Hand-In Exercise 2: Robustness and Implementation of Control Systems

Emil Asgeirsson (emasg18), Kasper Mikkelsen (kamik18) & Mads Christiansen (mach018)

1 Stability Margins

The system equation G(s) and the controller K(s) is given by:

$$K(s) = K_p \frac{s + \frac{1}{T_i}}{s} \qquad G(s) = \frac{k}{\tau s + 1} \tag{1}$$

1.1 Phase and Gain Margins

To determine the phase and gain margins of the system we will use the open loop system to determine the margins of the closed loop system. The transfer function used for the determining the phase and gain margins is

$$H(s) = G(s)K(s) \tag{2}$$

which is plotted in Figure 1. We look at the phase margin and gain margin where the curve of $KG(j\omega)$ meets a magnitude of 0 dB.

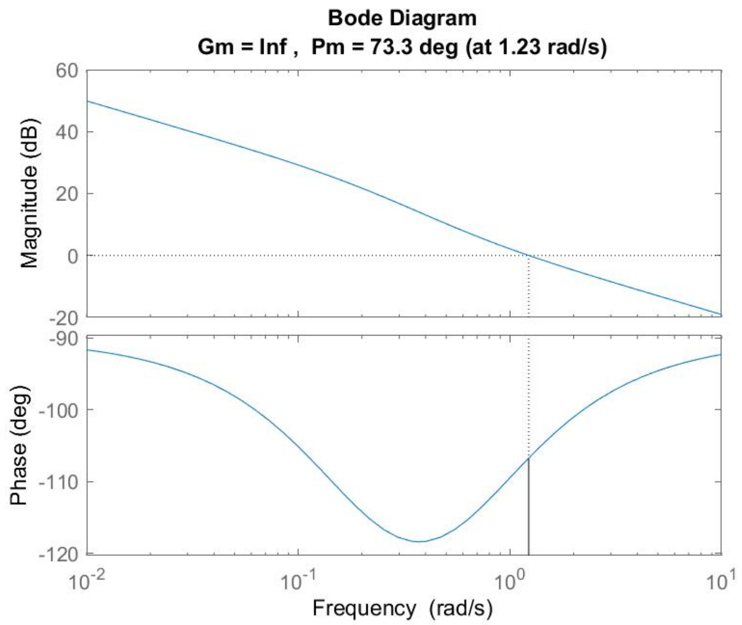

Figure 1: Closed Loop Margin

The phase margin (Pm) and gain margin (Gm) of the closed loop system are

$$Pm \approx 73.3^\circ$$
$$Gm = \infty \ dB$$

Gm is ∞ because the system can never become unstable.

1.2 Vector Margin

Figure 2 shows the Nyquist plot and from this we can determine the vector margin.

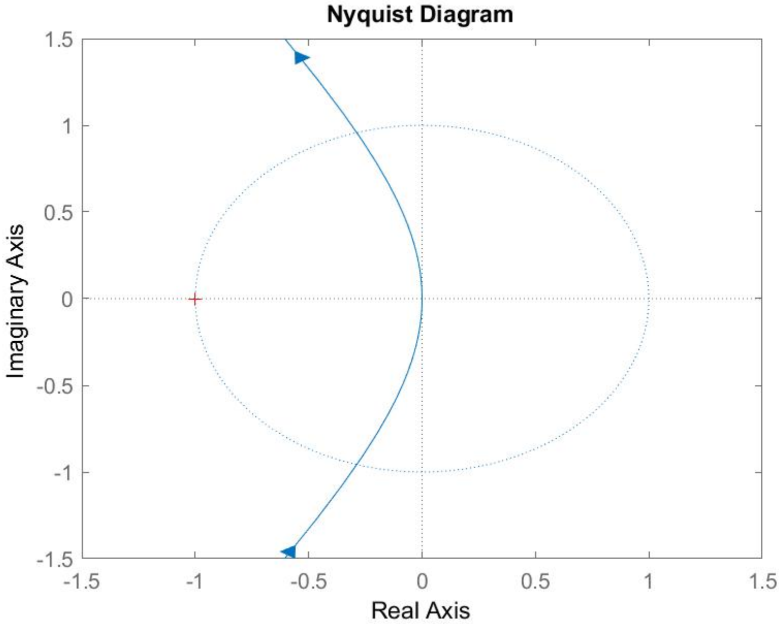

Figure 2: Nyquist plot

Looking at Figure 2 we can see that the plots forms a concave curve in left plane. The curve never reaches the -1 point and therefore we can determine that the system is stable since it never reaches -180°. The gain margin then teels us how far away we are from the -1 point. It is calculated by $\frac{1}{Gm}$. Since our gain is ∞ then this number becomes zero and we will have a length of 1 between 0 and -1 on the real axis. This means that the vector margin is -1. The direction of the arrows on the nyquist plot indicates that we are going in a counter clockwise direction. This is due to the system having 3 zeros and 4 poles.

1.3 Relation between Margins

In this section we are first going to define the phase margin and then the gain margin. After defining the phase margin and gain margin, we will describe how they relate to the vector margin.

