

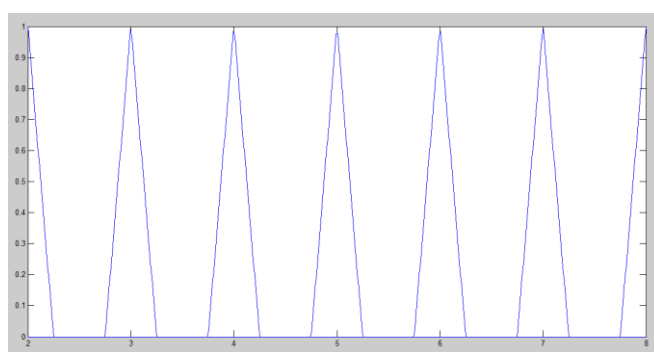
کمترین مربعات خطا

روش بازگشتی کمترین مربعات خطا

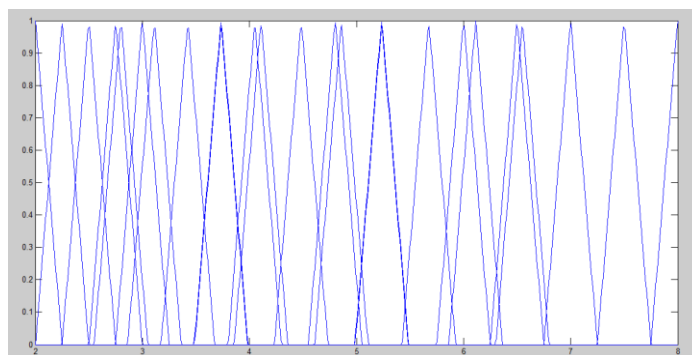
مراکز گروه های فازی سیستم زیر را بوسیله ی روش کمترین مجموع مربعات خطا به گونه ای تصحیح کنید که خطای تقریب کاهش یابد.

$$y = x_1^2 + x_1x_2$$

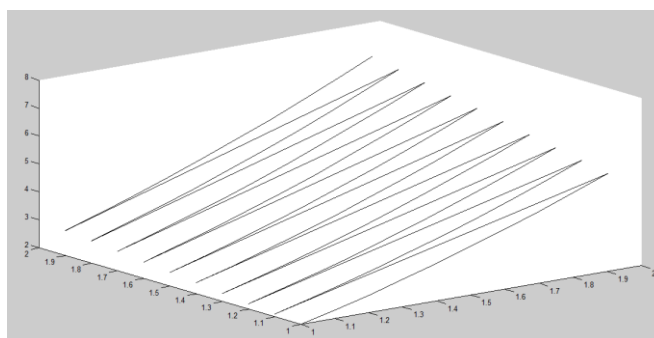
نمایش مراکز در نظر گرفته شده برای توابع تعلق خروجی قبل از اعمال روش کمترین مجموع مربعات خطا:



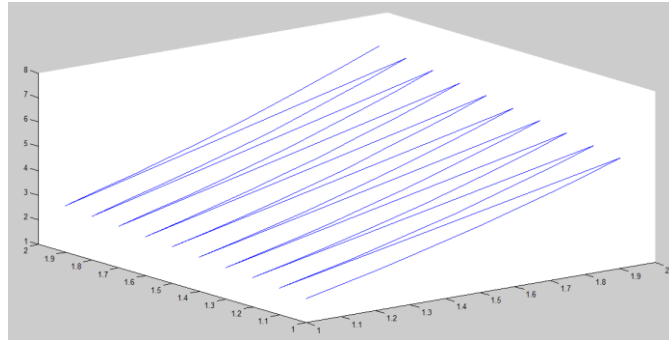
نمایش مراکز در نظر گرفته شده برای توابع تعلق خروجی بعد از اعمال روش کمترین مجموع مربعات خطا:



نمایش سیستم اصلی:

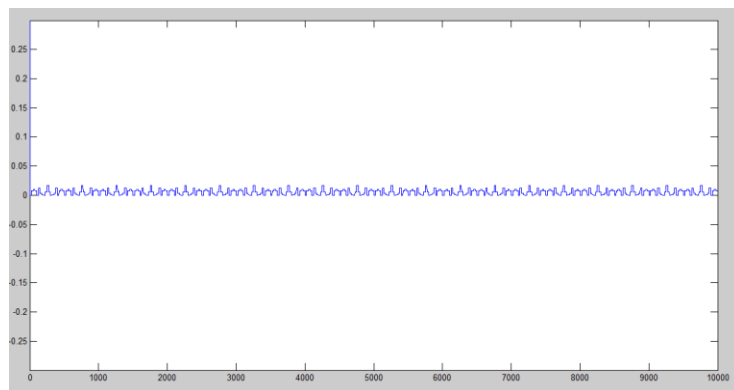


نمایش سیستم تقریب زده:



خطای تقریب سیستم از مقدار ۰.۶۴۲۶ در تقریب مدل سیستم به ۰.۰۱۶۷ بعد از اعمال روش کمترین مجموع مربعات خطا تقلیل پیدا می کند.

نمایش خطا:



برنامه نوشته شده در متلب به صورت زیر می باشد:

لازم به ذکر است که ماتریس assist قرار داده شده در ابتدای برنامه از برنامه های قبلی تقریب مدل سیستم در این قسمت قرار داده شده است.

```
clc
clear all
close all

assist=[1.0000    1.0000    1.0000   10.0000   10.0000    1.0000    1.0000    1.0000    2.0000    2.0000    1.0000    1.0000;
2.0000    1.0000    1.2000   10.0000   20.0000    1.0000    0.8000    0.8000    2.2000    2.0000    1.0000    1.2500;
3.0000    1.0000    1.4000   10.0000   30.0000    1.0000    0.6000    0.6000    2.4000    2.0000    1.0000    1.5000;
4.0000    1.0000    1.6000   10.0000   40.0000    1.0000    0.6000    0.6000    2.6000    3.0000    1.0000    1.5000;
5.0000    1.0000    1.8000   10.0000   50.0000    1.0000    0.8000    0.8000    2.8000    3.0000    1.0000    1.7500;
6.0000    1.0000    2.0000   10.0000   60.0000    1.0000    1.0000    1.0000    3.0000    3.0000    1.0000    2.0000;
7.0000    1.2000    1.0000   20.0000   10.0000    0.8000    1.0000    0.8000    2.6400    3.0000    1.2500    1.0000;
8.0000    1.2000    1.2000   20.0000   20.0000    0.8000    1.0000    0.8000    2.8800    3.0000    1.2500    1.2500;
9.0000    1.2000    1.4000   20.0000   30.0000    0.8000    1.0000    0.8000    3.1200    3.0000    1.2500    1.5000;
10.0000   1.2000    1.6000   20.0000   40.0000    0.8000    1.0000    0.8000    3.3600    3.0000    1.2500    1.5000;
11.0000   1.2000    1.8000   20.0000   50.0000    0.8000    0.8000    0.6400    3.6000    4.0000    1.2500    1.7500;
12.0000   1.2000    2.0000   20.0000   60.0000    0.8000    1.0000    0.8000    3.8400    4.0000    1.2500    2.0000;
13.0000   1.4000    1.0000   30.0000   10.0000    0.6000    1.0000    0.6000    3.3600    3.0000    1.5000    1.0000;
14.0000   1.4000    1.2000   30.0000   20.0000    0.6000    1.0000    0.6000    3.6400    4.0000    1.5000    1.2500;
15.0000   1.4000    1.4000   30.0000   30.0000    0.6000    1.0000    0.6000    3.9200    4.0000    1.5000    1.5000;
16.0000   1.4000    1.6000   30.0000   40.0000    0.6000    1.0000    0.6000    4.2000    4.0000    1.5000    1.5000;
17.0000   1.4000    1.8000   30.0000   50.0000    0.6000    0.8000    0.4800    4.4800    4.0000    1.5000    1.7500;
18.0000   1.4000    2.0000   30.0000   60.0000    0.6000    1.0000    0.6000    4.7600    5.0000    1.5000    2.0000;
19.0000   1.6000    1.0000   10.0000   10.0000    0.6000    1.0000    0.6000    4.1600    4.0000    1.5000    1.0000;
20.0000   1.6000    1.2000   30.0000   20.0000    0.6000    1.0000    0.6000    4.4800    4.0000    1.5000    1.2500;
```

