ภาคผนวก

มหาวิทยาลัยเชียงใหม่
Chiang Mai University

โปรแกรม M_file

```
%disp('*********************************')
%disp('*System  control simulate! ver_#13    *')
%disp('*       For MATLAB V5 & V4           *')
%disp('*                          T.Chidentree *')
%disp('*********************************')
clear
clc
%      Set initial condition
%
t=1;
ess=0.000000001;

yd=0;   % yd=y(t-1)
ud=0;   % ud=u(t-1)
ecd=0;  % ecd=ec(t-1)
hidi=10;hidc=25;                % Hidden  unit
Incf=[1 1 1;1 1 1;1 1 1] ; Onc=[1];  % for NNC
Ini=[1 1;1 1];Oni=[1];    ;       % for NNI

eni=input(' Maximum epoch for NNI <1-50> ');
tpi=[1 eni ess 0.01 1.05 0.7 0.98 1.04];

Dfi='1-(exp(x)-exp(-x)).^2./(exp(x)+exp(-x)).^2'; %
diff"tansig"
dfi=Dfi;

tmo=input('Monitor time period <sec>   ');
sn=input('Sampling number between 1sec <Hz> ');
Ts=1/sn; % Sampling rate
tmax=tmo*sn ;

lrc=input('Please input Learning rate of NNC<1e-20>  ') ;

lr=MENU(' Adaptive Learning rate ',' USE ',' No USE ');
if lr==1
  disp('Adaptive Learning rate <ON>')
  dLrc=0.65;  % decrease learning rate facter
  iLrc=1.025; %increase learning rate facter
  er=1.01;    % error ratio
elseif lr==2
  disp('Adaptive Learning rate <Off>')
  dLrc=1 ; % decrease learning rate facter
  iLrc=1 ; %increase learning rate facter
  er=1   ; % error ratio
else
  disp('ERROR --> off Adaptive Learning rate')
  dLrc=1 ; % decrease learning rate facter
  iLrc=1 ; %increase learning rate facter
  er=1   ; % error ratio
end

Vp= MENU('Plant parameter','No!need to vary','Vary by
small random value one time','Vary by small random value
many time');
Vp=Vp-1;
if Vp>0
  rw=input(' Ratio for reduce random value <1-100> ');
end

if Vp>=1  %  Vary plant parameter.
 disp(' ')
 disp(' Vary plant parameter by RANDOM at first start. ')
 disp(' >>>')
%      Plant modeling --> vary plant Character
 q1=randn(1)/rw;
 q2=randn(1)/rw;
 q3=randn(1)/rw;
 num=[4.3+q1];
 den=[1 (7.13+q2) (6.33+q3)];
 [a,b,c,d]=tf2ss(num,den);
 [aa,bb]=c2d(a,b,Ts); % Sampling time=Ts sec.
 xd=[0;0];

else %       Safe operation

%            Plant modeling
 num=[4.3];
 den=[1 7.13 6.33];
 [a,b,c,d]=tf2ss(num,den);
 [aa,bb]=c2d(a,b,Ts); % Sampling time=Ts sec.
 xd=[0;0];

end
plant=tf(num,den)

%*****************************************
**************
a1=2/(tmax+1);
b1=-a1;
a2=b1;
b2=-a2*tmax;
rr= MENU('Reference Signal','User define','Step on-
off','Ramp  signal','Sine slope_up','Random ','Ramp slope
up-down','Unit step','Sine 1 period','Step level down','Step
signal up','No used','Sine control frequency');
rr=rr-1;
disp('')
 if rr==7 ; % sine
   ns=linspace(0,2*pi,tmax);
   rsine=sin(ns);
 end

 if rr==11 ; % sine #11
   fsin=input('Input frequency of sin from reference signal
<0.1>.  ');
 end

weig=MENU('Initial weight by...','Memory','Random');
weig=weig-1;
if weig==0
 disp('LOADING !!!')
 load s10y1w01
end

if weig==1
 disp('RANDOM !!!')
 [wi1,bi1,wi2,bi2]=initff(Ini,hidi,'purelin',1,'purelin');
