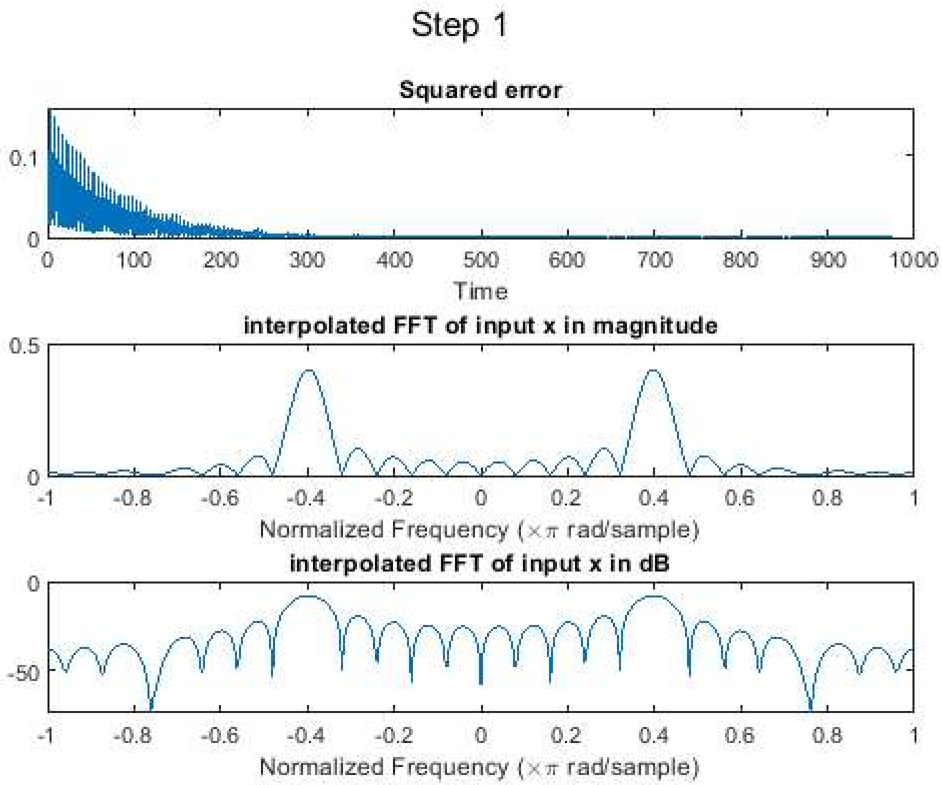
Contents


```
close all
clear
clc
```

Step 1

```
M = 25;
L = 1000;
mu = 0.001;
a = 0.1;

w      = zeros(M,1);
yout   = zeros(M,1);
errout = zeros(M,1);

x = cos(2*pi*0.2*(0:(L-1))) + cos( 2*pi*0.38*(0:(L-1)))+a*randn(1,L);  % this is x
d = 0.4*cos(2*pi*0.2*(0:(L-1)) + pi/5);   % this is d

for n = 1:L-M
    xn = x(n:n+M-1)';
    yn = w'*xn;
    yout(n) = yn;
    en = d(n) - yn;
    errout(n) = en;
    w = w + mu * en * xn;

end
N  = L;
fv = (0:(N-1)) - floor(N/2);
fv = 2*fv/N;
fvM0 = (0:(512-1)) - floor(512/2);
fvM0 = 2*fvM0/512;

figure()
sgtitle('Step 1')
subplot(3,1,1)
plot(errout.*errout);
xlabel('Time')
title('Squared error')
subplot(3,1,2)
plot(fvM0, abs(fftshift(fft(w,512))));
xlabel('Normalized Frequency (\times\pi rad/sample)')
title('interpolated FFT of input x in magnitude')
subplot(3,1,3)
plot(fvM0, mag2db(abs(fftshift(fft(w,512)))));
xlabel('Normalized Frequency (\times\pi rad/sample)')
title('interpolated FFT of input x in dB')
```

