Name: Swati Verma Reg. No.:200906448 Section: C Roll No: 66

Point by Point Solution of Swing Equation

Abstract

The swing equation plays an important role in the model and analysis of power system dynamics, including small-signal stability and transient stability. Because it has the same form as in several different fields, such as the forced pendulum in mechanics, the classical mechanistic description of superconducting Josephson junctions in physics, and the classical second-order phase lock loop in electronics, it has attracted general interest in science and engineering. Here, using the incremental harmonic balance (IHB) method its approximate limit cycle solution is obtained. The problem of more distorted limit cycle when the parameters are closer to the homoclinic bifurcation curve can be easily solved by incorporating harmonic higher orders in the IHB method. In this way, we can perfectly predict the homoclinic bifurcation curve. Moreover, the method is extended to study the generalized fluctuation equation including the excitation voltage dynamics.

Introduction

The importance of power system stability is increasingly becoming one of the most limiting for system performance. In recent times, the subject of great interest to power system planners and operations is Power system stability. In practical power systems, it is expected that power generation must be equal to power demand including losses to ensure constant speed and frequency of operation. For a successful operation of a power system, an engineer's ability to provide reliable and interrupted service to the load is very much a key part.

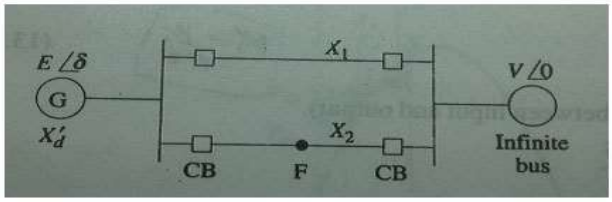
The following is the general objective of this purposed system:

• To evaluate the step by step for swing curve using point-by-point method.

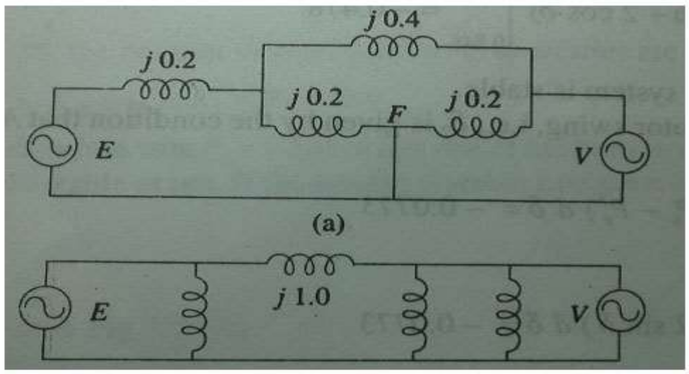
• To determine the power system stability problem for one-machine system based on MATLAB program software.

Problem

One line diagram



Impedance diagram during fault



A f = 50hz generator of 50 MVA is supplying 50 MW with inertia constant 'H' = 2.7 MJ/MVA at rated speed. E = 1.05 p.u., V = 1 p.u., X1 = X2 = 0.4 p.u. three phase faults at line 2.
(a) plot swing curve for a sustained fault up to a time of 5 secs.
(b) plot swing curve if fault is cleared by isolating line in 0.1 seconds.
(c) Find the critical clearing angle

MATLAB code

```
%% MVA base = 5
E = 50; V =1; Xd = 0.2; X1 =0.4; X2 = 0.4; H = 2.7;

% prefault condition
d = 0: pi/10: pi;
d1 =d;
d2 = d;

M = 2.7/ (180*50);                   % angular momentum = H/180*f
P = (1.05/0.4) *sin(d);           % Initial power curve
Po = 1 ;                             % power output in pu = 50 MW/50 MVA
del = asind(0.4/1.05);              % initial load angle in degrees //Pe =
(E*V/X) sin(delo)
% During fault

Pe2 = 1.05*sin(d1);                 % Power curve during fault

%Post fault condition
Pe3 = (1.05/0.6)*sin(d2);           % Power curve after clearing fault
```

