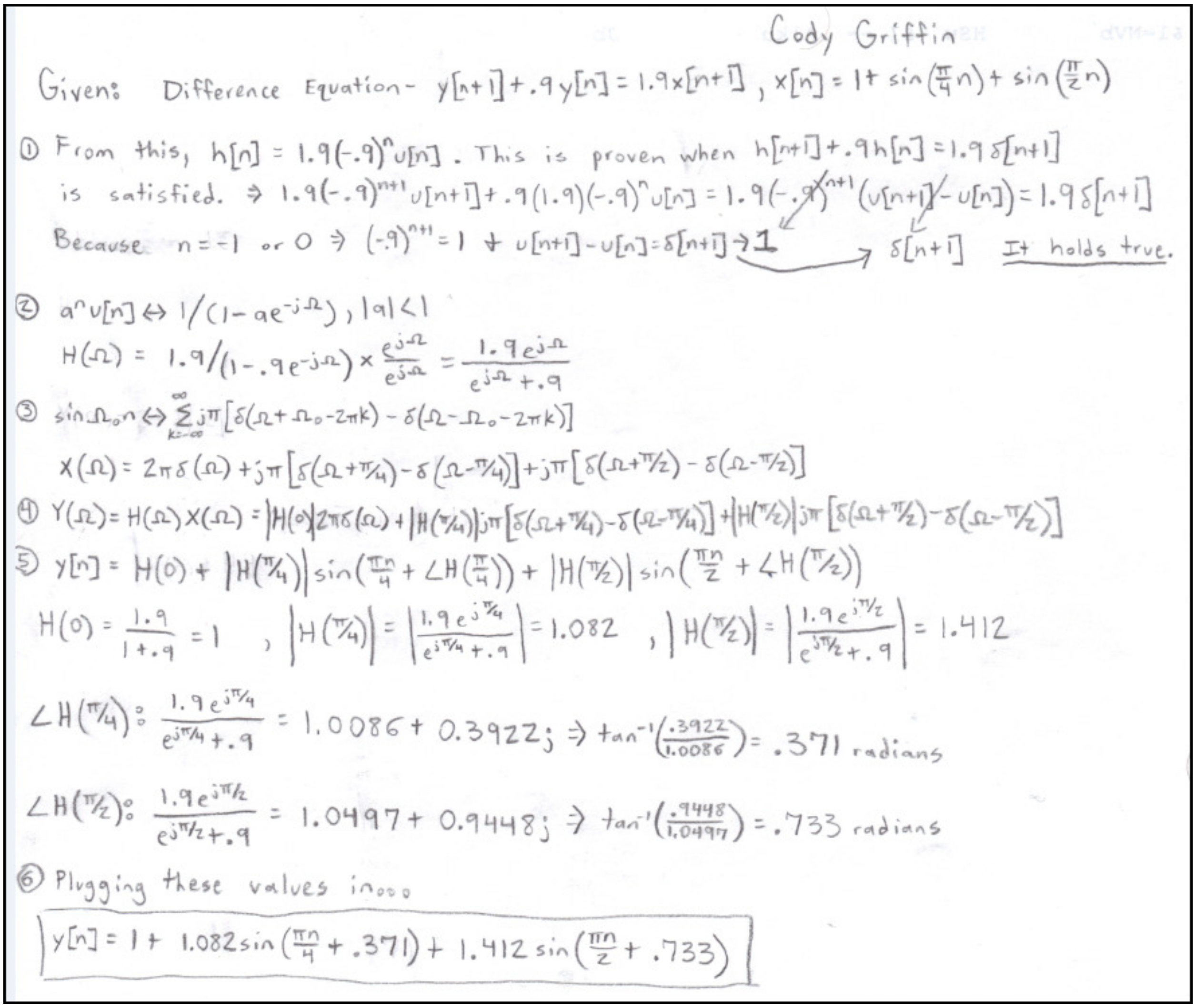
Cody Griffin (crg121)
ECE 3163 - Signals and Systems
March 27, 2009



Cody Griffin

Given: Difference Equation – $y[n+1] + .9y[n] = 1.9x[n+1]$, $x[n] = 1 + \sin(\frac{\pi}{4}n) + \sin(\frac{\pi}{2}n)$

① From this, $h[n] = 1.9(-.9)^n u[n]$. This is proven when $h[n+1] + .9h[n] = 1.9\,\delta[n+1]$ is satisfied. $\Rightarrow 1.9(-.9)^{n+1}u[n+1] + .9(1.9)(-.9)^n u[n] = 1.9(-.9)^{n+1}(u[n+1] - u[n]) = 1.9\,\delta[n+1]$

Because $n = -1$ or $0 \Rightarrow (-.9)^{n+1} = 1$ & $u[n+1] - u[n] = \delta[n+1] \to 1 \to \delta[n+1]$ It holds true.