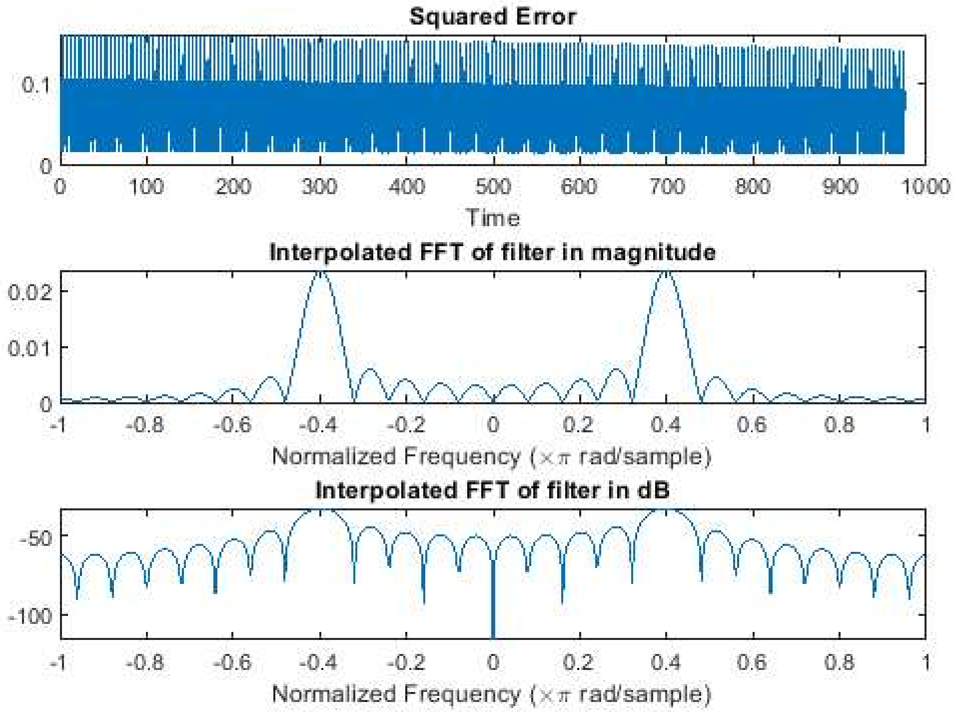


Step 2

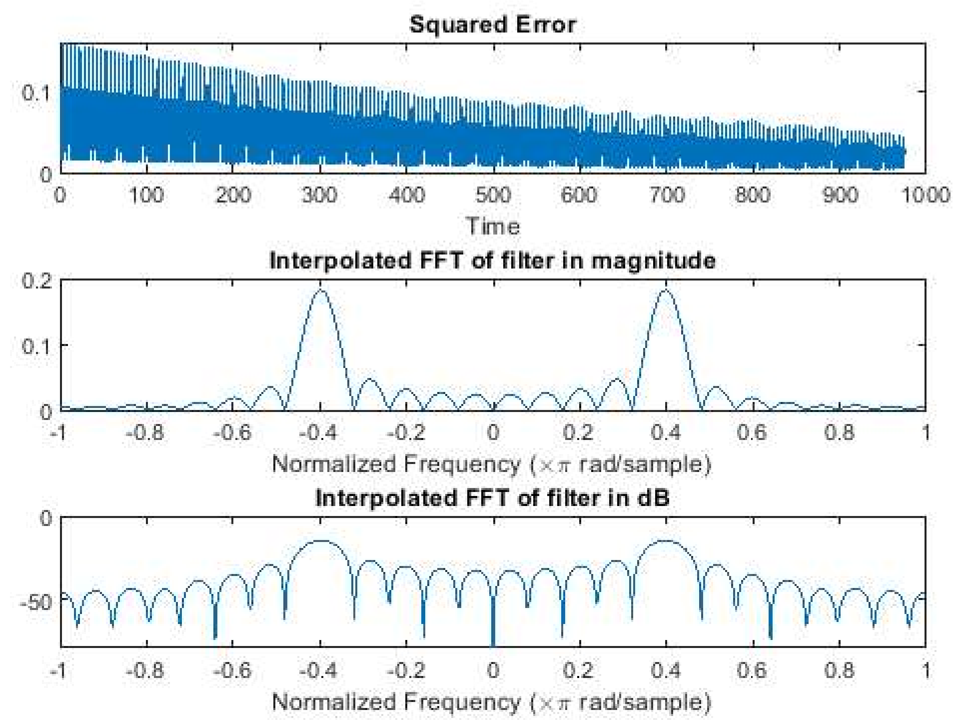
```
i=0;
for mu = [0.00001 0.0001 0.01 0.1 0.001 0.0001]
    M = 25;
    L = 1000;
    a = 0.1;

    if i == 4
        a = 0.5;
    end

    if i == 5
        L = 5000;
        a = 0.1;
    end

    w      = zeros(M,1);
    yout   = zeros(M,1);
    errout = zeros(M,1);

    x = cos(2*pi*0.2*(0:(L-1))) + cos( 2*pi*0.38*(0:(L-1)))+a*randn(1,L);  % this is x
    d = 0.4*cos(2*pi*0.2*(0:(L-1)) + pi/5);   % this is d

    for n = 1:L-M
        xn = x(n:n+M-1)';
        yn = w'*xn;
        yout(n) = yn;
        en = d(n) - yn;
        errout(n) = en;
        w = w + mu * en * xn;
    end
```

```
    N  = L;
    fv = (0:(N-1)) - floor(N/2);
    fv = 2*fv/N;
    fvM0 = (0:(512-1)) - floor(512/2);
    fvM0 = 2*fvM0/512;

    figure()
    sgtitle("Step 2 : mu = " + mu + ", a = " + a + ", L = " + L)
    subplot(3,1,1)
    plot(errout.*errout);
    xlabel('Time')
    title('Squared Error')
    subplot(3,1,2)
    plot(fvM0, abs(fftshift(fft(w,512))));
    xlabel('Normalized Frequency (\times\pi rad/sample)')
    title('Interpolated FFT of filter in magnitude')
    subplot(3,1,3)
    plot(fvM0, mag2db(abs(fftshift(fft(w,512)))));
    xlabel('Normalized Frequency (\times\pi rad/sample)')
    title('Interpolated FFT of filter in dB')
    i = i+1;
end
```

