\_1(Y\_Bus, V\_Desired, Delta\_in\_Rad, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type); % Putting Values after 4 Iterations of NR as Input for Fast Decoupled

V\_FD = transpose(V\_FD);

Delta\_in\_Rad\_FD = transpose(Delta\_in\_Rad\_FD);

[V, Delta\_in\_Rad, Iteration] = Fast\_Decoupled\_Function(Y\_Bus, V\_FD, Delta\_in\_Rad\_FD, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type);

end

## Function - Newton-Raphson:

%% EE 523 Assignment 01 - Sajjad Uddin Mahmud - Spring 2023 - WSU

%% Basic Initialization

clc;

clear all;

close all;

%% Setting Up The Input Data As Per Assignment

Problem = 'A';

if Problem == 'A'

Excel\_Worksheet = 'Problem\_A';

elseif Problem == 'B'

Excel\_Worksheet = 'Problem\_B';

elseif Problem == 'C'

Excel\_Worksheet = 'Problem\_C';

end

%% Reading From Bus Data

%% Bus Number

All\_Bus\_Number = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'A4:A14'); % Reading All Bus ID Data

Total\_Bus = length(All\_Bus\_Number); % Calculating Total Bus Number

%% Bus Type

All\_Bus\_Type = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'G4:G14'); % Reading All Bus Type Data

PQ\_Bus\_Type = 0;

PQ\_Bus\_Type\_1 = 1;

PV\_Bus\_Type = 2;

Slack\_Bus\_Type = 3;

Bus = All\_Bus\_Number(find(All\_Bus\_Type ~= Slack\_Bus\_Type)); % Bus Type Data Except the Slack Bus

Bus\_Type = All\_Bus\_Type(Bus); % Bus Type Data Except the Slack Bus

%% Bus Information

% Slack\_Bus\_Number = 1

Base\_MVA = 100;

V\_Desired = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'O4:O14'); % Given Desired Voltage

Delta\_in\_Rad = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'I4:I14'); % Given Voltage Angle

P\_Load = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'J4:J14')/Base\_MVA; % Load MW pu

Q\_Load = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'K4:K14')/Base\_MVA; % Load MVAR pu

P\_Gen = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'L4:L14')/Base\_MVA; % Generator MW pu

Q\_Gen = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'M4:M14')/Base\_MVA; % Generator MVAR pu

%% Reading from Branch Data

%% Branch Number

From\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'A18:A27');

To\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'B18:B27');

%% Bus Shunt Conductance and Shunt Susceptance

G\_Shunt\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'R4:R14');

B\_Shunt\_Bus = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'S4:S14');

%% Calculating Bus Shunt Admittance

Y\_Shunt\_Bus = G\_Shunt\_Bus + j.\*B\_Shunt\_Bus;

%% Branch Resistance Per Unit

R\_Branch = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'G18:G27');

%% Branch Reactance Per Unit

X\_Branch = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'H18:H27');

%% Line Charging B Per Unit

B\_Branch = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'I18:I27');

%% Transformer Turns Ratio

XFR\_TurnRatio = xlsread('Kundur\_11Bus\_System.xlsx',Excel\_Worksheet,'O18:O27');

%% Calculating Branch Impedence and Admittance

for i=1:length(From\_Bus)

Z\_Branch(i) = R\_Branch(i) + j \* X\_Branch(i); % Per Unit Impedance

Y\_Branch(i) = 1 / Z\_Branch(i); % Per Unit Admittance

end

%% Tap Consideration

Tap\_Consideration = 1; % 0 = Without Taps, 1 = With Taps

if (Tap\_Consideration == 0)

for i = 1:length(XFR\_TurnRatio)

XFR\_TurnRatio(i) = 0; % If We Do Not Consider Tap, All the Turn Ratio of Transformer are 0

end

end

%% Calculating Y Bus Matrix:

% Initialization

Y\_Bus = zeros(Total\_Bus,Total\_Bus);

% LOOP: Computing Off-Diagonal Elements

for i=1:length(Y\_Branch)

if (XFR\_TurnRatio(i)==0)

Y\_Bus(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i);

Y\_Bus(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i);

Y\_Bus\_Diag(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i);

Y\_Bus\_Diag(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i);

else

T = (1/(XFR\_TurnRatio(i)));

Y\_Bus(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i) \* (T);

Y\_Bus(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i) \* (T);

Y\_Bus\_Diag(From\_Bus(i),To\_Bus(i)) = - Y\_Branch(i);

Y\_Bus\_Diag(To\_Bus(i),From\_Bus(i)) = - Y\_Branch(i) \* (T^2);

end

end

% LOOP: Computing Diagonal Elements

Y\_Bus\_Sum = sum(Y\_Bus\_Diag);

for i=1:Total\_Bus

Y\_Bus(i,i) = -Y\_Bus\_Sum(i) + Y\_Shunt\_Bus(i); % Adding Shunt Capacitance

end

% LOOP: Adding Line Charaging Capacitance

for i=1:length(From\_Bus)

Y\_Bus(From\_Bus(i),From\_Bus(i)) = Y\_Bus(From\_Bus(i),From\_Bus(i)) + j \* (B\_Branch(i) / 2);