21.0000	1.6000	1.4000	30.0000	30.0000	0.6000	1.0000	0.6000	4.8000	5.0000	1.5000	1.5000;
22.0000	1.6000	1.6000	30.0000	30.0000	0.6000	1.0000	0.6000	5.1200	5.0000	1.5000	1.5000;
23.0000	1.6000	1.8000	30.0000	40.0000	0.6000	0.8000	0.4800	5.4400	5.0000	1.5000	1.7500;
24.0000	1.6000	2.0000	30.0000	50.0000	0.6000	1.0000	0.6000	5.7600	6.0000	1.5000	2.0000;
25.0000	1.8000	1.0000	40.0000	10.0000	0.8000	1.0000	0.8000	5.0400	5.0000	1.7500	1.0000;
26.0000	1.8000	1.2000	40.0000	20.0000	0.8000	1.0000	0.8000	5.4000	5.0000	1.7500	1.2500;
27.0000	1.8000	1.4000	40.0000	30.0000	0.8000	1.0000	0.8000	5.7600	6.0000	1.7500	1.5000;
28.0000	1.8000	1.6000	40.0000	30.0000	0.8000	1.0000	0.8000	6.1200	6.0000	1.7500	1.5000;
29.0000	1.8000	1.8000	40.0000	40.0000	0.8000	0.8000	0.6400	6.4800	6.0000	1.7500	1.7500;
30.0000	1.8000	2.0000	40.0000	50.0000	0.8000	1.0000	0.8000	6.8400	7.0000	1.7500	2.0000;
31.0000	2.0000	1.0000	50.0000	10.0000	1.0000	1.0000	1.0000	6.0000	6.0000	2.0000	1.0000;
32.0000	2.0000	1.2000	50.0000	20.0000	1.0000	1.0000	1.0000	6.4000	6.0000	2.0000	1.2500;
33.0000	2.0000	1.4000	50.0000	30.0000	1.0000	1.0000	1.0000	6.8000	7.0000	2.0000	1.5000;
34.0000	2.0000	1.6000	50.0000	30.0000	1.0000	1.0000	1.0000	7.2000	7.0000	2.0000	1.5000;
35.0000	2.0000	1.8000	50.0000	40.0000	1.0000	0.8000	0.8000	7.6000	8.0000	2.0000	1.7500;
36.0000	2.0000	2.0000	50.0000	50.0000	1.0000	1.0000	1.0000	8.0000	8.0000	2.0000	2.0000];

```

b=0.25;
ppp=length(asist);
figure
y=linspace(2,8,500);
for i=1:ppp
    plot(y,trimf(y,[asist(i,10)-b asist(i,10) asist(i,10)+b]))
    hold on
end
%hade aghale morabaat
%-----
ppp=length(asist);
nvorodi=2;
%p nemidonam chie vali p=m
%majmoe hadeaghal morabaat
%p soton haye matrix Z ya B ast
%m satr haye matrix Z ya B ast
for k=1:ppp
    for sib=1:ppp
        asist;
        %sorate b(jozve)---->esmesh inja sorat_b
        sorat_b(k,sib)=1;
        for j=1:nvorodi
            wert=trimf(asist(k,1+j),[asist(sib,10+j)-b asist(sib,10+j)
asist(sib,10+j)+b]);
            sorat_b(k,sib)=sorat_b(k,sib)*wert;
        end
        sorat_b(k,sib);
        %makhraj_b----->makhraje b(jozve)
        %B(jozve)----->B
        makhraj_b(k,sib)=0;
        q=1;
        for i=1:ppp
            for j=1:nvorodi
                wert=trimf(asist(k,1+j),[asist(i,10+j)-b asist(i,10+j)
asist(i,10+j)+b]);
                q=q*wert;
            end
            makhraj_b(k,sib)=makhraj_b(k,sib)+q;
            q=1;
        end
        makhraj_b(k,sib);
        B(k,sib)=sorat_b(k,sib)/makhraj_b(k,sib);
    end
    Y(k,1)=asist(k,9);
end
teta=(inv(B'*B))*B'*Y;
asist(:,10)=teta

f=1;
for i=1:ppp

