2 Theoretical Background

Developing the controller for the crane system consists of two parts

- Stability analysis
- Controller development

2.1 Stability analysis

The concept of stability refers to the ability of a system to maintain a desired state or equilibrium in the face of internal or external disturbances. Stability is a fundamental property in various fields, including physics, engineering, economics, and biology, and it plays a crucial role in understanding and predicting the behavior of complex systems. When it comes to the stability analyzing of non-linear systems, Lyapunov method comes in to the play.
Lyapunov stability is a concept in the field of control theory and dynamical systems that helps analyze and classify the behavior of a system over time. It is named after the Russian mathematician Aleksandr Lyapunov, who made significant contributions to the theory of stability. In simple terms, Lyapunov stability refers to the stability of an equilibrium point or a trajectory of a dynamical system. An equilibrium point is a state at which the system remains unchanged over time, while a trajectory represents the time evolution of the system's state variables.
Consider a differential equation $\dot{x} = f(x)$ has the equilibrium point of x^* and has a unique and continuous solution for all initial condition in $U_1(0)$. If the Lyapunov function of f(x) is V(x), which is continuous and continuously differentiable in $U_2(0) \subseteq U_1(0)$ and fulfills;
1. V(0)=0
2. $V(x) > 0$; $x \neq 0$
3. $\dot{V}(x) \leq 0$; $x \neq 0$
Then the system is stable in the sense of Lyapunov.
Moreover, if the $\dot{V}(x) \leq 0$ for all x, the system is asymptotically stable [1].

2.2 Controller development

2.2.1 Feedback Linearization

Feedback linearization is a control theory approach that converts a nonlinear system into a linear one by applying an appropriate feedback control rule. This transformation enables the use of well-established linear control methods to develop nonlinear controllers. The feedback linearization method consists of two major steps.
The first step is to identify an appropriate variable modification that changes the original system dynamics into a "locally controllable" form. This transformation is accomplished by picking a collection of new variables that cancel out the system dynamics' non linearities. As a result, the system's dynamics are linear in terms of the additional variables.
After the system has been translated into a linear form, basic linear control techniques may be used to create a feedback control law. The control rule is intended to stabilize the linearized system while achieving the required control goals. This control rule is usually written in terms

of the original system variables, and it may include feedback of system states, derivatives, or other measures.
The feedback control rule is based on the system's linearized model, yet it still acts on the original nonlinear system. By successfully canceling out the non linearities through feedback, the control rule is able to stabilize and govern the nonlinear system [2].
Consider non-linear system of the form of,

$$\dot{x} = a(x) + b(x).u$$

$$y = c(x)$$

Assume this system can be transform in to:

$$\begin{bmatrix} \dot{x_1} \\ . \\ . \\ \dot{x}_{n-1} \\ \dot{x_n} \end{bmatrix} = \begin{bmatrix} x_2 \\ . \\ . \\ x_n \\ \alpha(x) \end{bmatrix} + \begin{bmatrix} 0 \\ . \\ . \\ 0 \\ \beta(x) \end{bmatrix} .u$$

$$y = x_1$$

The control law can write as;

$$u = \frac{\alpha(x) + \sum_{i=1}^{n} a_{i-1}x_i}{\beta(x)} + \frac{V}{\beta(x)}\omega$$

$$u = \frac{\alpha(x) + \sum_{i=1}^{n} a_{i-1}x_i}{\beta(x)} + \frac{V}{\beta(x)}\omega$$

ω = reference input
a = free choosable parameters and,
V = yields
Now $\dot{x}$ can write as,

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & . & . & 0 \\ 0 & 0 & 1 & . & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & 1 \\ -a_0 & -a_1 & -a_2 & . & . & -a_{n-1} \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ . \\ . \\ 0 \\ V \end{bmatrix} \omega$$

However, to derive the state transformation, apply Lie derivatives to the output function c,

$$L_f c(x) = \frac{\partial c(x)}{\partial x} a(x)$$

Now by taking the derivation of output y,

$$\dot{y} = \frac{\partial c(x)}{\partial x} \dot{x}$$

$$\dot{y} = \frac{\partial c(x)}{\partial x} a(x) + \frac{\partial c(x)}{\partial x} b(x).u$$

Which is

$$\dot{y} = L_a c(x) + L_b c(x).u$$

In most cases ,$L_b c(x) = 0$. Therefore,

$$\dot{y} = L_a c(x)$$

By doing this for higher order derivatives;

$$y^{\delta-1} = L_a^{\delta} c(x) + L_a L_b^{\delta-1} c(x).u$$

where i=0,...,$\delta - 2$ δ = relative degree of the system
Consider the case where the relative degree equals to the system dimensions, $\delta = n$.

$$z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ . \\ . \\ z_n \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \\ \ddot{y} \\ . \\ . \\ y^{(n-1)} \end{bmatrix} = \begin{bmatrix} c(x) \\ L_a c(x) \\ L_a^2 c(x) \\ . \\ . \\ L_a^{n-1} c(x) \end{bmatrix} = t(x)$$