Y\_Bus(To\_Bus(i),To\_Bus(i)) = Y\_Bus(To\_Bus(i),To\_Bus(i)) + j \* (B\_Branch(i) / 2);

end

% Converting Y Bus Data into Polar Form

Rho = abs(Y\_Bus); % Magnitude of Y Bus Entries

Theta = angle(Y\_Bus); % Angle of Y Bus Entries in radian

B = imag(Y\_Bus); % Imaginary Part of Y Bus Entries

G = real(Y\_Bus); % Real Part of Y Bus Entries

% End of Y Bus Formation. Y Bus is Ready

%% Power Flow

%% Method

Task = 1; % Newton-Raphson = 1; Fast Decoupled = 2

%% Power Flow - Newton-Raphson Method

if Task == 1

[V, Delta\_in\_Rad, Iteration] = Newton\_Raphson\_Function(Y\_Bus, V\_Desired, Delta\_in\_Rad, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type);

%% Power Flow - Fast Decoupled

elseif Task == 2

[V\_FD, Delta\_in\_Rad\_FD, Iteration\_FD] = Newton\_Raphson\_Function\_1(Y\_Bus, V\_Desired, Delta\_in\_Rad, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type); % Putting Values after 4 Iterations of NR as Input for Fast Decoupled

V\_FD = transpose(V\_FD);

Delta\_in\_Rad\_FD = transpose(Delta\_in\_Rad\_FD);

[V, Delta\_in\_Rad, Iteration] = Fast\_Decoupled\_Function(Y\_Bus, V\_FD, Delta\_in\_Rad\_FD, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type);

end

## Function - Newton-Raphson (For Getting Input of Fast Decoupled):

%% Power Flow Function: Newton-Raphson

function [V, Delta\_in\_Rad, Iteration] = Newton\_Raphson\_Function\_1(Y\_Bus, V\_Desired, Delta\_in\_Rad, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type)

%% Basic Initialization

Total\_Bus = length(All\_Bus\_Number);

PQ\_Bus\_Type = 0;

PQ\_Bus\_Type\_1 = 1;

PV\_Bus\_Type = 2;

Slack\_Bus\_Type = 3;

Bus = All\_Bus\_Number(find(All\_Bus\_Type ~= Slack\_Bus\_Type)); % Bus Type Data Except the Slack Bus

Bus\_Type = All\_Bus\_Type(Bus); % Bus Type Data Except the Slack Bus

Base\_MVA = 100;

%% Power Flow

% Schedule Real and Reactive Power

P\_Scheduled = transpose(P\_Gen - P\_Load);

Q\_Scheduled = transpose(Q\_Gen - Q\_Load);

% Initial Voltage Magnitude

V = ones(1,length(All\_Bus\_Number));

V(1,find(V\_Desired)) = V\_Desired(find(V\_Desired),1);

% Initialization

Iteration = 0;

Tolerance = 0.01;

while 1

%% Calculating Real Power

% Initialization

P\_Calculated = zeros(1,Total\_Bus);

% LOOP: Computing Real Power

for i=1:Total\_Bus

for n=1:Total\_Bus

P\_Calculated(i) = P\_Calculated(i) + (abs(abs(Y\_Bus(i,n)) \* V(i) \* V(n))) \* (cos(angle(Y\_Bus(i,n)) + Delta\_in\_Rad(n) - Delta\_in\_Rad(i)));

end

end

%% Calculating Reactive Power

% Initialization

Q\_Calculated = zeros(1,Total\_Bus);

% LOOP: Computing Reactive Power

for i=1:Total\_Bus

for n=1:Total\_Bus

Q\_Calculated(i) = Q\_Calculated(i) + (abs(abs(Y\_Bus(i,n)) \* V(i) \* V(n))) \* (sin(angle(Y\_Bus(i,n)) + Delta\_in\_Rad(n) - Delta\_in\_Rad(i)));

end

Q\_Calculated(i) = - Q\_Calculated(i);

end

%% Calculating Mismatch

Delta\_P = P\_Scheduled - P\_Calculated;

Delta\_Q = Q\_Scheduled - Q\_Calculated;

% Initializating Jacobian Matrix

J11 = zeros(length(Bus));

J12 = zeros(length(Bus));

J21 = zeros(length(Bus));

J22 = zeros(length(Bus));

% LOOP: Computing Jacobian Matrix for All the Buses Except Slack Bus

for i=1:length(Bus)

for j=1:length(Bus)

if (i==j)

J11(i,j) = - Q\_Calculated(Bus(i)) - ((V(Bus(i)))^2) \* (imag(Y\_Bus(Bus(i),Bus(i))));

J21(i,j) = P\_Calculated(Bus(i)) - ((V(Bus(i)))^2) \* (real(Y\_Bus(Bus(i),Bus(i))));

J12(i,j) = P\_Calculated(Bus(i)) + ((V(Bus(i)))^2) \* (real(Y\_Bus(Bus(i),Bus(i))));

J22(i,j) = Q\_Calculated(Bus(i)) - ((V(Bus(i)))^2) \* (imag(Y\_Bus(Bus(i),Bus(i))));

else

J11(i,j) = - abs(V(Bus(i)) \* V(Bus(j)) \* abs(Y\_Bus(Bus(i),Bus(j)))) \* sin(angle(Y\_Bus(Bus(i),Bus(j))) + Delta\_in\_Rad(Bus(j)) - Delta\_in\_Rad(Bus(i)));

J21(i,j) = - abs(V(Bus(i)) \* V(Bus(j)) \* abs(Y\_Bus(Bus(i),Bus(j)))) \* cos(angle(Y\_Bus(Bus(i),Bus(j))) + Delta\_in\_Rad(Bus(j)) - Delta\_in\_Rad(Bus(i)));

J12(i,j) = - J21(i,j);

J22(i,j) = J11(i,j);

end

end

end

% Removing Rows and Columns from Jacobian for PV Bus

PV = find(Bus\_Type==PV\_Bus\_Type);

J12(:,PV) = [];

J21(PV,:) = [];

