Applications of DSP

Project 5

UNIVERSITY OF CENTRAL FLORIDA
DEPARTMENT OF ELECTRICAL ENGINEERING

April 23, 2017
Authored by: **Rajib Dey**

Applications of DSP

Project 5

Question 1:

Given,
The system with Gaussian white noise x(n),
Input signal $\boldsymbol{\eta(n)} = \mathbf{0}$

Need to find Weiner weights by computing $\boldsymbol{w}* = [\boldsymbol{R}]^{-1}[\boldsymbol{P}]$ using 100 and 1000 samples of the input signals.

Initializing the problem with c= [1,2,3], the unknown channel F(z) is represented by $F(z) = c0 + c1 * z^{-1} + c2 * z^{-2}$.

From the unknown channel, the output is d(n), which is added to the input signal.
We get the error e(n) by subtracting the modeling system: $A(z) = w0 + w1 * z^{-1} + w2 * z^{-2}$.
Then generated the R (autocorrelation matrix of the input signal) and P matrix (cross correlation of the input and the desired signal) by convoluting the given c vector with a random vector [1xn].

From the MATLAB console window, for 100 and 1000 samples, the Weiner weight output is:

```
the result of Problem1 for n=100

w_Problem1 =

    0.9931
    1.8961
    2.8622

the result of Problem1 for n=1000

w_Problem1 =

    0.9998
    1.9959
    2.9928
```

As I have generated a Random number in MATLAB the output is always different.

<u>**Question 2:**</u>

<u>Given,</u>

Adaptive system where the error e(n) from the last system is fed back through the LMS Algorithm block to the modeling system.

Need to find iteratively w0, w1, w2 using the LMS algorithm

Initializing the number of samples, n as 150. μ is 0.02.

Implemented the LMS algorithm using the following formula:

$y(k) = w0(k) * x(k) + w1(k) * x(k-1) + w2(k) * x(k-2)$

Error $e(k) = d(k) - y(k)$.

