Boran Kılıç - 22103444

EEE 431- Digital Communications
Computational Assignment 2

Q1- Filtering of Random Processes

This part the assignment includes generating a random process, estimating its PSD and filtering.

a)

Given $W_n = 0.9 W_{n-1} + V_n$, $W_0 = 0$

$W_n = 0.9 W_{n-1} + V_n$

$W_{n-1} = 0.9 W_{n-2} + V_{n-1}$ $\Rightarrow$

$\vdots$

$W_1 = 0.9 W_0 + V_0$

$W_0 = 0$

$W_n = (0.9)^{n-1} V_1 + (0.9)^{n-2} V_2 + \dots + V_n$

$\vdots$

$W_3 = 0.81 V_1 + 0.9 V_2 + V_3$

$W_2 = 0.9 V_1 + V_2$

$W_1 = V_1$

$W_0 = 0$

$$\Rightarrow W_n = \sum_{k=0}^{n-1} (0.9)^k V_{n-k} , \quad n = 1, 2, 3, \dots$$

$$W_0 = 0$$

→ V_n is a iid Gaussian Noise therefore WSS.

→ W_n is a linear combination of V_n
which means W_n is also Gaussian. Therefore, also WSS.

* Check mean:
V_n zero mean iid Gaussian, therefore

$$E[W_n] = \sum_{k=0}^{n-1} (0.9)^k E[V_{n-k}] = 0$$

So mean is time-invariant.

* Check Autocorrelation:

$$R_W(m,n) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (0.9)^i (0.9)^j \underbrace{E[V_{m-i} V_{n-j}]}_{R_V(m-i,\, n-j) = R_V(m-n-i+j) = R_V(\tau)}$$

$m-i = n-j$
$n-m = i-j = \tau$

Since V_n iid $N(0,1)$

$$R_V(\tau) = \delta(\tau)$$

$$\Rightarrow R_W(n,m) = R(\tau)$$

$\therefore$ WSS ✓

b)

$n = 1, 2, \ldots, N$ where $N = 10^6$ realizations of W_n are generated and plotted using the definition of W_n.

$$W_n = 0.9W_{n-1} + V_n$$

where V_n is a zero mean IID Gaussian noise.

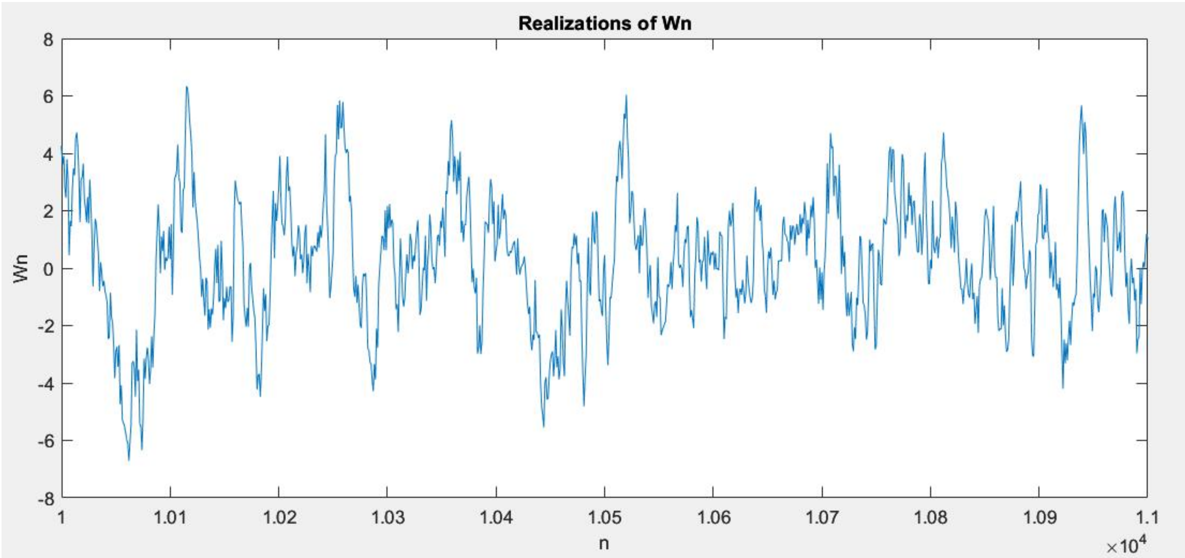


Figure 1: 1000 realization of W_n

c)