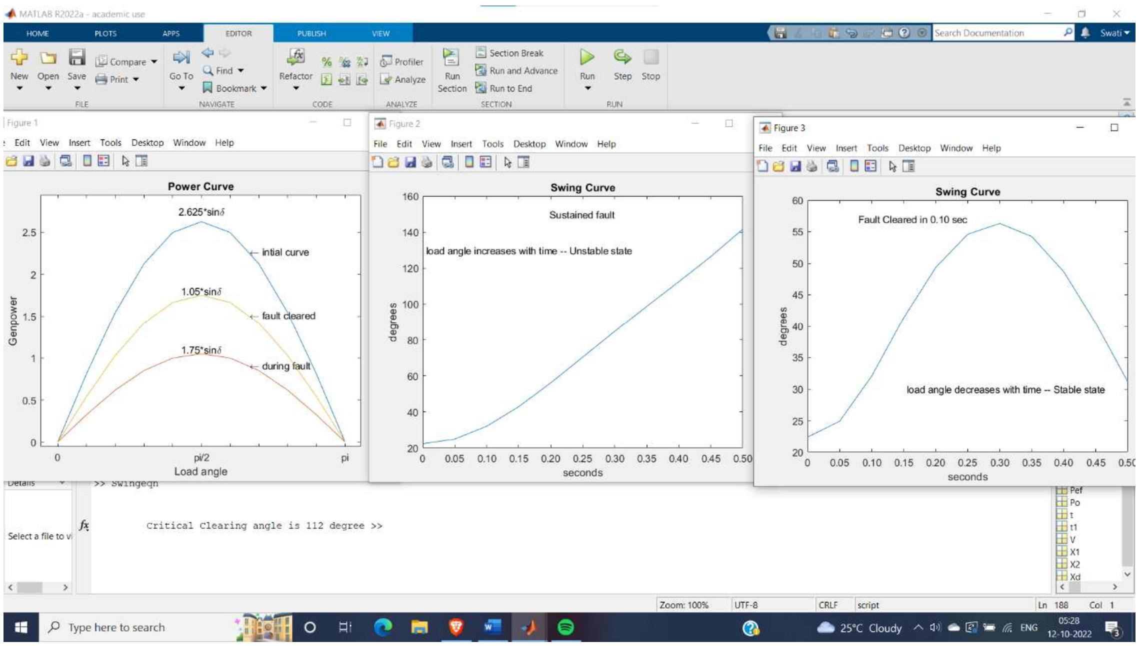
```
%%  Primary Power curve plot Figure-1

plot(d,P);
set(gca,'XTick',0:pi/10:pi);
set(gca,'XTickLabel',{'0','','','','','pi/2','','','','','pi'});
title('Power Curve');
xlabel('Load angle');
ylabel('Genpower');
text((2/3)*pi,(1.05/0.4)*sin((2/3)*pi),'\leftarrow intial
curve','HorizontalAlignment','left');
text(pi/2,2.75,'2.625*sin\delta','HorizontalAlignment','center');
hold all
plot(d1,Pe2);
text((2/3)*pi,1.05*sin((2/3)*pi),'\leftarrow during
fault','HorizontalAlignment','left');
text(pi/2,1.80,'1.05*sin\delta','HorizontalAlignment','center');
plot(d2,Pe3);
text((2/3)*pi,(1.05/0.6)*sin((2/3)*pi),'\leftarrow fault
cleared','HorizontalAlignment','left');
text(pi/2,1.1,'1.75*sin\delta','HorizontalAlignment','center');
hold off

%% ------------

t = 0.05;        % time step preferably 0.05 seconds
t1 = 0:t:0.5;


%% (a) sustained fault at t = 0

% For discontinuity at t = 0, we take the average of accelerating power
% Before and after the fault
% At t = 0-, Pa1 = 0
% At t = 0+. Pa2 = Pi - Pe2
% at t = 0 ,Pa =Pa1+Pa2/2

Pao = (1 - (1.05*sind(del)))/2;      % at the instant of fault d1 = del
Pa(1) = Pao;
cdel(1) = 0;
d1 = t^2/M;

for i = 1:11

    if i == 1

        d2(i) = d1*Pa(i);
        d(i) = del;

    else
      cdel(i) = cdel(i-1)+d2(i-1);

      d(i)  = d(i-1)+cdel(i);
```

```
      Pe(i) = 1.05*sind(d(i));

      Pa(i) = 1 - Pe(i);

      d2(i) = d1*Pa(i);
    end
end
%% swing curve 1 plot

figure (2);
plot(t1,d);
set(gca,'Xtick',0:0.05:0.5);
set(gca,'XtickLabel',{'0','0.05','0.10','0.15','0.20','0.25','0.30','0.35'
,'0.40','0.45','0.50'});
title('Swing Curve');
xlabel('seconds');
ylabel('degrees');
text(0.30,150,'  Sustained fault','HorizontalAlignment','right');
text(0.001,130,' load angle increases with time -- Unstable
state','HorizontalAlignment','left');

%% (b) Fault cleared in 0.10 seconds ,2nd step ---- 3rd element [1]0
[2]0.05,[3]0.10

Pafo = (1 - (1.05*sind(del)))/2;     % at the instant of fault del1 = delo
Paf(1) = Pao;
cdelf(1) = 0;
d1f = t^2/M;

for i = 1:2

    if i == 1

      d2f(i) = d1*Pa(i);
      delf(i) = del;

   else
      cdelf(i) = cdelf(i-1)+d2f(i-1);

      delf(i)  = delf(i-1)+cdelf(i);

      Pef(i) = 1.05*sind(delf(i));