J22(:,PV) = [];

J22(PV,:) = [];

J = [J11 J12; J21 J22];

% Delta

Delta\_J = Delta\_in\_Rad(find(All\_Bus\_Type ~= Slack\_Bus\_Type));

V\_J = V(find((All\_Bus\_Type == PQ\_Bus\_Type)));

Delta\_P\_J = Delta\_P(find(All\_Bus\_Type ~= Slack\_Bus\_Type));

Delta\_Q\_J = Delta\_Q(find((All\_Bus\_Type == PQ\_Bus\_Type)));

Delta\_P\_Q = [transpose(Delta\_P\_J);transpose(Delta\_Q\_J)];

%% Updating V and Delta through LU Factorization

% Function Calling: LU Factorization Using Dolittle's Method

[V\_Delta\_Corrected] = LU\_Factorization\_Dolittle\_Function(J,Delta\_P\_Q);

% LOOP: Sorting the Voltages and Angles after LU Factorization

for i=1:length(V\_Delta\_Corrected)

if (i <= length(Delta\_P\_J))

Delta\_Corrected(i) = V\_Delta\_Corrected(i);

else

V\_Corrected(i-length(Delta\_P\_J)) = V\_Delta\_Corrected(i);

end

end

% Updating Voltages and Angles

Delta\_Updated = Delta\_J + Delta\_Corrected;

V\_Updated = V\_J .\* (1 + V\_Corrected);

% Preparing for Next Iteration

V\_i = (find((All\_Bus\_Type == PQ\_Bus\_Type)));

Delta\_i = find(All\_Bus\_Type ~= Slack\_Bus\_Type);

for i=1:length(Delta\_i)

Delta\_New(Delta\_i(i)) = Delta\_Updated(i);

end

for i=1:length(V\_i)

V\_Desired(V\_i(i)) = V\_Updated(i);

end

V = transpose(V\_Desired);

Delta\_in\_Rad = Delta\_New;

Delta\_in\_Degree = (180 / pi) \* Delta\_in\_Rad;

Iteration = Iteration + 1;

%% Output

%fprintf("YBus: \n")

%Y\_Bus

%fprintf("Number of Iteration: \n");

%Iteration

%fprintf("Voltage Magnitude: \n")

%V

%fprintf("Voltage Angles in Degree: \n")

%Delta\_in\_Degree

%fprintf("Real Power in MW: \n")

%P\_Calculated \* Base\_MVA

%fprintf("Reactive Power in MVAR: \n")

%Q\_Calculated \* Base\_MVA

% Checking Iteration Limit

if (Iteration == 4)

break;

end

end

% This does not converge; this is just to get a certain iterated values which are used as a input of Fast Decoupled Function to make that converge.

## Function - Fast Decoupled:

%% Power Flow Function: Fast Decoupled

function [V, Delta\_in\_Rad, Iteration] = Fast\_Decoupled\_Function(Y\_Bus, V\_Desired, Delta\_in\_Rad, P\_Gen, P\_Load, Q\_Gen, Q\_Load, All\_Bus\_Number, All\_Bus\_Type)

%% Basic Initialization

Total\_Bus = length(All\_Bus\_Number);

PQ\_Bus\_Type = 0;

PQ\_Bus\_Type\_1 = 1;

PV\_Bus\_Type = 2;

Slack\_Bus\_Type = 3;

Bus = All\_Bus\_Number(find(All\_Bus\_Type ~= Slack\_Bus\_Type)); % Bus Type Data Except the Slack Bus

Bus\_Type = All\_Bus\_Type(Bus); % Bus Type Data Except the Slack Bus

Base\_MVA = 100;

%% Power Flow

% Schedule Real and Reactive Power

P\_Scheduled = transpose(P\_Gen - P\_Load);

Q\_Scheduled = transpose(Q\_Gen - Q\_Load);

% Initial Voltage Magnitude

V = ones(1,length(All\_Bus\_Number));

V(1,find(V\_Desired)) = V\_Desired(find(V\_Desired),1);

% Initialization

Iteration = 0;

Tolerance = 0.01;

while 1

%% Calculating Real Power

% Initialization

P\_Calculated = zeros(1,Total\_Bus);

% LOOP: Computing Real Power

for i=1:Total\_Bus

for n=1:Total\_Bus

P\_Calculated(i) = P\_Calculated(i) + (abs(abs(Y\_Bus(i,n)) \* V(i) \* V(n))) \* (cos(angle(Y\_Bus(i,n)) + Delta\_in\_Rad(n) - Delta\_in\_Rad(i)));

end

end

%% Calculating Reactive Power

% Initialization

Q\_Calculated = zeros(1,Total\_Bus);

% LOOP: Computing Reactive Power

for i=1:Total\_Bus

for n=1:Total\_Bus

Q\_Calculated(i) = Q\_Calculated(i) + (abs(abs(Y\_Bus(i,n)) \* V(i) \* V(n))) \* (sin(angle(Y\_Bus(i,n)) + Delta\_in\_Rad(n) - Delta\_in\_Rad(i)));

end

Q\_Calculated(i) = - Q\_Calculated(i);

end

%% Calculating Mismatch

Delta\_P = P\_Scheduled - P\_Calculated;

Delta\_Q = Q\_Scheduled - Q\_Calculated;

% Initialization of Jacobian in Fast Decoupled Method; J12=J21=0

J11 = zeros(length(Bus));

J22 = zeros(length(Bus));

% LOOP: Computing Jacobian Matrix for All the Buses Except Slack Bus

for i=1:length(Bus)

for j=1:length(Bus)

J11(i,j) = - (imag(Y\_Bus(Bus(i),Bus(j))));