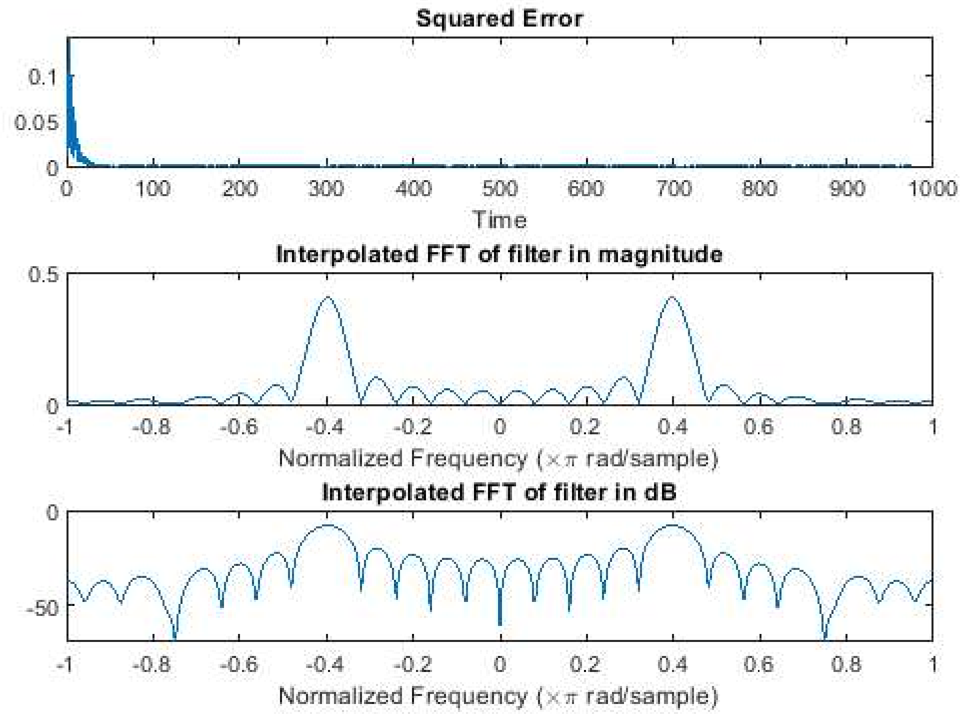
Step 2 : mu = 1e-05, a = 0.1, L = 1000



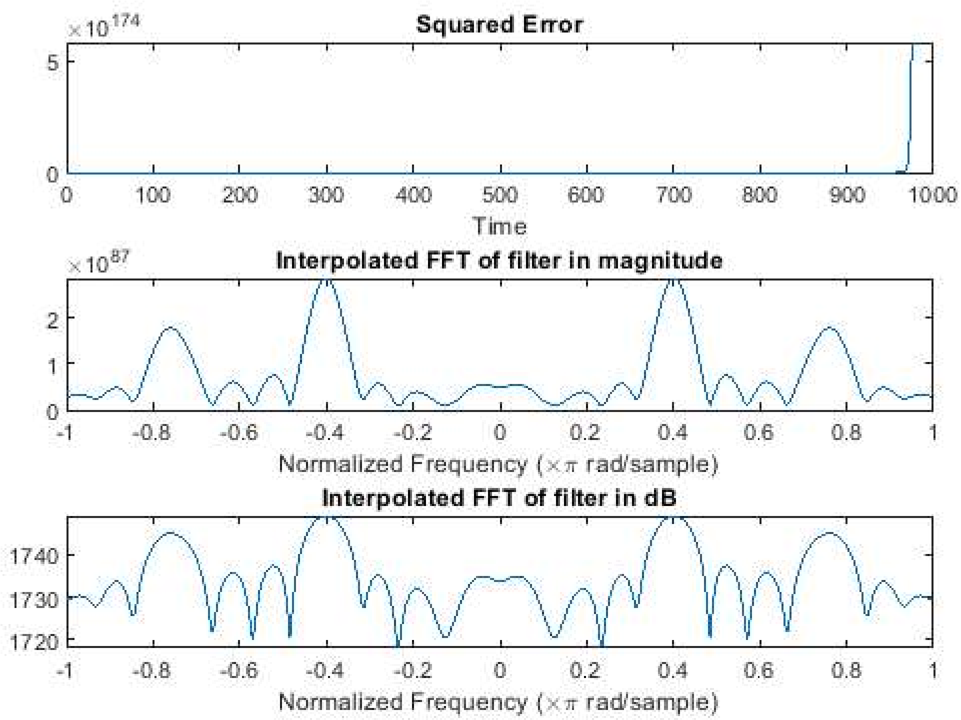
Step 2 : mu = 0.0001, a = 0.1, L = 1000

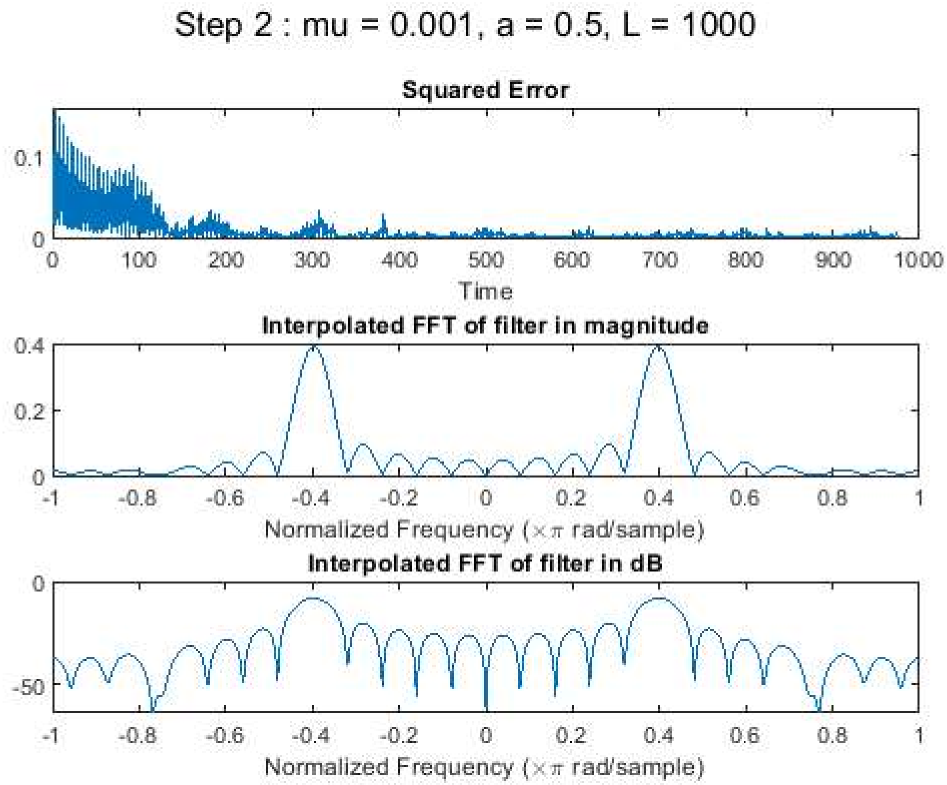


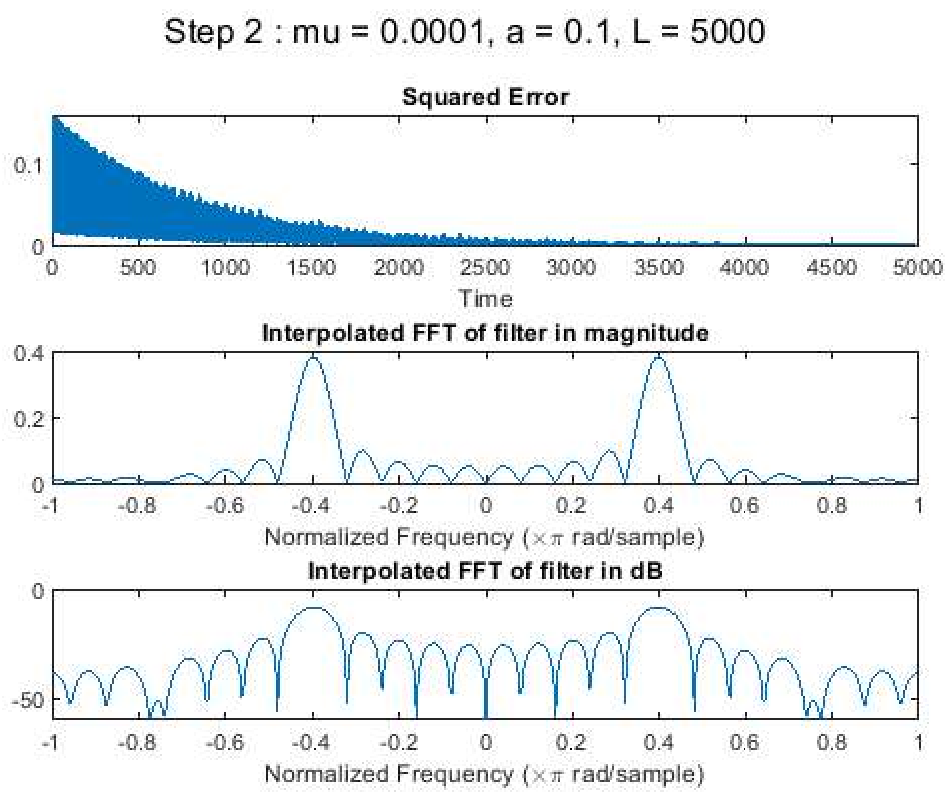
Step 2 : mu = 0.01, a = 0.1, L = 1000



Step 2 : mu = 0.1, a = 0.1, L = 1000





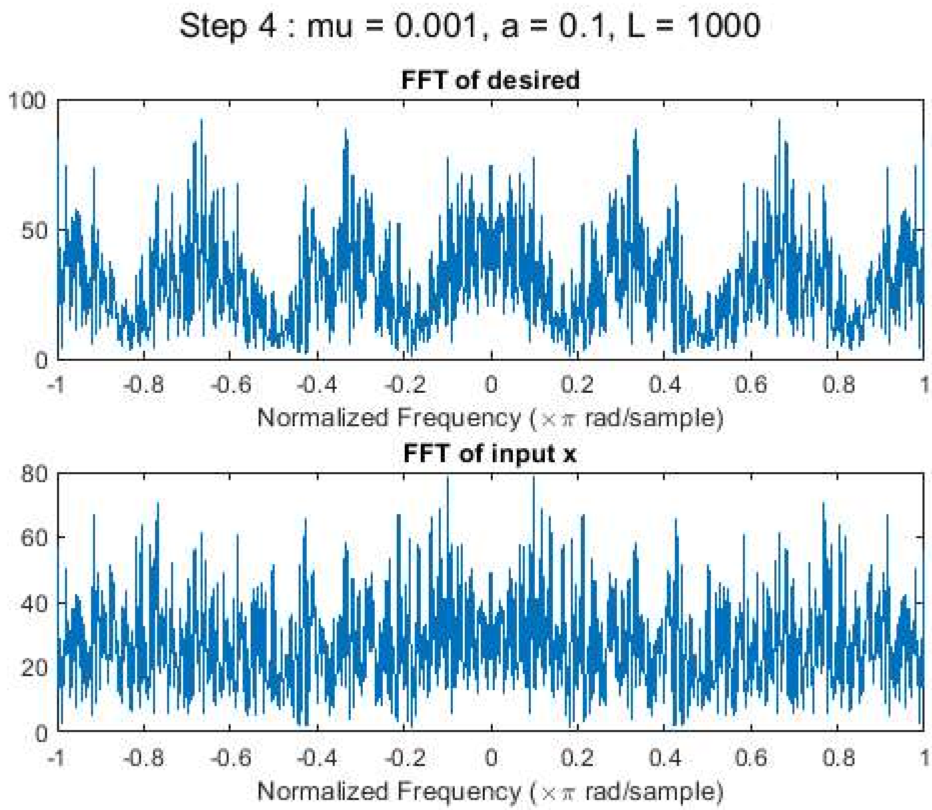


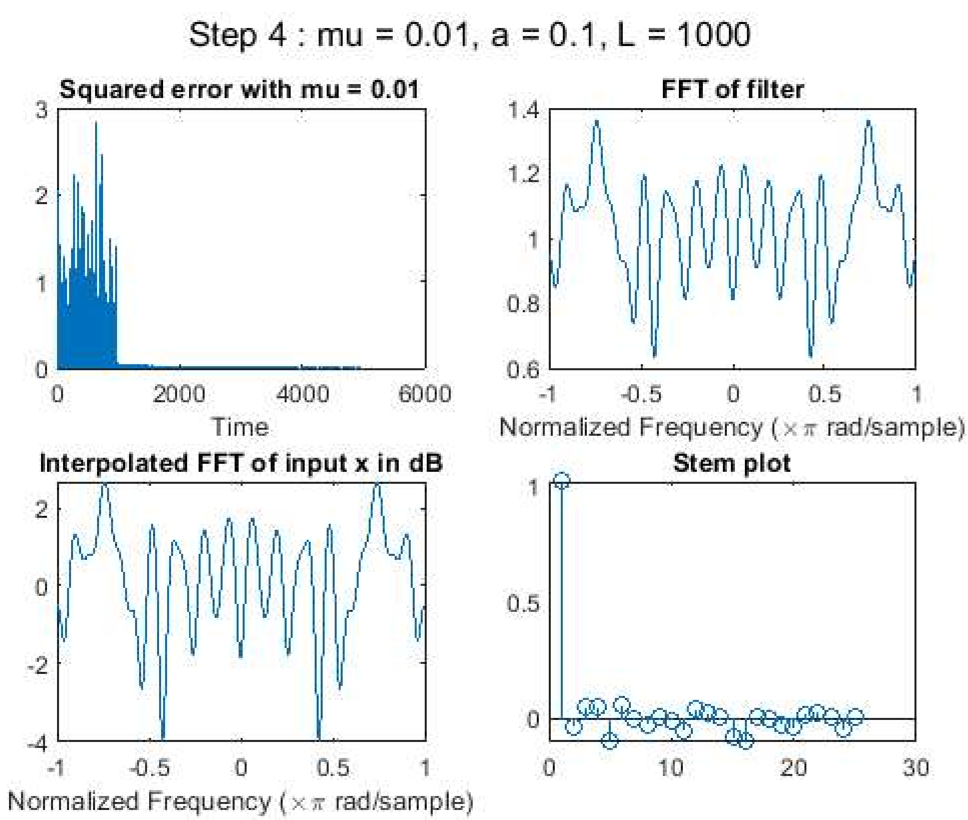
Step 3

```
M  = 25;
L  = 1000;
a  = 0.1;
```

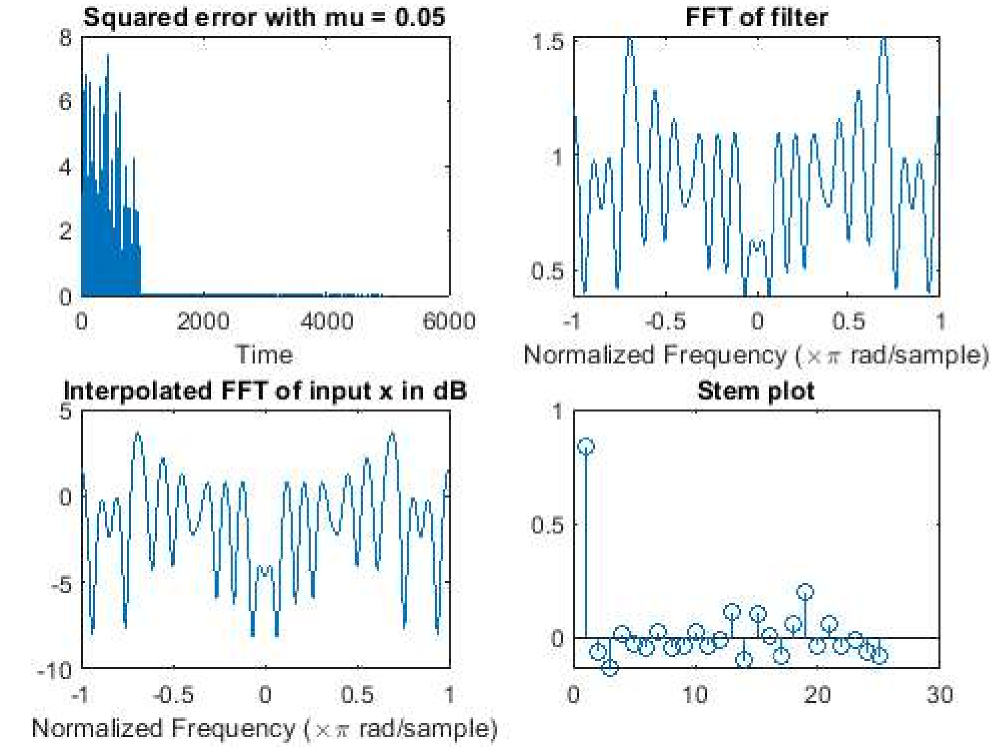
```
i  = 0;
N  = L;
fv = (0:(N-1)) - floor(N/2);
fv = 2*fv/N;
fvM0 = (0:(512-1)) - floor(512/2);
fvM0 = 2*fvM0/512;

for mu = [0.001 0.01 0.05 0.08 0.0001 0.01]
    x = randn(1,L);  % this is x