② $a^n u[n] \leftrightarrow 1/(1 - ae^{-j\Omega})$, $|a| < 1$

$H(\Omega) = 1.9/(1 - .9e^{-j\Omega}) \times \frac{e^{j\Omega}}{e^{j\Omega}} = \frac{1.9e^{j\Omega}}{e^{j\Omega} + .9}$

③ $\sin\Omega_0 n \leftrightarrow \sum_{k=-\infty}^{\infty} j\pi[\delta(\Omega + \Omega_0 - 2\pi k) - \delta(\Omega - \Omega_0 - 2\pi k)]$

$X(\Omega) = 2\pi\delta(\Omega) + j\pi[\delta(\Omega + \pi/4) - \delta(\Omega - \pi/4)] + j\pi[\delta(\Omega + \pi/2) - \delta(\Omega - \pi/2)]$

④ $Y(\Omega) = H(\Omega)X(\Omega) = |H(0)|2\pi\delta(\Omega) + |H(\pi/4)|j\pi[\delta(\Omega + \pi/4) - \delta(\Omega - \pi/4)] + |H(\pi/2)|j\pi[\delta(\Omega + \pi/2) - \delta(\Omega - \pi/2)]$

⑤ $y[n] = H(0) + |H(\pi/4)|\sin(\frac{\pi n}{4} + \angle H(\frac{\pi}{4})) + |H(\pi/2)|\sin(\frac{\pi n}{2} + \angle H(\pi/2))$

$H(0) = \frac{1.9}{1 + .9} = 1$, $|H(\pi/4)| = \left|\frac{1.9e^{j\pi/4}}{e^{j\pi/4} + .9}\right| = 1.082$, $|H(\pi/2)| = \left|\frac{1.9e^{j\pi/2}}{e^{j\pi/2} + .9}\right| = 1.412$

$\angle H(\pi/4)$: $\frac{1.9e^{j\pi/4}}{e^{j\pi/4} + .9} = 1.0086 + 0.3922j \Rightarrow \tan^{-1}(\frac{.3922}{1.0086}) = .371$ radians

$\angle H(\pi/2)$: $\frac{1.9e^{j\pi/2}}{e^{j\pi/2} + .9} = 1.0497 + 0.9448j \Rightarrow \tan^{-1}(\frac{.9448}{1.0497}) = .733$ radians

⑥ Plugging these values in...

$y[n] = 1 + 1.082\sin(\frac{\pi n}{4} + .371) + 1.412\sin(\frac{\pi n}{2} + .733)$

Figure 1.1: My handwritten work to compute the output y[n] when x[n] is the input to the given difference equation. In the first step, I showed how the given h[n], the impulse response to the difference equation, is verified. This was part a of question 5.45. Secondly, I computed the DTFT of h[n] to get h(Ω), the impulse response in the frequency domain. In the third step, I calculated the DTFT of x[n] to get x(Ω). In step four, I multiplied x(Ω) and h(Ω) to compute y(Ω). This multiplication in the frequency domain is equivalent to the convolution in the time domain. The next step is the longest step of the process. I began by computing the inverse DTFT of y(Ω) to get y[n]. Once I had the format of y[n], I had to finish the equation by computing H(Ω) at the specified frequencies. I computed H(0), H(π/4), and H(π/2). I then simply had to calculate the angles of H(π/4) and H(π/2) to complete the equation for y[n]. Putting it all together in step 6, y[n] = 1 + 1.082sin(πn/4 + 0.371) + 1.412sin(πn/2 + 0.733).