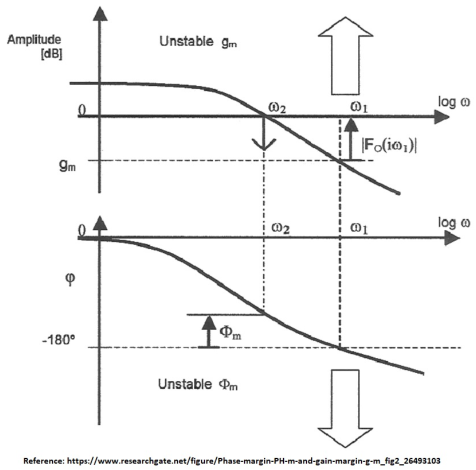

Figure 3: Phase and Gain Margins

The phase margin is used to describe the stability margin in a system. It is described by the amount the phase of $KG(j\omega)$ exceeds -180 °when $|KG(j\omega)| = 1$ (0 if plotted in dB). The phase needs to be positive for the system to be stable and that means that the phase deceeds -180 °.

The gain margin describes how much the gain can be increased before the system reaches instability. Most often this can be read from the bode plot by measuring the vertical distance between the the $|KG(j\omega)|$ curve and magnitude the frequency where $\angle G(j\omega) = -180°$. If there is no frequency the crosses -180 °then the gain margin is infinite.

The phase and gain margin combined is called a vector margin. The vector margin gives a better understanding of stability in complicated cases. In regards to this case we can easily determine the systems stability using only the bode plot. The gain margin tells you how much you can scale the nyquist plot before it becomes unstable. It's calculated by $\frac{1}{Gm}$. The phase margin tells us how much we can rotate the Plot before it becomes unstable.. Since the vector margin is a single parameter, it is easier to determine the stability that using the gain and phase margin in combination.

2 Anti-Windup

2.1 Anti-Windup for PI-Controller

The chosen anti-windup is back calculation. The concept of using back calculation as anti-windup, is to evaluate the controller output before and after the saturation. This form of anti-windup recompute the value of the integral gain, by taking the difference of the controller output, and when it saturates. This will cause the controller to reach the desired position without overshooting.

Another anti-windup for the PI-controller is clamping, where it detects if an integrator overflow has occurred. In case of an integrator overflow it will switch the integrator value to zero, and if that is not the case, it will return the actual value.

The implementation of back calculation anti-windup, is seen on figure 4. In the saturation block is the upper limit defined to 3000, and the lower limit to -3000.

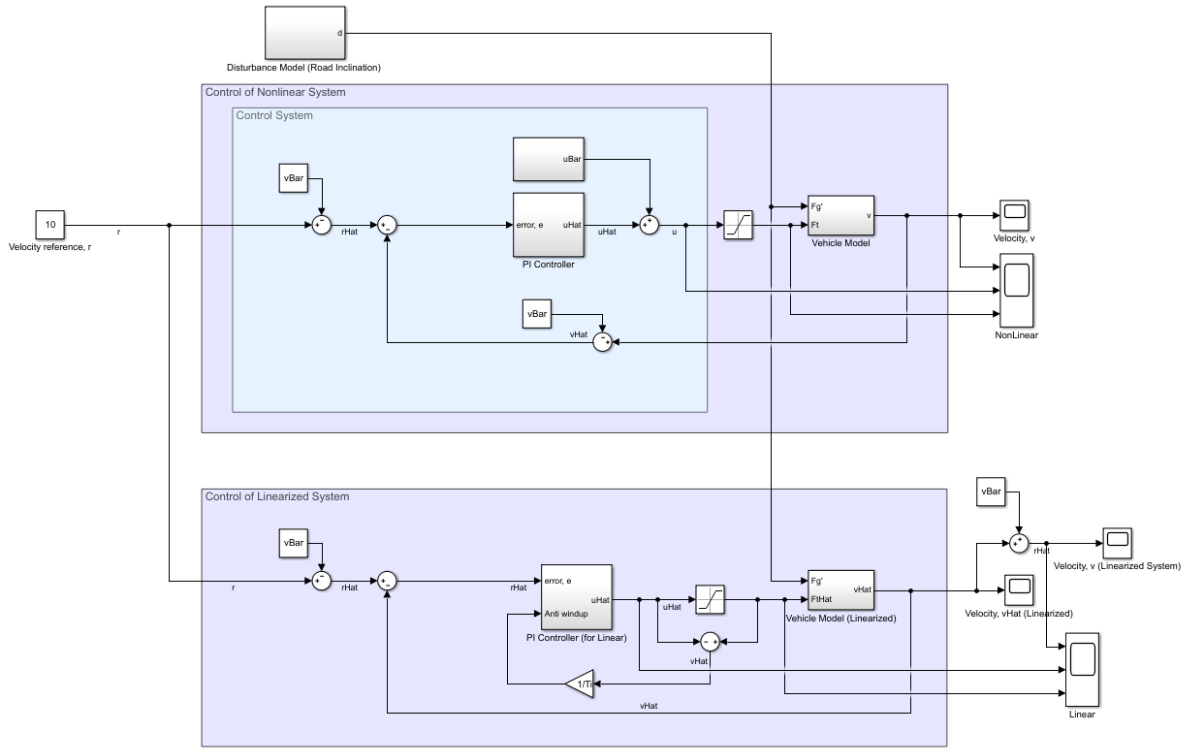

Figure 4: Block diagram of back calculation anti-windup. The upper part is the non linearized model, and the lower part is the linearized model.