 [wc1,bc1,wc2,bc2]=initff(Incf,hidc,'purelin',1,'purelin');
end

messag=sprintf('System parameter# Epoch of NNI=%%g:
Learning rate=%%g:  Monitor time=%%g<sec> :
Sampling rate=%%g<sec>');
fprintf(messag,eni,lrc,tmo,Ts)
MENU('SIMULATE START...','RUN');
clc
Tstart=clock;

%                  Start Main loop

while  t<=tmax

tr=(t)/sn ;% real time for system

%              Generate  r(t)
if rr==0
 if t<tmax/4
   if t==1
     ru1=input('Input Ref >> Level #1 ?  ');
   end
```

```
    r(t)=ru1;
  elseif t<(tmax/4)*2
    if t==tmax/4
       ru2=input('Input Ref >> Level #2 ? ');
    end
    r(t)=ru2;
  elseif t<(tmax/4)*3
     if t==(tmax/4)*2
       ru3=input('Input Ref >> Level #3 ? ');
    end
    r(t)=ru3;
  else
     if t==(tmax/4)*3
       ru4=input('Input Ref >> Level #4 ? ');
    end
    r(t)=ru4;
  end
end

if rr==1
 if t<=(tmax-1)/2
  r(t)=1;       % constant value#1
 end
 if t>(tmax-1)/2
  r(t)=0;       % constant value#0
 end
end
if rr==2
r(t)=(t+1)/tmax;   % ramp
end
if rr==3
r(t)=sin((t+1)/tmax); % sine
end
if rr==4
r(t)=rand; % random
end
if rr==5;  % Ramp slope up-down
 if t<=(tmax-1)/2
    r(t)=(a1*t)+b1;  % slope up
  end
  if t>(tmax-1)/2
    r(t)=(a2*t)+b2;  % slope down
  end
end
if rr==6; % Unit step
 r(t)=1;
end
if rr==7;% Sine 1 period
 r(t)=rsine(t);
end

if rr==8;% step level down.
  if t<=tmax/4
    r(t)=1;
  elseif t<=(tmax/4)*2
    r(t)=0.75;
  elseif t<=(tmax/4)*3
    r(t)=0.25;
  else
    r(t)=0;
  end
end

if rr==9;% step level up.
  if t<=tmax/4
    r(t)=0;
  elseif t<=(tmax/4)*2
    r(t)=0.25;
  elseif t<=(tmax/4)*3
    r(t)=0.75;
  else
    r(t)=1.0;
  end
end

if rr==11;% Sine by frequency.
  r(t)=sin(2*pi*fsin*tr);
end
%
%****************************************
%
%                    Find   u(t)
%
Inc=[r(t) yd ud]';
u(t)=simuff(Inc,wc1,bc1,'tansig',wc2,bc2,'purelin');
%
%****************************************
%
%                    Find   ym(t)
%
Ini=[u(t) yd]';
ym(t)=simuff(Ini,wi1,bi1,'tansig',wi2,bi2,'purelin');
%
%****************************************
%
%                    Find   y(t)
%
if Vp==2  %  Vary plant parameter every time.
  disp('')
  disp(' Vary plant parameter !!! ')

%      Plant modeling --> vary every time!!!
  q1=randn(1)/rw;
  q2=randn(1)/rw;
  q3=randn(1)/rw;
  num=[4.3+q1];
  den=[1 (7.13+q2) (6.33+q3)];
  [a,b,c,d]=tf2ss(num,den);
  [aa,bb]=c2d(a,b,Ts); % Sampling time=Ts sec.
end

  xx=(aa*xd)+(bb*u(t));
  ys=(c*xx)+d;
  xd=xx;
  xdc=xx; % NNC phase
y(t)=ys;
to=tr;

%
%************************************
%
%                    Find  error
%
ec(t)=r(t)-y(t);
em=y(t)-ym(t);
%
if abs(em)>=ess
%**************************************
%                    Train&Update NNI
%
clf
[wi1,bi1,wi2,bi2]=mniV5x0(wi1,bi1,'tansig',wi2,bi2,'pureli
n',Ini,y(t),tpi);
%
end
%**************************************
%
%                    Find yp [plant information]
%
Sji=(wi1*Ini)+bi1;       % find input layer summation
w1i1=wi1(:,1);           % get only u(t) weight
x=Sji;
```