$n = 1, 2, \ldots, N$ where $N = 10^6$ realizations of Y_n are generated and plotted using the definition of Y_n

$$Y_n = (W_n + W_{n-1} + \cdots + W_{n-19})/20$$

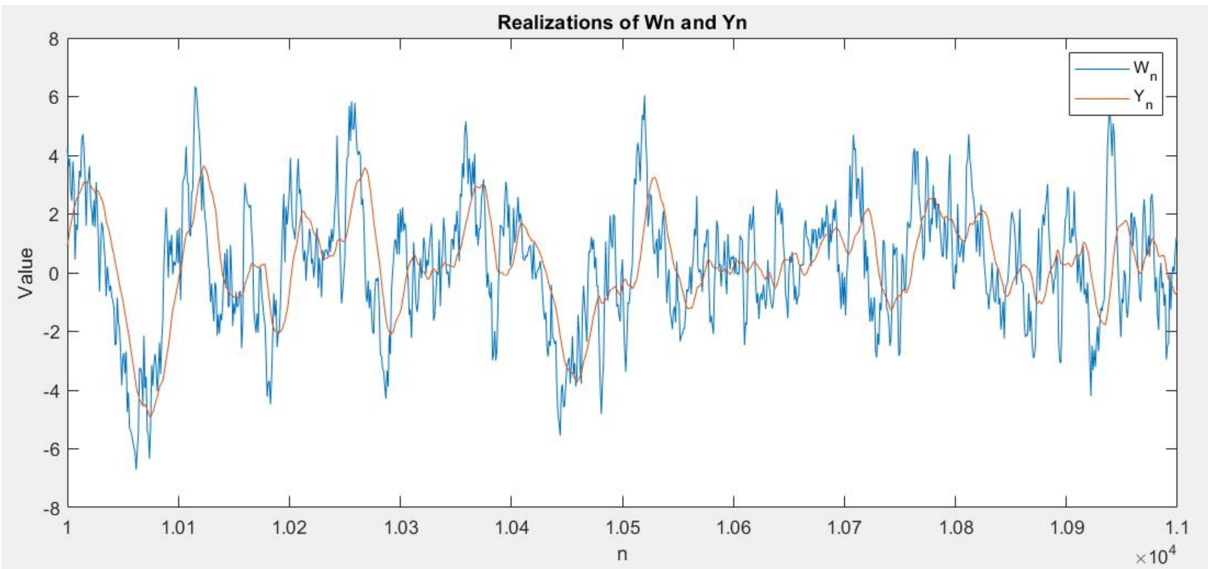


Figure 2: 1000 realization of Y_n on top of W_n

From *Figure 2*, visually it is evident that variance of Y_n is less W_n. Intuitevely, this is due to Y_n being the moving average of W_n for every 20 samples. Taking average of a signal brings it closer to its mean acting like a low-pass filter. For instance if Y_n was defined over the whole signal instead of just 20 samples we would get the mean 0 (or empirical mean that is very close to zero).

To show rigorously let $var(W_n) = \sigma_W^2$, $var(Y_n) = \sigma_Y^2$

$$Y_n = \frac{1}{20} \sum_{k=0}^{19} W_{n-k}$$

Then:

$$\sigma_Y^2 = var(Y_n) = Var\left(\frac{1}{20} \sum_{k=0}^{19} W_{n-k}\right)$$

$$\sigma_Y^2 = \frac{1}{400} \sum_{i=0}^{19} \sum_{j=0}^{19} cov(W_{n-i}, W_{n-j})$$

W_n is AR(1) process:

$$Cov(W_{n-i}, W_{n-j}) = \sigma_w^2 \cdot \rho^{|i-j|} \quad \text{with } \rho = 0.9$$

Therefore:

$$\sigma_Y^2 = \frac{1}{400} \sum_{i=0}^{19} \sum_{j=0}^{19} \sigma_w^2 \cdot 0.9^{(i-j)} < \sigma_w^2$$

d)

Power Spectral Densities (PSD) of W_n and Y_n are estimeted using the *pwelch* function in MATLAB.

When *pwelch* function was directly applied the output was noisy. In order to get a better approximation, windowing, overlapping, and zero-padding functionalities of the *pwelch* function are used.

- A Hamming window of length 1024 is applied to reduce spectral leakage.
- 50% overlap (512 samples) is used to provide smoother and less noisy estimates by averaging more data segments.
- The FFT length (nfft) is set to 2048, introducing zero-padding which improves the frequency resolution of the PSD estimate.

PSD estimates are plotted.

