ECEN 220 Lab 1

Niels Clayton
300437590

Part 1: Signal Periodicity

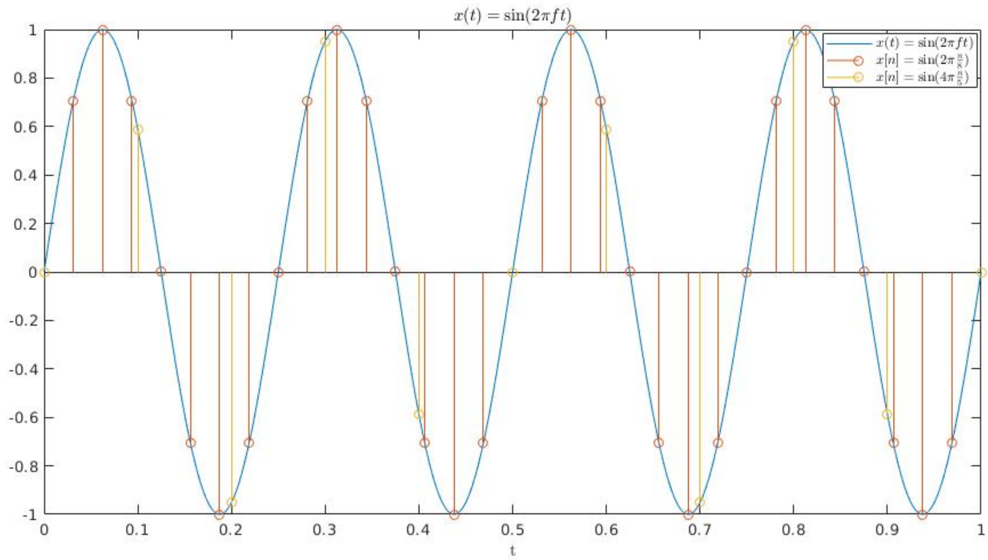

Figure 1: continuous and discrete time plots

The period of a continuous time signal can found using $\frac{1}{f_0} = \frac{1}{4}$.
The period of a discrete time signal can be found using $\frac{2\pi}{\omega_0} = \frac{N}{M}$, Where N is the period as a number of samples, and M is the number of samples that fit into one cycle of the discrete signal. The period of one sample is one over the sampling frequency.

Using this we can find that:

$x(t)$ has a period of $\frac{1}{4}$ or $0.25s$
$x_1[n]$ has a period of 8 samples
$x_2[n]$ has a period of 5 samples

Using the sampling frequencies of x_1 and x_2 respectively, we can calculate the time period in terms of seconds.

$x_1[n] : 8 \times \frac{1}{32} = 0.25s$
$x_2[n] : 5 \times \frac{1}{10} = 0.5s$

From this we can see that x_1 has the same period as $x(t)$, this can be seen in figure 1 above. It can also be seen that x_2 has a period twice that of $x(t)$, meaning that it repeats ever 2 periods of $x(t)$

Part 2: Linearity

If a system $f(x)$ is linear, The following rule will be true:

$$f(x_1 + x_2) = f(x_1) + f(x_2)$$

When this test was applied to the system $y[n] = 2^{x[n]}$ the two different outputs were plotted against each other in figure 2.
Since the two discrete time plots do not match all values, it can be concluded that the system $y[n] = 2^{x[n]}$ is not linear.