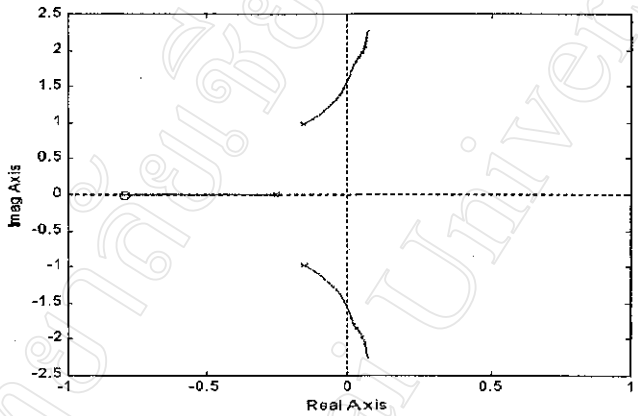
```
dSji=eval(dfi);
Yp=wi2*(w1i1.*dSji);
%-------------------------------------------------------------

%************************************
if abs(ec(t))>=ess
%************************************
%                Train&Update NNC

ecp=ec(t);

% Adaptive learning rate
if abs(ec(t))<=abs(ecd*er)
  lrc=lrc*iLrc;
else
  lrc=lrc*dLrc;
end
% [backpropagation phase plus plant information]

% find input layer summation
Sjc=(wc1*Inc)+bc1;
x=Sjc;
SSjc=tansig(Sjc);
dSjc=eval(dfi);

% ****Output to Hidden layer****
deltaWoc=lrc*Yp*ecp*(SSjc');
deltaBoc=lrc*Yp*ecp;

% ****Hidden to Input layer****
deltaWic=lrc*Yp*ecp*(wc2'.*dSjc)*Inc';
deltaBic=lrc*Yp*ecp*(wc2'.*dSjc);

% ****Up date****
wc2=wc2+deltaWoc;
bc2=bc2+deltaBoc;
wc1=wc1+deltaWic;
bc1=bc1+deltaBic;

message=sprintf('N_Controller at time =%%g:
error=%%g : Yp=%%g');
fprintf(message,to,ec(t),Yp)
disp(': @ Ver 0.13: Plant#Yun .Li ')
disp('>>> # ')

else
message=sprintf('N_Controller at time =%%g:
error=%%g : Yp=%%g');
fprintf(message,to,ec(t),Yp)
disp(': @ Ver 0.13: Plant#Yun .Li ')
disp('>>> [Not need to training phase.] ')


end % if control
%************************************
%*                Delay Data                *
%************************************
yd=y(t);
ud=u(t);
ecd=ec(t);
%************************************
trr(t)=tr;
yp(t)=Yp;
LRC(t)=lrc;
t=t+1;

end          % end for while main

%************************************
%                Display                *
%************************************
ts=trr;

figure(1)
subplot(2,2,1)
  plot(ts,r,'b');grid
  title('reference signal')
  xlabel('Time <sec>')
subplot(2,2,2)
  plot(ts,y,'k');grid
  title('output signal')
  axis([0 tmo (min(y)-0.05) (max(y)+0.05)])
  xlabel('Time <sec>')
subplot(2,2,3)
  plot(ts,ym,'m');grid
  title('output model signal')
  axis([0 tmo (min(y)-0.05) (max(y)+0.05)])
  xlabel('Time <sec>')
subplot(2,2,4)
  plot(ts,u,'r');grid
  title('plant control signal')
  xlabel('Time <sec>')

figure(2)
subplot(3,1,1)
  plot(ts,ec,'r');grid
  title('ERROR from Control system')
  xlabel('Time <sec>')
subplot(3,1,2)
  plot(ts,yp,'g');grid
  title('Plant information <yp>')
  xlabel('Time <sec>')
subplot(3,1,3)
  plot(ts,LRC,'b') ;grid
  title(' Learning Rate of NNC ')
  xlabel('Time <sec>')

figure(3)
  plot(ts,y,'k');grid
  title('Plant output signal <By NNI & NNC>')
  axis([0 tmo (min(y)-0.05) (max(y)+0.05)])
  xlabel('Time <sec>')

figure(4)
  plot(ts,ym,'m');grid
  title('Output model signal from NNI')
  axis([0 tmo (min(y)-0.05) (max(y)+0.05)])
  xlabel('Time <sec>')

figure(5)
  plot(ts,yp,'b');grid
  title('Plant information <yp>')
  xlabel('Time <sec>')

%**********************
%       Save weight
%**********************
if  weig==1;
wsav=menu(' Save this weight','Save','NO');
  if wsav==1;
    disp('Saving weight .....in file
C:\MATLAB\n12w03.mat ')
    save n12w02 wc1 wc2 wi1 wi2 bc1 bc2 bi1 bi2
  end
end
disp('              *** END Program ***                    ')
T_End=etime(clock,Tstart)
E_sumsqr=sumsqr(ec)
```