      Paf(i) = 1 - Pef(i);

      d2f(i) = d1*Paf(i);
    end

end

% after clearing fault, power curve shift to Pe3

for i = 3:11
```

```
    if i == 3

     cdelf(i) = cdelf(i-1) +d2f(i-1);
     delf(i) = delf(i-1) +cdelf(i);
     Pef(i) = 1.05*sind(delf(i));
     Paf(i) = 1 - Pef(i);
     a1 = Paf(i);
     d2f(i) = d1*Paf(i);
      a2 = d2f(i);

     Pef(i) = 1.75*sind(delf(i));
     Paf(i) = 1 - Pef(i);
     d2f(i) = d1*Paf(i);

     Paf(i) = (Paf(i)+ a1)/2;
     d2f(i) = (d2f(i) + a2)/2;

    else


     cdelf(i) = cdelf(i-1)+d2f(i-1);

     delf(i)  = delf(i-1)+cdelf(i);

     Pef(i) = 1.75*sind(delf(i));

     Paf(i) = 1 - Pef(i);

     d2f(i) = d1*Paf(i);
    end
end

%% ------

figure (3);
plot(t1,delf);
set(gca,'Xtick',0:0.05:0.5);
set(gca,'XtickLabel',{'0','0.05','0.10','0.15','0.20','0.25','0.30','0.35'
,'0.40','0.45','0.50'});
title('Swing Curve');
xlabel('seconds');
ylabel('degrees');
text(0.25,57,' Fault Cleared in 0.10 sec','HorizontalAlignment','right');
text(0.15,30,' load angle decreases with time -- Stable
state','HorizontalAlignment','left');

%% (c) critical clearing angle

del = degtorad(del);       % initial load angle in rad
delm = pi - asin(1/1.75);   % angle of max swing

c1 = ((delm-del)-(1.05*cos(del))+(1.75*cos(delm)))/(1.75-1.05);
```

```
cclang = acos(c1);                              % critical clearing
angle in rad
cclang = radtodeg(cclang);                      % critical clearing
angle in degree
cclang = int16(cclang);                         % converting to integer
fprintf('\n\n\t\t Critical Clearing angle is %d degree ',cclang);
```

Output



Conclusion

In this report, the power system stability problem is considered, and point by point method is used for efficient analysis of power system behavior for various types of operating condition. Hence, this equation is very helpful to determine the stability condition of any complex power system through calculating the angle delta and also having the equal area criteria so, any of transients can be eliminated from this. Finally, the result is displayed on the screen using MATLAB program software.

REFERENCES:

1. Nagrath, I.J., and Kothari, D.P., Power System Engineering, New Delhi, Tada McGraw-Hill,1995.