If t(x) is continuously differentiable,
$t^{-1}(t(x)) = x$

By differentiation the transformation equation,

$$\dot{z} = \dot{t}(x) = \frac{\partial t(x)}{\partial x}\dot{x} = \begin{bmatrix} L_a c(x) \\ L_a^2 c(x) \\ . \\ . \\ L_a^{n-1} c(x) \\ L_a^n + L_b L_a^{n-1} c(x).u \end{bmatrix}$$

Now the non-linear system is transformed into;

$$\begin{bmatrix} \dot{z_1} \\ . \\ . \\ . \\ \dot{z}_{n-1} \\ \dot{z_n} \end{bmatrix} = \begin{bmatrix} z_2 \\ . \\ . \\ . \\ z_n \\ L_a^n c(x) \end{bmatrix} + \begin{bmatrix} 0 \\ . \\ . \\ . \\ 0 \\ L_a L_b^{n-1} c(x) \end{bmatrix} .u$$

$$y = z_1$$

As we derived, the system can be controlled by the control law.

$$u(x, \omega) = -r(x) + v(x).\omega$$

where the controller,

$$r(x) = \frac{L_a^n c(x) + k^T z}{L_b L_a^{n-1} c(x)}$$

$$k^T = \begin{bmatrix} a_0 & a_1 & . & . & . & a_{n-1} \end{bmatrix}$$

the prefilter,

$$v(x) = \frac{V}{L_b L_a^{n-1} c(x)}$$

Since,

$$z = t(x) = \begin{bmatrix} c(x) \\ L_a c(x) \\ . \\ . \\ . \\ L_a^{n-1} c(x) \end{bmatrix}$$

We can obtain the control law as,

$$u = -\frac{L_a^n c(x) + a_{n-1} L_a^{n-1} c(x) + ... + a_1 L_a c(x) + a_0 c(x)}{L_b L_a^{n-1} c(x)} + \frac{V}{L_b L_a^{n-1} c(x)} \omega$$

2.2.2 Linear Model Predictive Control

Model Predictive Control (MPC) is a sophisticated control approach that makes predictions and optimizes control actions over a finite time horizon by using mathematical models of the system. It has grown in prominence across a wide range of sectors because of its capacity to manage complicated dynamic systems with restrictions and uncertainties.

MPC operates on a receding horizon, which means it solves an optimization problem repeatedly to discover the best control actions at each time step. The optimization problem takes the system model, current state measurements, and intended objectives or setpoints into consideration. MPC may anticipate system behavior and make proactive control choices by considering a prediction horizon, which is a defined future time span. One of MPC's primary characteristics is its capacity to deal with restrictions. MPC can include limitations on the system's inputs, states, and outputs, ensuring that control actions meet operational limits and safety criteria. MPC facilitates effective handling of restrictions such as equipment saturation, process boundaries, and safety limits by explicitly considering constraints. It may alter control actions dynamically to keep within the set limitations. Furthermore, MPC allows for the inclusion of various objectives or setpoints. This feature enables the simultaneous optimization of many variables while taking into consideration their respective relevance or trade-offs. It makes it easier to accomplish optimal control actions that satisfy performance objectives, such as tracking setpoints, optimizing energy usage, or decreasing process variability. MPC is used in a wide range of sectors, including process industries, automotive systems, robotics, energy management, and building control. It has been shown to be successful in regulating complex systems with nonlinearities, temporal delays, and variable interactions. MPC can increase system performance, stability, and adaptability to changing operational circumstances. MPC implementation, on the other hand, necessitates an appropriate mathematical model of the system, which may incorporate system identification techniques or knowledge of the underlying physics. The correctness of the model is critical to the performance of MPC. Furthermore, MPC may be computationally demanding, particularly for large-scale systems, and real-time implementation necessitates fast algorithms and technology[3].

3 Stability Analysis

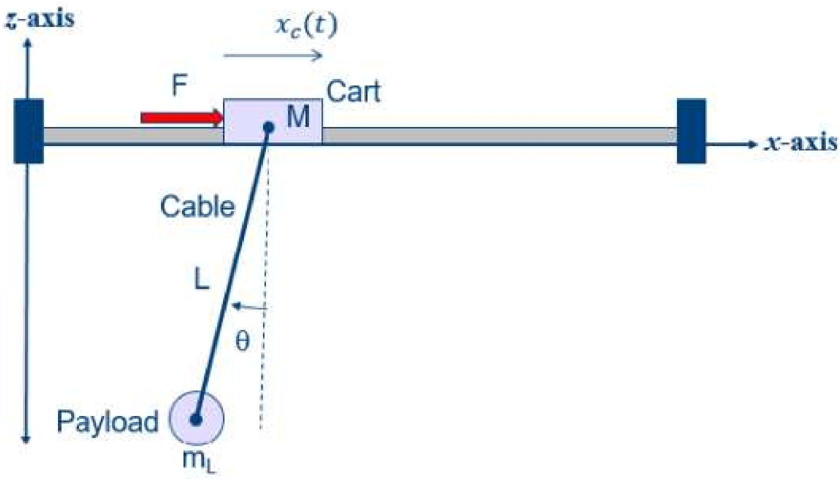

Figure 3.1: ANTI-SWAY SYSTEM

Differential equations for the motion of the system where M, mL, and F are all positive quantities;