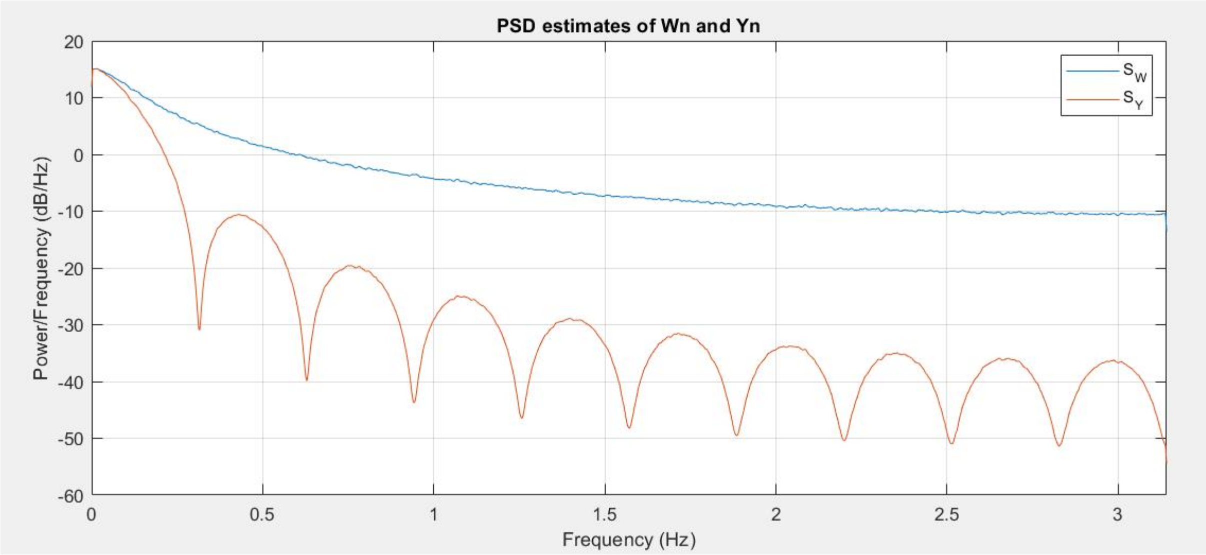


Figure 3: PSD estimates of Y_n and W_n

Comparing the PSDs of *Wn* and *Yn* in Figure 3: the PSD of Y_n has deep notches because Y_n is an average of 20 consecutive W_n samples, effectively acting like a moving average filter, which introduces periodic nulls in the frequency domain, while W_n is a correlated AR(1) process whose PSD smoothly decays without such sharp periodic cancellations.

The impulse response and magnitude response of the system is found analytically.

One way to find the impulse response of a system is to input impulse to the system. The system is given by:

$$y[n] = \frac{1}{20} \sum_{n=0}^{19} w[n-k]$$

So let $w[n-k] = \delta[n-k]$ impulse response can be found as:

$$h[n] = \frac{1}{20} \sum_{n=0}^{19} \delta[n-k]$$

Taking DTFT of $h[n]$ in order to find the magnitude response

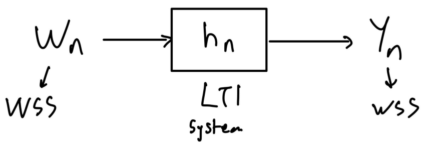
$$H(f) = \frac{1}{20} \sum_{n=0}^{19} e^{-j2\pi n f}$$

$$= \frac{1}{20} \left(e^{-j2\pi 0 f} + e^{-j2\pi 1 f} + \dots + e^{-j2\pi 18 f} + e^{-j2\pi 19 f} \right)$$

$$= \frac{1}{20} \frac{1 - e^{-j2\pi 20 f}}{1 - e^{-j2\pi f}}$$

$$|H(f)| = \frac{1}{20} \left| \frac{1 - e^{-j2\pi 20 f}}{1 - e^{-j2\pi f}} \right|$$

Using the PSD estimates the magnitude response of an LTI system can estimated with the following approach:



$$S_Y(f) = |H(f)|^2 S_W(f)$$

$$|H(f)| = \sqrt{\frac{S_W(f)}{S_Y(f)}}$$

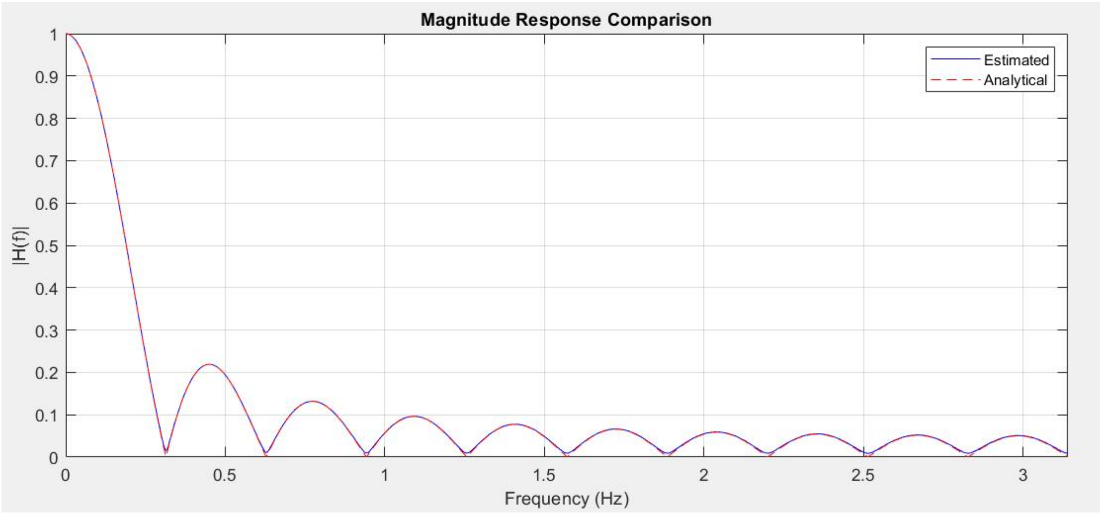


Figure 4: Estimated and Analytical magnitude responses |H(f)|

In Figure 4, it is visible that the estimated and analytical magnitude responses matches exactly which means this is good approximation.

e)

White Gaussian noise of length N is generated and plotted.

