

Experiment-1

EE:2801 DSP-Lab

Indian Institute of Technology, Hyderabad

Jay Vikrant

EE22BTECH11025

I. QUESTION

Simulate convolution and correlation between two signals in Matlab and C. Take input as

x	0.3426	3.5784	2.7694	-1.3499	3.0349	0.7254	-0.0631
h	0.7147	-0.2050	-0.1241	1.4897	1.4090		

II. SOLUTION

Here is the C code,

```
#include <stdio.h>

void calc_conv(double *x, double *h, int l1, int l2) // function to compute convolution
{
    int l = l1 + l2 - 1;
    double y[l];
    for (int n = 0; n < l; n++)
    {
        y[n] = 0;
        for (int k = 0; k < l1; k++)
        {
            if (n - k >= 0 && n - k < l2) /*keeping the h(n-k) sequence same even though it has
                been folded and shifted */
            {
                y[n] = y[n] + x[k] * h[n - k]; /* computing \sigma x(k)*h(n-k) */
            }
        }
    }

    for (int i = 0; i < l1 + l2 - 1; i++)
    {
        printf("%.4f", y[i]);
    }
    printf("\n");
}

double flipr(double *x, int l1, int index) // function for folding the sequence i.e finding x(-n)
{

```

```

    double x0[l1];

    for (int i = 0; i < l1; i++)
    {
        x0[(l1 - 1) - i] = x[i];
    }
    return x0[index];
}

void calc_xcorr(double *x, double *h, int l1, int l2) // function to compute corss correlation
{
    double h0[l2];
    for (int i = 0; i < l2; i++)
    {
        h0[i] = fliplr(h, l2, i);
    }
    calc_conv(x, h0, l1, l2);
}

void calc_acorr(double *x, int l1) // function to compute auto correlation
{
    double x0[l1];
    for (int i = 0; i < l1; i++)
    {
        x0[i] = fliplr(x, l1, i);
    }
    calc_conv(x, x0, l1, l1);
}

int main()
{
    /*int l1, l2;
    double x[l1], h[l2];
    printf("Enter the length of the first sequence: ");
    scanf("%d", &l1);
    printf("Enter the length of the second sequence: ");
    scanf("%d", &l2);

    printf("Enter the first sequence x:\n");
    for (int i = 0; i < l1; i++) {
        scanf("%d", &x[i]);
    }

    printf("Enter the second sequence h:\n");
    for (int i = 0; i < l2; i++) {
        scanf("%d", &h[i]);
    } */

    // for our question i am taking the given input directly but the above commented part can also be

```

used

```

int l1 = 7;
int l2 = 5;
double x[7] = {0.3426, 3.5784, 2.7694, -1.3499, 3.0349, 0.7254, -0.0631};
double h[5] = {0.7147, -0.2050, -0.1241, 1.4897, 1.4090};

printf("Convolution_of_x_and_h_resulted:\n");
calc_conv(x, h, l1, l2);

printf("\n");

printf("Cross-correlation_of_x_and_h_resulted:\n");
calc_xcorr(x, h, l1, l2);

printf("\n");

printf("auto-correlation_of_x_resulted:\n");
calc_acorr(x, l1);

printf("\n");

printf("auto-correlation_of_h_resulted:\n");
calc_acorr(h, l2);

printf("\n");

return 0;
}

```

The following got computed in C,
Now, for the Matlab simulation,

```

function main()

x = [0.3426 3.5784 2.7694 -1.3499 3.0349 0.7254 -0.0631];
h = [0.7147 -0.2050 -0.1241 1.4897 1.4090];

disp('convolution_x_and_h_is_given_by');
disp(conv(x,h));

disp('corss-correlation_of_x_and_h_is_given_by');
disp(xcorr(x,h));

disp('auto-correlation_x_is_given_by');
disp(acorr(x));

disp('auto-correlation_h_is_given_by');
disp(acorr(h));
end

```

/home/jay/Desktop/Dsp-lab/C/main

Convolution of x and h resulted:

```
0.2449  2.4872  1.2032  -1.4662  7.9156  9.2314  1.3207  2.5420  5.3646
0.9281  -0.0889
```

Cross-correlation of x and h resulted:

```
0.4827  5.5523  9.1903  1.7093  1.4328  7.7005  2.8711  -1.7710  2.0282
0.5314  -0.0451
```

auto-correlation of x resulted:

```
-0.0216  0.0227  3.4608  12.4917  3.3524  5.4565  32.1549  5.4565  3.352
4  12.4917  3.4608  0.0227  -0.0216
```

auto-correlation of h resulted:

```
1.0070  0.7758  -0.5689  1.7930  4.7727  1.7930  -0.5689  0.7758  1.0070
```

```
Process returned 0 (0x0)   execution time : 0.001 s
Press ENTER to continue.
```

%Convolution

function y = conv(x,h) *% y here is the output*

% Get input sequences

%x = input('Enter the first sequence x: ');

%h = input('Enter the second sequence h: ');

%dont mind the above comments just for testing and debugging

% Calculate output length

l = length(x) + length(h) - 1;

% Initialize output sequence

y = zeros(1, l);

% Perform convolution manually

for n = 1:l

for k = 1:length(x)

if (n - k + 1) >= 1 && (n - k + 1) <= length(h)

 y(n) = y(n) + x(k) * h(n - k + 1); *% This is just /sigma x(k)*h(n-k)*

end

end

end

end

% cross_correlation

function y1 = xcorr(x,h) *% y1 here is the output*

```

h0=flipr(h); %to rearrange  $h(n)$  as  $h(-n)$ 
y1 = conv(x,h0);

```

```
end
```

```
% auto_correlation
```

```
function y2 = acorr(x)
```

```

h0=flipr(x); %to rearrange  $x(n)$  as  $x(-n)$ 
y2 = conv(x,h0);

```

```
end
```

The following got computed in Matlab,

```

Command Window

>> main
convolution x and h is given by
    0.2449    2.4872    1.2032   -1.4662    7.9156    9.2314    1.3207    2.5420    5.3646    0.9281   -0.0889

cross-correlation of x and h is given by
    0.4827    5.5523    9.1903    1.7093    1.4328    7.7005    2.8711   -1.7710    2.0282    0.5314   -0.0451

auto-correlation x is given by
   -0.0216    0.0227    3.4608   12.4917    3.3524    5.4565   32.1549    5.4565    3.3524   12.4917    3.4608    0.0227   -0.0216

auto-correlation h is given by
    1.0070    0.7758   -0.5689    1.7930    4.7727    1.7930   -0.5689    0.7758    1.0070

fx >>

```