$$(M + m_L)\ddot{X}_c - m_L\dot{\theta}^2 sin\theta + m_L\ddot{\theta}cos\theta = F \tag{3.1}$$

$$m_L L\ddot{X}_c cos\theta + m_L L^2\ddot{\theta} + m_L gLsin\theta = 0 \tag{3.2}$$

Now consider the f(x) as the state varible vector for the system;

$$f(x) = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} \tag{3.3}$$

From equation (3.1) and (3.2);

$$\ddot{x} = \frac{F + m_L L\dot{\theta}^2 sin\theta + m_L Lgcos\theta sin\theta}{(M + m_L) - m_L Lcos^2\theta} \tag{3.4}$$

$$\ddot{\theta} = \frac{Fcos\theta + m_L L\theta^2 sin\theta cos\theta + (M + m_L)gsin\theta}{m_L Lcos^2\theta - (M + m_L)} \tag{3.5}$$

Now f(x) can write as;

$$f(x) = \begin{bmatrix} x_2 \\ \frac{F+m_L Lx_4^2 sinx_3+m_L Lgsinx_3cosx_3}{(M+m_L)-m_L Lcos^2x_3} \\ x_4 \\ \frac{Fcosx_3+m_L Lx_4^2 sinx_3cosx_3+(M+m_L)gsinx_3}{m_L Lcos^2x_3-(M+m_L)} \end{bmatrix} \tag{3.6}$$

Consider x^* is the equilibrium point of the system,

$$x^* = \begin{pmatrix} 0, & 0, & 0, & 0 \end{pmatrix} \tag{3.7}$$

and,

$$\Omega = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} : \quad -\pi < x_3 < \pi, \quad |x_1|, |x_2|, |x_4| < k \right\} \tag{3.8}$$

Where Ω is the Lyapunoc region.

For this system, Lyapunov function can write as the total energy of the crane system. Therefore; V(x)=K.E. + P.E. Where, K.E. = Kinetic energy P.E. = Potential energy Therefore;

$$V(x) = \frac{1}{2}Mx_1^2 + \frac{1}{2}m_L[x_1^2 + 2x_1x_4Lcosx_3 + L^2x_4^2] + m_LgL(1 - cosx_3) \tag{3.9}$$

Since,
$\dot{V}(x) = \bigtriangledown V(x) * f(x)$
$\dot{V}(x)$ can write as;

$$\dot{V}(x) = \frac{\partial \dot{V}(x)}{\partial x_1} * \dot{x_1} + \frac{\partial \dot{V}(x)}{\partial x_2} * \dot{x_2} + \frac{\partial \dot{V}(x)}{\partial x_3} * \dot{x_3} + \frac{\partial \dot{V}(x)}{\partial x_4} * \dot{x_4} \tag{3.10}$$

Where;

$$f(x) = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{F + m_LLx_4^2sinx_3 + m_LLgsinx_3cosx_3}{(M+m_L) - m_LLcos^2x_3} \\ x_4 \\ \frac{Fcosx_3 + m_LLx_4^2sinx_3cosx_3 + (M+m_L)gsinx_3}{m_LLcos^2x_3 - (M+m_L)} \end{bmatrix} \tag{3.11}$$

and,

$$\bigtriangledown V(x) = \begin{bmatrix} (M + m_L)x_1 + m_LLx_4cosx_3 \\ 0 \\ -m_LLx_1x_4sinx_3 + m_LgLsinx_3 \\ m_Lx_1cosx_3 + m_LL^2x_4 \end{bmatrix} \tag{3.12}$$

Since,
1. V(x*)=0,
2. $V(x) > 0$; for all x in Ω except x^* and,
3. $\dot{V}(x) \leq 0$; for all x in Ω except x^*
The above non-linear system is stable in the sense of Lyapunove[4].

4 Controller Design

As its visible in the previous section, the system is Lyapunov stable. This means the system is stable inside a region. The figure – shows the initial system designed in Simulink environment. An when its simulated under the given conditions, its visible that there is a sway angle, but its not changing during the course of time. This explains the Lyapunov stability mentioned in the previous section. The figure (4.1) explains the result.