2.2 Simulation 1: Duration 20 s and Reference 15 m/s

The results of the simulation running over 20 seconds with a reference of 15 m/s, is shown as a linearized model, figure 5, as well as the non linearized model, figure 6. Using the anti-windup in the linearized model gives better results, than using the non linearized model.

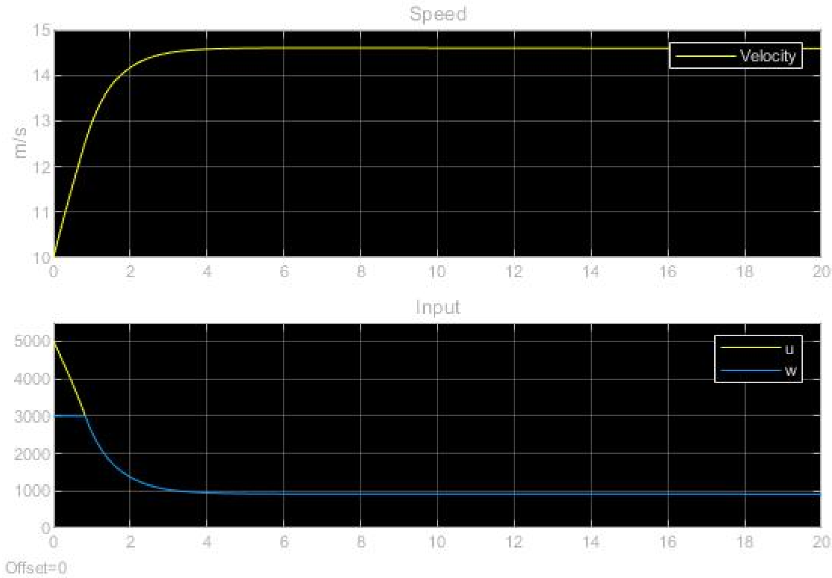

Figure 5: Linearized model over 20 seconds, with a reference of 15 m/s.

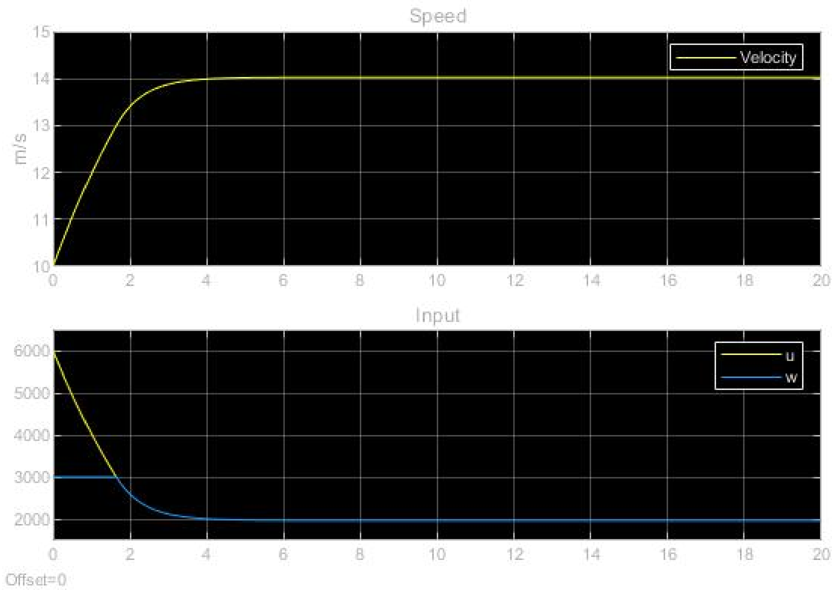

Figure 6: Non linearized model over 20 seconds, with a reference of 15 m/s.

2.3 Simulation 2: Duration 150 s and Reference 10 m/s

The results of the simulation running over 150 seconds with a reference of 10 m/s, is shown as a linearized model, figure 7, as well as the non linearized model, figure 8.

As seen on both figures, the road starts to incline after 50 seconds, and at 100 seconds, the incline is at 200 %, witch is the same as approximately 1.1 rad. Due to the restriction of a maximum input at 3000, the car fails to hold the speed, in both the linearized and non linearized model.

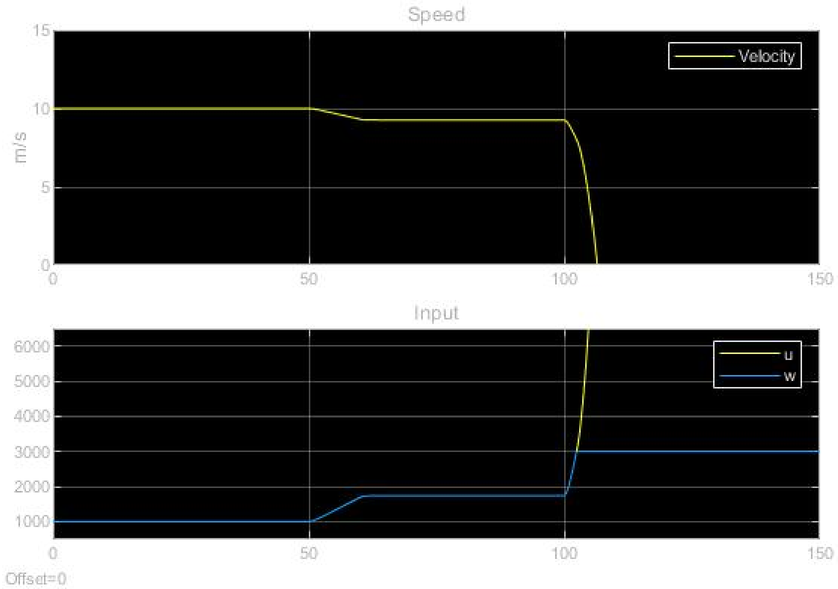

Figure 7: Non linearized model over 150 seconds, with a reference of 10 m/s.