ผลการตอบสนองที่ได้จากส่วนดำเนินที่เป็นเชิงเส้น order 3

ส่วนดำเนินการใช้ในการทดลองเป็นส่วนดำเนินการที่จำลองมาจากระบบควบคุมหัวอ่านของจานแม่เหล็ก โดยมีสมการการถ่ายโอนดังนี้

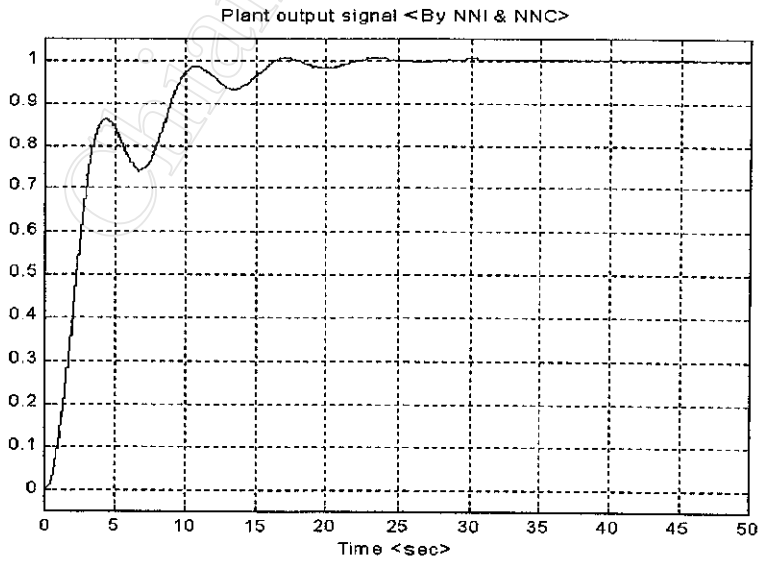
$$\frac{Y(s)}{U(s)} = \frac{0.313s + 0.2504}{s^3 + 0.55s^2 + 1.075s + 0.25}$$