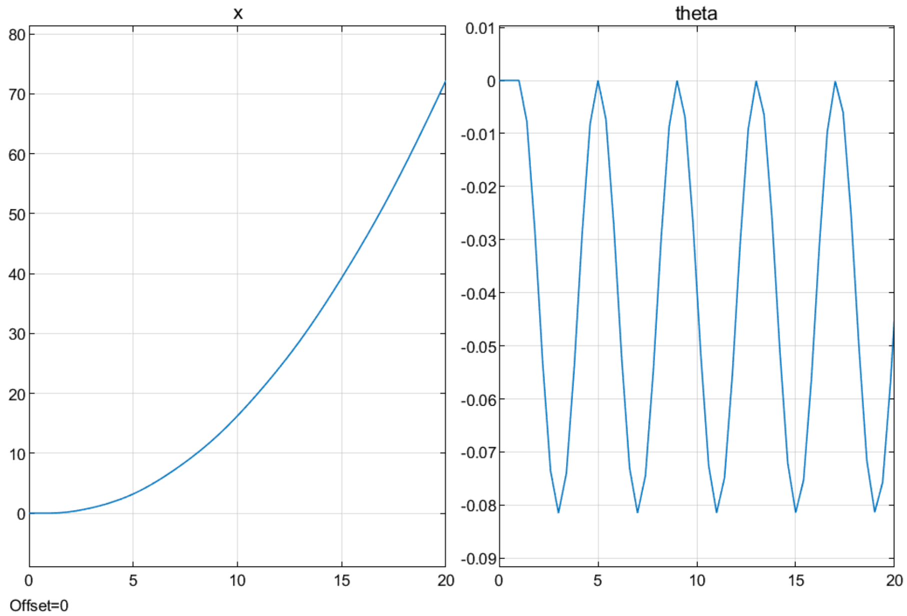

Figure 4.1: BEHAVIOUR OF x AND theta IN UNCONTROLLED CONDITIONS

Hence, it is required to develop a controller to control the sway angle as it was discussed in the first report, sway angle is a severe disadvantage to the performance of the system. The assumptions made when checking the stability of the system directly affect the performance of the controller. The objective of the controller is to reduce the sway angle to be minimum and, in this case, reducing it to zero because assumptions were made that there is no friction or drag involved in the system.

The first approach used in designing the controller was feedback linearization method. This method focuses on linearizing the output of the system and transforming it to a linear system and design the controller according to that. The first stem is the develop the state equations for the system. This is done by using equations (1.1) and (1.2).

$$x = x_1$$
$$\dot{x} = x_2$$
$$\Theta = x_3$$
$$\dot{\Theta} = x_4$$

Using state variables given above, the system can be transformed in to state space .

$$\dot{x}_1 = x_2 \tag{4.1}$$

$$\dot{x}_2 = \frac{1}{k}(mgsinx_3cosx_3 + m_L Lx_4 + F) \tag{4.2}$$

$$\dot{x}_3 = x_4 \tag{4.3}$$

$$\dot{x}_4 = \frac{1}{kL}((M + m_L)gsinx_3 + mLx_4 sinx_3 cosx_3 + cosx_3 F) \tag{4.4}$$

And term k is,

$$k = (M + m_L) - m_L cos^2 x_3$$

Using the obtained equations (4.1), (4.2), (4.3) and (4.4), the f(x) and g(x) matrices were generated.

$$f(x) = \begin{bmatrix} x_2 \\ \frac{1}{k}(mgsinx_3 cosx_3 + m_L Lx_4) \\ x_4 \\ \frac{1}{kL}((M + m_L)gsinx_3 + mLx_4 sinx_3 cosx_3) \end{bmatrix} \tag{4.5}$$

$$g(x) = \begin{bmatrix} 0 \\ \frac{1}{k} \\ 0 \\ \frac{cosx_3}{kL} \end{bmatrix} \tag{4.6}$$

The output of the system is considered as x (cart position). In state variable notation, this is x_1. Using the, the h(x) matrix was generated.

$$h(x) = \begin{bmatrix} x_1 & 0 & 0 & 0 \end{bmatrix} \tag{4.7}$$

The next step of feedback linearization method is to find the relevant Lie derivatives using the output function h(x). The Lie derivative is a notation for a combination of partial derivative and a multiplication. It can be shown in equation (4.8).

$$L_a c(x) = \frac{\partial c(x)}{\partial x} a(x) \tag{4.8}$$

This process starts with the feedback. Following are the calculations.

$$L_f{}^0 h(x) = h(x) = x_1 \tag{4.9}$$

The we check weather,

$$L_g L_f{}^0 h(x) \tag{4.10}$$

if this value is equal to zero, further lie derivatives exist. Hence, it is required to calculate this in each step of calculating Lie derivatives.
Using equation (4.10) the value for (4.9) was calculated.

$$L_g L_f{}^0 h(x) = \frac{\partial(\frac{\partial h(x)}{\partial x})}{\partial x} g(x) = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} g(x) = 0 \tag{4.11}$$

This means there are further Lie derivatives exist.