    h = [1 0 0 0 0 0 0.5];
    xUnk = conv(h,x);
    d = xUnk(1:end - length(h)+1);
    if i == 5
        d = xUnk(length(h):end);
    end

    for n = 1 : L-M
        xn = x(n:n+M-1)';
        yn = w'*xn;
        yout(n) = yn;
        en = d(n) - yn;
        errout(n) = en;
        w = w + mu * en * xn;
    end

    if i == 0
        figure()
        sgtitle("Step 4 : mu = " + mu + ", a = " + a + ", L = " + L)
        subplot(2,1,1)
        plot(fv, abs(fftshift(fft(d,L))));
        xlabel('Normalized Frequency (\times\pi rad/sample)')
        title('FFT of desired')

        subplot(2,1,2)
        plot(fv, abs(fftshift(fft(x,L))));
        xlabel('Normalized Frequency (\times\pi rad/sample)')
         title('FFT of input x')

    else
        figure()
        sgtitle("Step 4 : mu = " + mu + ", a = " + a + ", L = " + L)

        subplot(2,2,1)
        plot(errout.*errout);
        xlabel('Time')
        title("Squared error with mu = " + mu)

        subplot(2,2,2)
        plot(fvM0, abs(fftshift(fft(w,512))));
        xlabel('Normalized Frequency (\times\pi rad/sample)')
        title('FFT of filter')

        subplot(2,2,3)
        plot(fvM0, mag2db(abs(fftshift(fft(w,512)))));
        xlabel('Normalized Frequency (\times\pi rad/sample)')
        title('Interpolated FFT of input x in dB')

        subplot(2,2,4)
        stem(w);
        title('Stem plot')
    end
```

```
    i = i + 1;
end
```