J22(i,j) = - (imag(Y\_Bus(Bus(i),Bus(j))));

end

end

% Removing Rows and Columns from Jacobian for PV Bus

PV = find(Bus\_Type == PV\_Bus\_Type);

J22(:,PV) = [];

J22(PV,:) = [];

% Delta

Delta\_J = Delta\_in\_Rad(find(All\_Bus\_Type ~= Slack\_Bus\_Type));

V\_J = V(find((All\_Bus\_Type == PQ\_Bus\_Type)));

Delta\_P\_J = Delta\_P(find(All\_Bus\_Type ~= Slack\_Bus\_Type));

Delta\_Q\_J = Delta\_Q(find((All\_Bus\_Type == PQ\_Bus\_Type)));

%% Updating V and Delta through LU Factorization

% Function Calling: LU Factorization Using Dolittle's Method

[Delta\_Corrected] = LU\_Factorization\_Dolittle\_Function(J11,Delta\_P\_J);

[V\_Corrected] = LU\_Factorization\_Dolittle\_Function(J22,Delta\_Q\_J);

% Updating Voltages and Angles

Delta\_Updated = Delta\_J + transpose(Delta\_Corrected);

V\_Updated = V\_J.\*(1 + transpose(V\_Corrected));

% Preparing for Next Iteration

V\_i = (find((All\_Bus\_Type == PQ\_Bus\_Type)));

Delta\_i = find(All\_Bus\_Type ~= Slack\_Bus\_Type);

for i=1:length(Delta\_i)

Delta\_New(Delta\_i(i)) = Delta\_Updated(i);

end

for i=1:length(V\_i)

V\_Desired(V\_i(i)) = V\_Updated(i);

end

V = transpose(V\_Desired);

Delta\_in\_Rad = Delta\_New;

Delta\_in\_Degree = (180 / pi) \* Delta\_in\_Rad;

Iteration = Iteration + 1;

%% Output

%fprintf("YBus: \n")

%Y\_Bus

fprintf("Number of Iteration: \n")

Iteration

fprintf("Voltage Magnitude: \n")

V

fprintf("Voltage Angles in Degree: \n")

Delta\_in\_Degree

fprintf("Real Power in MW: \n")

P\_Calculated \* Base\_MVA

fprintf("Reactive Power in MVAR: \n")

Q\_Calculated \* Base\_MVA

% Checking Tolerance Limit

if (max(abs(Delta\_P\_J)) < Tolerance & max(abs(Delta\_Q\_J)) < Tolerance)

break;

end

end

end

## Function – LU Factorization:

%% LU Factorization Function: Dolittle's Algorithm

function [ X\_Matrix ] = LU\_Factorization\_Dolittle\_Function(A\_Matrix,B\_Matrix)

% Getting the Size of Input Matrix

Length\_A = length(A\_Matrix);

% Initializing The Lower and Upper Triangular Matrices

Lower\_Triangular\_Matrix = zeros(Length\_A,Length\_A);

Upper\_Triangular\_Matrix = zeros(Length\_A,Length\_A);

% LOOP: Assigning 1 into All Diagonal Elements of Lower Traingular Matrix

for j = 1:Length\_A

Lower\_Triangular\_Matrix(j,j) = 1;

end

% Computing 1st Row of Upper Traingular Matrix

Upper\_Triangular\_Matrix(1,:) = A\_Matrix(1,:);

% Computing 1st Column of Lower Traingular Matrix

Lower\_Triangular\_Matrix(:,1) = A\_Matrix(:,1)/Upper\_Triangular\_Matrix(1,1);

% LOOP: Computing All Other Rows and Column of Upper and Lower Traingular Matrix

for j = 2:Length\_A

for k = j:Length\_A

Upper\_Triangular\_Matrix(j,k) = A\_Matrix(j,k) - Lower\_Triangular\_Matrix(j,1:j-1) \* Upper\_Triangular\_Matrix(1:j-1,k);

end

for l = j+1:Length\_A

Lower\_Triangular\_Matrix(l,j) = (A\_Matrix(l,j) - Lower\_Triangular\_Matrix(l,1:j-1) \* Upper\_Triangular\_Matrix(1:j-1,j)) / Upper\_Triangular\_Matrix(j,j);

end

end

% Output

% A\_Matrix

% Lower\_Triangular\_Matrix

% Upper\_Triangular\_Matrix

% Verification

% A\_Matrix - (Lower\_Triangular\_Matrix \* Upper\_Triangular\_Matrix)

%% Forward Substitution

% Initialization of Y Matrix

Y\_Matrix = zeros(Length\_A,1);

% Computing First Value of Y Matrix

Y\_Matrix(1) = B\_Matrix(1) / Lower\_Triangular\_Matrix(1,1);

% LOOP: Computing Rest of the Entries of Y Matrix

for j = 2:Length\_A

Y\_Matrix(j) = (B\_Matrix(j) - Lower\_Triangular\_Matrix(j,1:j-1) \* Y\_Matrix(1:j-1)) / Lower\_Triangular\_Matrix(j,j);

end

% Output

% Y\_Matrix

%% Backward Substitution

% Initialization of X Matrix

X\_Matrix = zeros(Length\_A,1);

% Computing Last Value of X Matrix

X\_Matrix(Length\_A) = Y\_Matrix(Length\_A) / Upper\_Triangular\_Matrix(Length\_A,Length\_A);

% LOOP: Computing Rest of the Entries of X Matrix

for j = Length\_A-1:-1:1

X\_Matrix(j) = (Y\_Matrix(j) - Upper\_Triangular\_Matrix(j,j+1:Length\_A) \* X\_Matrix(j+1:Length\_A)) / Upper\_Triangular\_Matrix(j,j);

end

% Output

% X\_Matrix

end

Results:

# Problem A:

## Newton-Raphson Method

Number of Iteration:

Iteration =

1

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0401 1.0376 1.0455 1.0651 1.0608 1.0462 1.0447

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -7.9641 -7.9641 -7.9641 -7.9641 -7.9641 -7.9641 -7.9641 -7.9641 -7.9641 -7.9641

Real Power in MW:

ans =

1.0e-11 \*

0.0767 0.0744 0.0767 0.0744 0.0577 -0.1599 -0.2109 -0.0133 -0.2842 -0.1599 0.0577

Reactive Power in MVAR:

ans =

185.0299 60.4790 185.0299 60.4790 -181.8282 -62.9427 -220.1250 -38.5000 -370.1250 -62.9427 -181.8282

Number of Iteration:

Iteration =

2

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0359 1.0349 1.0424 1.0612 1.0574 1.0438 1.0427

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -7.3821 -17.2845 -26.8624 -5.5302 -13.7736 -20.8314 -30.8859 -40.6743 -33.2014 -23.6779

Real Power in MW:

ans =

888.8507 0.0000 0.0000 0.0000 -887.7857 -9.1911 4.4969 4.5886 14.5387 -14.5481 -0.6001

Reactive Power in MVAR:

ans =

-0.6797 -166.6613 -90.7605 -218.7274 133.3290 76.0008 -195.6295 2.2105 -271.1066 77.7264 83.6692

Number of Iteration:

Iteration =

3

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0096 0.9847 0.9711 0.9641 0.9813 0.9897 1.0111

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.4562 -24.9306 -35.0354 -6.3806 -16.1891 -24.3288 -37.2777 -49.9540 -41.7316 -31.5442

Real Power in MW:

ans =

1.0e+03 \*

0.6157 0.6968 0.7161 0.6970 -0.0022 0.0114 -0.9495 0.0061 -1.7389 0.0128 0.0025

Reactive Power in MVAR:

ans =

-6.4035 -111.9341 -38.0511 -165.7143 50.4980 164.1559 10.8884 59.9932 20.6614 191.3774 100.0547

Number of Iteration:

Iteration =

4

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0065 0.9782 0.9613 0.9491 0.9716 0.9836 1.0083

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.7581 -26.9988 -37.1876 -6.4733 -16.5527 -24.9547 -38.7910 -52.3550 -43.9452 -33.6381

Real Power in MW:

ans =

1.0e+03 \*

0.6920 0.6982 0.7183 0.6980 -0.0056 -0.0096 -0.9608 0.0003 -1.7401 -0.0077 -0.0025

Reactive Power in MVAR:

ans =

164.3868 194.2299 157.9330 163.5236 3.1602 9.8272 -91.3397 7.0366 -92.4274 9.9196 3.0792

Number of Iteration:

Iteration =

5

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0064 0.9780 0.9609 0.9485 0.9713 0.9834 1.0082

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.7682 -27.0919 -37.2828 -6.4765 -16.5647 -24.9765 -38.8539 -52.4584 -44.0422 -33.7318

Real Power in MW:

ans =

1.0e+03 \*

0.6998 0.7000 0.7190 0.7000 -0.0001 -0.0004 -0.9670 0.0000 -1.7659 -0.0003 -0.0001

Reactive Power in MVAR:

ans =

184.6540 233.5748 175.7289 201.1635 0.0528 0.1733 -99.7298 0.3787 -99.8528 0.1037 0.0138

Number of Iteration:

Iteration =

6

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0064 0.9780 0.9609 0.9485 0.9713 0.9834 1.0082

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.7682 -27.0921 -37.2830 -6.4765 -16.5647 -24.9765 -38.8540 -52.4586 -44.0423 -33.7320

Real Power in MW:

ans =

1.0e+03 \*

0.7001 0.7000 0.7190 0.7000 -0.0000 -0.0000 -0.9670 0.0000 -1.7670 -0.0000 -0.0000

Reactive Power in MVAR:

ans =

185.3269 234.9460 176.2398 202.3625 0.0001 0.0002 -99.9995 0.0009 -99.9998 0.0001 0.0000

# Problem A:

## Fast Decoupled Method

As Fast Decoupled method was not converging with initial condition, first I have run Newton-Raphson and took the 4th iteration results as the input of the Fast Decoupled function.

Number of Iteration:

Iteration =

1

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0064 0.9782 0.9611 0.9489 0.9715 0.9835 1.0082

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.7926 -27.0332 -37.2220 -6.5077 -16.5871 -24.9892 -38.8254 -52.3895 -43.9796 -33.6725

Real Power in MW:

ans =

1.0e+03 \*

0.6998 0.7000 0.7190 0.7000 -0.0001 -0.0004 -0.9670 0.0000 -1.7659 -0.0003 -0.0001

Reactive Power in MVAR:

ans =

184.6540 233.5748 175.7289 201.1635 0.0528 0.1733 -99.7298 0.3787 -99.8528 0.1037 0.0138

Number of Iteration:

Iteration =

2

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0064 0.9782 0.9611 0.9489 0.9715 0.9835 1.0082

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.7708 -27.0930 -37.2829 -6.4770 -16.5663 -24.9761 -38.8520 -52.4548 -44.0413 -33.7326

Real Power in MW:

ans =

1.0e+03 \*

0.7035 0.6999 0.7190 0.6999 -0.0039 -0.0006 -0.9669 0.0000 -1.7656 -0.0004 -0.0001