$$L_f{}^1 h(x) = x_2 \tag{4.12}$$

$$L_g L_f{}^1 h(x) = \frac{1}{k} \tag{4.13}$$

From equation (4.13) it is visible that it is no longer equal to zero. This shows that there are no further Lie derivatives.
The last Lie derivative is 1 and it is equal to the relative degree of the system. In feedback linearization, there are two condition.
1. Relative degree = System dimension
2. Relative degree ¡ System dimension
In this case the the relative degree which is 1 is less than the system dimension which is 4. This means, the system has internal dynamics. To find internal dynamics, we choose two arbitrary functions which will satisfy the equation (4.14)

$$L_g t(x) = 0 \tag{4.14}$$

Where t(x) is the arbitrary function. In this case it is required to find two arbitrary functions as there are only two functions available within the system. Those are,
$z_1 = L_f{}^0 h(x) = x_1$
$z_2 = L_f{}^1 h(x) = x_2$
But selecting two arbitrary functions is a complex task for this system as the system it self is complex. Hence, another method was used in controller designing.

4.1 Linear Model Predict Controller

This controller type was selected because it required to linearize the inputs and outputs of the system at a known point. Positively, the cran all the output variable x and theta can be linearized at the initial position as initially both cart position and the sway angle are zero. By using leanirizing techniques, it was possible to linearfize the inputs and the two outputs of the system Following that, MATLAB provided a MPC toolbox which make the process easy to design the controller[5, 3]. Figure (4.2) shows the process of designing the controller.

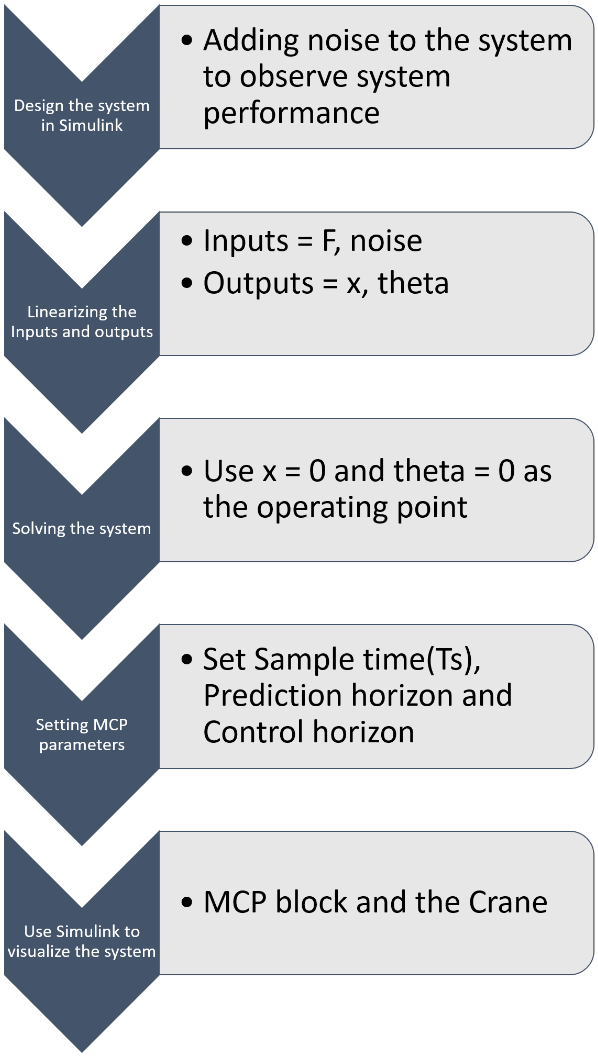

Figure 4.2: CONTROLLER DESIGN PROCESS

The figure (4.3) shows the block diagram of the controller.

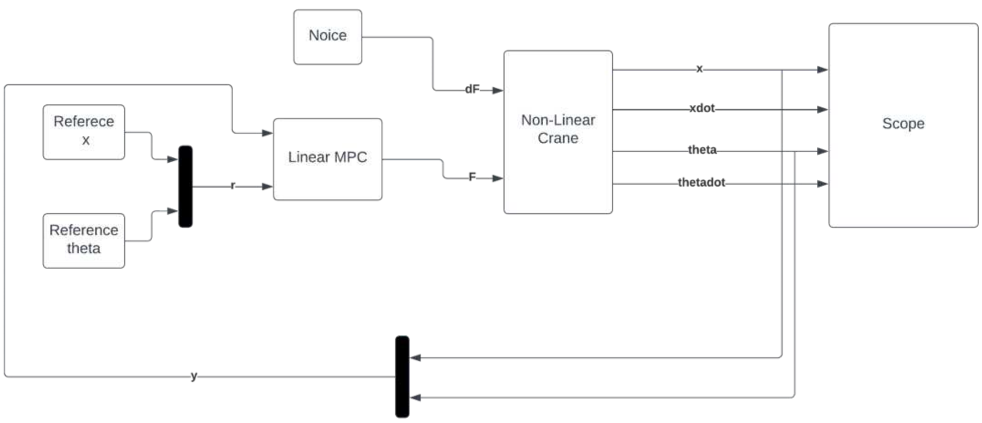

Figure 4.3: CONTROLLER BLOCK DIAGRAM

5 Results

The desinged conotrolle was abled to successfully controlle the sway angel of the system. Follwing are the final parameters of the system

- Sample time (Ts) = 0.3
- Prediction horizon = 100
- Control horizon = 10
- Objective weight for x = 0.5
- Objective weight for theta = 0.5

Figure (5.1) shows the final result of the syste where sway angle goes to zero.

