

نام و نام خانوادگی : پدرام غیاثوند

نام استاد : دکتر معقولی

گزارش سمینار درس سیستم های فازی

عنوان مقاله شبیه سازی شده :

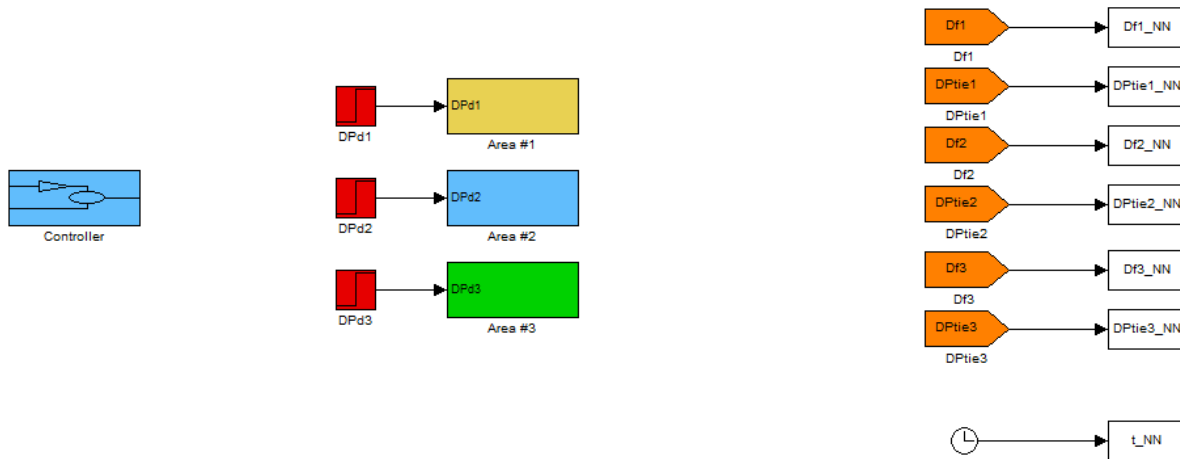
Load Frequency Control of a Multi-Area Power System An Adaptive Fuzzy Logic Approach

توضیح فایل های ارسال شده و نحوه اجرای برنامه :

- 1- فایل های Area1.m , Area2.m و Area3.m که حاوی مدل سه ناحیه سیستم قدرت هستند.
- 2- فایل های NN_C1.m , NN_C2.m و NN_C3.m که کنترل کننده های عصبی را پیاده سازی می کنند.
- 3- فایل های NN_C1.mat , NN_C2.mat و NN_C3.mat که حاوی ساختار کنترل کننده های عصبی هستند.
- 4- فایل های سیمولینک LFC_PID.mdl و LFC_NN.mdl که کنترل PID و کنترل شبکه عصبی را در محیط سیمولینک پیاده سازی می کنند.
- 5- فایل Main.m که فایل اصلی بوده و با اجرای آن کلیه نتایج شبیه سازی بر روی نمودارهایی ترسیم می شوند. (نتایج نهایی در انتها موجود است).

1- فایل های سیمولینک :

شکل زیر نمای یکی از فایل های سیمولینک را نشان می دهد. فایل دیگر مشابه همین فایل است با این تفاوت که حاوی کنترل کننده دیگری است.



در این شکل ناحیه ها و کنترل کننده در بلوک های مختلفی ایجاد شده اند.