```

```

sum1=0;
sum2=0;
    for j=1:ppp
        www1=trimf(asist(i,2),[asist(j,11)-b asist(j,11)
asist(j,11)+b]);
        www2=trimf(asist(i,3),[asist(j,12)-b asist(j,12)
asist(j,12)+b]);
        ww=www1*www2;

        sum1=sum1+ww.*asist(j,10);
        sum2=sum2+ww;
    end
    fx1(f)=sum1./sum2;
    fx2(f)=asist(i,2).^2+asist(i,2)*asist(i,3);
    ee(f)=abs(fx1(f)-fx2(f));
    f=f+1;
end
f=1;
flag1=1;
flag2=1;
rt1=1;
rt2=1;
while(flag1)
    while(flag2)

        for i=1:ppp
            sum1=0;
            sum2=0;
            for j=1:ppp
                www1=trimf(rt1,[asist(j,11)-b asist(j,11) asist(j,11)+b]);
                www2=trimf(rt2,[asist(j,12)-b asist(j,12) asist(j,12)+b]);
                ww=www1*www2;

                sum1=sum1+ww.*asist(j,10);
                sum2=sum2+ww;
            end
            fx1(f)=sum1./sum2;
            fx2(f)=rt1.^2+rt2*rt1;

            r1(f)=rt1;
            r2(f)=rt2;

            ee(f)=abs(fx1(f)-fx2(f));
            ee';
            f=f+1;
        end
        rt1=rt1+0.1;
        if(rt1>=2)
            flag2=0;
        end
        rt1;
    end
    flag2=1;
    rt2;
    rt1=1;
    rt2=rt2+0.1;
    if(rt2>=2)
        flag1=0;
    end
end
end

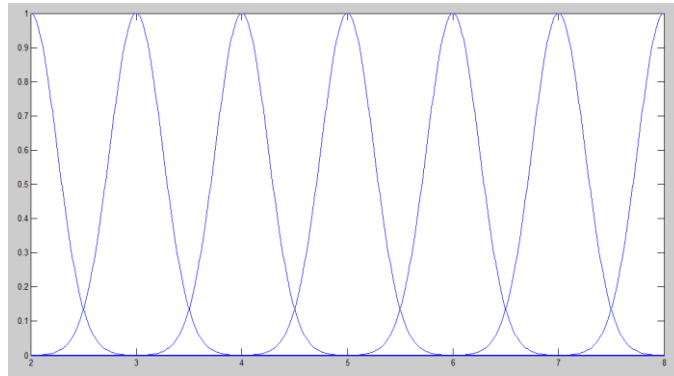
```

```
figure
error_max=max(ee)
plot3(r1,r2,fx1)
figure
plot3(r1,r2,fx2,'k')
figure
y=linspace(2,8,500);
for i=1:ppp
    plot(y,trimf(y,[asist(i,10)-b asist(i,10) asist(i,10)+b]))
    hold on
end
```

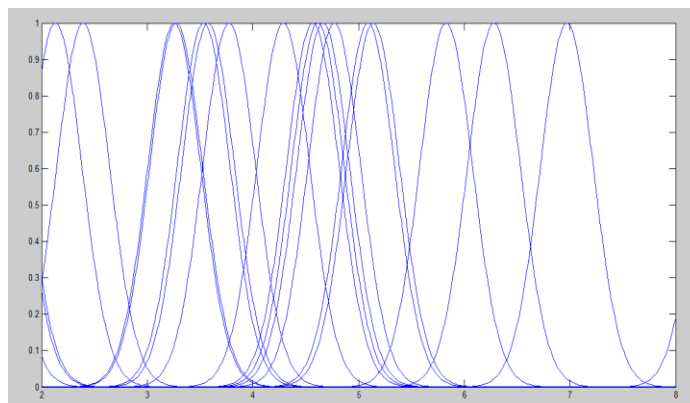
مراکز گروه های فازی سیستم زیر را بوسیله ی روش بازگشتی کمترین مجموع مربعات خطا به گونه ای تصحیح کنید که خطای تقریب کاهش یابد.