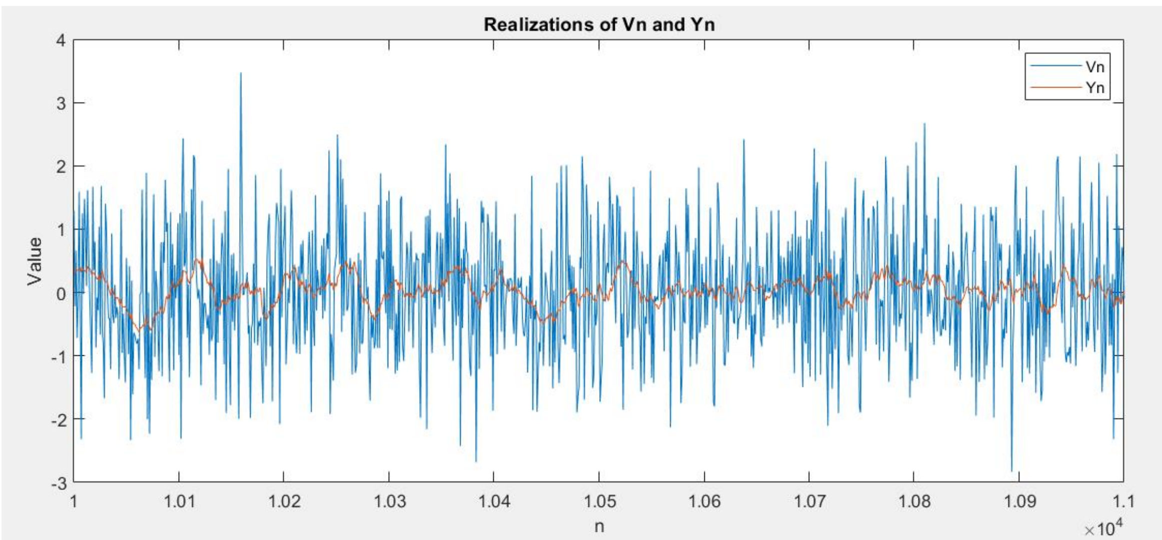


Figure 5: 1000 realization of Y_n on top of V_n

Similar to part (c) the variance of Y_n is smaller than variance of V_n. It can be proven analytically as follows:

Let $var(V_n) = \sigma_V^2$, $var(Y_n) = \sigma_Y^2$ $\quad V_n \sim N(0,1)$

$$Y_n = \frac{1}{20} \sum_{k=0}^{19} V_{n-k}$$

Then:

$$\sigma_Y^2 = var(Y_n) = Var\left(\frac{1}{20} \sum_{k=0}^{19} V_{n-k}\right)$$

$$\sigma_Y^2 = \frac{1}{20^2} 20 \sigma_V^2 = 0.05 \sigma_V^2 < \sigma_V^2$$

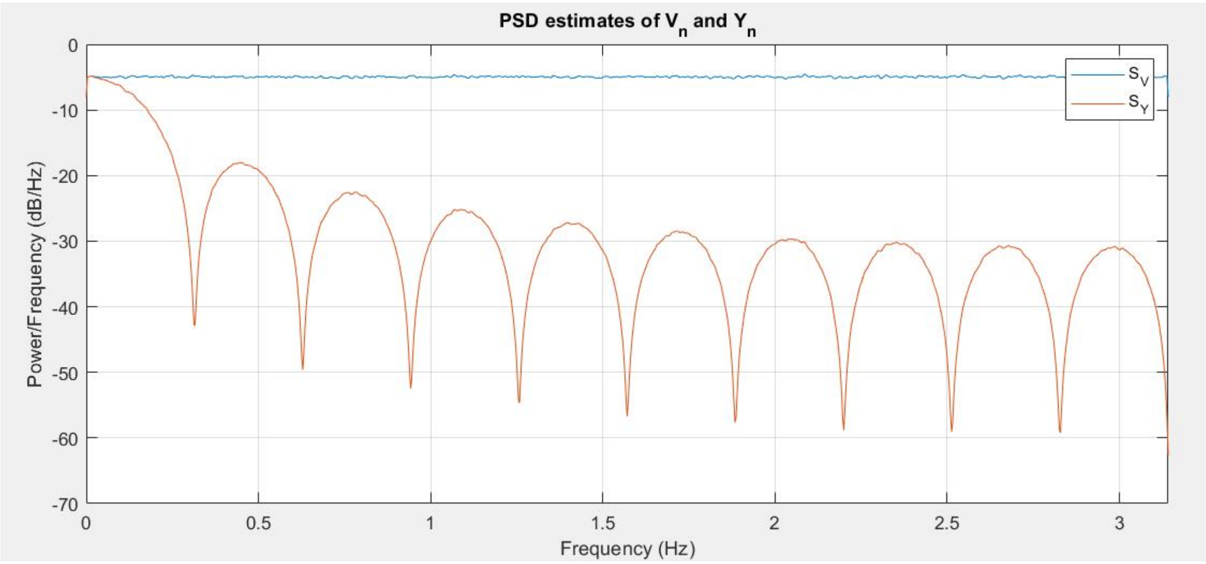


Figure 6: PSD estimates of V_n and Y_n

Comparing the PSDs of *Vn* and *Yn* in Figure 6: the PSD of V_n is flat because it is white Gaussian noise (equal power at all frequencies), while the PSD of Y_n shows periodic notches due to the averaging over 20 samples, acting like a moving average filter that shapes the spectrum.

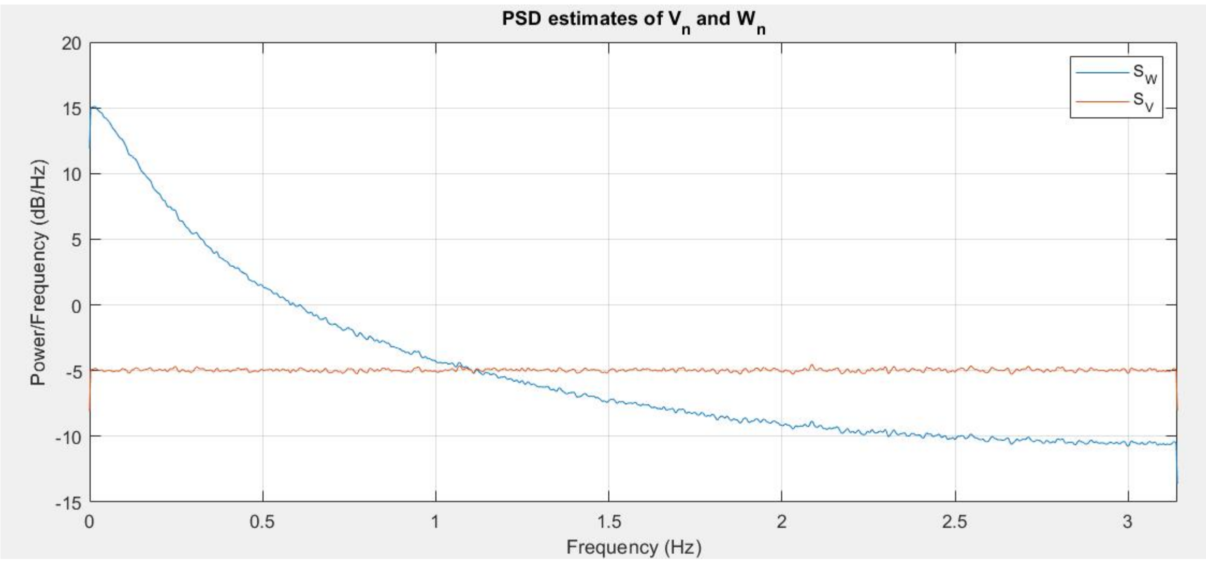


Figure 7: PSD estimates of W_n and V_n

Comparing the PSDs of *Vn* and *Wn* in Figure 7: the PSD of V_n is flat because it is white noise, while the PSD of W_n decays with frequency due to its AR(1) structure, showing that W_n has more low-frequency energy compared to V_n.

Magnitude response of the system is both found analytically and estimated using the same approach in part (d).