$w0(k+1) = w0(k) + 2\mu * e(k) * x(k)$

$w0(k+1) = w1(k) + 2\mu * e(k) * x(k-1)$

$w0(k+1) = w2(k) + 2\mu * e(k) * x(k-2)$

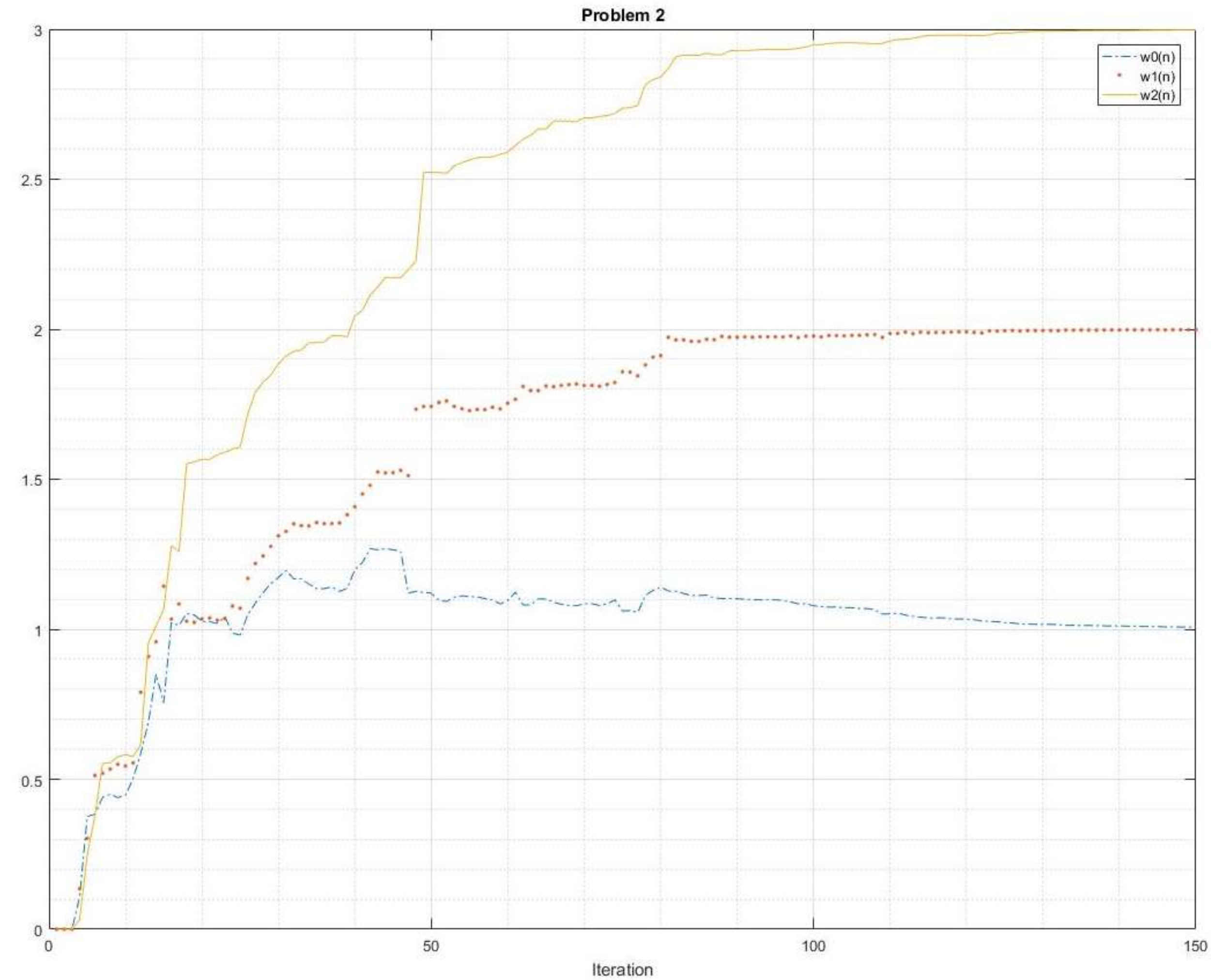
From the MATLAB console window:

```
the result of Problem1 for n=150

w_Problem2 =

    1.0068    1.9982    2.9980
```

Weight vs Iteration plot:



Question 3:

Given,

The same system setup as in problem 2. As discussed by Dr. Wasfy in the class, from the LMS algorithm

$$w_{n+1}(i) = w_n(i) + 2\mu\, e(n)x(n-i),\ i = 0 \rightarrow L$$

Need to find iteratively the weights w0, w1 and w2 using the Homogeneous Adaptive Algorithm.

Using Taylor's series expansion, we get:

$$e_{n+1} = e_n + \sum_{i=0}^{L} \frac{\partial e(n)}{\partial wi} * \Delta wi$$

Coupling the two equations we get

$$\mu = (1)/(2\sum_{i=0}^{L} x2(n - i))$$

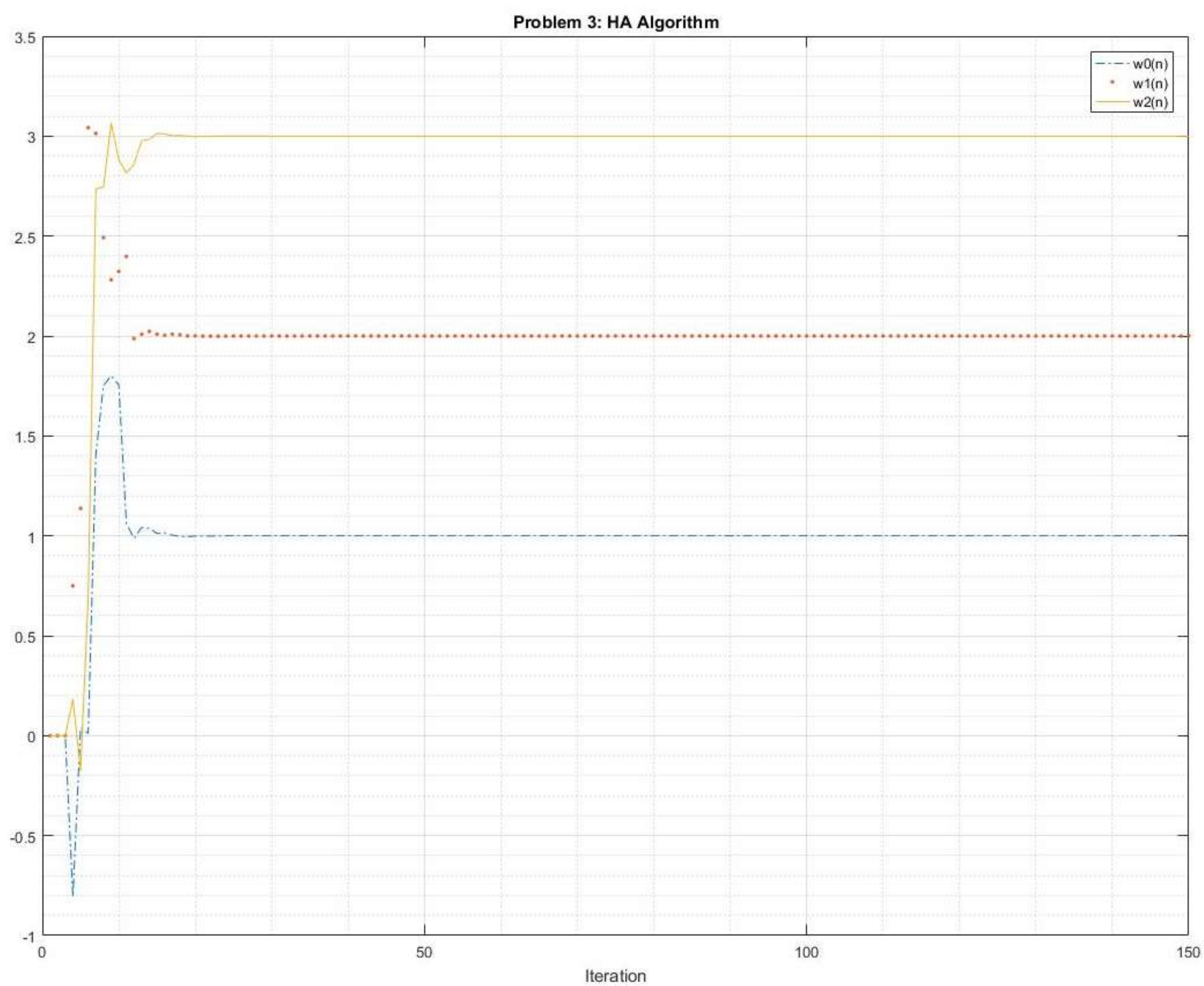
From the MATLAB console window:

```
the result of Problem1 for n=150

w_Problem3 =

    1.0000    2.0000    3.0000
```

Weight vs Iteration plot:



Question 4:

Given,

Uncorrelated input signal: $\boldsymbol{\eta(n)} = \mathbf{0.2} \cdot \frac{\sin(2\pi n)}{16}$

Need to find, iteratively the three weights using LMS algorithm.

The equation used here:

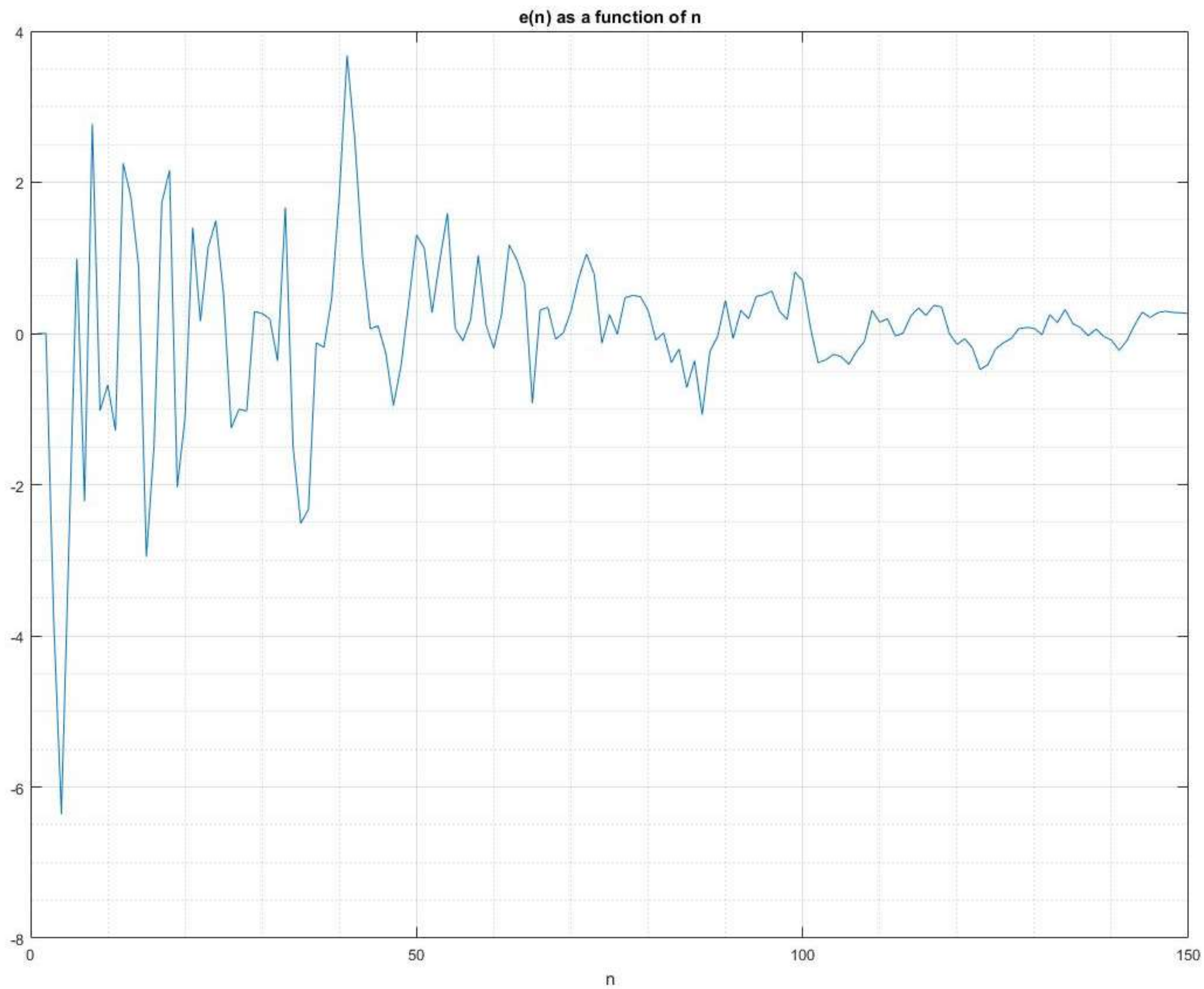
$e(k) = d(k) + \eta k - y(k).$

From the MATLAB console window:

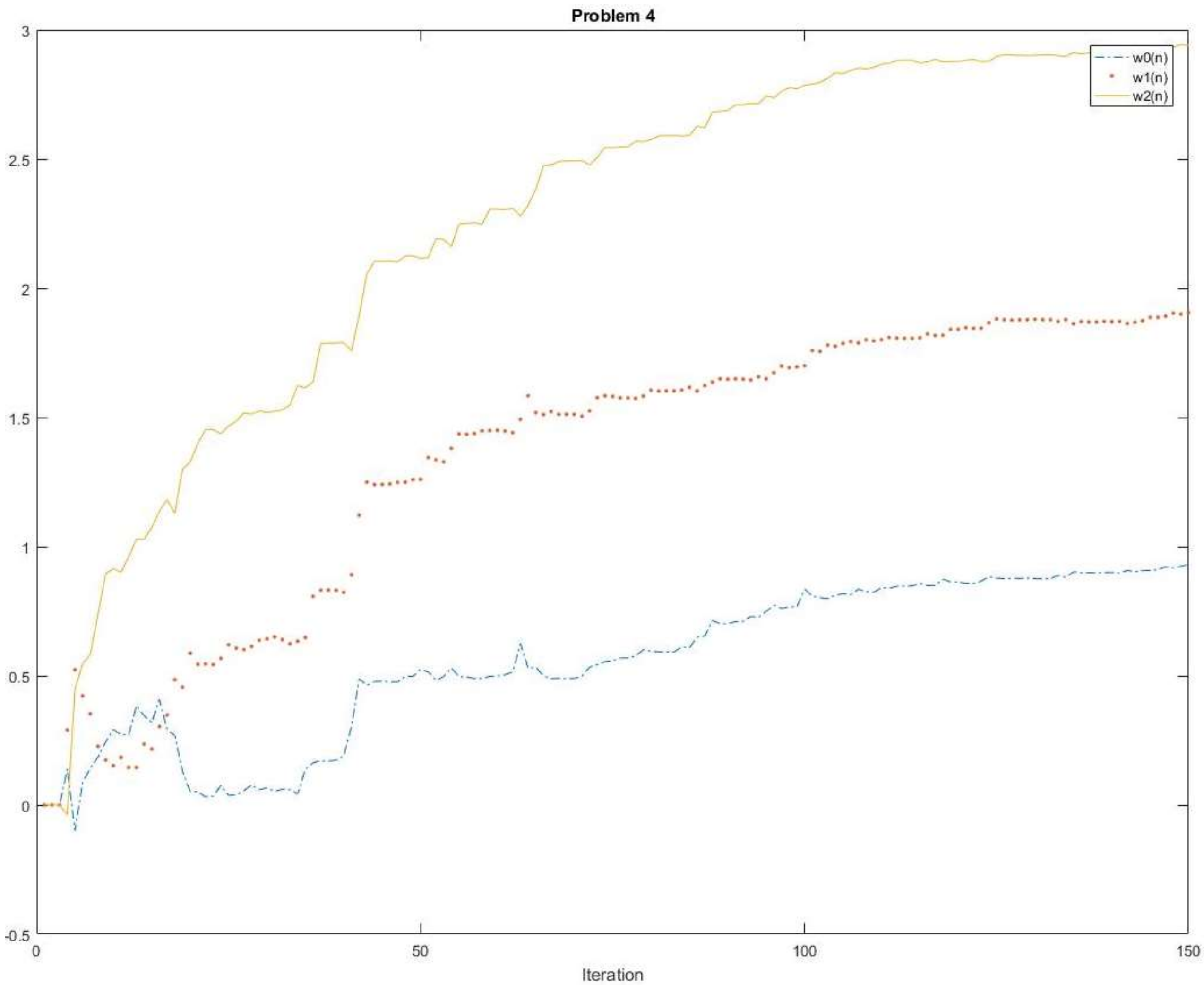
```
the result of Problem1 for n=150

w_Problem4 =

    0.9354    1.9130    2.9459
```



Weight vs Iteration plot:



<u>MATLAB Code</u>

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% EEL 5513 Applications of DSP
% Project 5
% Submitted by Rajib Dey
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Code for Qu. no. 1---------------------------------------

% Initializing
clear;
clc;
c=[1,2,3];
```

```
% for n=100
n=100;
x=randn(1,n);
d=conv(x,c);
d=d(1:n);

% Generating R & P Matrix
for k=1:3
    x_temp=[zeros(1,k-1),x(1:n-k+1)];
    r(k)=x*x_temp'/n;
    P(k)=d*x_temp'/n;
end
for i=1:3
    for j=1:3
        R(i,j)=r(abs(i-j)+1);
    end
end
% Displaying the result
display('the result of Problem1 for n=100');
w_Problem1=inv(R)*P'
% for n=1000
n=1000;
x=randn(1,n);
d=conv(x,c);
d=d(1:n);
% Generating R & P Matrix
for k=1:3
x_temp=[zeros(1,k-1),x(1:n-k+1)];
r(k)=x*x_temp'/n;
P(k)=d*x_temp'/n;
end
for i=1:3
for j=1:3
R(i,j)=r(abs(i-j)+1);
end
end
% Displaying the result
display('the result of Problem1 for n=1000');
w_Problem1=inv(R)*P'