$$y = x_1^2 + x_1x_2$$

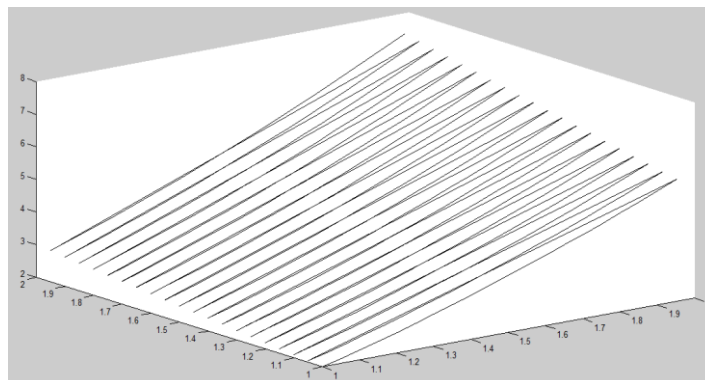
نمایش مراکز فرضی برای توابع تعلق خروجی قبل از اعمال روش بازگشتی کمترین مجموع مربعات خطا:



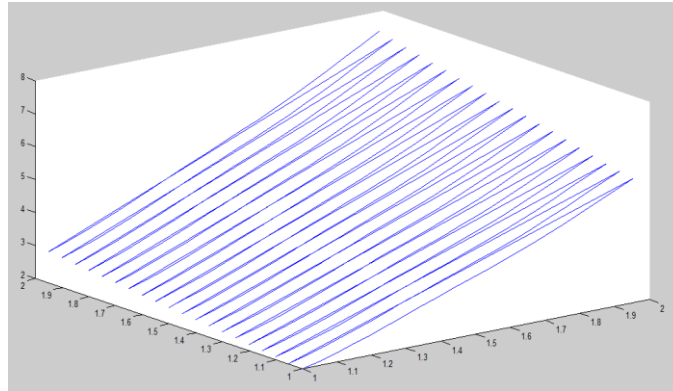
نمایش مراکز فرضی برای توابع تعلق خروجی بعد از اعمال روش بازگشتی کمترین مجموع مربعات خطا:



نمایش سیستم اصلی:

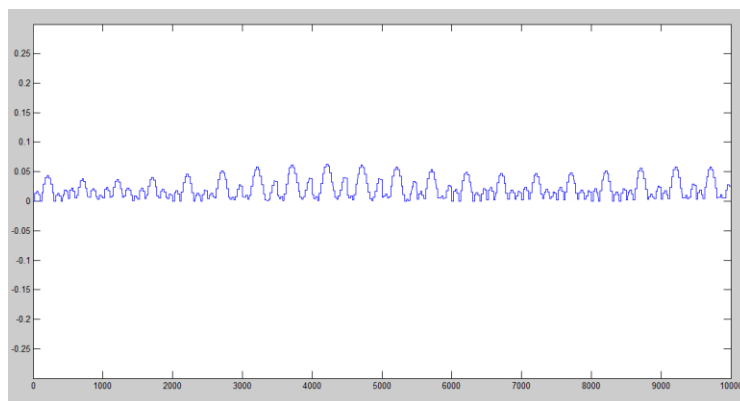


نمایش سیستم تقریب زده:



خطای تقریب سیستم از مقدار ۰.۶۴۲۶ در تقریب مدل سیستم به ۰.۰۶۲۸ بعد از اعمال روش بازگشتی کمترین مجموع مربعات خطا تقلیل پیدا می کند.