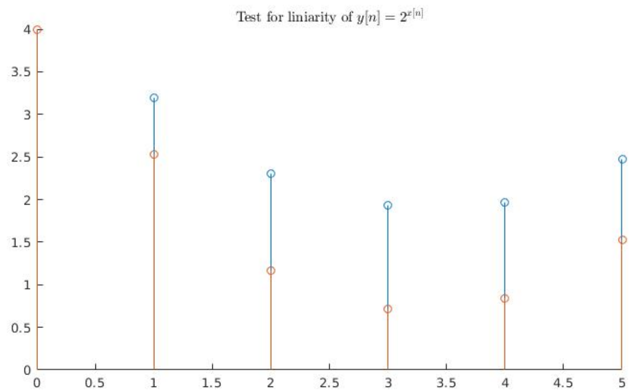

Figure 2: $f(x_1 + x_2) = f(x_1) + f(x_2)$

When this test was applied to the system $y[n] = nx[n]$ the two different outputs were plotted against each other in figure 3.
Since the two discrete time plots match entirely, it can be concluded that the system $y[n] = nx[n]$ is linear.

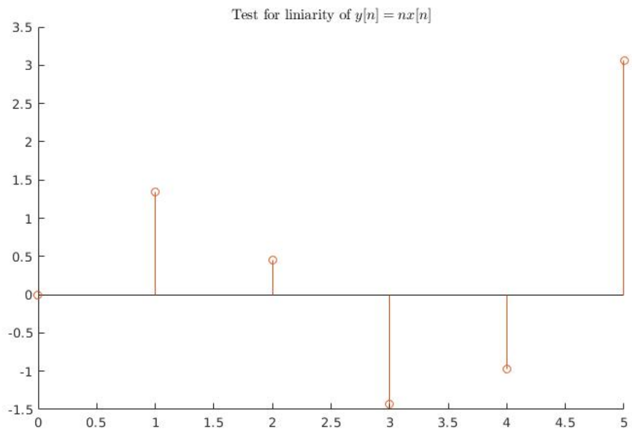

Figure 3: $f(x_1 + x_2) = f(x_1) + f(x_2)$

Convolution

The functions $h[n] = 0.7^n$ and $x[n] = u[n] - u[n-4]$ were convolved in matlab, and by hand, to produce the function $y[n]$.