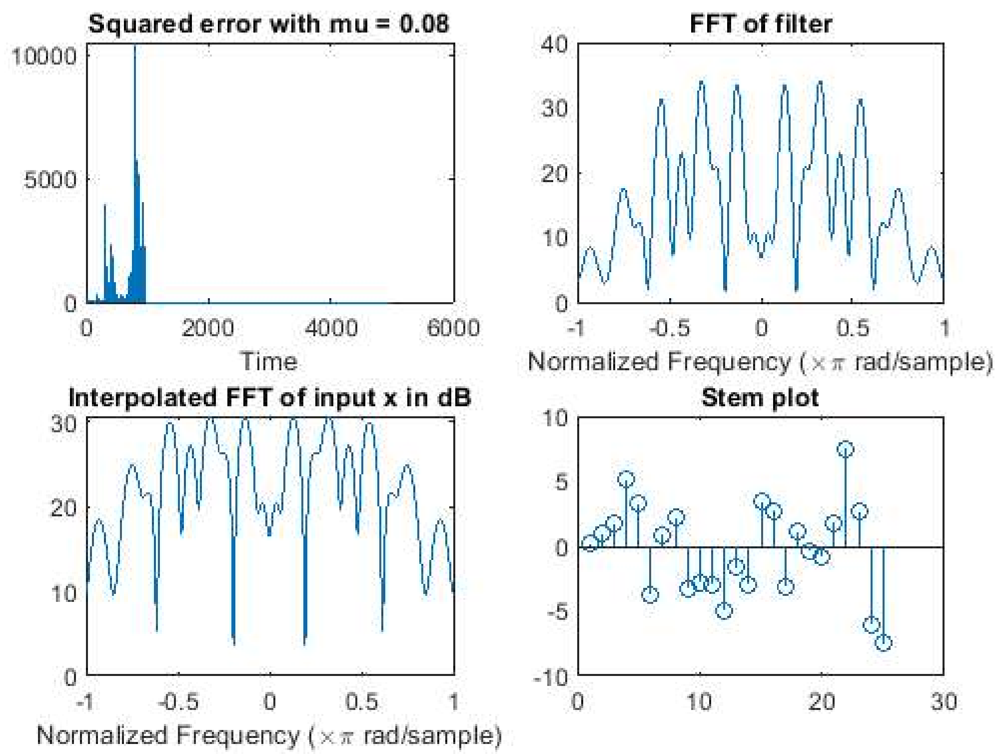

Step 4 : mu = 0.001, a = 0.1, L = 1000
FFT of desired
Normalized Frequency (×π rad/sample)
FFT of input x
Normalized Frequency (×π rad/sample)