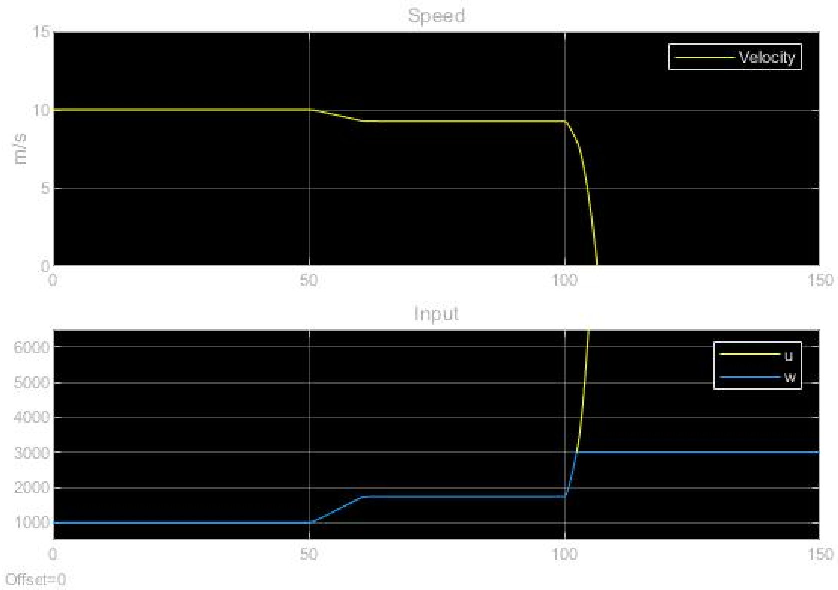

Figure 8: Non linearized model over 150 seconds, with a reference of 10 m/s.

This disturbance which cause the car to deaccelerate, is seen on figure 9

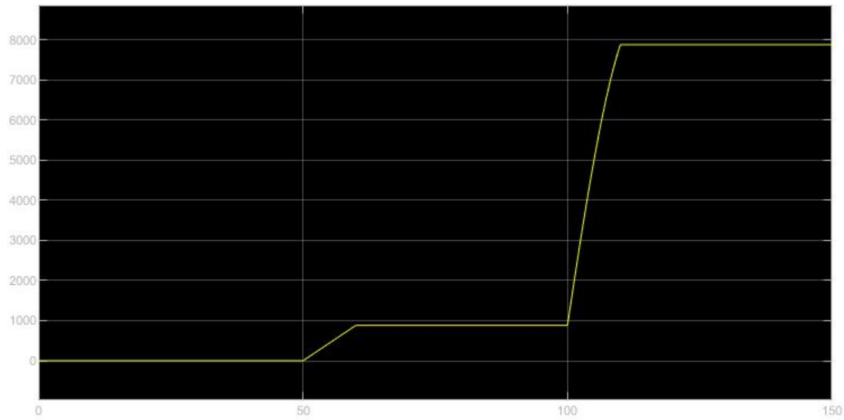

Figure 9: Disturbance.

3 Digitization

3.1 Discrete Time Controller

$$K(s) = K_p \frac{s + \frac{1}{T_i}}{s} \tag{3}$$

We are converting to an equivalent discrete controller by using the Trapezoidal rule.

$$K_d(z) = K(\frac{2}{T}\frac{z-1}{z+1}) = K_p \frac{\frac{2}{T}\frac{z-1}{z+1} + \frac{1}{T_i}}{\frac{2}{T}\frac{z-1}{z+1}} \tag{4}$$

$$= K_p \frac{\frac{z-1}{T(z+1)} + \frac{1}{2T_i}}{\frac{z-1}{T(z+1)}}$$

$$= K_p \frac{z - 1 + \frac{1}{2T_i}T(z+1)}{z-1}$$

$$= K_p \frac{z - 1 + \frac{1}{2T_i}Tz + \frac{1}{2T_i}T}{z-1}$$

$$= K_p \frac{(1 + \frac{1}{2T_i}T)z + \frac{1}{2T_i}T - 1}{z-1}$$

We insert the values of $T_i = 1.6$ and $K_p = 1000$ such that.

$$1000 \frac{\frac{2}{T}\frac{z-1}{z+1} + \frac{1}{1.6}}{\frac{2}{T}\frac{z-1}{z+1}} = 1000 \frac{(1 + 0.3125T)z + 0.3125T - 1}{z-1} \tag{5}$$

We determine an appropriate sampling frequency T by drawing and examining the bode-plot of the continuous open loop system at a magnitude of -3dB shown in figure 10.