Reactive Power in MVAR:

ans =

185.3031 234.0405 175.8874 201.5237 0.4127 -0.0095 -99.9792 0.0324 -100.0168 -0.0057 -0.0058

Number of Iteration:

Iteration =

3

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0064 0.9780 0.9609 0.9485 0.9713 0.9834 1.0082

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.7579 -27.0413 -37.2313 -6.4723 -16.5535 -24.9603 -38.8189 -52.4033 -43.9897 -33.6809

Real Power in MW:

ans =

1.0e+03 \*

0.7002 0.7000 0.7190 0.7000 0.0001 -0.0000 -0.9669 0.0001 -1.7672 -0.0000 -0.0000

Reactive Power in MVAR:

ans =

185.2359 234.1776 175.8904 201.5415 -0.3317 0.2503 -99.5279 0.5382 -99.4844 0.0685 0.0121

# Problem B:

## Newton-Raphson Method

Number of Iteration:

Iteration =

1

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0529 1.0687 1.1025 1.1085 1.0853 1.0596 1.0502

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -8.0698 -8.0698 -8.0698 -8.0698 -8.0698 -8.0698 -8.0698 -8.0698 -8.0698 -8.0698

Real Power in MW:

ans =

1.0e-11 \*

0.0767 0.0744 0.0767 0.0744 0.0577 -0.1599 -0.2842 -0.0155 -0.2132 -0.1599 0.0577

Reactive Power in MVAR:

ans =

185.0299 60.4790 185.0299 60.4790 -181.8282 -62.9427 -429.7500 -48.1250 -420.1250 -62.9427 -181.8282

Number of Iteration:

Iteration =

2

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0469 1.0624 1.0930 1.0989 1.0786 1.0553 1.0472

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -7.2930 -12.3140 -21.8067 -5.4874 -13.4981 -20.1224 -26.2115 -35.3291 -28.0655 -18.6741

Real Power in MW:

ans =

911.5820 0.0000 0.0000 0.0000 -918.1976 -28.9979 35.0530 6.4278 23.1118 -23.0541 -3.9005

Reactive Power in MVAR:

ans =

-76.6819 -355.1600 -124.4594 -299.7196 139.8339 82.3335 -171.8105 5.1446 -263.7234 80.4464 85.4798

Number of Iteration:

Iteration =

3

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0226 1.0164 1.0282 1.0188 1.0094 1.0053 1.0175

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.1288 -17.8981 -27.8995 -6.2424 -15.6529 -23.2235 -30.8826 -42.4449 -34.4940 -24.4710

Real Power in MW:

ans =

1.0e+03 \*

0.6175 0.6945 0.7155 0.6958 -0.0057 0.0041 -0.9416 0.0047 -1.7302 0.0088 0.0012

Reactive Power in MVAR:

ans =

-74.7544 -279.0003 -66.4655 -235.6717 50.8425 158.7622 -6.3054 45.2724 14.9952 188.2070 99.2678

Number of Iteration:

Iteration =

4

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0203 1.0119 1.0214 1.0097 1.0026 1.0008 1.0153

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.3387 -18.9954 -29.0640 -6.3057 -15.9065 -23.6544 -31.6656 -43.7938 -35.7046 -25.5884

Real Power in MW:

ans =

1.0e+03 \*

0.6858 0.6985 0.7183 0.6982 -0.0047 -0.0080 -0.9604 -0.0002 -1.7450 -0.0066 -0.0022

Reactive Power in MVAR:

ans =

83.1321 0.9693 118.6110 68.6800 2.7488 8.5346 -93.6781 4.4650 -93.2986 8.9904 2.8879

Number of Iteration:

Iteration =

5

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0203 1.0118 1.0213 1.0095 1.0025 1.0008 1.0153

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.3425 -19.0218 -29.0914 -6.3069 -15.9110 -23.6625 -31.6823 -43.8245 -35.7327 -25.6150

Real Power in MW:

ans =

1.0e+03 \*

0.6912 0.7000 0.7190 0.7000 -0.0001 -0.0002 -0.9669 -0.0000 -1.7665 -0.0002 -0.0000

Reactive Power in MVAR:

ans =

97.6241 28.7114 131.9440 95.9812 0.0285 0.0960 -99.8913 0.1405 -99.9105 0.0656 0.0096

>>

# Problem B:

## Fast Decoupled Method

As Fast Decoupled method was not converging with initial condition, first I have run Newton-Raphson and took the 4th iteration results as the input of the Fast Decoupled function.

Number of Iteration:

Iteration =

1

Voltage Magnitude:

V =

1.0300 1.0100 1.0300 1.0100 1.0203 1.0119 1.0214 1.0096 1.0026 1.0008 1.0153

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.3588 -19.0154 -29.0841 -6.3257 -15.9266 -23.6745 -31.6857 -43.8139 -35.7247 -25.6084

Real Power in MW:

ans =

1.0e+03 \*

0.6912 0.7000 0.7190 0.7000 -0.0001 -0.0002 -0.9669 -0.0000 -1.7665 -0.0002 -0.0000

Reactive Power in MVAR:

ans =

97.6241 28.7114 131.9440 95.9812 0.0285 0.0960 -99.8913 0.1405 -99.9105 0.0656 0.0096

# Problem C :

## Newton-Raphson Method

Number of Iteration:

Iteration =

1

Voltage Magnitude:

V =

1.0300 1.0100 1.0305 1.0105 1.0203 1.0119 1.0214 1.0098 1.0031 1.0013 1.0158

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.3407 -19.0201 -29.0897 -6.3051 -15.9093 -23.6608 -31.6806 -43.8228 -35.7310 -25.6133

Real Power in MW:

ans =

1.0e+03 \*

0.6913 0.7000 0.7190 0.7000 -0.0000 0.0000 -0.9670 -0.0000 -1.7671 0.0000 0.0000

Reactive Power in MVAR:

ans =

97.9126 29.2820 132.0794 96.2569 -0.0152 -0.2815 -99.7948 -0.0139 -100.1817 0.4484 -0.0194

>>

## Fast Decoupled Method

Here, it is not necessary to run Newton-Raphson first. Fast decoupled is converged with direct input.