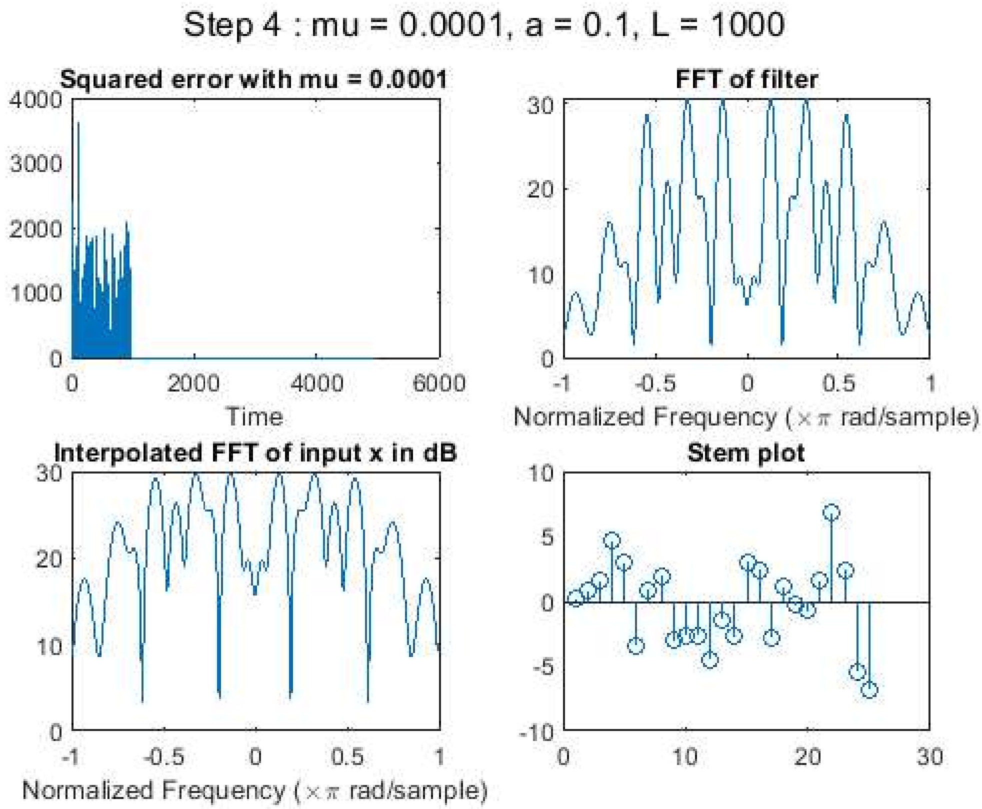

Step 4 : mu = 0.01, a = 0.1, L = 1000
Squared error with mu = 0.01
Time
FFT of filter
Normalized Frequency (×π rad/sample)
Interpolated FFT of input x in dB
Normalized Frequency (×π rad/sample)
Stem plot