نمایش خطا:



برنامه نوشته شده در متلب به صورت زیر می باشد:

لازم به ذکر است که ماتریس assist قرار داده شده در ابتدای برنامه از برنامه های قبلی تقریب مدل سیستم در این قسمت قرار داده شده است.

```
clc
clear all
close all
asist=[1.0000    1.0000    1.0000   10.0000   10.0000    1.0000         0         0    2.0000    2.0000    1.0000    1.0000;
2.0000    1.0000    1.2000   10.0000   20.0000    0.9802         0         0    2.2000    2.0000    1.0000    1.2500;
4.0000    1.0000    1.6000   10.0000   30.0000    1.0000         0         0    2.6000    3.0000    1.0000    1.5000;
5.0000    1.0000    1.8000   10.0000   40.0000    1.0000         0         0    2.8000    3.0000    1.0000    1.7500;
6.0000    1.0000    2.0000   10.0000   50.0000    1.0000         0         0    3.0000    3.0000    1.0000    2.0000;
7.0000    1.2000    1.0000   20.0000   10.0000    1.0000         0         0    2.6400    3.0000    1.2500    1.0000;
8.0000    1.2000    1.2000   20.0000   20.0000    1.0000         0         0    2.8800    3.0000    1.2500    1.2500;
10.0000    1.2000    1.6000   20.0000   30.0000    1.0000         0         0    3.3600    3.0000    1.2500    1.5000;
11.0000    1.2000    1.8000   20.0000   40.0000    0.9802         0         0    3.6000    4.0000    1.2500    1.7500;
12.0000    1.2000    2.0000   20.0000   50.0000    1.0000         0         0    3.8400    4.0000    1.2500    2.0000;
19.0000    1.6000    1.0000   30.0000   10.0000    1.0000         0         0    4.1600    4.0000    1.5000    1.0000;
20.0000    1.6000    1.2000   30.0000   20.0000    1.0000         0         0    4.4800    4.0000    1.5000    1.2500;
22.0000    1.6000    1.6000   30.0000   30.0000    1.0000         0         0    5.1200    5.0000    1.5000    1.5000;
```

```

23.0000    1.6000    1.8000    30.0000    40.0000    0.9802    0    0    5.4400    5.0000    1.5000    1.7500;
24.0000    1.6000    2.0000    30.0000    50.0000    1.0000    0    0    5.7600    6.0000    1.5000    2.0000;
25.0000    1.8000    1.0000    40.0000    10.0000    1.0000    0    0    5.0400    5.0000    1.7500    1.0000;
26.0000    1.8000    1.2000    40.0000    20.0000    0.9802    0    0    5.4000    5.0000    1.7500    1.2500;
28.0000    1.8000    1.6000    40.0000    30.0000    0.9231    0    0    6.1200    6.0000    1.7500    1.5000;
29.0000    1.8000    1.8000    40.0000    40.0000    0.9802    0    0    6.4800    6.0000    1.7500    1.7500;
30.0000    1.8000    2.0000    40.0000    50.0000    1.0000    0    0    6.8400    7.0000    1.7500    2.0000;
31.0000    2.0000    1.0000    50.0000    10.0000    1.0000    0    0    6.0000    6.0000    2.0000    1.0000;
32.0000    2.0000    1.2000    50.0000    20.0000    1.0000    0    0    6.4000    6.0000    2.0000    1.2500;
34.0000    2.0000    1.6000    50.0000    30.0000    1.0000    0    0    7.2000    7.0000    2.0000    1.5000;
35.0000    2.0000    1.8000    50.0000    40.0000    0.9802    0    0    7.6000    8.0000    2.0000    1.7500;
36.0000    2.0000    2.0000    50.0000    50.0000    1.0000    0    0    8.0000    8.0000    2.0000    2.0000];
b=0.25;
ppp=length(asist);
y=linspace(2,8,500);
for i=1:ppp
    plot(y,gaussmf(y,[b asist(i,10)]))
    hold on
end
disp(asist)
nvorodi=2;
%p nemidonam chie vali p=m
%majmoe hadeaghal morabaat
%p jozve soton haye matrix Z ya B ast
%m jozve(tedad ghanon)satr haye matrix Z ya B ast
for k=1:ppp
    for sib=1:ppp
        asist;
        %sorate b(jozve)---->esmesh inja sorat_b
        sorat_b(k,sib)=1;
        for j=1:nvorodi
            wert=gaussmf(asist(k,1+j),[b asist(sib,10+j)]);
            sorat_b(k,sib)=sorat_b(k,sib)*wert;
        end
        sorat_b(k,sib);
        %makhraj_b----->makhraje b(jozve)
        %B(jozve)----->B
        makhraj_b(k,sib)=0;
        q=1;
        for i=1:ppp
            for j=1:nvorodi
                wert=gaussmf(asist(k,1+j),[b asist(i,10+j)]);
                q=q*wert;
            end
            makhraj_b(k,sib)=makhraj_b(k,sib)+q;
            q=1;
        end
        makhraj_b(k,sib);
        bb(k,sib)=sorat_b(k,sib)/makhraj_b(k,sib);
    end
end
bb=bb';
zarib=100;
komaki_teta=asist(:,10);
p_p=0;
n=length(bb);
flg=1;
%in 2 be hichi rabt nadare va faghat bayad 2 bashe!
for k=1:2
    if(k-1==0)
        komaki_pp=zarib;
    else
        while(flg)
            k_p=komaki_pp*bb*(inv((bb'*komaki_pp*bb)+1));
            komaki_kp=k_p;
            teta_p=komaki_teta+k_p*(asist(:,9)-bb'*komaki_teta)