2- فایل Main.m :

```
%%
=====
clc;
clear all;
close all;

تعریف متغیرهایی جهت ذخیره سراسری کنترل کننده های عصبی
global NN_Controller1 NN_Controller2 NN_Controller3
بارگذاری کنترل کننده عصبی ناحیه 1

NNTEMP=load('NNC1');
NN_Controller1=NNTEMP.net;

بارگذاری کنترل کننده عصبی ناحیه 2

NNTEMP=load('NNC2');
NN_Controller2=NNTEMP.net;

بارگذاری کنترل کننده عصبی ناحیه 3

NNTEMP=load('NNC3');
NN_Controller3=NNTEMP.net;
=====
```

اجرای شبیه سازی در حالت اول

```
DPd1=0.3;
DPd2=0;
```

```
DPd3=0;
```

اجرای فایل های سیمولینک برای این حالت

```
sim('LFC_PID');
```

```
sim('LFC_NN');
```

ترسیم متغیرهای سیستم بر روی نمودارها

```
figure (1)
plot(t_PID,Df1_PID,'b','linewidth',2);hold on;
plot(t_NN,Df1_NN,'r','linewidth',2);grid on;
ylabel('DF1');
xlabel('time[s]');
legend('PID','NN');
axis([0 60 -0.002 0.002]);
figure (2)
plot(t_PID,DPTie1_PID,'b','linewidth',2);hold on;
plot(t_NN,DPTie1_NN,'r','linewidth',2);grid on;
ylabel('DPTie1');
xlabel('time[s]');
legend('PID','NN');
axis([0 60 -0.5 0.5]);
figure (3)
plot(t_PID,Df2_PID,'b','linewidth',2);hold on;
plot(t_NN,Df2_NN,'r','linewidth',2);grid on;
ylabel('DF1');
xlabel('time[s]');
legend('PID','NN');
axis([0 60 -0.002 0.002]);
figure (4)
plot(t_PID,DPTie2_PID,'b','linewidth',2);hold on;
plot(t_NN,DPTie2_NN,'r','linewidth',2);grid on;
ylabel('DPTie1');
xlabel('time[s]');
legend('PID','NN');
axis([0 60 -0.5 0.5]);
figure (5)
plot(t_PID,Df3_PID,'b','linewidth',2);hold on;
plot(t_NN,Df3_NN,'r','linewidth',2);grid on;
ylabel('DF1');
xlabel('time[s]');
legend('PID','NN');
axis([0 60 -0.002 0.002]);
figure (6)
plot(t_PID,DPTie3_PID,'b','linewidth',2);hold on;
plot(t_NN,DPTie3_NN,'r','linewidth',2);grid on;
ylabel('DPTie1');
xlabel('time[s]');
legend('PID','NN');
axis([0 60 -0.5 0.5]);
%=====
```

اجرای شبیه سازی برای حالت دوم

```
DPd1=0.3;
```

```
DPd2=0.1;
```

```
DPd3=0.01;
```

اجرای فایل های سیمولینک برای حالت دوم

```
sim('LFC_PID');  
sim('LFC_NN');
```

ترسیم نتایج

```
figure (7)  
plot(t_PID,Df1_PID,'b','linewidth',2);hold on;  
plot(t_NN,Df1_NN,'r','linewidth',2);grid on;  
ylabel('DF1');  
xlabel('time[s]');  
legend('PID','NN');  
axis([0 60 -0.002 0.002]);  
figure (8)  
plot(t_PID,DPTie1_PID,'b','linewidth',2);hold on;  
plot(t_NN,DPTie1_NN,'r','linewidth',2);grid on;  
ylabel('DPTie1');  
xlabel('time[s]');  
legend('PID','NN');  
axis([0 60 -0.5 0.5]);  
figure (9)  
plot(t_PID,Df2_PID,'b','linewidth',2);hold on;  
plot(t_NN,Df2_NN,'r','linewidth',2);grid on;  
ylabel('DF1');  
xlabel('time[s]');  
legend('PID','NN');  
axis([0 60 -0.002 0.002]);  
figure (10)  
plot(t_PID,DPTie2_PID,'b','linewidth',2);hold on;  
plot(t_NN,DPTie2_NN,'r','linewidth',2);grid on;  
ylabel('DPTie1');  
xlabel('time[s]');  
legend('PID','NN');  
axis([0 60 -0.5 0.5]);  
figure (11)  
plot(t_PID,Df3_PID,'b','linewidth',2);hold on;  
plot(t_NN,Df3_NN,'r','linewidth',2);grid on;  
ylabel('DF1');  
xlabel('time[s]');  
legend('PID','NN');  
axis([0 60 -0.002 0.002]);  
figure (12)  
plot(t_PID,DPTie3_PID,'b','linewidth',2);hold on;  
plot(t_NN,DPTie3_NN,'r','linewidth',2);grid on;  
ylabel('DPTie1');  
xlabel('time[s]');  
legend('PID','NN');  
axis([0 60 -0.5 0.5]);  
%=====
```