Number of Iteration:

Iteration =

1

Voltage Magnitude:

V =

1.0300 1.0081 1.0251 1.0052 1.0196 1.0100 1.0191 1.0063 0.9978 0.9961 1.0105

Voltage Angles in Degree:

Delta\_in\_Degree =

0 -9.3428 -19.0221 -29.0917 -6.3071 -15.9113 -23.6628 -31.6826 -43.8248 -35.7330 -25.6153

Real Power in MW:

ans =

1.0e+03 \*

0.6913 0.7000 0.7190 0.7000 -0.0000 0.0000 -0.9670 -0.0000 -1.7671 0.0000 0.0000

Reactive Power in MVAR:

ans =

97.9126 29.2820 132.0794 96.2569 -0.0152 -0.2815 -99.7948 -0.0139 -100.1817 0.4484 -0.0194

>>

Input Data:

# Problem A:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BUS DATA FOLLOWS | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BUS | | | | LOAD FLOW AREA | LOSS ZONE | TYPE | V\_MAG | V\_ANG in RAD | LOAD\_MW | LOAD\_MVA | G\_MW | G\_MVAR | BASE\_KV | V\_DESIRED | MAX MVAR/VOLT LIMIT | MIN MVAR/VOLT LIMIT | SHUNT\_G | SHUNT\_B | REMOTE CONTROLLED BUS |
| 1 | Bus | 1 | HV | 1 | 1 | 3 | 1.06 | 0 | 0 | 0 | 700 | 185 | 0 | 1.03 | 0 | 0 | 0 | 0 | 0 |
| 2 | Bus | 2 | HV | 1 | 1 | 2 | 1.045 | 0 | 0 | 0 | 700 | 235 | 0 | 1.01 | 0 | 0 | 0 | 0 | 0 |
| 3 | Bus | 3 | HV | 1 | 1 | 2 | 1.01 | 0 | 0 | 0 | 719 | 176 | 0 | 1.03 | 0 | 0 | 0 | 0 | 0 |
| 4 | Bus | 4 | HV | 1 | 1 | 2 | 1.019 | 0 | 0 | 0 | 700 | 202 | 0 | 1.01 | 0 | 0 | 0 | 0 | 0 |
| 5 | Bus | 5 | HV | 1 | 1 | 0 | 1.02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 6 | Bus | 6 | LV | 1 | 1 | 0 | 1.07 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | Bus | 7 | ZV | 1 | 1 | 0 | 1.062 | 0 | 967 | 100 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0 |
| 8 | Bus | 8 | TV | 1 | 1 | 0 | 1.09 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 9 | Bus | 9 | LV | 1 | 1 | 0 | 1.056 | 0 | 1767 | 100 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 3.5 | 0 |
| 10 | Bus | 10 | LV | 1 | 1 | 0 | 1.051 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 11 | Bus | 11 | LV | 1 | 1 | 0 | 1.057 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BRANCH DATA FOLLOWS | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| FROM | TO | LOAD FLOW AREA | LOSS ZONE | CIRCUIT | TYPE | R | X | B | LINE MVA RATING 1 | LINE MVA RATING 2 | LINE MVA RATING 3 | CONTROL BUS | SIDE | XFR FINAL TURN RATIO | XFR FINAL ANGLE | MIN TAP.PHASE SHIFT | MAX TAP/PHASE SHIFT | STEP SIZE | MIN VOLT/MVAR MW LIMIT | MAX VOLT/MVAR MW LIMIT |
| 1 | 5 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 6 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 11 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 10 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 6 | 1 | 1 | 1 | 0 | 0.0025 | 0.025 | 0.04375 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 7 | 1 | 1 | 1 | 0 | 0.001 | 0.01 | 0.0175 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 8 | 1 | 1 | 1 | 0 | 0.0055 | 0.055 | 0.385 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 9 | 1 | 1 | 1 | 0 | 0.0055 | 0.055 | 0.385 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 10 | 1 | 1 | 1 | 0 | 0.001 | 0.01 | 0.0175 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 11 | 1 | 1 | 1 | 0 | 0.0025 | 0.025 | 0.04375 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# Problem B:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BUS DATA FOLLOWS | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BUS | | | | LOAD FLOW AREA | LOSS ZONE | TYPE | V\_MAG | V\_ANG in RAD | LOAD\_MW | LOAD\_MVA | G\_MW | G\_MVAR | BASE\_KV | V\_DESIRED | MAX MVAR/VOLT LIMIT | MIN MVAR/VOLT LIMIT | SHUNT\_G | SHUNT\_B | REMOTE CONTROLLED BUS |
| 1 | Bus | 1 | HV | 1 | 1 | 3 | 1.06 | 0 | 0 | 0 | 700 | 185 | 0 | 1.03 | 0 | 0 | 0 | 0 | 0 |
| 2 | Bus | 2 | HV | 1 | 1 | 2 | 1.045 | 0 | 0 | 0 | 700 | 235 | 0 | 1.01 | 0 | 0 | 0 | 0 | 0 |
| 3 | Bus | 3 | HV | 1 | 1 | 2 | 1.01 | 0 | 0 | 0 | 719 | 176 | 0 | 1.03 | 0 | 0 | 0 | 0 | 0 |
| 4 | Bus | 4 | HV | 1 | 1 | 2 | 1.019 | 0 | 0 | 0 | 700 | 202 | 0 | 1.01 | 0 | 0 | 0 | 0 | 0 |
| 5 | Bus | 5 | HV | 1 | 1 | 0 | 1.02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 6 | Bus | 6 | LV | 1 | 1 | 0 | 1.07 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | Bus | 7 | ZV | 1 | 1 | 0 | 1.062 | 0 | 967 | 100 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 0 |
| 8 | Bus | 8 | TV | 1 | 1 | 0 | 1.09 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 9 | Bus | 9 | LV | 1 | 1 | 0 | 1.056 | 0 | 1767 | 100 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 0 |
| 10 | Bus | 10 | LV | 1 | 1 | 0 | 1.051 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 11 | Bus | 11 | LV | 1 | 1 | 0 | 1.057 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BRANCH DATA FOLLOWS | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| FROM | TO | LOAD FLOW AREA | LOSS ZONE | CIRCUIT | TYPE | R | X | B | LINE MVA RATING 1 | LINE MVA RATING 2 | LINE MVA RATING 3 | CONTROL BUS | SIDE | XFR FINAL TURN RATIO | XFR FINAL ANGLE | MIN TAP.PHASE SHIFT | MAX TAP/PHASE SHIFT | STEP SIZE | MIN VOLT/MVAR MW LIMIT | MAX VOLT/MVAR MW LIMIT |
| 1 | 5 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 6 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 11 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 10 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 6 | 1 | 1 | 1 | 0 | 0.0025 | 0.025 | 0.04375 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 7 | 1 | 1 | 1 | 0 | 0.001 | 0.01 | 0.0175 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 8 | 1 | 1 | 1 | 0 | 0.00367 | 0.0367 | 0.5775 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 9 | 1 | 1 | 1 | 0 | 0.0055 | 0.055 | 0.385 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 10 | 1 | 1 | 1 | 0 | 0.001 | 0.01 | 0.0175 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 11 | 1 | 1 | 1 | 0 | 0.0025 | 0.025 | 0.04375 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

