# Experiment 6 Difference equations and analysis of LTI systems

|  |
| --- |
| Objective: To explore how the frequency response of a system may be computed based on its z-transform system function. The relationship between the system function and its impulse response is also examined. |

## Background information

Difference equation of a digital system is given in general form as follows:



MATLAB function ‘filter’ can be used to calculate the output signal *y*(*n*) for a given input signal *x*(*n*). This function needs the coefficients on the left and right hand sides of the above equation in two separate arrays and an input signal to calculate the output signal (refer to MATLAB help for more information). Rearrange this equation so that all the *y*(*n*) and *x*(*n*) terms are placed on the left hand and right hand sides of the equation respectively:



Procedure 1: Difference equation with no feedback

1. We first use a simple difference equation with no feedback terms. That is,  in the above equation. Let the difference equation of the system be:

*y*(*n*) = 0.5*x*(*n*) + 0.1*x*(*n* – 1) – 0.5*x*(*n* – 2)

To use the MATLAB function ‘filter’, declare all the coefficients into matrices as:

>>a=[1];

>>b=[0.5 0.1 -0.5];

1. We now need to define an input signal. For instance, in order to input a unit impulse signal into the system, we generate an impulse signal as follows:

>>impulse=[1 0 0 0 0 0 0 0];

We can now calculate the output signal:

>>y=filter(b,a,impulse)

What is the output?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Compute the impulse response of the system manually. Are they the same?

Now, we modify the signal ‘impulse’ by padding more zeros:

>>impulse=[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];

Do you think a different output will be produced?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Now apply a new input signal to the filter, calculate the output and name it ‘y2’:

>>x=[0.1 0.2 0.3 0 0 0 0 0];

>>y2=filter(b,a,x)

Delay the input signal x by two samples, i.e., change x to:

>>x=[0 0 0.1 0.2 0.3 0 0 0];

Execute the program again and observe the result. What is the difference between new result and those obtained in step 1-3?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Do you think this filter is time-invariant?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Change the input signal to {0.3, 0.5, 0.4, 0, 0, 0, 0, 0}. Simulate the filter again and record the output and name it ‘y3’.
2. Change the input signal again to {0.4, 0.7, 0.7, 0, 0, 0, 0, 0}. Notice that this input signal is actually the sum of the input signals of steps 1-3 and 1-5. Simulate the filter again and record the output. Name it ‘y4’.

Can you see a relationship between ‘y2’, ‘y3’ and ‘y4’? Do you think this filter is linear?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Try applying different input signals to the filter and verify your results by manual calculation.
2. Change the difference equation to:

*y*(*n*) = *x*(*n*) + 4*x*(*n* – 1) – 2*x*(*n* – 2) + 3*x*(*n* – 3) – *x*(*n* – 4)

What is its impulse response?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What is the output if the input signal *x*(*n*) = {1, –2, 0, 1, 0.2, –1}?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Procedure 2: Difference equation with feedback**

1. Assume that a digital filter with feedback, is given as follows:

*y*(*n*) = 0.1*x*(*n*) + 0.2*x*(*n* – 1) + 0.09*x*(*n* – 2) + 0.94*y*(*n* – 1) – 0.33*y*(*n* – 2)

Enter the coefficients into the MATLAB workspace:

>>a=[1 -0.94 0.33];

>>b=[0.1 0.2 0.09];

Determine the impulse response of the filter and verify your result with manual computation.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Procedure 3: Convolution**

1. Convolution of two sequences can be calculated easily using MATLAB command ‘conv’. View the description of the command:

>>help conv

1. We can calculate the convolution of two sequences used in Example 2.5 with MATLAB:

>>x=[1 1 0.5];

>>h=[0.5 0.5];

>>y=conv(x,h)

What is the result of convolution?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Enter the following commands and find the impulse response of the digital system given in Procedure 2.

>>imp=[1, zeros(1,20)];

>>a=[1 -0.94 0.33];

>>b=[0.1 0.2 0.09];

>>h=filter(b,a,imp)

Array ‘h’ is now holding the impulse response of the system in Procedure 2.

**Exercise:**

1. Implement the following digital system, which is given by the difference equation using the command, conv:



Plot the impulse and unit response h(n) for n=0..4.

1. Implement the following digital system, which is given by the difference equation using the command, filter :

.

Plot the impulse response h(n) and unit response h1(n) for n=0...9.