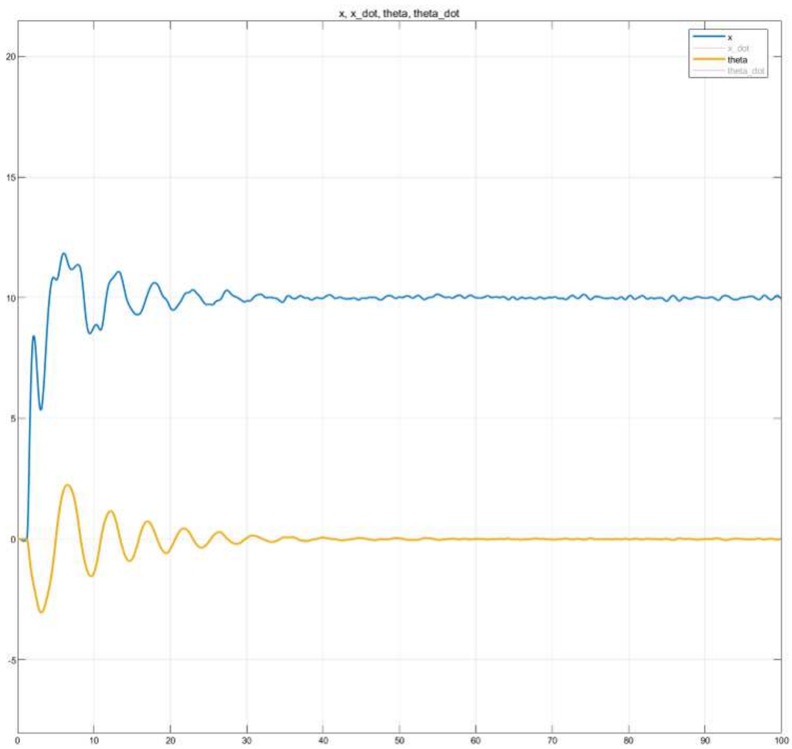

Figure 5.1: RESULTS

If all other parameters like velocity of the cart and angular velocity of the load are concidered, the figure (5.2) is obtained.

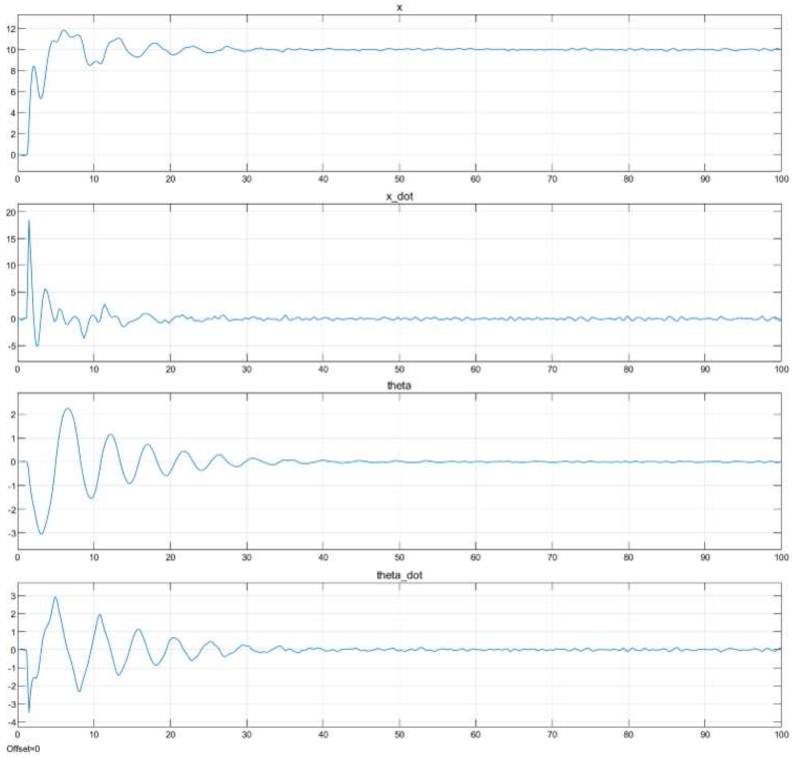

Figure 5.2: OVERALL RESULTS

The reason for the noise in the output is because as mentioned before in figure (4.3), a external noise was added to the system as a white noise to observe the controller performance under disturbances. The system can handle between 0.1 and 1.1 level of noise from the white noise generator and keep the sway angle extremely close to zero.