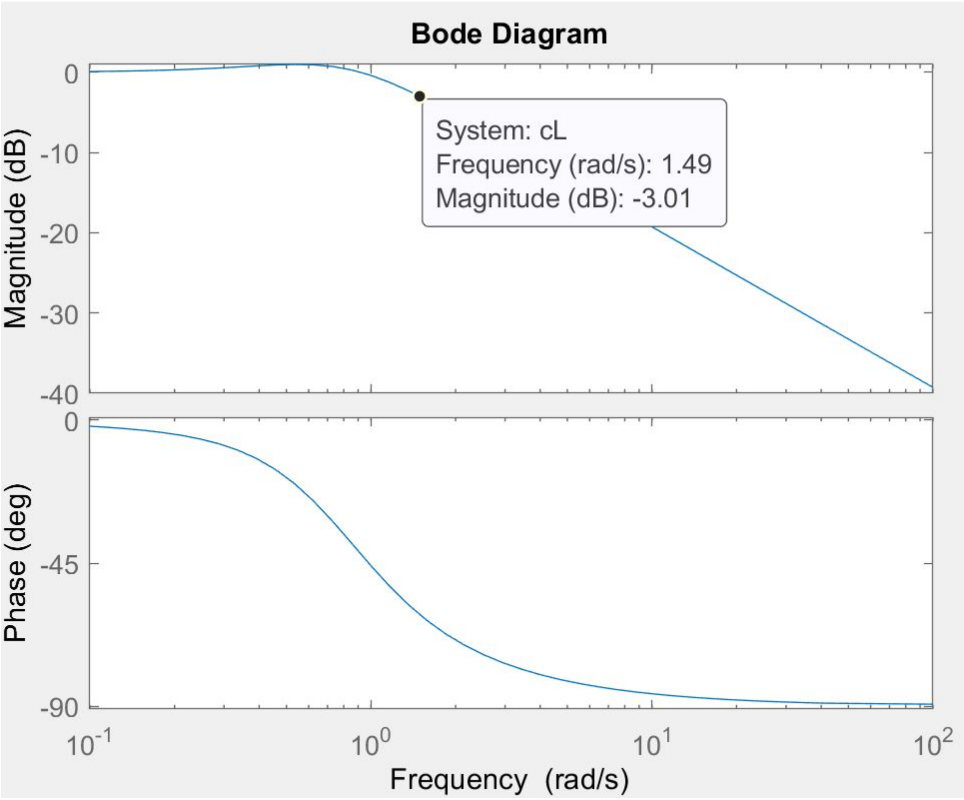

Figure 10: bode-plot of the Continuous closed loop system, examined at -3dB

-3dB is the maximum gain we can choose for calculating the sampling frequency so that the system is not unstable.
First we read the bandwidth which is in rad/s from the examined bode-plot and convert it to Hz.

$$f = \frac{1.49 \cdot 30}{2 \cdot \pi} = 7.1142 \text{ Hz}$$

Then using the rule of thumb, that the sampling frequency should be 30 times the bandwidth, we get a sampling frequency of.

$$T = \frac{1}{f} = 0.1406 \text{ s}$$

Which we use for the discrete controller.

$$1000 \frac{(1 + 0.3125 \cdot 0.1406)z + 0.3125 \cdot 0.1406 - 1}{z - 1} = \frac{1043.9375z - 956.0625}{z - 1}$$

To calculate the difference equation we use.

$$u(k) = \sum_{i=1}^{N} a_i u(k-i) + \sum_{j=0}^{M} b_j e(k-j)$$

Where $i \to N$ and $j \to M$ denote the order of z and a_i and b_j denote the factors of the corresponding order. on discrete controller $K(z)$ and get the difference equation.

$$u(k) = u[k-1] + 1043.9375e[k] - 956.0625e[k-1]$$

The controller as a discrete transfer function is $(u(z) = K(z)e(z))$

$$K(z) = \frac{1043.9375z - 956.0625}{z - 1}$$

The controller is implemented as the difference equation

$$u_k = u[k-1] + 1043.9375e[k] - 956.0625e[k-1]$$

3.2 Implementation in Simulink

The discretized controller is implemented in Simulink using the difference equation of the controller defined in 3.1. The implementation is done using the delay block and setting the Model configuration parameters to use a solver with type Fixed-step and a step-size of 0.1406 that corresponds to the sampling frequency T found in section 3.1. The diagram is seen in Figure 11.