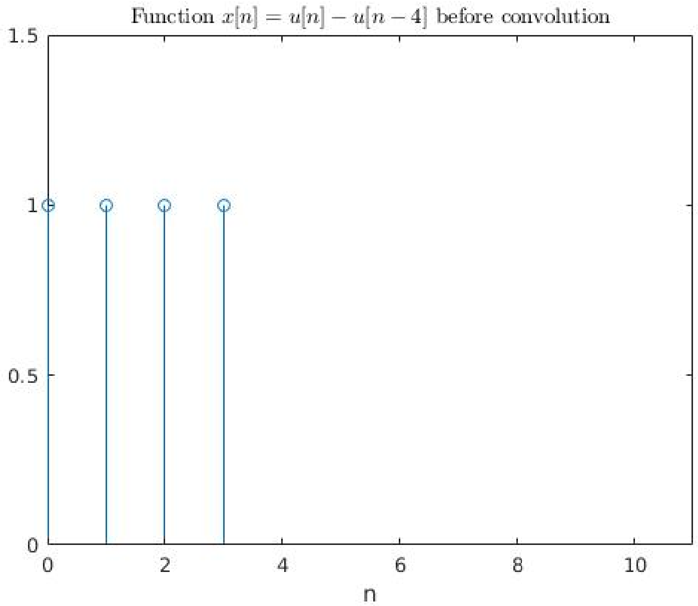

Figure 4: $h[n] = 0.7^n$

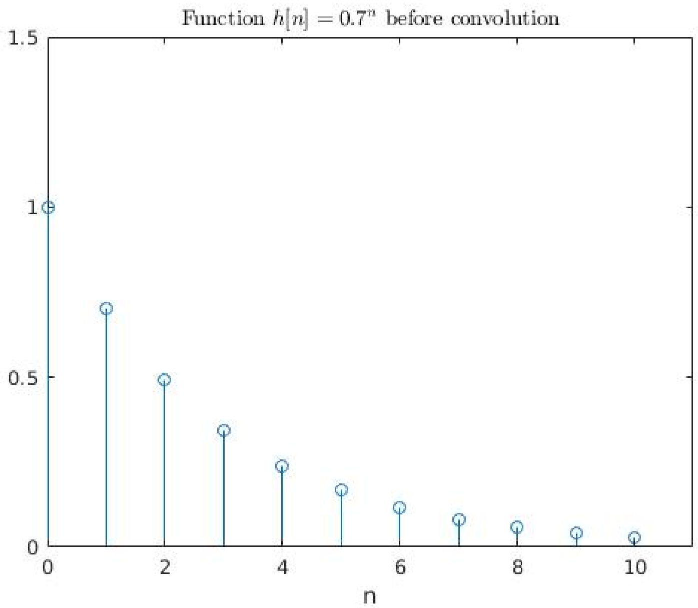

Figure 5: $x[n] = u[n] - u[n-4]$

The output of this convolution done in matlab can be seen in figure 6.

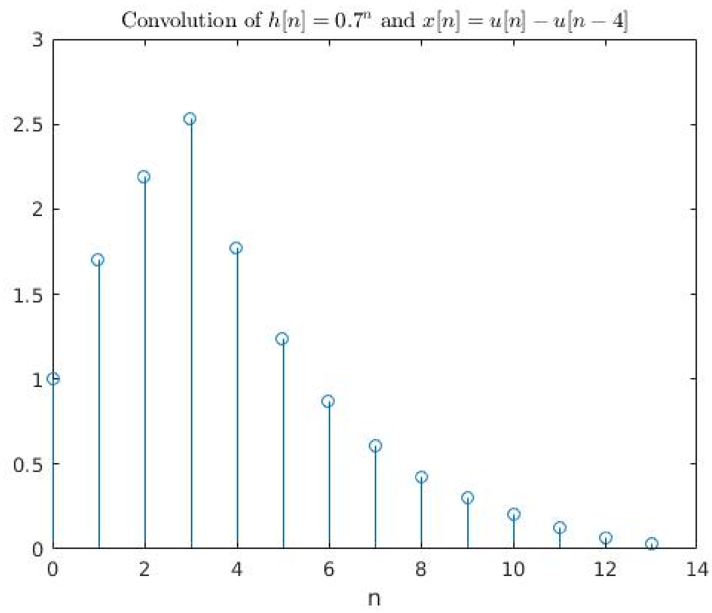

Figure 6: Convolution of $x[n]$ with $h[n]$

The output of the convolution done by hand can be seen to be exactly the same in figure 7, with the working below in figure 8.