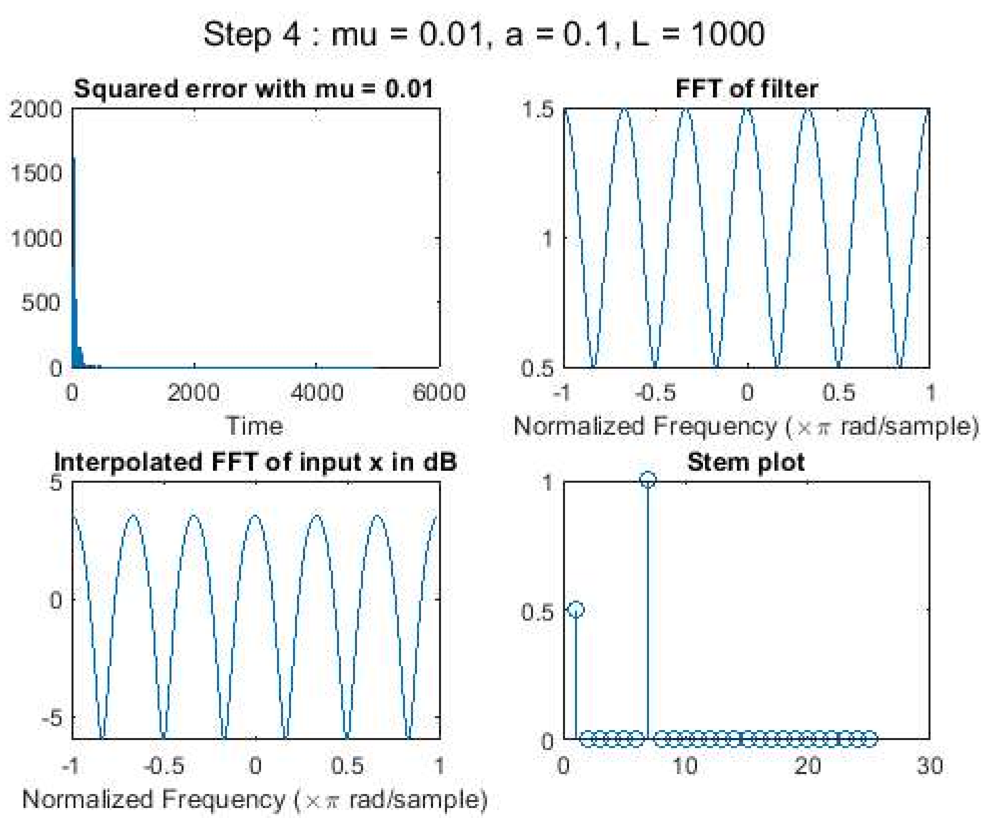
Step 4 : mu = 0.05, a = 0.1, L = 1000



Step 4 : mu = 0.08, a = 0.1, L = 1000





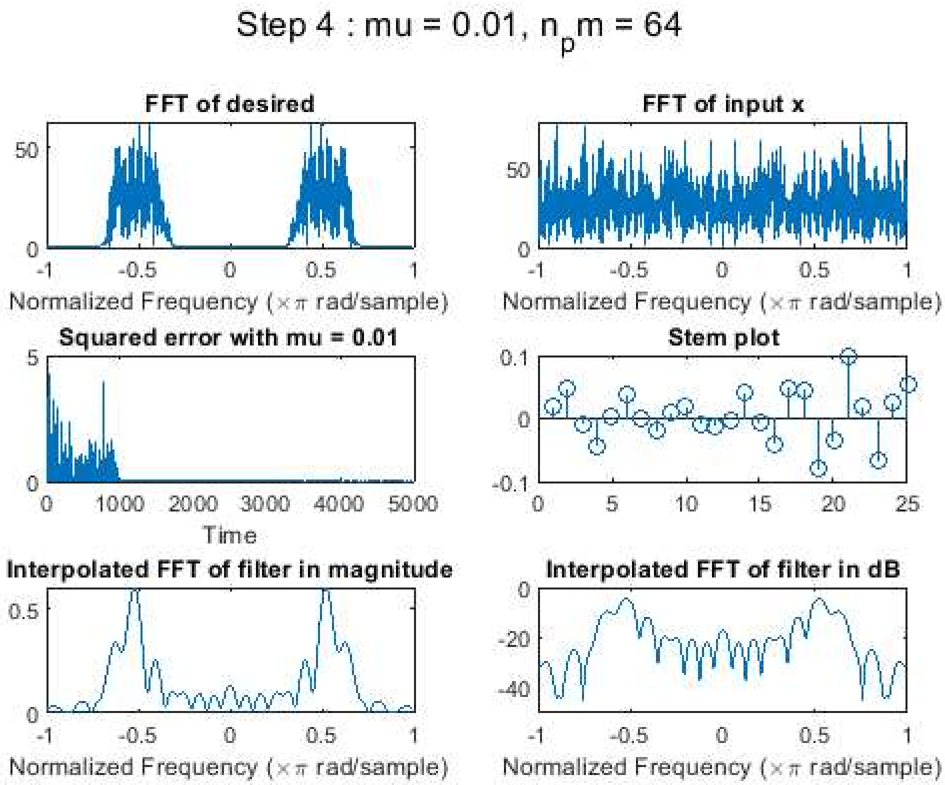


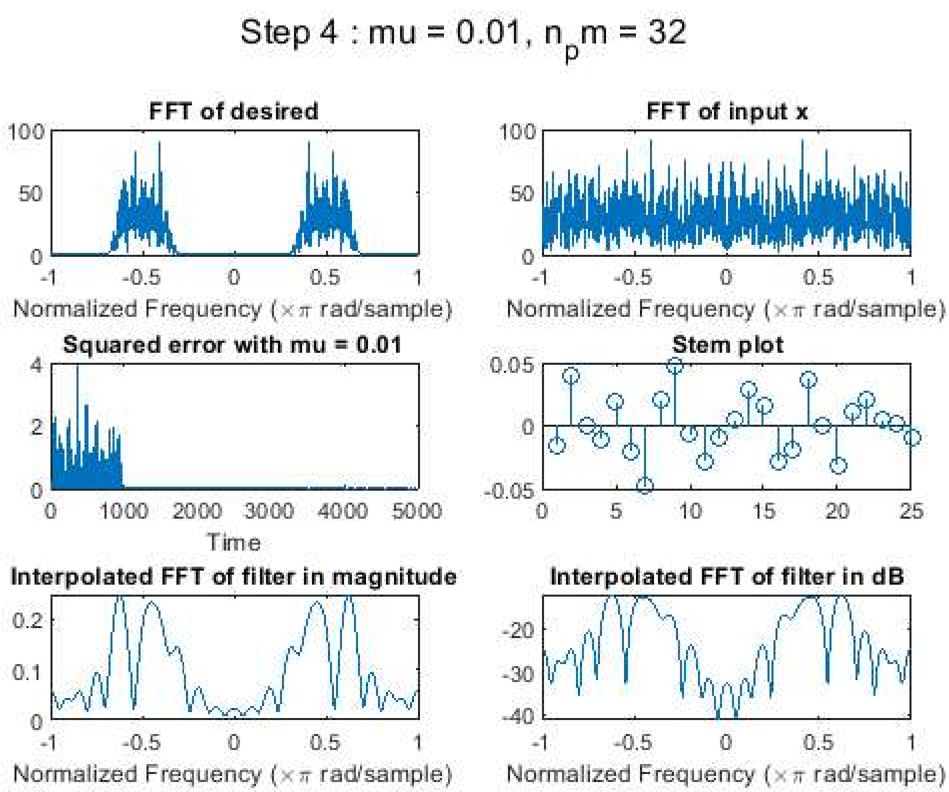
Step 4

```
M   = 25;
L   = 1000;
a   = 0.1;
```

```
i  = 0;
mu = 0.01;
N  = L;
fv = (0:(N-1)) - floor(N/2);
fv = 2*fv/N;
fvM0 = (0:(512-1)) - floor(512/2);
fvM0 = 2*fvM0/512;

for n = 1:2
    f_pm = [0, 1200, 1600, 2400, 2800, 4000]/4000;
    a_pm = [0,0,1,1,0,0]; % band pass
    n_pm = 64;
    b_pm = firpm(n_pm, f_pm, a_pm);
    x = randn(1,L); % create x input for both filters
    x_unk = conv(b_pm, x);
    d = x_unk(1: end- length(b_pm)+1); % make d the same length as x

    if n == 2
        n_pm = 32;
    end

    h = [1 0 0 0 0 0 0.5];
    xUnk = conv(h,x);

    for n = 1 : L-M
        xn = x(n:n+M-1)';
        yn = w'*xn;
        yout(n) = yn;
        en = d(n) - yn;
        errout(n) = en;
        w = w + mu * en * xn;
    end

    figure()
    sgtitle("Step 4 : mu = " + mu + ", n_pm = " + n_pm)
    subplot(3,2,1)
    plot(fv, abs(fftshift(fft(d))));
    xlabel('Normalized Frequency (\times\pi rad/sample)')
    title('FFT of desired')

    subplot(3,2,2)
    plot(fv, abs(fftshift(fft(x))));
    xlabel('Normalized Frequency (\times\pi rad/sample)')
    title('FFT of input x')

    subplot(3,2,3)
    plot(errout.*errout);
    xlabel('Time')
    title('Squared error with mu = 0.01' )

    subplot(3,2,4)
    stem(w);
    title('Stem plot')

    subplot(3,2,5)
    plot(fvM0, abs(fftshift(fft(w,512))));
    xlabel('Normalized Frequency (\times\pi rad/sample)')
    title('Interpolated FFT of filter in magnitude')

    subplot(3,2,6)
    plot(fvM0, mag2db(abs(fftshift(fft(w,512)))));
```

```
    xlabel('Normalized Frequency (\times\pi rad/sample)')
    title('Interpolated FFT of filter in dB')

    i = i + 1;
end
```





Published with MATLAB® R2020a