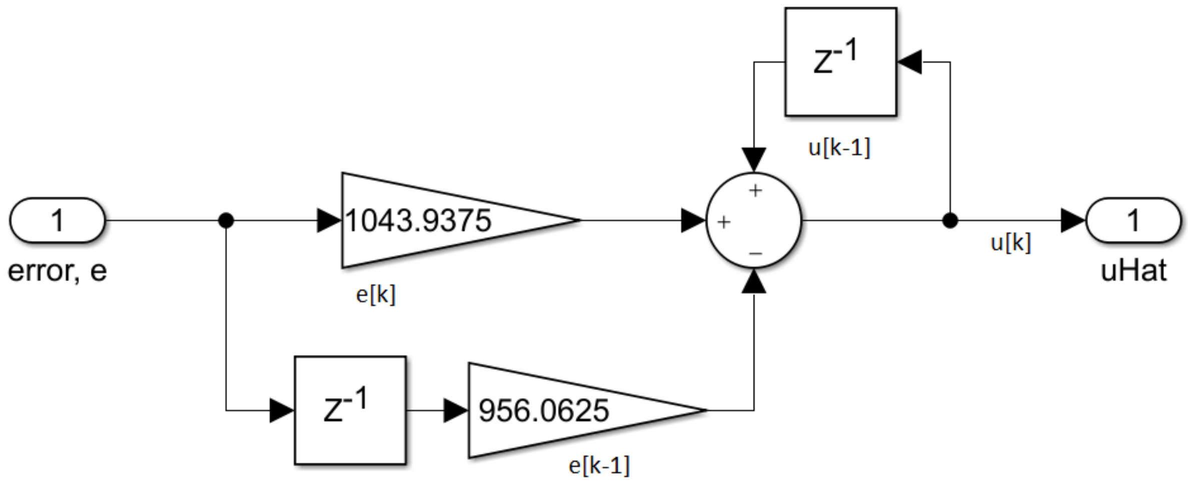

Figure 11: Block diagram of discrete controller implementation in simulink

3.3 Stability Margins

The determination of the stability margins of the system with plant model $\tilde{G}(s)$. Below is shown a Bode plot as Figure 12. Comparing the bode plot with Figure 1 we can see that the Pm is reduced by 5°. And if the gain is reduced below -50 dB we can see that we are very close to instability, but system will never become unstable. The transfer function used is G(s) and the controller K(s) is given by:

$$H(s) = K(s)\tilde{G}(s) = K(s)G(s)Gd(s) \tag{6}$$

$$H(s) = K_p \frac{s + \frac{1}{T_i}}{s} \cdot \frac{k}{\tau s + 1} \cdot \frac{1}{\frac{Ts}{2}s + 1} = \frac{Kp \cdot k(s + \frac{1}{T_i})}{(\tau s^2 + s)(\frac{T_s s^2}{2} + s)} \tag{7}$$