วิเคราะห์คุณสมบัติของส่วนดำเนินการโดยอาศัยทางเดินรากได้ดังนี้

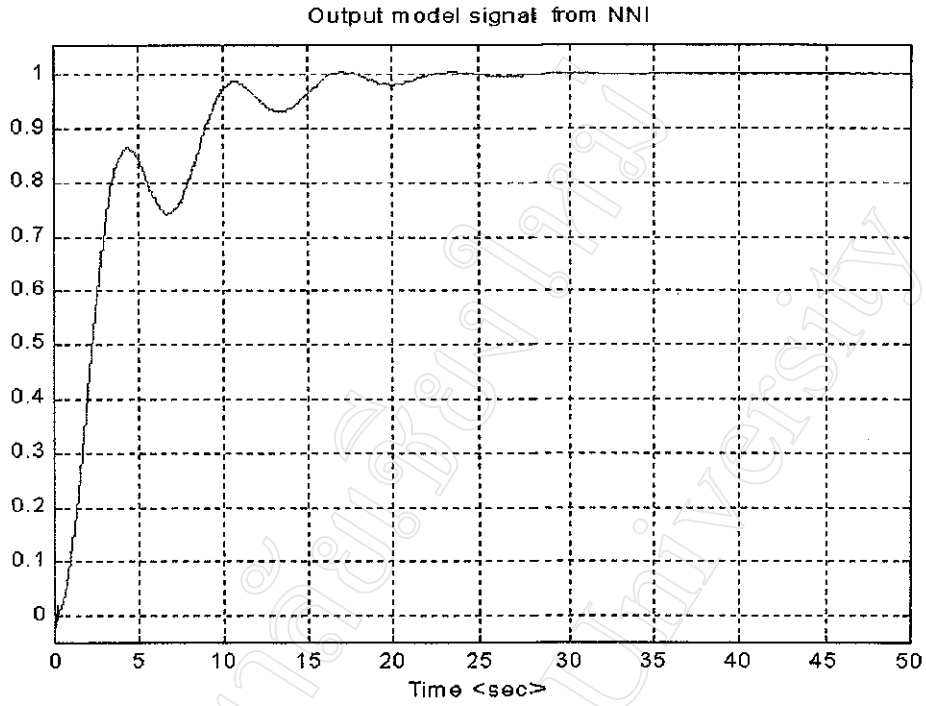


ทางเดินรากที่ได้จากส่วนดำเนินการ

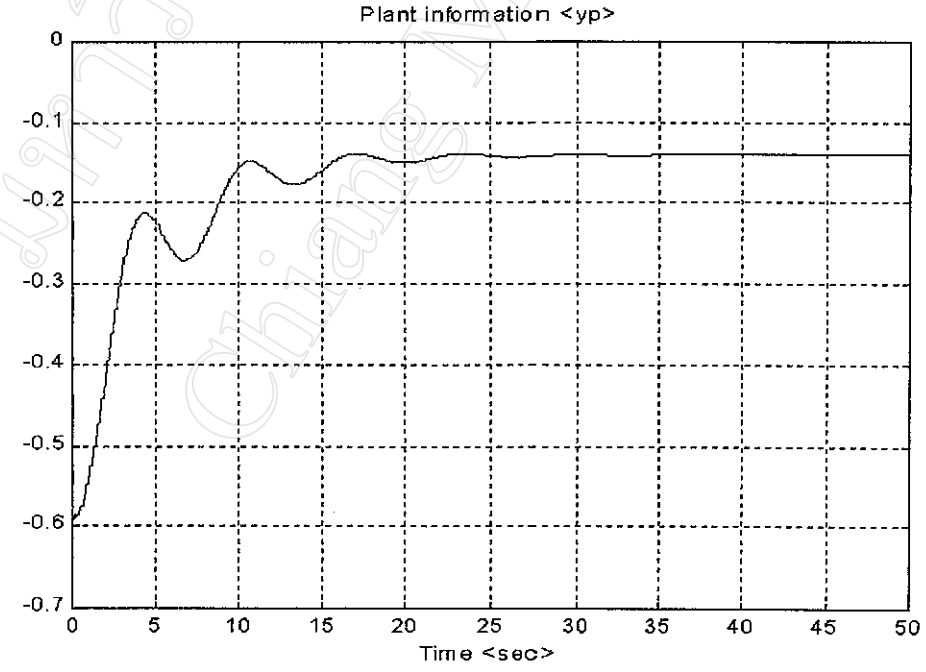
ผลการตอบสนองที่ได้จากระบบควบคุมที่นำเสนอกับส่วนดำเนินการ โดยให้สัญญาณอ้างอิงเป็น unit step สามารถแสดงได้ดังนี้



สัญญาณขาออก y(t)



สัญญาณที่ได้จากโครงข่ายประสาทเทียมที่ใช้ระบุส่วนดำเนินการ



สัญญาณ plant information

ประวัติผู้เขียน

ชื่อ	นายชิเดนทรีย์ ตรีสัตยพันธุ์
วัน เดือน ปี เกิด	30 ธันวาคม 2518
ประวัติการศึกษา	สำเร็จการศึกษามัธยมศึกษาตอนปลาย โรงเรียนกัลยาณีศรีธรรมราช นครศรีธรรมราช ปีการศึกษา 2536 สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า มหาวิทยาลัยเชียงใหม่ ปีการศึกษา 2540

มหาวิทยาลัยเชียงใหม่ Chiang Mai University