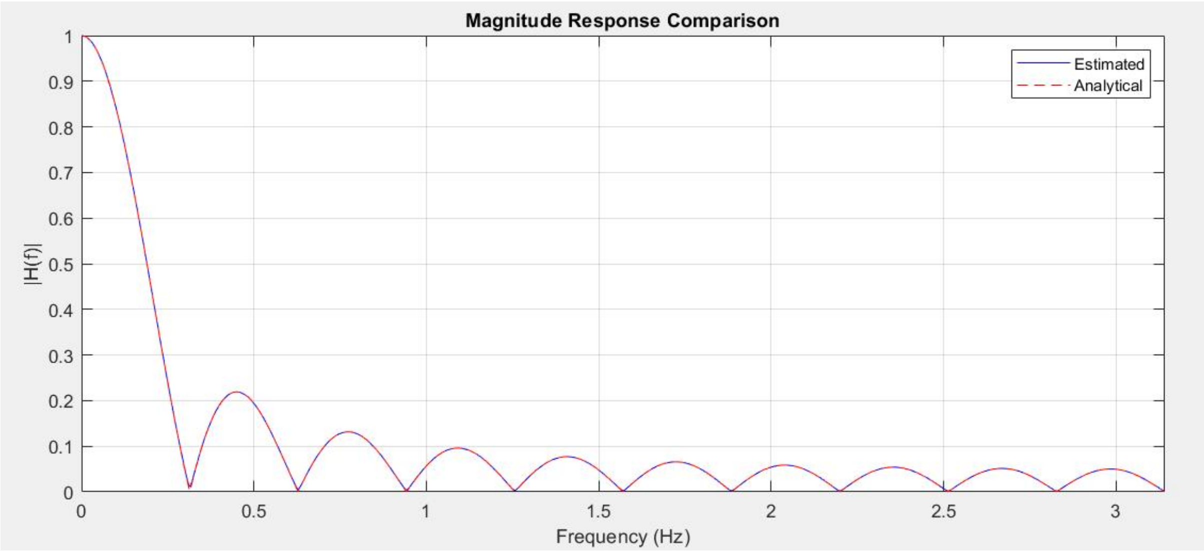


Figure 8: Estimated and Analytical magnitude responses |H(f)|

Similar to part (d), Figure 8 depicts that the estimated and analytical magnitude responses matches exactly which means this is good approximation.

Q2- DSB-SC and Frequency Offset

This part the assignment includes implementation of Double Sideband-Suppressed Carrier (DSB-SC) amplitude modulation with various phase offsets.

a)

The message signal is given as:

$$m(t) = sin(2\pi 100t)$$

Figure 9 shows the message signal that is wanted to trasmitted.

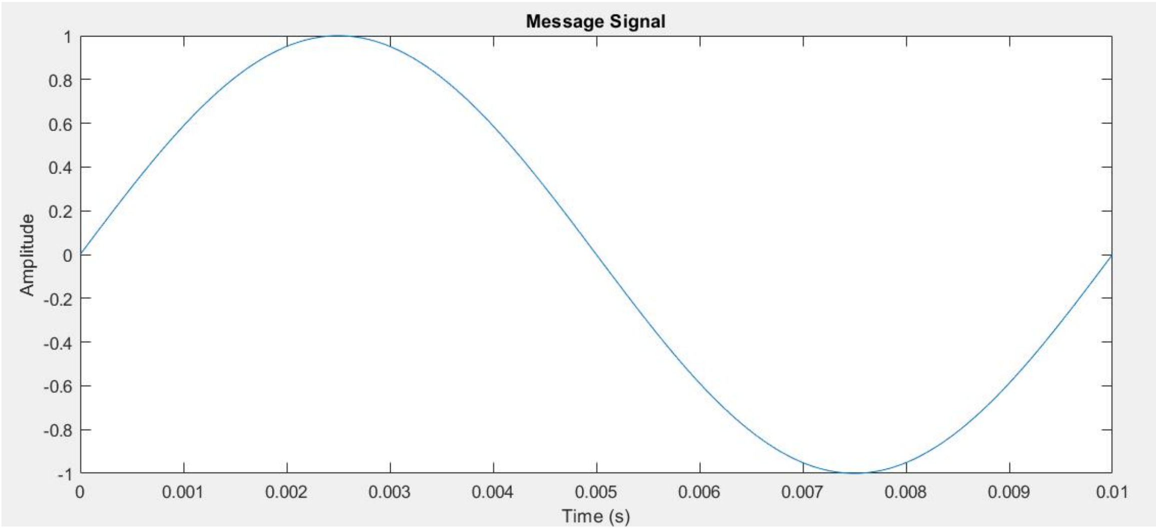


Figure 9: Message signal m(t)

This message signal is modulated with carrier frequency $f_c = 5\ kHz$ using DSB-SC method. Modulated signal:

$$x(t) = A_c m(t) cos(2\pi f_c t)$$

Where amplitude of the carrier determined as Ac = 1. Figure 10 shows the modulated signal.

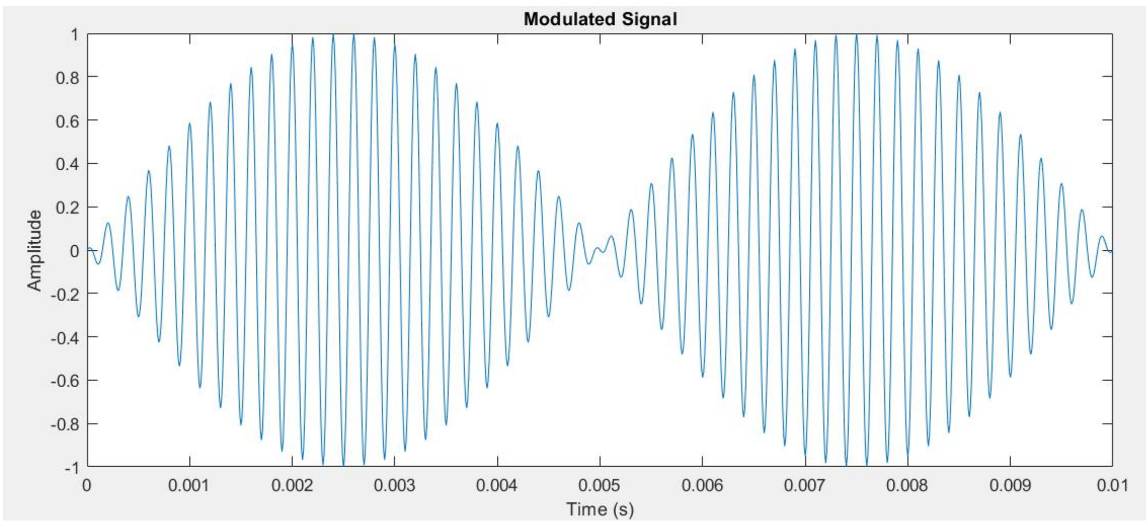
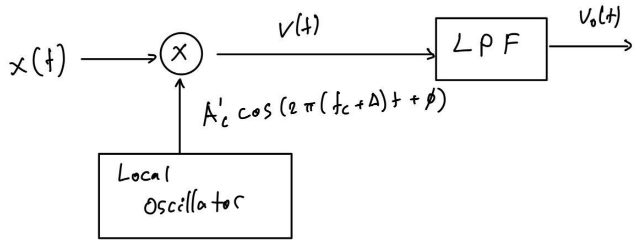