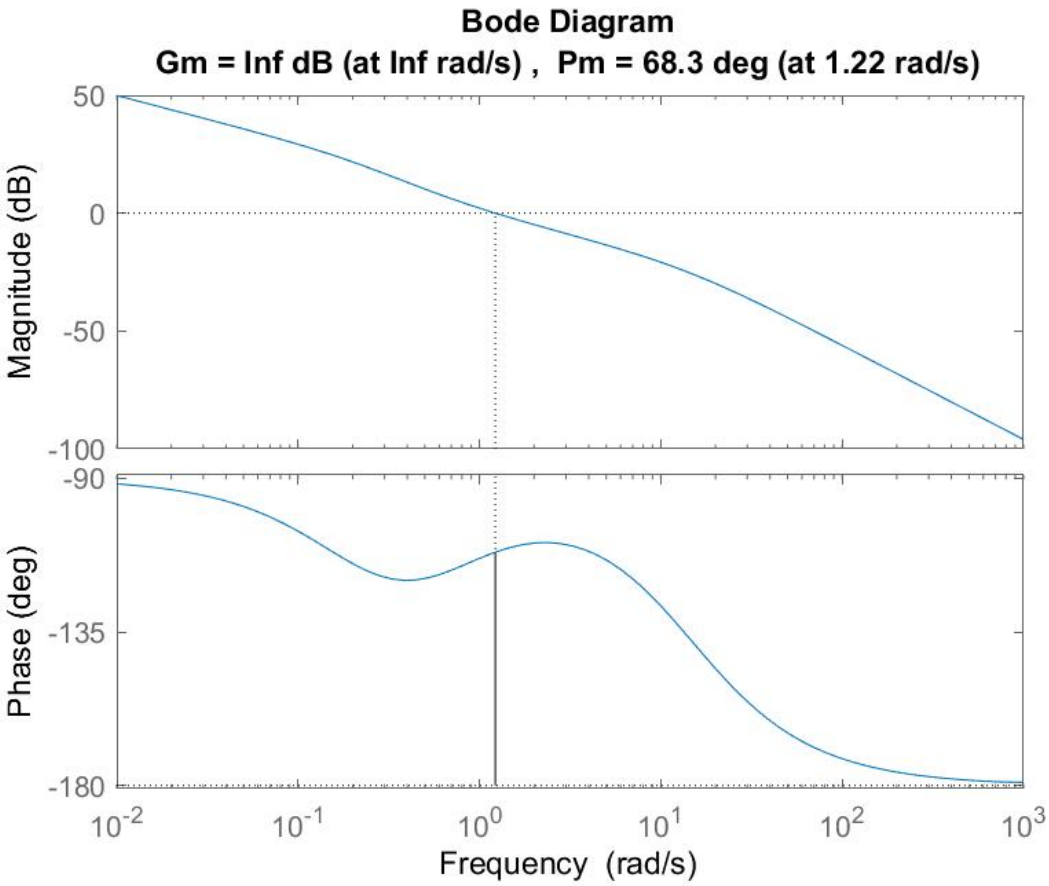

Figure 12: Bode plot for the discrete system

The phase margin (Pm) and gain margin (Gm) of the closed loop system are

$$Pm \approx 68.3°$$
$$Gm = \infty \ dB$$

3.4 Simulation 1: Duration 20 s and Reference 15 m/s

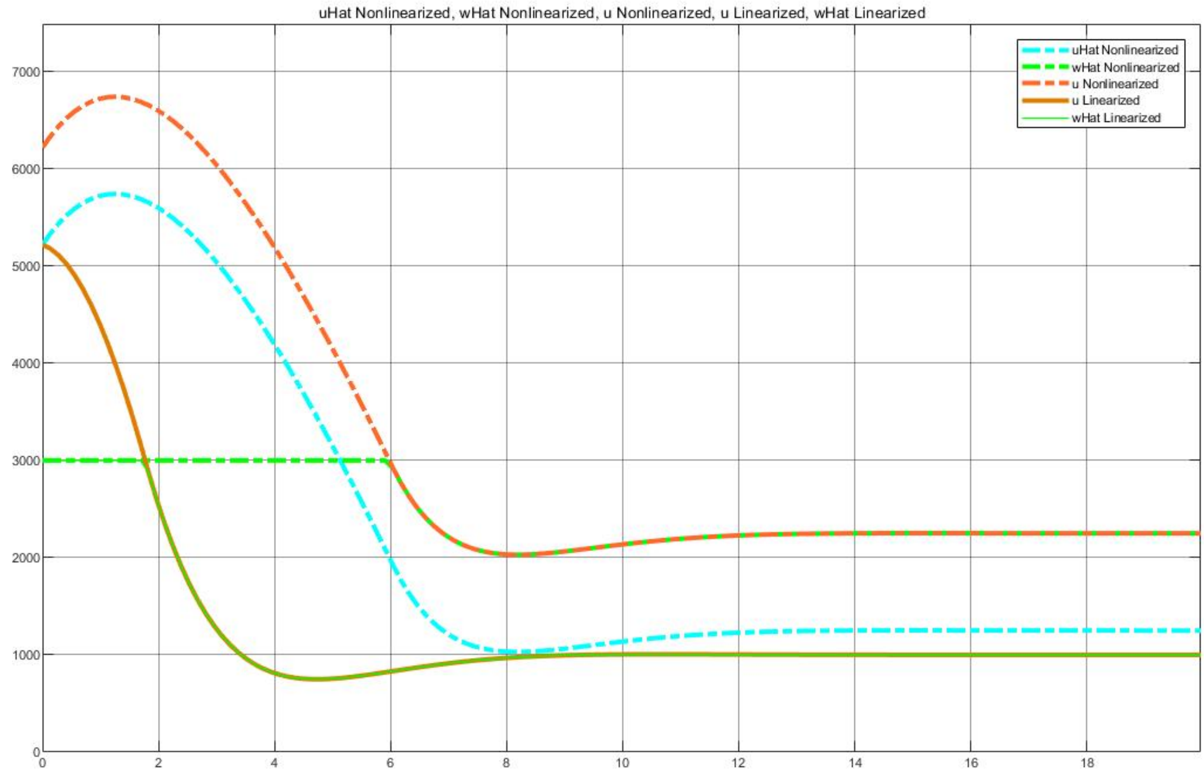

Figure 13: Plot of $\hat{u}$, $\hat{w}$, and u nonlinearized, and $\hat{w}$, and u Linearized.