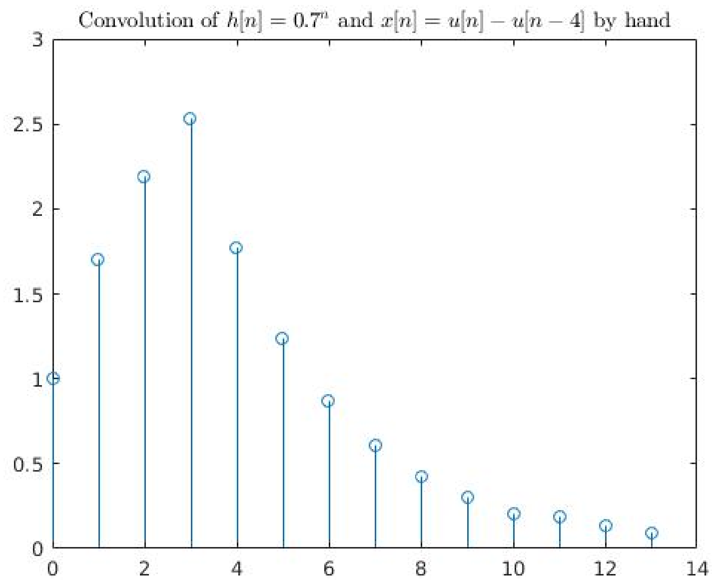

Figure 7: Convolution of $x[n]$ with $h[n]$ bone by hand

$$h[n] = 0.7^n \quad , \quad x[n] = u[n] - u[n-4]$$

Region 1 : $n < 0$: $y[n] = 0$

Region 2 : $0 \leqslant n < 3$

$$y[n] = \sum_{k=0}^{n} 0.7^k = \frac{1 - 0.7^{n+1}}{1 - 0.7}$$

Region 3 : $3 \leqslant n \leqslant 10$

$$y[n] = \sum_{k=n-3}^{n} 0.7^k = \frac{0.7^{n-3} - 0.7^{n+1}}{1 - 0.7}$$

Region 4 : $10 \leqslant n \leqslant 13$

$$y[n] = \sum_{k=n-3}^{10} 0.7^k = \frac{0.7^{n-3} - 0.7^{11}}{1 - 0.7}$$

Figure 8: Convolution of $x[n]$ with $h[n]$ working

Matlab Code

```
%% Question 1
clear variables; clc;

hold on
t = linspace(0, 1, 1000); % Continuous time sampleing
f_0 = 4;                  % Frequency
x = sin(2*pi*f_0*t);      % Function

figure(1);
plot(t,x)
title('$x(t)=\sin(2\pi ft)$','interpreter','latex')
xlabel('t','interpreter','latex')

n = 0:1:32; % Discrete time sampling
```

```
x_1 = sin(2*pi*n/8);
stem(n/32, x_1);

n = 0:1:10; % Discrete time sampling
x_1 = sin(4*pi*n/5);
stem(n/10, x_1);
legend('$x(t)=\sin(2\pi ft)$','$x[n]=\sin(2\pi \frac{n}{8})$','$x[n]=\sin(4\pi \frac{n}{5})$','interpreter','latex')
hold off


%% Question 2
clear variables; clc;

n = 0:1:5
x_1 = power(0.8, n);
x_2 = cos(n);
y_1 = power(2, x_1);
y_2 = power(2, x_2);
y_out_1 = y_1+y_2;

x_3 = x_1+x_2;
y_out_2 = power(2, x_3);

figure(2);
hold on
title('Test for liniarity of $y[n]=2^{x[n]}$','interpreter','latex')
stem(n,y_out_1)
stem(n,y_out_2)
hold off

y_1 = n.*x_1;
y_2 = n.*x_2;
y_out_1 = y_1+y_2;
y_out_2 = n.*x_3;
figure(3)
hold on
title('Test for liniarity of $y[n]=nx[n]$','interpreter','latex')
stem(n,y_out_1)
stem(n,y_out_2)
hold off

%% Question 3

clear variables; clc;

n = [0:1:10];
n2 = [0:1:3];
n3 = [0:1:13];
h = power(0.7, n)
x = (n2>=0) - ((n2-4)>=0);

```

```
y = conv(h,x);

figure(4)
stem(n2,x)
xlabel("n")
axis([0 11 0 1.5])
title('Function $x[n] = u[n] - u[n-4]$ before convolution','interpreter','latex')

figure(5)
stem(n,h)
xlabel("n")
axis([0 11 0 1.5])
title('Function $h[n] = 0.7^n$ before convolution','interpreter','latex')


figure(6)
stem(n3,y)
title('Convolution of $h[n] = 0.7^n$ and $x[n] = u[n] - u[n-4]$','interpreter','latex')
xlabel('n')

figure(7)


syms n_2
y_hand = piecewise(0<=n_2<=3, (1-0.7.^(n_2+1))/0.3, 3<n_2<=10, ((0.7.^(n_2-3))-0.7.^(n_2+1))/0.3, 10<n_2<=13,((0.7.^(n_2-3))-0.7.^(n_2+11))/0.3)


n_3 = 0:1:13
y_hand_2 = subs(y_hand, n_2, n_3)
stem(n_3, y_hand_2)
title('Convolution of $h[n] = 0.7^n$ and $x[n] = u[n] - u[n-4]$ by hand','interpreter','latex')
```