% Code for Qu. no. 2---------------------------------------
clear;
c=[1,2,3];
% Initializing for n=150
n=150;
x=randn(1,n);
d=conv(x,c);
d=d(1:n);
mu=0.02;
w0(1)=0;
w1(1)=0;
w2(1)=0;
w0(2)=0;
w1(2)=0;
w2(2)=0;
```

```
w0(3)=0;
w1(3)=0;
w2(3)=0;

% LMS Algorithm
for k=3:n
    y(k)=w0(k)*x(k)+w1(k)*x(k-1)+w2(k)*x(k-2);
    e(k)=d(k)-y(k);
    w0(k+1)=w0(k)+2*mu*e(k)*x(k);
    w1(k+1)=w1(k)+2*mu*e(k)*x(k-1);
    w2(k+1)=w2(k)+2*mu*e(k)*x(k-2);
end
% Plot
i=1:n;
figure;
plot(i,w0(i),'-.',i,w1(i),'.',i,w2(i),'-');
legend('w0(n)','w1(n)','w2(n)');
xlabel('Iteration');
title('Problem 2');
grid on;
grid minor;
% Displaying the result
display('the result of Problem1 for n=150');
w_Problem2=[w0(k+1),w1(k+1),w2(k+1)]

% Code for Qu. no. 3---------------------------------------
clear;
c=[1,2,3];
% Initializing for n=150
n=150;
x=randn(1,n);
d=conv(x,c);
d=d(1:n);
mu=0.02;
w0(1)=0;
w1(1)=0;
w2(1)=0;
w0(2)=0;
w1(2)=0;
w2(2)=0;
w0(3)=0;
w1(3)=0;
w2(3)=0;

% HA Algorithm
for k=3:n
    y(k)=w0(k)*x(k)+w1(k)*x(k-1)+w2(k)*x(k-2);
    e(k)=d(k)-y(k);
    mu=1/(2*(x(k)^2+x(k-1)^2+x(k-2)^2));

    w0(k+1)=w0(k)+2*mu*e(k)*x(k);
    w1(k+1)=w1(k)+2*mu*e(k)*x(k-1);
    w2(k+1)=w2(k)+2*mu*e(k)*x(k-2);
end

% Plotting
```

```
i=1:n;
figure;
plot(i,w0(i),'-.',i,w1(i),'.',i,w2(i),'-');
legend('w0(n)','w1(n)','w2(n)');
xlabel('Iteration');
title('Problem 3: HA Algorithm');
grid on;
grid minor;
% Displaying the result
display('the result of Problem1 for n=150');
w_Problem3=[w0(k+1),w1(k+1),w2(k+1)]

% Code for Qu. no. 4---------------------------------------
clear;
c=[1,2,3];
% Initializing for n=150
n=150;
x=randn(1,n);
d=conv(x,c);
d=d(1:n);
mu=0.02;
w0(1)=0;
w1(1)=0;
w2(1)=0;
w0(2)=0;
w1(2)=0;
w2(2)=0;
w0(3)=0;
w1(3)=0;
w2(3)=0;
% LMS Algorithm
for k=3:n
    y(k)=w0(k)*x(k)+w1(k)*x(k-1)+w2(k)*x(k-2);
    eta(k)=0.2*sin(2*pi*k/16);
    e(k)=d(k)+eta(k)-y(k);
    w0(k+1)=w0(k)+2*mu*e(k)*x(k);
    w1(k+1)=w1(k)+2*mu*e(k)*x(k-1);
    w2(k+1)=w2(k)+2*mu*e(k)*x(k-2);
end
% Plotting
i=1:n;
figure;
plot(i,w0(i),'-.',i,w1(i),'.',i,w2(i),'-');
legend('w0(n)','w1(n)','w2(n)');
xlabel('Iteration');
title('Problem 4');
figure;
plot (e(i))
xlabel ('n');
title('e(n) as a function of n');
grid on;
grid minor;
% Displaying the result
display('the result of Problem1 for n=150');
w_Problem4=[w0(k+1),w1(k+1),w2(k+1)]
```