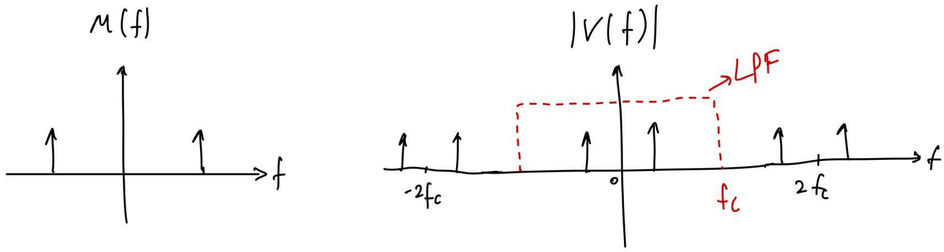
Figure 10: Modulated signal x(t)

b)

The message signal is demodulated using coherent detection technique.

Coherent Detection:





Cut-off frequency of the LPF is chosen to be same as carrier frequency f_c and that filter s realized with a butterworth filter in MATLAB.

Recieved signal : $x(t) = A_c\, m(t) \cos(2\pi f_c t)$

$$v(t) = A_c A_c' m(t) \cos(2\pi f_c t) \cos(2\pi(f_c+\Delta)t)$$

$$v(t) = \frac{A_c A_c'}{2} m(t) \left[\cos(2\pi\Delta t) + \cos(2\pi(2f_c+\Delta)t)\right]$$

$A_c = 1$, $A_c' = 2$

After LPF: $$v_o(t) = \frac{A_c A_c'}{2} m(t) \cos(2\pi\Delta t) = m(t)\cos(2\pi\Delta t)$$

Therefore CFO causes the recovered signal to be the modulated version of the message signal.

In this part it is assumed that the local oscillator generates signal that has carrier frequency off-set $\Delta \in \{0, 50, 100, 500, 1000\}$.

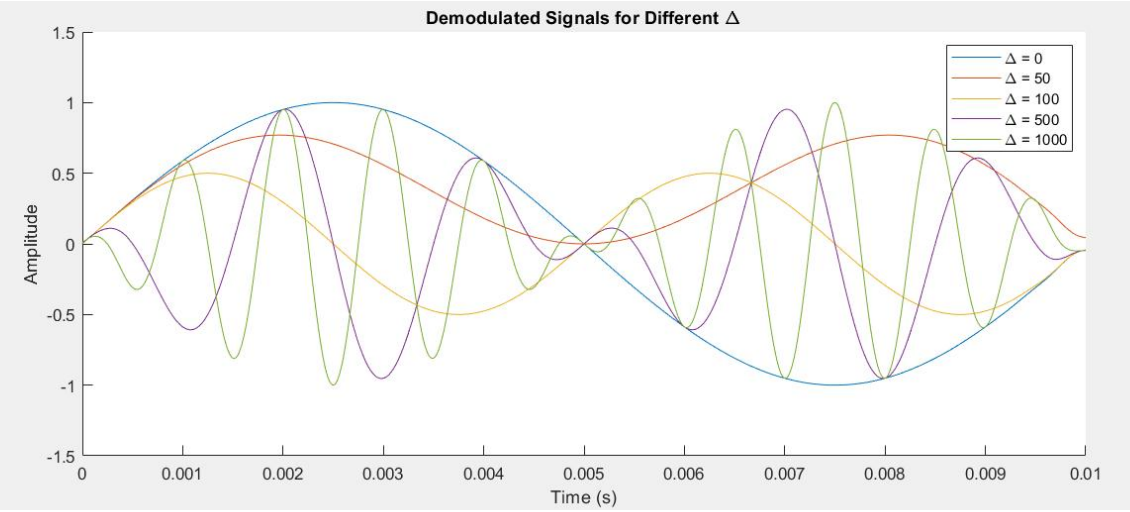


Figure 11: Demodulated signals for different Δ

Figure 11 depicts that as the frequency offset Δ increases, the demodulated signals become increasingly distorted compared to the original message signal. When Δ = 0, the signal is perfectly recovered. However, as Δ grows larger, for instance Δ = 500 or Δ = 1000, the signals exhibit faster oscillations and amplitude variations, indicating a significant degradation in coherent demodulation performance due to the carrier frequency mismatch.