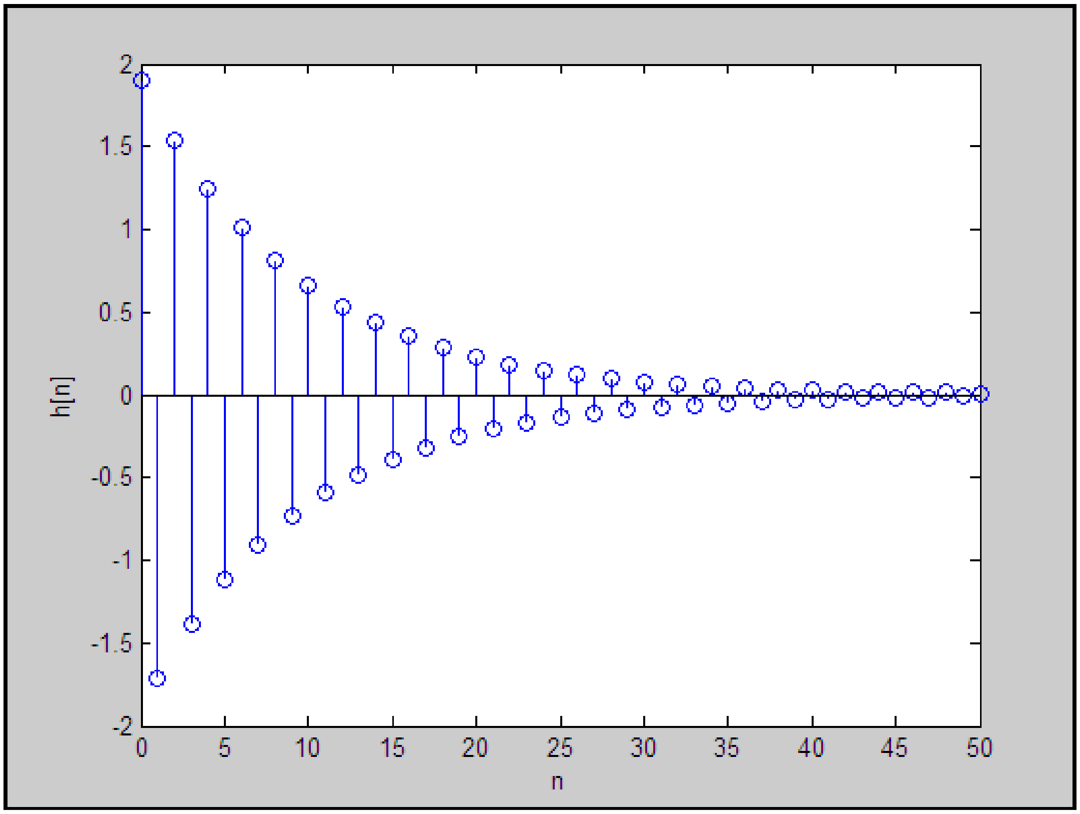


Figure 1.2: $h[n] = 1.9(-0.9)^n u[n]$: The impulse response to the given difference equation.

