

# ECE 311 Lab 3

Jacob Hutter

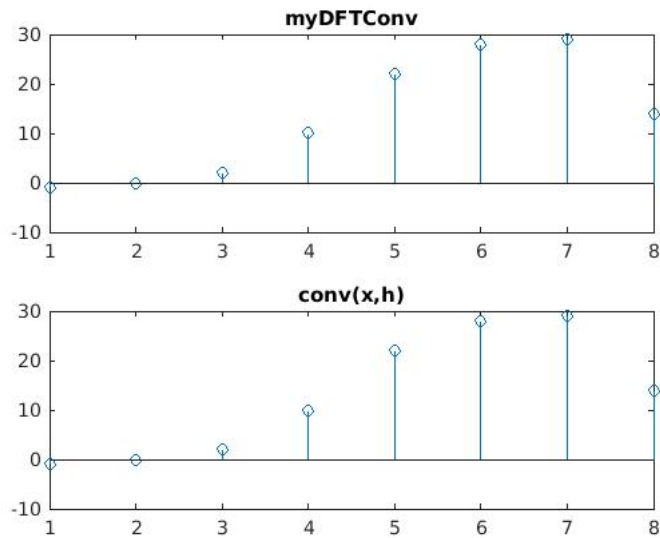
March 1, 2017

```

1 function [ y ] = myDFTConv( x, h )
3 %transfer over to frequency domain
  l = size(x,2) + size(h,2) - 1;
5 x_w = fft(x,l);
  h_w = fft(h,l);
7
9 %element multiplication to get y_w
  y_w = x_w.*h_w;
11 %return to discrete domain
  y = ifft(y_w);
13
15 %plot my result
  figure;
  subplot(211);
  stem(y);
  title('myDFTConv');
19
21 %plot key
  subplot(212);
  stem(conv(x,h));
  title('conv(x,h)');
23
25 end

```

myDFTConv.m



Because of the use of fft and ifft, the time coplexity is also  $O(N\log(N))$ .

```

1 function [ y_n ] = sys1( a,N )
3 %get delta function
  x_n = zeros(1,N);
5 x_n(1) = 1;

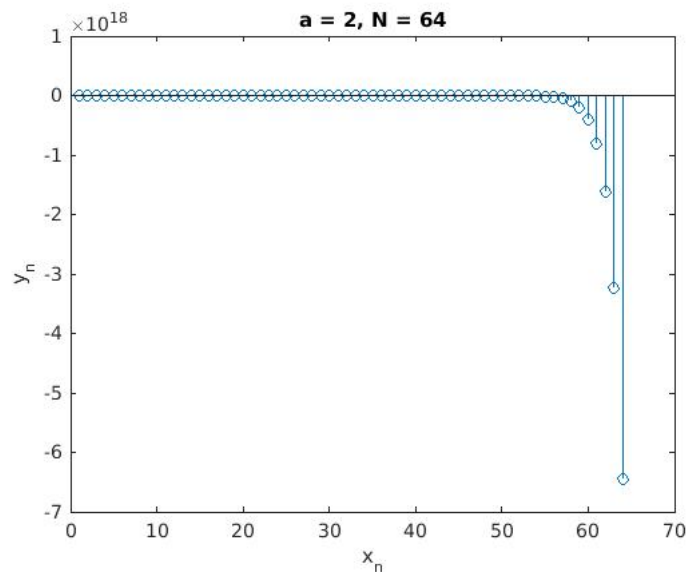
7 %get shifted delta function
  x_n_1 = zeros(1,N);
9 x_n_1(2) = 1;

11 y_n = zeros(1,N+1);

13 for i = 1:N,
    y_n(i+1) = a.*y_n(i) + 0.3.*x_n(i) -2.*x_n_1(i);
15 end
  y_n = y_n(2:length(y_n)); % remove first element to get to N
17 figure;
  stem(y_n);
19 title('a = 2, N = 64');
  ylabel('y_n');
21 xlabel('x_n');
  end

```

sys1.m



Notice that this function is in the order of  $-e^x$  therefore, it is not stable  $\lim_{x_n \rightarrow \infty} y_n = -\infty$ . The system is causal, however, by the finite difference equation, output only depends on past or current inputs.

```

1 function [ y_n ] = sys2( x_n, a )
N = 64; % length always 64
3 y_n = zeros(1,N+1);

5 for i = 1:64,
    y_n(i+1) = a*y_n(i) + x_n(i).^2;
7 end
    y_n = y_n(2:length(y_n)); % remove first element to shorten to
    64
9
    figure;
11 stem(y_n);
    title('difference equation with a = 2');
13 xlabel('n');
    ylabel('y(n)');
15 end

```

sys2.m

