clc();

clear all;

close all;

%number of system points

N=2000;

inp=randn(N,1);

n=randn(N,1);

[b,a]=butter(2,0.25);

Gz=tf(b,a,-1);%transfer function of filter

%y(n)=b1\*u(n)+b2\*u(n-1)+b3\*u(n-2)-a1\*y(n-1)-a2\*y(n-2)

h=[b -a(2:length(a))];

%channel system order fixed as we have 5 elements(2 in a and 3 in b)

inporder=3;

outorder=2;

sysorder=inporder+outorder;

y=lsim(Gz,inp);

%add noise

n=n\*std(y)/(15\*std(n));

d=y+n;

totallength=size(d,1);

%take 50points for training(N-inporder 47=50-3)

N=50;

%begin of the alg

%forgetting factor

lamda=0.999;

%initial p matrix

delta=1e2;

P=delta\*eye(sysorder);

w=zeros(sysorder,1);

for n=inporder:N

%u(n),u(n-1),u(n-2)

u=inp(n:-1:n-inporder+1);

%d(n-1),d(n-2)

outp=d(n-1:-1:n-outorder);

u=[u;outp];

phi=u'\*P;

k=phi'/(lamda+phi\*u);

y(n)=w'\*u;

e(n)=d(n)-y(n);

w=w+k\*e(n);

P=(P-k\*phi)/lamda;

%just for plotting

Recordedw(1:sysorder,n)=w;

end

%check for results

for n=N+1:totallength

%u(n),u(n-1),u(n-2)

u=inp(n:-1:n-inporder+1);

%d(n-1),d(n-2)

outp=d(n-1:-1:n-outorder);

u=[u;outp];

y(n)=w'\*u;

e(n)=d(n)-y(n);

end

hold on

plot(d)

plot(y,'r');

title('system output');

xlabel('samples');

ylabel('true and estimated output');

legend('channel response with noise','channel response');

figure

semilogy((abs(e)));

title('error curve');

xlabel('samples');

ylabel('error value');

figure

plot(h,'r+')

hold on

plot(w,'.');

legend('filter weights','estimated filter weights');

title('comparison of the filter weights and estimated weights');

hold off

figure

plot(Recordedw(1:sysorder,sysorder:N)');

title('estimated weights convergence');

xlabel('samples');

ylabel('weights value');

%axis([1 N-sysorder min(min(Recordedw(1:sysorder,sysorder:N)')) max(max(Recordedw(1:sysorder,sysorder:N)')) ]);