6 Conclusion

6.1 Controller for non-linear system

The purpose of this controller is to control the sway angle. Designing a linear MPC was possible because both the outputs concidred for the MPC were know at there initial conditions. Which are the cart position x and load sway angle theta. Both of these are at zero in the beginning. So, the system can be linearized at this point for the controller.
When designing the controller, 3 parameters were considered.
1. Sample time
2. Predict horizon
3. Control horizon
The tuning was done in at three instances where changing one parameter while keeping other two fixes. The referecne values for parameters were taken as the values for the stable system.
1. Sample time = 0.3
2. Predict horizon = 100
3. Control horizon = 10

1. Change predict horizon
Figure (6.1) shows the response to the change in predict horizon.

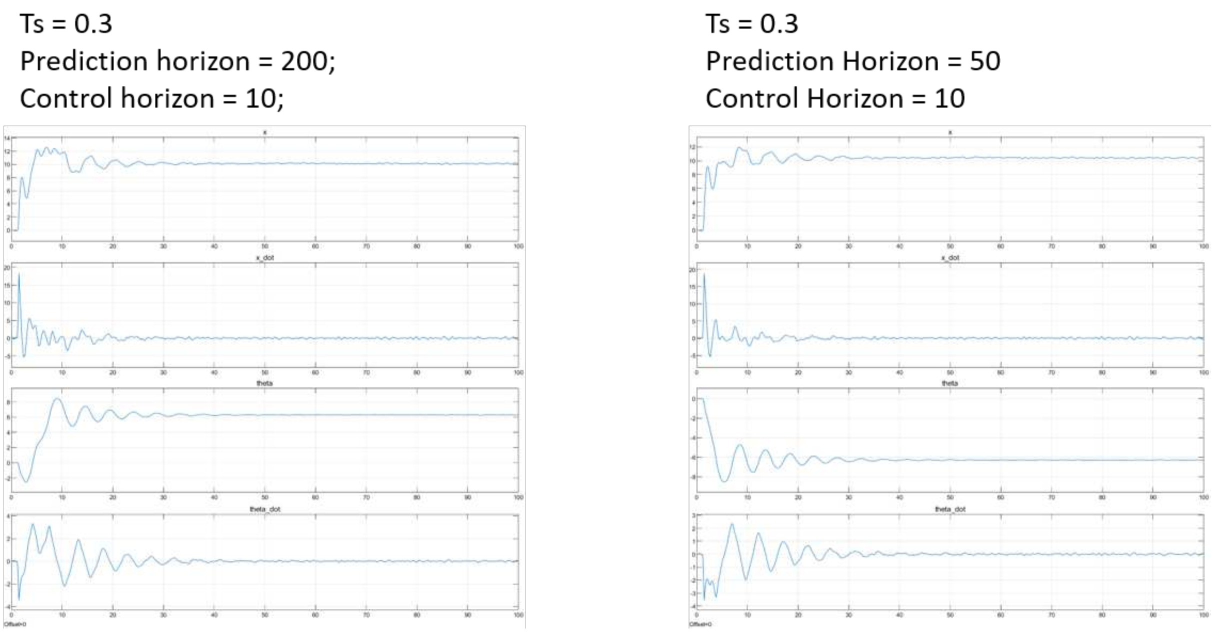

Figure 6.1: PREDICT HORIZON CHANGE

Changing the predict horizon resulted in changing the stable angle. The sway angle does not go to zero but was stable at an angle.

2. Change control horizon

Figure (6.2) shows the response to the change in control horizon.