```



```

        if(teta_p-komaki_teta<0.000000000000000000000001)
            flg=0;
        end
        komaki_teta=teta_p;
        p_p=komaki_pp-
        (komaki_pp*bb)*(inv((b'*komaki_pp*b)+1))*(b'*komaki_pp);
        komaki_pp=p_p;
    end
end
f=1;
asist(:,10)=teta_p;
for i=1:ppp
    sum1=0;
    sum2=0;
    for j=1:ppp
        www1=gaussmf(asist(i,2),[b asist(j,11)]);
        www2=gaussmf(asist(i,3),[b asist(j,12)]);
        ww=www1*www2;
        sum1=sum1+ww.*asist(j,10);
        sum2=sum2+ww;
    end
    fx1(f)=sum1./sum2;
    fx2(f)=asist(i,2).^2+asist(i,2)*asist(i,3);
    eee(f)=abs(fx1(f)-fx2(f));

    f=f+1;
end
f=1;
flag1=1;
flag2=1;
rt1=1;
rt2=1;
while(flag1)
    while(flag2)
        for i=1:ppp
            sum1=0;
            sum2=0;
            for j=1:ppp
                www1=gaussmf(rt1,[b asist(j,11)]);
                www2=gaussmf(rt2,[b asist(j,12)]);
                ww=www1*www2;

                sum1=sum1+ww.*asist(j,10);
                sum2=sum2+ww;
            end
            fx1(f)=sum1./sum2;
            fx2(f)=rt1.^2+rt2*rt1;

            r1(f)=rt1;
            r2(f)=rt2;
            ee(f)=abs(fx1(f)-fx2(f));
            ee';
            f=f+1;
        end
        rt1=rt1+0.05;
        if(rt1>=2)
            flag2=0;
        end
        rt1;
    end
end

```

```
        flag2=1;
        rt2;
        rt1=1;
        rt2=rt2+0.05;
        if(rt2>=2)
            flag1=0;
        end
    end
    figure
    error_max=max(ee)
    plot3(r1,r2,fx1)
    figure
    plot3(r1,r2,fx2,'k')
    figure
    y=linspace(2,8,500);
    for i=1:ppp
        plot(y,gaussmf(y,[b asist(i,10)]))
        hold on
    end
```

دانشکده برق و رباتیک

سیستم های کنترل فازی

شبیه سازی مرتبط با فصل ۱۴

کمترین مربعات خطا

روش بازگشتی کمترین مربعات خطا