3- فایل های Area :

در این فایل ها برای هر ناحیه، معادلات فضای حالت داده شده در مقاله پیاده سازی شده اند.

```
function dx=AREA1(In)
```

```
%-----  
تفکیک ورودی های تابع
```

```
u1=In(1);  
DPd1=In(2);  
Df1=In(3);  
DPt11=In(4);  
DPg11=In(5);  
DPt21=In(6);  
DPr1=In(7);  
DPg21=In(8);  
DPtie1=In(9);  
Df2=In(10);  
Df3=In(11);
```

بردار حالت سیستم

```
x=[Df1;DPt11;DPg11;DPt21;DPr1;DPg21;DPtie1];
```

```
%-----  
تعریف پارامترها و ماتریس های فضای حالت
```

```
D=0.24;Kr=0.3;M=167;  
R=0.04;Tt1=0.4;Tg1=0.1;  
Tt2=1;Tg2=0.1;Tr=1;  
T12=8.4;T13=2.3;  
Dr1=0;  
Ab11=[-D/M,1/M,0,1/M,0,0,-1/M;  
0,-1/Tt1,1/Tt1,0,0,0,0;  
-1/R/Tg1,0,-1/Tg1,0,0,0,0;  
0,0,0,-1/Tt2,1/Tt2,0,0;  
-Kr/R/Tg2,0,0,0,-1/Tr,-Kr/Tg2+1/Tr,0;  
-1/R/Tg2,0,0,0,0,-1/Tg2,0;  
2*pi*(T12+T13),0,0,0,0,0,0];  
Ab12=[0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
-2*pi*T12,0,0,0,0,0,0];  
Ab13=[0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
0,0,0,0,0,0,0;  
-2*pi*T13,0,0,0,0,0,0];  
Bb1=[0;0;1/Tg1;0;0;0;0];  
Fb1=[1/M;0;0;0;0;0;0];  
Eb1=[0;0;0;0;0;1;0];  
Cb1=[1,0,0,0,0,0,0];  
v1=0.1;B1=0.03;  
g11=1/M*[v1;0;0;0;0;0;0];  
g21=(1/Tt1)*[0;B1;0;0;0;0;0];
```

```
%-----  
محاسبه سمت راست معادلات حالت و ارسال به سیمولینک جهت انتگرال گیری
```

```
dx=Ab11*x+Bb1*u1+Eb1*Dr1+Ab12*[Df2;0;0;0;0;0;0;0]+Ab13*[Df3;0;0;0;0;0;0;0]+g11+g2
1-Fb1*DPd1;
```

دو تابع دیگر مشابه هستند.

4- فایل های NN_C :

در این فایل ها کنترل کننده های عصبی طراحی شده بارگذاری شده و برای ورودی فعلی ارزیابی شده و سیگنال کنترلی به سیستم ارسال می شود.

```
function u1=NN_C1(In)
global NN_Controller1
u1=sim(NN_Controller1,In);
```

ارزیابی کنترل کننده عصبی

دو فایل دیگر مشابه هستند.

با اجرای شبیه سازی نتایج زیر حاصل می شوند. 6 نمودار اول برای مورد اول و 6 نمودار دوم برای مورد دوم هستند.

(در مجموع 12 نمودار)