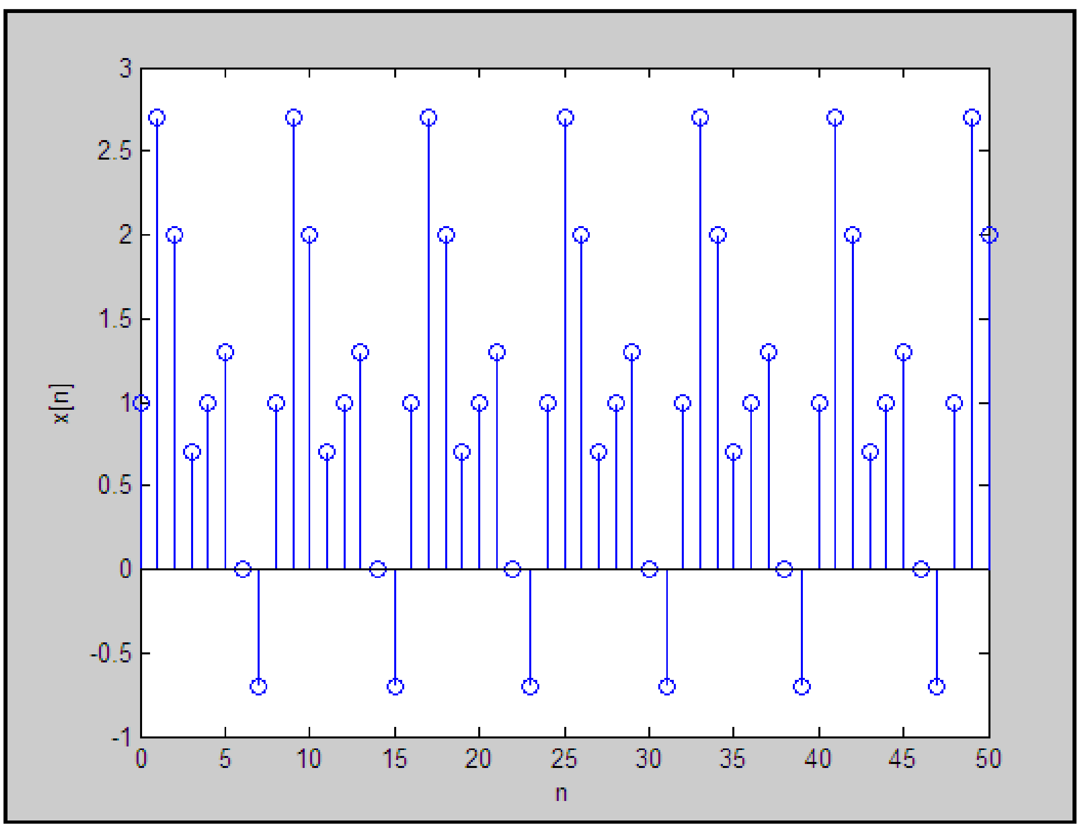


Figure 1.3: $x[n] = 1 + \sin(\pi n/4) + \sin(\pi n/2)$

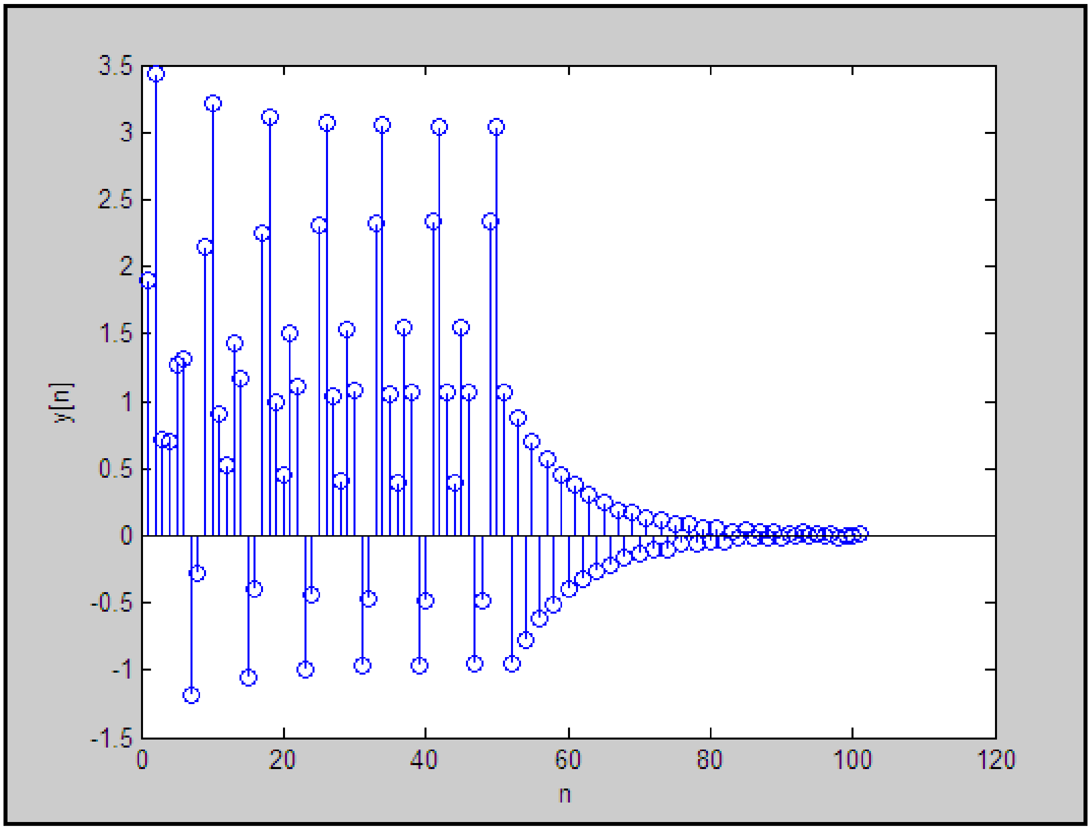


Figure 1.4: MATLAB y[n]: The output response of the given difference equation to the input x[n].