c)

Carrier Frequency Offset (CFO) is a mismatch between the transmitter's and receiver's carrier frequencies during modulation and demodulation. In real-world systems, it is almost impossible for the transmitter and receiver oscillators to be perfectly synchronized. The effect of CFO may cause from the Doppler effects between source receiver. Simply Δ can be considered as a random variable therefore estimation methods, such as ML estimate, can be applied.

One common method is to use a Phase-Locked Loop (PLL), which dynamically tracks the incoming carrier frequency and phase, adjusting the local oscillator accordingly to minimize the offset [1]. Adaptive filtering techniques, like the Kalman filter, are also used when Δ varies randomly, providing real-time estimates and corrections.

In advanced systems like OFDM (used in Wi-Fi, 5G), CFO estimation algorithms are built into the receiver to track and correct frequency offsets at both coarse and fine levels [2]. Overall, CFO elimination is essential to ensure reliable communication.

d)

In this part, local signal generator adds a phase difference $\phi \in \{0, \pi/6, \pi/3, \pi/2\}$

Recieved signal : $x(t) = A_c\, m(t) \cos(2\pi f_c t)$

$$V(t) = A_c A_c' \cos(2\pi f_c t + \phi)\, m(t) + \frac{1}{2} A_c A_c' \cos\phi\, m(t)$$

$A_c = 1$, $A_c' = 2$

After LPF: $V_0(t) = \frac{A_c A_c'}{2} \cos\phi\, m(t) = \cos\phi\, m(t)$

Since the captured signal after LPF

$$v_0(t) = \cos(\phi) m(t)$$

The phase difference ϕ results a change in the amplitude of the signal. Figure 12 shows that as ϕ increases the amplitude of the detected signal gets suppressed and for large ϕ the amplitude is effectively zero.

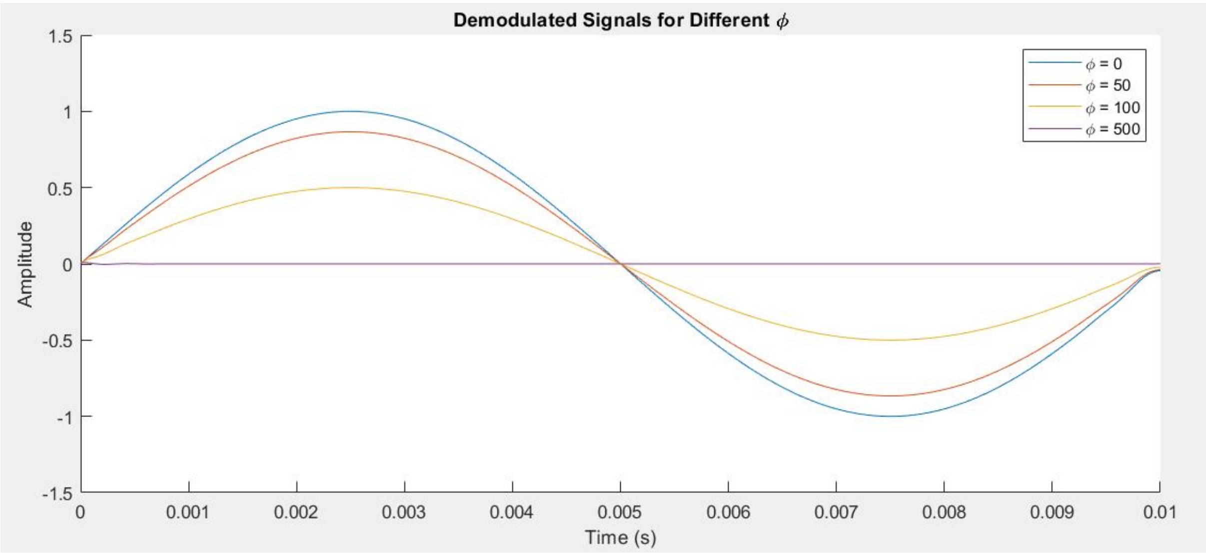


Figure 12: Demodulated signals for different ϕ

References

[1] J. G. Proakis and M. Salehi, "Section 8.8," in Fundamentals of Communication Systems, 2nd ed. Pearson, 2013.

[2] P. K. Nishad and P. Singh, "Carrier frequency offset estimation in OFDM systems," 2013 IEEE Conference on Information & Communication Technologies, Thuckalay, India, 2013, pp. 885-889, doi: 10.1109/CICT.2013.6558220.