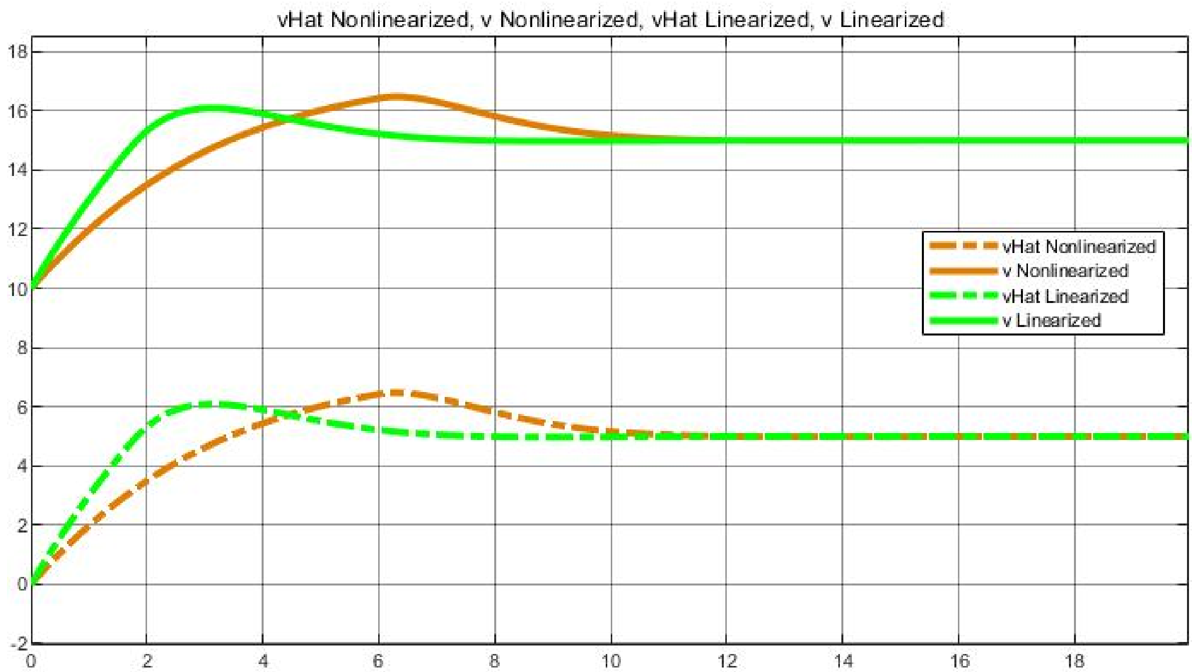

Figure 14: Plot of $\hat{v}$ and v nonlinearized and $\hat{v}$ and v Linearized

3.5 Simulation 2: Duration 150 s and Reference 10 m/s

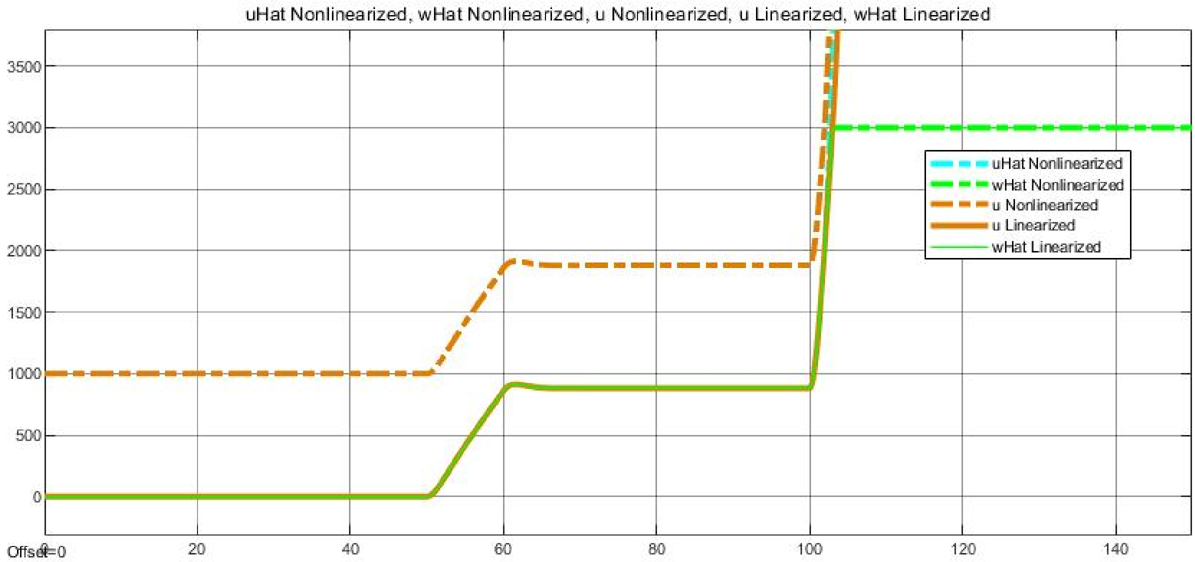

Figure 15: Plot of $\hat{u}$, $\hat{w}$, and u nonlinearized, and $\hat{w}$, and u Linearized.

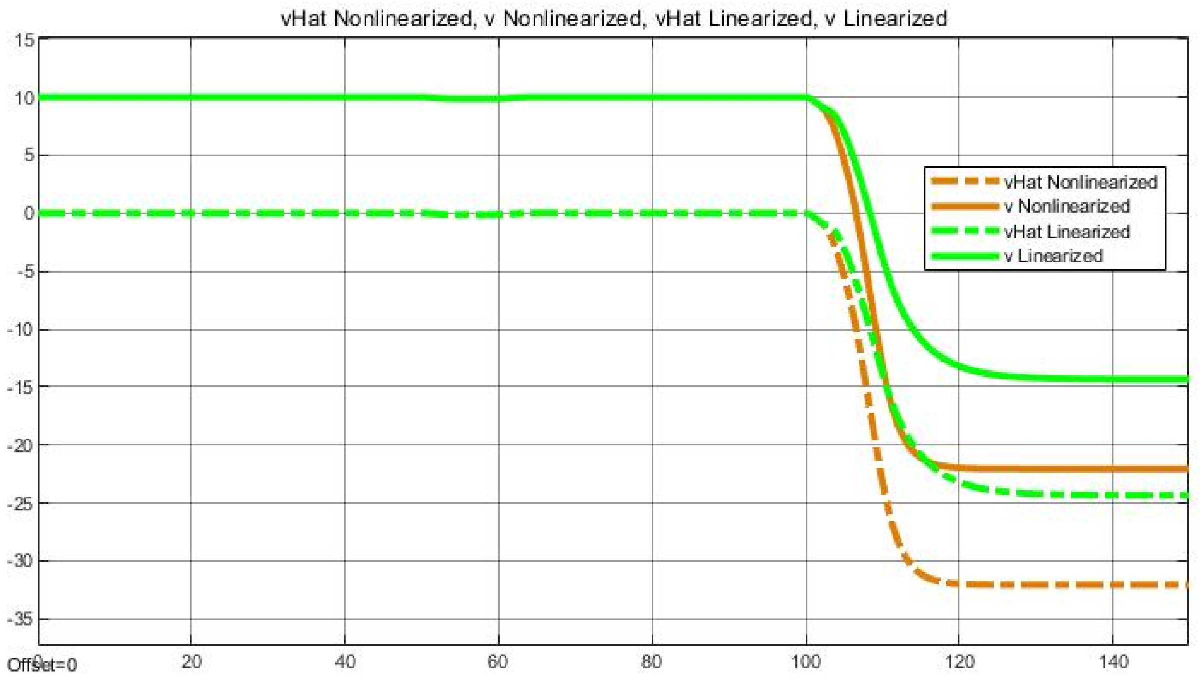

Figure 16: Plot of $\hat{v}$ and v nonlinearized and $\hat{v}$ and v Linearized