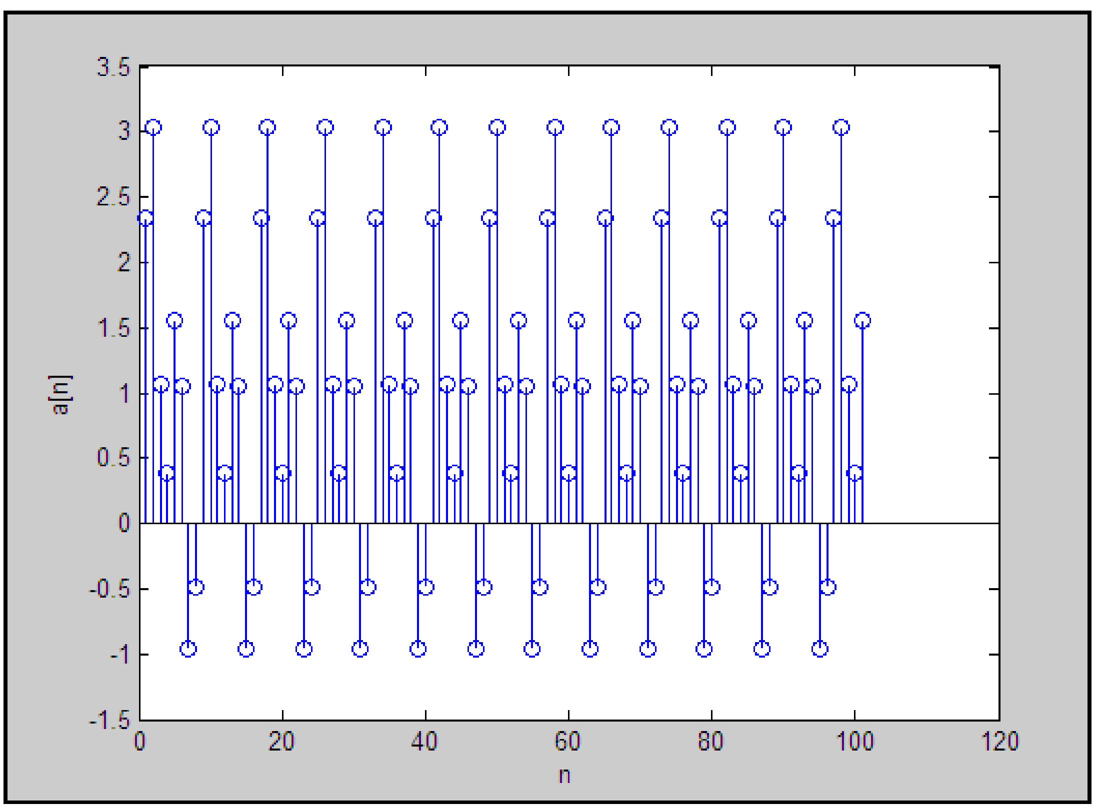


Figure 1.5: Theoretical y[n]

```
>> n = 0:1:50;
>> h = 0:1:50;
>> x = 0:1:50;
>> a = 0:1:100;
>> for j = 1:51,
h(j) = 1.9*(-0.9)^(j-1);
x(j) = 1+sin(0.25*pi()*(j-1))+sin(0.5*pi()*(j-1));
n(j) = (j-1);
end;
>> y = conv(x,h);
>> stem(n,x);
>> xlabel('n');
>> ylabel('x[n]');
>> stem(n,h);
>> xlabel('n');
>> ylabel('h[n]');
>> stem(y);
>> xlabel('n');
>> ylabel('y[n]');
>> for j = 1:101,
a(j) = 1+1.082*sin(pi()/4*(j-1)+.371)+1.412*sin(pi()/2*(j-1)+.733);
end;
>> stem(a);
>> xlabel('n');
>> ylabel('a[n]');
```

Figure 1.6: MATLAB Code. I began by declaring arrays to hold the values of h[n] and x[n]. I also declared an array to hold my theoretical solution, a[n]. I then used a for loop to populate the arrays with the values of h[n] and x[n] for n from 0 to 50 for a total of 51 values in each array. The plots of h[n] and x[n] can be seen in Figure 1.2 and 1.3 respectively. Notice the *u*[n] is disregarded in the x[n] equation because it is zero for all negative values of n. Then I computed the convolution of h[n] with x[n] which gives the output response y[n] (Figure 1.4). I finally populated the array a[n] with my theoretical values with n ranging from 0 to 100 and plotted it (Figure 1.5).

Conclusion:

From my handwritten work and my MATLAB simulation, it can be seen that y[n] is periodic. It is periodic with period n = 8 as seen in Figure 1.5. This is because the input x[n] is periodic with period n = 8 as seen in Figure 1.3. The difference between the graph of my analytical solution and my MATLAB solution is that the MATLAB simulation does not take into consideration that x[n] continues to n = infinity. In MATLAB, my samples of x[n] and h[n] are finite so the convolution of the two signals eventually decays to zero. In the real world with an x[n] that continues infinitely, my analytical solution holds true. Notice that the MATLAB simulation of y[n] does not seem to level out until a few periods into the convolution. This is because the discrete convolution is really a summation that approaches my analytical value as h[n] goes to zero. h[n] visibly goes to zero at approximately n = 32 (or 4 periods of x[n]) and it can be seen that y[n] has approximately approached its limit after 4 periods. In conclusion, my MATLAB simulation clearly shows that the output y[n] of the difference equation when the input is x[n] is the convolution of h[n], the impulse response of the difference equation, with x[n].