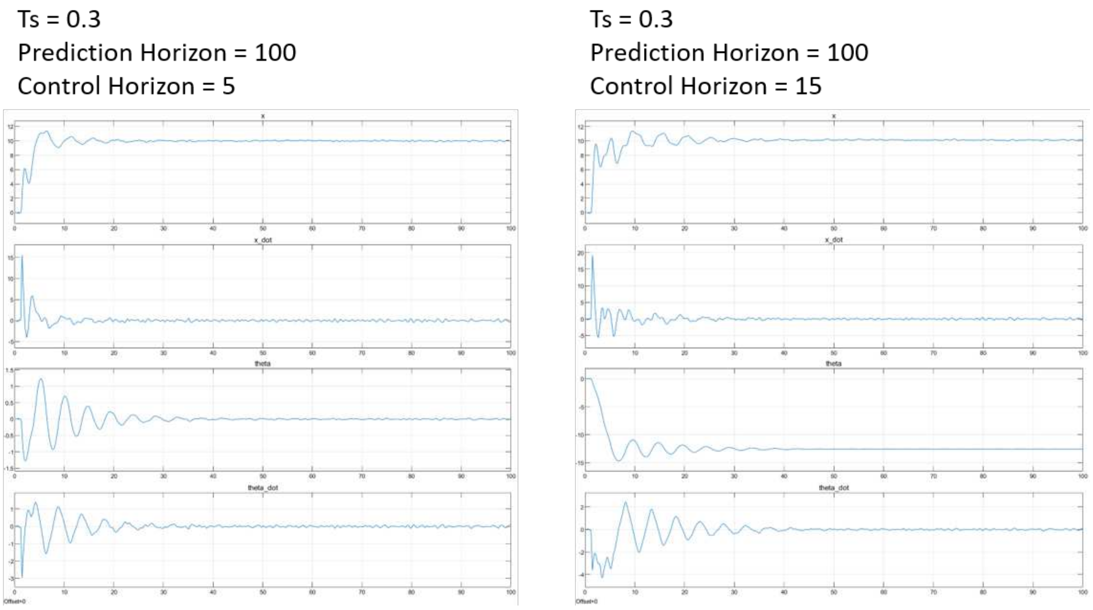

Figure 6.2: CONTROL HORIZON CHANGE

Changing the control horizon resulted in changing the stable angle only when increasing it. The sway angle does not go to zero but was stable at an angle.

3. Change sample time

Figure (6.3) shows the response to the change in sample time.

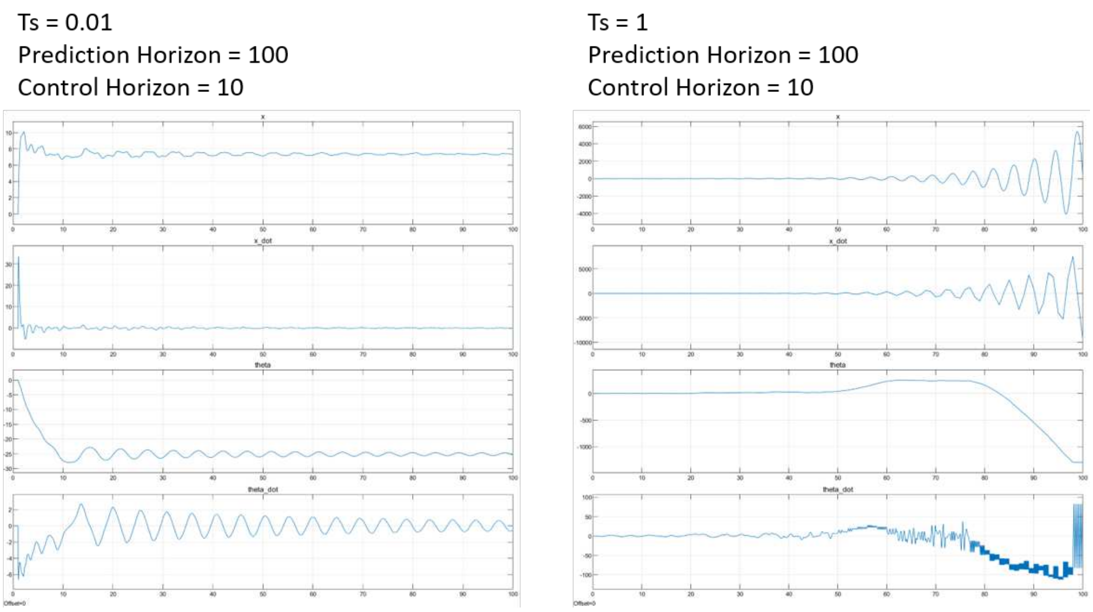

Figure 6.3: SAMPLE TIME CHANGE

Changing the sample time resulted in destabilizing the entire system while increasing it.

Bibliography

[1] Richard M. Murray, Zexiang Li, and Shankar Sastry. *A mathematical introduction to robotic manipulation*. Boca Raton: CRC Press, 1994. ISBN: 9781315136370.

[2] Weon Kim. "Feedback Linearization of Nonlinear Systems: Robustness and Adaptive Control". In: *LSU Historical Dissertations and Theses* (1991).

[3] *What is Model Predictive Control? - MATLAB & Simulink - MathWorks Deutschland*. 6/20/2023. URL: %5Curl%7Bhttps://de.mathworks.com/help/mpc/gs/what-is-mpc.html%7D.

[4] MIT OpenCourseWare. *Resources — Dynamics of Nonlinear Systems — Electrical Engineering and Computer Science — MIT OpenCourseWare*. 6/13/2023. URL: %5Curl%7Bhttps://ocw.mit.edu/courses/6-243j-dynamics-of-nonlinear-systems-fall-2003/download/%7D.

[5] *Model predictive controller - MATLAB - MathWorks Deutschland*. 6/20/2023. URL: %5Curl%7Bhttps://de.mathworks.com/help/mpc/ref/mpc.html%7D.