## Problem C:

# Problem B:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BUS DATA FOLLOWS | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| BUS | | | | LOAD FLOW AREA | LOSS ZONE | TYPE | V\_MAG | V\_ANG in RAD | LOAD\_MW | LOAD\_MVA | G\_MW | G\_MVAR | BASE\_KV | V\_DESIRED | MAX MVAR/VOLT LIMIT | MIN MVAR/VOLT LIMIT | SHUNT\_G | SHUNT\_B | REMOTE CONTROLLED BUS |
| 1 | Bus | 1 | HV | 1 | 1 | 3 | 1.06 | 20.2 | 0 | 0 | 700 | 185 | 0 | 1.03 | 0 | 0 | 0 | 0 | 0 |
| 2 | Bus | 2 | HV | 1 | 1 | 2 | 1.045 | 0 | -700 | -28.37 | 0 | 0 | 0 | 1.01 | 0 | 0 | 0 | 0 | 0 |
| 3 | Bus | 3 | HV | 1 | 1 | 2 | 1.01 | 0 | -719 | -131.883 | 0 | 0 | 0 | 1.03 | 0 | 0 | 0 | 0 | 0 |
| 4 | Bus | 4 | HV | 1 | 1 | 2 | 1.019 | 0 | -700 | -95.8365 | 0 | 0 | 0 | 1.01 | 0 | 0 | 0 | 0 | 0 |
| 5 | Bus | 5 | HV | 1 | 1 | 0 | 1.02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 6 | Bus | 6 | LV | 1 | 1 | 0 | 1.07 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | Bus | 7 | ZV | 1 | 1 | 0 | 1.062 | 0 | 967 | 100 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 0 |
| 8 | Bus | 8 | TV | 1 | 1 | 0 | 1.09 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 9 | Bus | 9 | LV | 1 | 1 | 0 | 1.056 | 0 | 1767 | 100 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 0 |
| 10 | Bus | 10 | LV | 1 | 1 | 0 | 1.051 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 11 | Bus | 11 | LV | 1 | 1 | 0 | 1.057 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BRANCH DATA FOLLOWS | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| FROM | TO | LOAD FLOW AREA | LOSS ZONE | CIRCUIT | TYPE | R | X | B | LINE MVA RATING 1 | LINE MVA RATING 2 | LINE MVA RATING 3 | CONTROL BUS | SIDE | XFR FINAL TURN RATIO | XFR FINAL ANGLE | MIN TAP.PHASE SHIFT | MAX TAP/PHASE SHIFT | STEP SIZE | MIN VOLT/MVAR MW LIMIT | MAX VOLT/MVAR MW LIMIT |
| 1 | 5 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 6 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 11 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 10 | 1 | 1 | 1 | 0 | 0 | 0.0167 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 6 | 1 | 1 | 1 | 0 | 0.0025 | 0.025 | 0.04375 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 7 | 1 | 1 | 1 | 0 | 0.001 | 0.01 | 0.0175 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 8 | 1 | 1 | 1 | 0 | 0.00367 | 0.0367 | 0.5775 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 9 | 1 | 1 | 1 | 0 | 0.0055 | 0.055 | 0.385 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 10 | 1 | 1 | 1 | 0 | 0.001 | 0.01 | 0.0175 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 11 | 1 | 1 | 1 | 0 | 0.0025 | 0.025 | 0.04375 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |