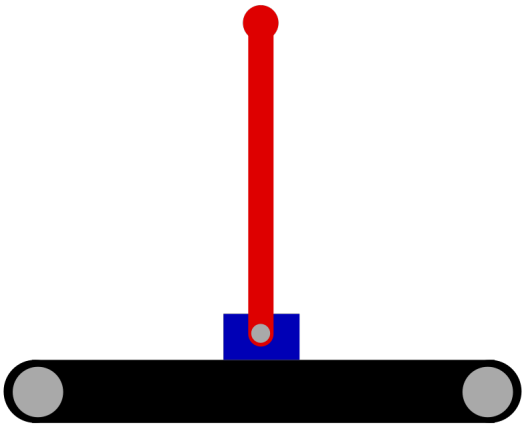
The Inverted Pendulum

Semesterproject in Control and

Simulation of autonomous systems



Project group 5

Alex Ellegaard - aelle20 Anders Lind-Thomsen - andli20

Simon Christensen - simch20 Peter Frydensberg - pefry20

Thomas Therkelsen - ththe20 Victoria Jørgensen - vijoe20

Supervisor

Christian Schlette

BEng in Robot Systems

TEK MMMI

University of Southern Denmark

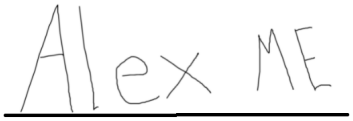
May 21st 2022

Abstract

Preface

Special thanks should be given to Associate Professors Aljaz Kramberger & Christoffer Sloth, for exemplary guidance and councelling regarding the control systems design in this project.

Alex Ellegaard



Simon Bork

Anders Lind-Thomsen

Figures/anders.underskrift.png

Thomas Therkelsen

Victoria Jørgensen

Peter Frydensberg

Contents

1 Introduction

The fourth industrial revolution, as some refer to it, is constantly evolving, as can be seen in the automation of many industrial and everyday processes. One of the many advantages of automating a process is that the mistakes that humans make during the manual process are eliminated. In light of these things, a wide range of robots are constructed for all kinds of purposes.

In this project, the robot is going to be based upon a PLC which will be controlling an inverted-pendulum-cart-system, which is inherently unstable. This is marginally resemblant to the classic kids' game of trying to keep a pencil upright on the palm, by adjusting for the movement of the pencil by moving your hand around. This is a case in which you can notice the robot eliminating human error. This is due to humans being unable to accurately measure and react to the movement of the pencil. - Whereas the robot can quantify the error precisely and adjust for it in a short amount of time.

Stabilizing an inverted pendulum on a cart, an inherently unstable system is a textbook control problem. This is one that many will face in their journey through the world of control systems. This version of the problem is taken one step further, as it's not just an inverted pendulum on a cart; It's an inverted pendulum on a cart, strapped to a conveyor belt, moved by a DC motor, and controlled by a PLC.

1.1 Problem

From the given project description, a problem is formulated from which a fully fledged control system can be developed.

- How is the dynamical model of the system as well as a control system mathematically modelled as well as simulated such that it can be implemented on the physical system?
- How is the PLC used to implement the control system model on the physical system, to be able to control it?
- How is the physical system controlled using the implemented control system, and how will it need to be tweaked to optimize performance?

1.2 Specifications

1.3 Report structure

2 Related Work

This chapter surveys and compares previous work in the field of controlling the classic control systems problem of an inverted pendulum on a cart.

2.1 Modelling of the system

The problem is commonly approached using one of two methods of modelling the system being either **Newton–Euler**-based modelling or **Euler–Lagrange**-based modelling. Given that the system is inherently unstable and it consisting of non-linear elements, a linearized model of the system is desirable. This is usually done using a first order **Taylor approximation**, which is either applied on the equations of motion of the system, or the State-Space model of the system. As well as a linearization of the system model, some approximations such as $\sin(\theta) \approx \theta$, $\cos(\theta) \approx \theta$ and $\dot{\theta}^2 \approx 0$ all for sufficiently small θ.

2.2 Controlling the system

2.2.1 Classical Control

There are a few common approaches to designing a control system for the inverted pendulum. The most simple way of controlling an inverted-pendulum-cart-system is by making just a single PID controller that corrects the pendulum angle.

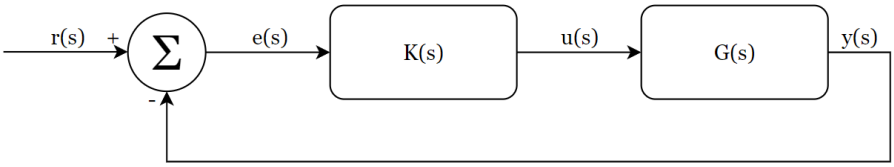


Figure 1: Example of PID Control structure

Another way to do it is to design a parallel control structure using two PID controllers with one correcting the pendulum angle and the other correcting the cart position.

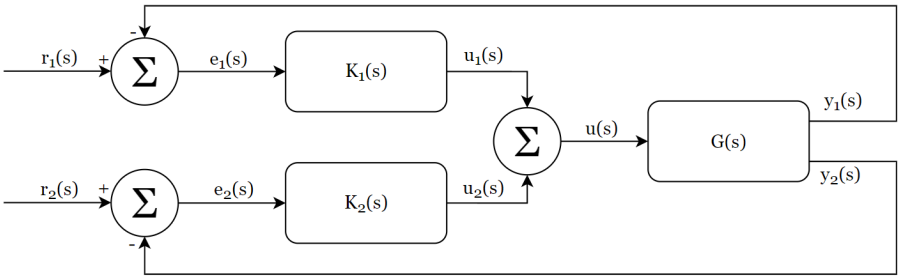


Figure 2: Example of Parallel PID Control structure

2.2.2 Modern Control

A modern control system can be designed to control the pendulum angle and cart position.

2.3 Our contribution

Our contribution to this problem will be based upon Euler–Lagrange modelling, based upon which a few approaches will be made to determine the one with the best performance.

3 Mathematical Modelling

3.1 Euler-Lagrange Modelling

In order to model the system, Euler-lagrange modelling is utilized. The equations of motions are derived from the following formulars:

$$\mathscr{L} = E_{kin} - E_{pot} \tag{1}$$

Where $\mathscr{L}$ is the lagrangian.

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathscr{L}}{\partial \dot{q}} - \frac{\partial \mathscr{L}}{\partial q} = Q \tag{2}$$

Where q is a vector of the generalized coordinates $q = [x, \theta]^T$, and Q is a vector of generalized forces, in this case friction and input force $Q = [-b_c \cdot \dot{x} + u, -b_p \cdot \dot{\theta}]^T$.

The kinetic energy of the system is given by:

$$E_{kin} = \frac{1}{2} \cdot (M + m) \cdot \dot{x}^2 - m \cdot \dot{x} \cdot l \cdot \dot{\theta} \cdot \cos(\theta) + \frac{1}{2} \cdot (m \cdot l^2 + I) \cdot \dot{\theta}^2 \tag{3}$$

And the potential energy of the system is given by:

$$E_{pot} = m \cdot g \cdot l \cdot \cos\theta \tag{4}$$

Where x decsribes the position of the cart along the x-axis $[m]$, $\dot{x}$ is the velocity $[m/s]$ and $\ddot{x}$ is the acceleration $[m/s^2]$. θ decsribes the angle of the pendulum $[rad]$, $\dot{\theta}$ is the angular-velocity $[rad/s]$ and $\ddot{\theta}$ is the angular-acceleration $[rad/s^2]$. M describes the mass of the cart $[kg]$, m describes the mass of the pendulum $[kg]$, l is the length from the cart to the center of mass of the pendulum $[m]$. By using equation 2, the motions of equation written as first order equations become:

$$(M + m) \cdot \ddot{x} - m \cdot l \cdot \cos(\theta) \cdot \ddot{\theta} + m \cdot l \cdot \sin(\theta) \cdot \dot{\theta}^2 = u - b_c \cdot \dot{x} \tag{5}$$

$$(m \cdot l^2 + I) \cdot \ddot{\theta} - m \cdot l \cdot \cos(\theta) \cdot \ddot{x} - m \cdot g \cdot l \cdot \sin(\theta) = -b_p \cdot \dot{\theta} \tag{6}$$

These equations are non-linear because of the trigonometric functions and the '$\dot{\theta}^2$'. They therefore have to be linearized, before linear methods can be used on the system. Here two methods of linearization are presented: The first method uses the jacobian matrix with linearization around an equilibrium point. The equilibrium point is a system state, around which, we want our system to move. In this case, this point corresponds to the upright pendulum-position, and an arbitrary cart-position. This method also brings the system directly on state space form which is used for modern control-strategies.

The second method uses knowledge about the numerical sizes of the system variables as well as trigono-

metric properties. This is a more simple linearization method.

Method 1: In order to lienarize with the jaboian matrix, an equilibrium point needs to established. This point is given by: $(\bar{u}, \bar{x}, \dot{\bar{x}}, \bar{\theta}, \dot{\bar{\theta}}) = (0, 0, 0, 0, 0)$, since the pendulum should be in upright position ($\bar{\theta}$) with ideally no angular velocity ($\dot{\bar{\theta}}$). In this position, the cart should be standing still ($\dot{\bar{x}}$), $\bar{u}$) in an arbitrary position, here set to 0 for simplicty ($\bar{x} = 0$). In order to use the jacobian, the 2 non-linear second order equations of motion need to be reformulated into 4 first order equations. Matlab is here used to isolate the different state-variables:

$$v = \dot{x} \tag{7}$$

$$\dot{v} = -\frac{I \cdot b_c \cdot v - u \cdot I - u \cdot l \cdot m + \omega^2 \cdot l^2 \cdot m^2 \cdot \sin(\theta) + b_c \cdot v \cdot l \cdot m - g \cdot l^2 \cdot m^2 \cdot \cos(\theta) \cdot \sin(\theta) + I \cdot \omega^2 \cdot l \cdot m \cdot \sin(}{-l^2 \cdot m^2 \cdot \cos(\theta)^2 + l \cdot m^2 + M \cdot l \cdot m + I \cdot m + I \cdot M} \tag{8}$$

$$\omega = \dot{\theta} \tag{9}$$

$$\dot{\omega} = -\frac{b_p \cdot \omega \cdot m + b_p \cdot \omega \cdot m - u \cdot l \cdot m \cdot \cos(\theta) - g \cdot l \cdot m^2 \cdot \sin(\theta) + \omega^2 \cdot l^2 \cdot m^2 \cdot \cos(\theta) \cdot \sin(\theta) + b_c \cdot v \cdot l \cdot m \cdot \cos(\theta}{-l^2 \cdot m^2 \cdot \cos(\theta)^2 + l \cdot m^2 + M \cdot l \cdot m + I \cdot m + I \cdot M} \tag{10}$$

These equations are now used in the jacobian:

$$A_{linear} \fallingdotseq [\tag{11}$$

$$B_{linear} = \tag{12}$$

Method 2: This method utilizes knowledge about the system as well as trigonometric properties. we assume $\dot{\theta}^2 \approxeq 0$, since the angular velcoity is supposed to be much smaller than 1, and therefore, it becomes irrelevant when it is raised to the power of 2. Another assumption is that $\cos(\theta) \approxeq 1$. This is feacible since the pendulum is supposed to be around the angle $\theta = 0$, and $\cos(0) = 1$. The last assumption is that $\sin(\theta) \approxeq \theta$. This is true for small angles. By combining equations X AND X, and the assumptions, the system can be linearized, and the resulting equations become:

$$(M + m) \cdot \ddot{x} + m \cdot l \cdot \ddot{\theta} = u - b_c \cdot \dot{x} \tag{13}$$

$$(m \cdot l^2 + I) \cdot \ddot{\theta} + m \cdot l \cdot \ddot{x} + m \cdot g \cdot l \cdot \theta = 0 \tag{14}$$

These equations are the laplace-transformed in order to obtain transfer functions for the system. Two transfer-functions will be obtained, one that describes $\theta(s)$ with an input $u(s)$, and one that describes $x(s)$ with the same input $u(s)$. These transfer functions will be used with classical control. The laplace-transformed equations become:

$$(M + m) \cdot X(s) \cdot s^2 + m \cdot l \cdot \Theta(s) \cdot s^2 = U(s) - b_c \cdot X(s) \cdot s \tag{15}$$

$$(m \cdot l^2 + I) \cdot \Theta(s) \cdot s^2 + m \cdot l \cdot X(s) \cdot s^2 + m \cdot g \cdot l \cdot \Theta(s) = 0 \tag{16}$$

From here on, the two transfer functions are derived. The derivation IS IN THE APPENDIX. The transfer functions become:

$$G_p = \frac{\Theta(s)}{U(s)} = \frac{s^2}{-0.2527 \cdot s^3 - 6.826 \cdot s - 49.1} \tag{17}$$

$$G_c = \frac{X(s)}{U(s)} = \frac{-0.4667 \cdot s^2 - 9.82}{-0.2527 \cdot s^4 - 2.333 \cdot s^3 + 6.54 \cdot s^2 - 49.1 \cdot s} \tag{18}$$

These are used in classical control. In order to use modern control, the state space matrices for the system have to be determined. The general state space structure is as follows:

$$\dot{x} = A \cdot x + B \cdot u \tag{19}$$

$$y = C \cdot x + D \cdot u \tag{20}$$

Here, x is the state space vector, B is the input matrix, y describes the output of the measurements, C is the output matrix and D is the BLABKASLKDLKASBD In our case, $x = [x_c, \dot{x_c}, \theta, \dot{\theta}]^T$. The notation for the x-position of the cart is here changed from x to x_c to avoid confusion with the state space vector x.

The linearized equations of motion are used in the derivation of the state space matrices. A coefficient 'C' is defined to simplify the expression:

$$C = M + m - \frac{(m \cdot l)^2}{m \cdot l^2 + I} \tag{21}$$

$$\begin{bmatrix} \dot{x}_c \\ \ddot{x}_c \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{b}{C} & \frac{(m \cdot l)^2 \cdot g}{C \cdot (m \cdot l^2 + I)} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{b \cdot (m+M)}{C \cdot m \cdot l} - \frac{-b}{m \cdot l} & \frac{-(m+M) \cdot (m \cdot l)^2 \cdot g}{C \cdot (m \cdot l^2 + I) \cdot m \cdot l} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_c \\ \dot{x}_c \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{C} \\ 0 \\ \frac{1}{m \cdot l} - \frac{M+m}{C \cdot m \cdot l} \end{bmatrix} \cdot u \tag{22}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_c \\ \dot{x}_c \\ \theta \\ \dot{\theta} \end{bmatrix} \tag{23}$$

TEST OF THE MODELS

The system is now ready to be controlled with both modern- and classical control-strategies.

4 Control System Design

4.1 PID Control

To control the system we have chosen to implement a PID-controller with feedback on the system. The desired goals when implementing a PID-controller is typically fast rise time, low overshoot and close to no steady state error.

To design a PID controller, one must first consider the transfer function. When including low-pass filter for differential gain, it looks as follows on ideal form.

$$K(s) = K_p \cdot \left(1 + \frac{1}{s \cdot T_i} + \frac{s \cdot T_d}{1 + \frac{s \cdot T_d}{N}}\right) \tag{24}$$

Where K_p is the proportional gain, K_i is the integral gain, and K_d is the differential gain. The transfer function for the open-loop response, given by the angle of the pendulum is shown in Equation 17 and is defined as G_p. The loop gain and closed loop transfer function for the closed loop system are derived (full equations can be seen in appendix **??** due to size.

$$L_p(s) = G_p(s) \cdot K_p(s) \tag{25}$$

$$H_p(s) = \frac{L_p(s)}{1 + L_p(s)} \tag{26}$$

To illustrate where the closed-loop poles are placed in the s-plane when the gain k_p is changed, a root locus plot of ones closed-loop transfer function can be computed and analyzed. The root locus plot for the pendulum is seen below.

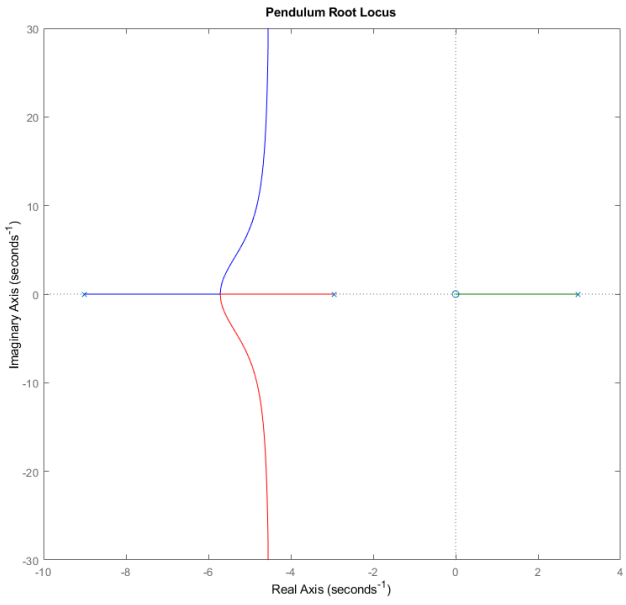


Figure 3: Pendulum Root Locus plot

4.2 Pendulum angle

By analyzing the root locus plot and doing some parameter finetuning afterwards, the following controller parameters are picked.

$$k_p = 617$$
$$k_i = 3.36 \cdot 10^3$$
$$k_d = 28.2$$

Resulting in the following responses for the pendulum.

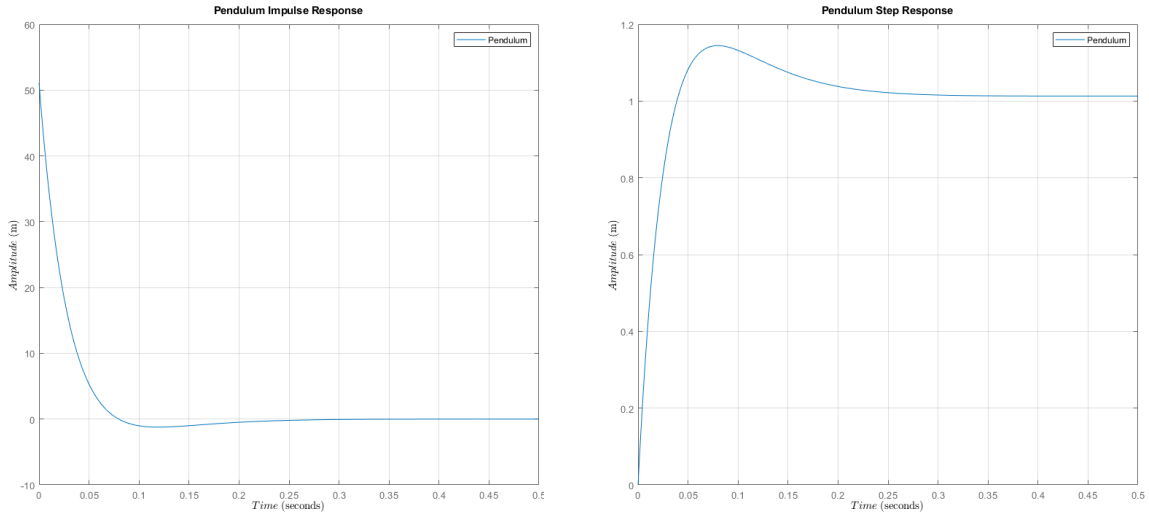


Figure 4: Pendulum system responses

The impulse responses show that the pendulum system recovers from the δ impulse within 300 $[ms]$. The step response shows that the system converges on its' reference with a slight steady state error of amplitude$1.5 \cdot 10^{-2}$. Thus it can be concluded that this performance is acceptable.

There is just the problem of the cart; On the real life cart system, there are hardware limits of ± 0.86 $[m]$. Meaning, a new approach must be taken to solve this problem.

4.3 Parallel Control

While a single PID controller could do the job in a context with no bounds on how far the cart can move, which is not the case in this project given the hardware limits on the motor-conveyor-cart system. Based on the previously defined PID control system, which only adjusts the pendulum angle, a parallel pid control system can be designed. A parallel structure is chosen because it is desired to not only control the pendulum angle, but also the cart position at the same time. - And because of its ability to automatically adjust for control signals that differ.

When dealing with parallel structure, one method to design the system is to just design each controller independently of eachother. In our case, the pendulum controller is already designed, so the cart controller is missing. This is utilizable due to the previously mentioned property of the parallel structure which is that you simply add the control signals and let it handle the differences internally.

The transfer function for the cart has been defined in Equation 17. Then the loop gain is derived for the cart

$$L_c = G_c \tag{27}$$

The root locus plot looks as follows

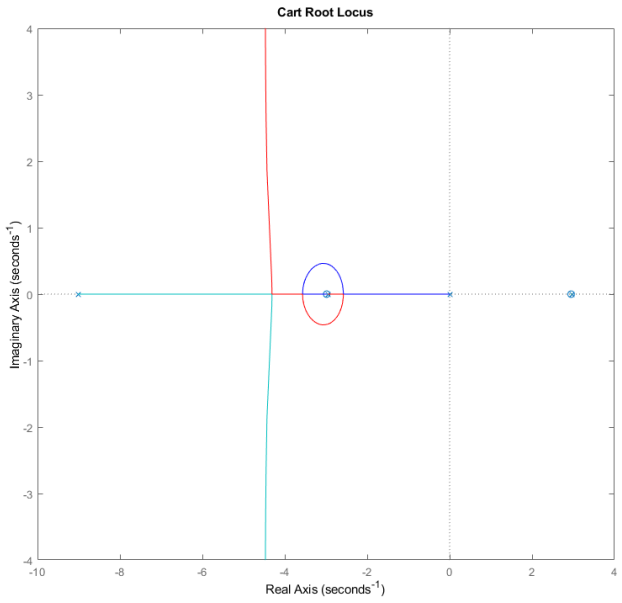


Figure 5: Cart Root Locus plot

The root locus is analyzed, and the controller gains are picked as

$$k_p = 120$$
$$k_i = \frac{-1}{120}$$
$$k_d = \frac{-30}{120}$$

Which gives the following impulse response, showing that the cart is relatively stable.

Figure 6: Cart Impulse Response

5 Simulink/Simscape Control Simulation

5.1 Rigidbody Model

To get an idea of how the system behaves under velocity control, it is modelled and controlled in Simulink, using a combination of Simscape and Simulink components. A good deal of inspiration is taken from the CTMS website[1], which is then changed quite a bit as their system is different than ours. In the model, the motor and belt are both omitted for simplification's sake.

5.1.1 Parameters

The parameters of the rigidbody system is given by

Cart

$$\begin{aligned} \text{length} &= 0.203\ [\text{m}] \\ \text{width} &= 0.0038\ [\text{m}] \\ \text{height} &= 0.0015\ [\text{m}] \\ \text{mass} &= 0.5\ [\text{kg}] \\ \text{dampening} &= 5\ \left[\text{N}\,(\text{m}\cdot\text{s})^{-1}\right] \end{aligned}$$

Pendulum

$$\begin{aligned} \text{length} &= 0.350\ [\text{m}] \\ \text{thickness} &= 0.006\ [\text{m}] \\ \text{mass} &= 0.0084\ [\text{kg}] \\ \text{dampening} &= 0.0012\ \left[\text{N}\,(\text{m}\cdot\text{s})^{-1}\right] \end{aligned}$$

Given the parameters, two rigidbodies are modelled using Simscape frames, rigid transforms, joints, solids, etc. A visualisation of this model is shown below.

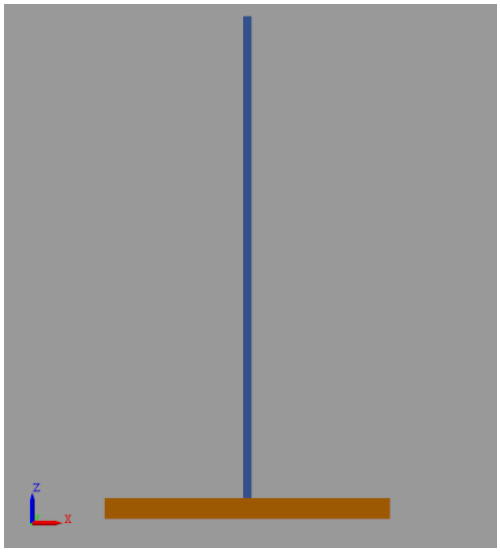


Figure 7: Rigidbody Model

5.2 System Model

Converters are added to the system model, to allow for reading of parameters and controlling the system using Simulink components later on.

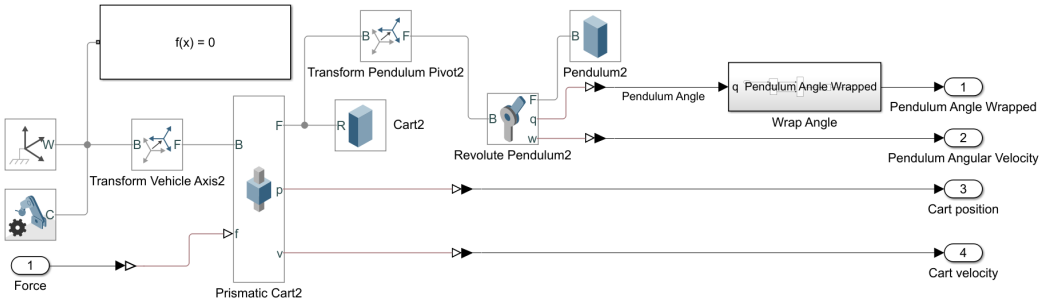


Figure 8: Pendulum-Cart Simscape Subsystem

As seen in figure 9, the pendulum eventually converges to π [rad] which affects the cart with some force, making it also converge at a given translational position.

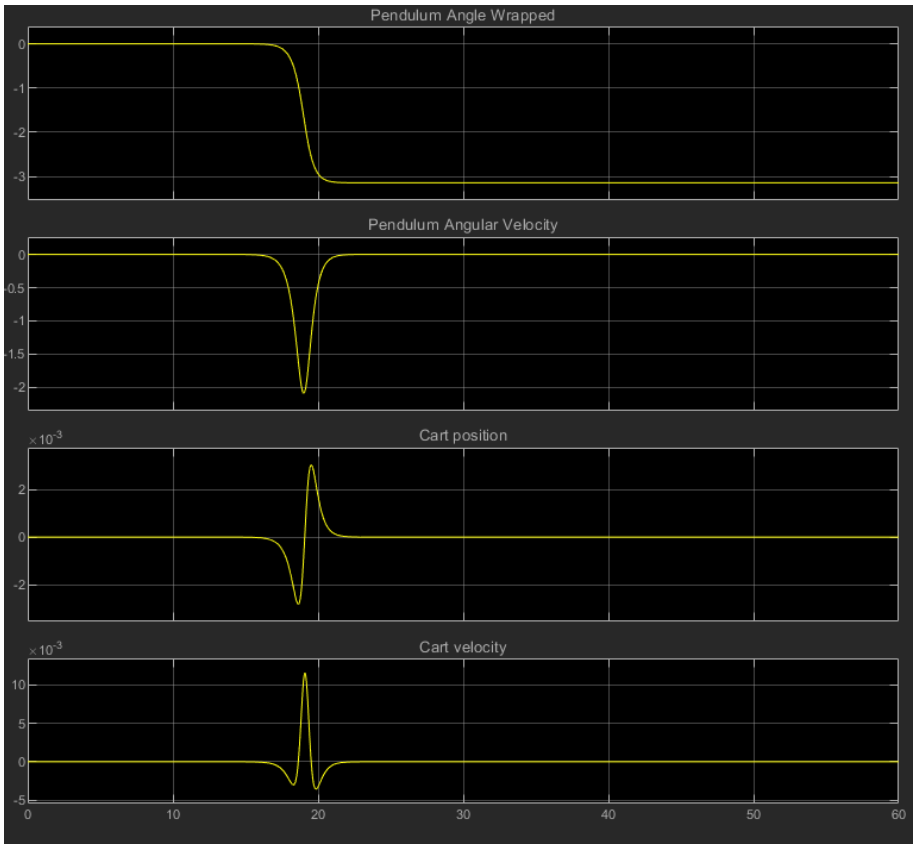


Figure 9: Rigidbody System Disturbanceless

5.2.1 Parallel PID Control

A parallel PID controller system is made using the previously defined Pendulum-Cart Subsystem as the Plant, and a few Simulink components to allow controlling of the system.

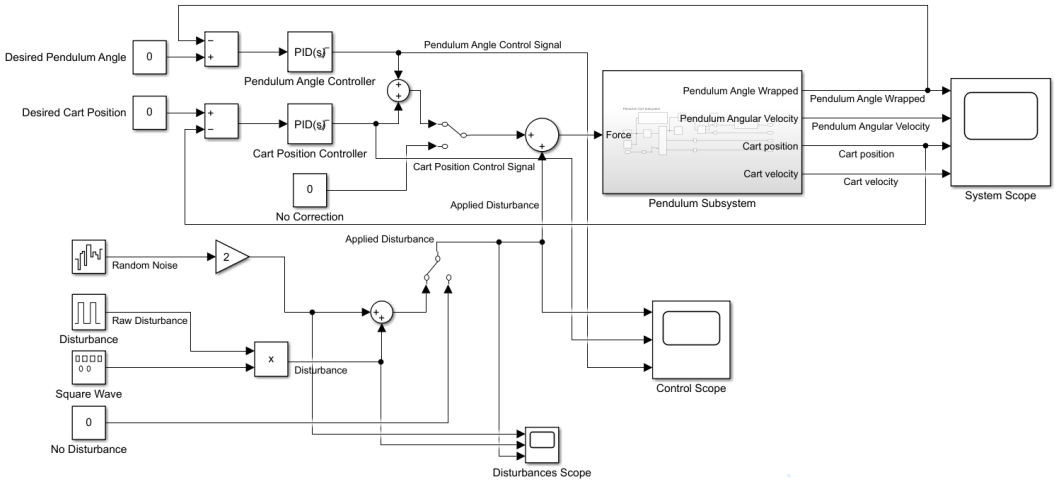


Figure 10: Simscape/Simulink Parallel Control System

The raw disturbance is of an arbitrarily high amplitude which is then multiplied by a square wave with an amplitude of 1, with some added random noise. The former results in a pseudo-random disturbance with amplitude between -10 and 10 [kN] as seen in figure 14. The random noise is not visible in the applied disturbance scope due to the difference in amplitude being 10^3.

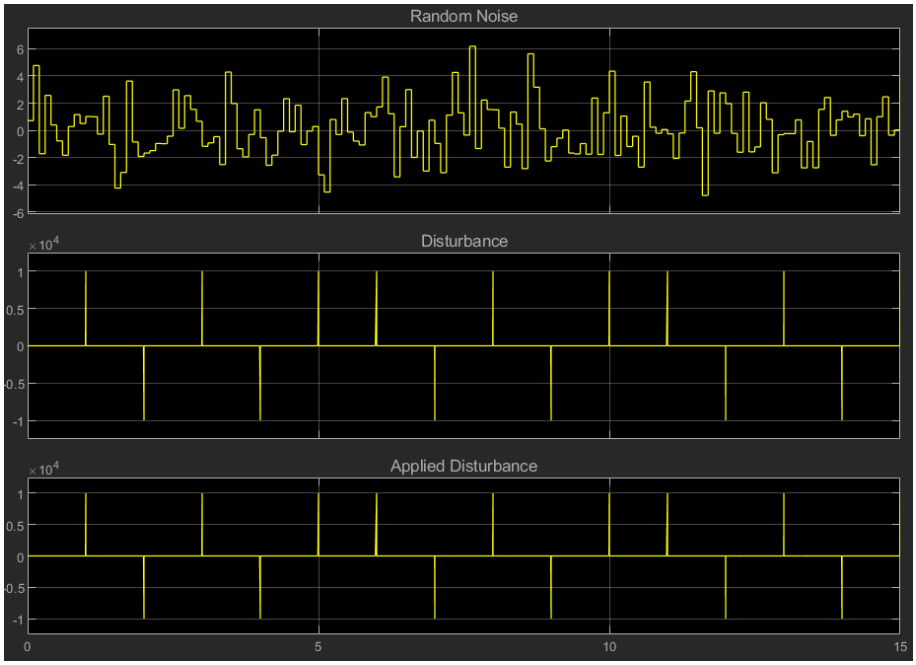


Figure 11: Disturbance Graphs

The PID-Controllers used in the parallel control structure are tuned independently of eachother, done by disconnecting one controller while the other is being tuned, and vice versa. The controllers have built in anti-windup of type back-calculation with a coefficient of 1, and saturation with the bounds -10 and 10. It may be worth noting, that the pendulum angle is wrapped to be between $-\pi$ and π [rad]. The simulated control system's behaviour can be seen below.

Applying the aforementioned disturbances to the defined control system, the pendulum angle always eventually converges to 0 [rad] with a maximum divergence of magnitude $2 \cdot 10^{-1}$ as seen in the figure below.

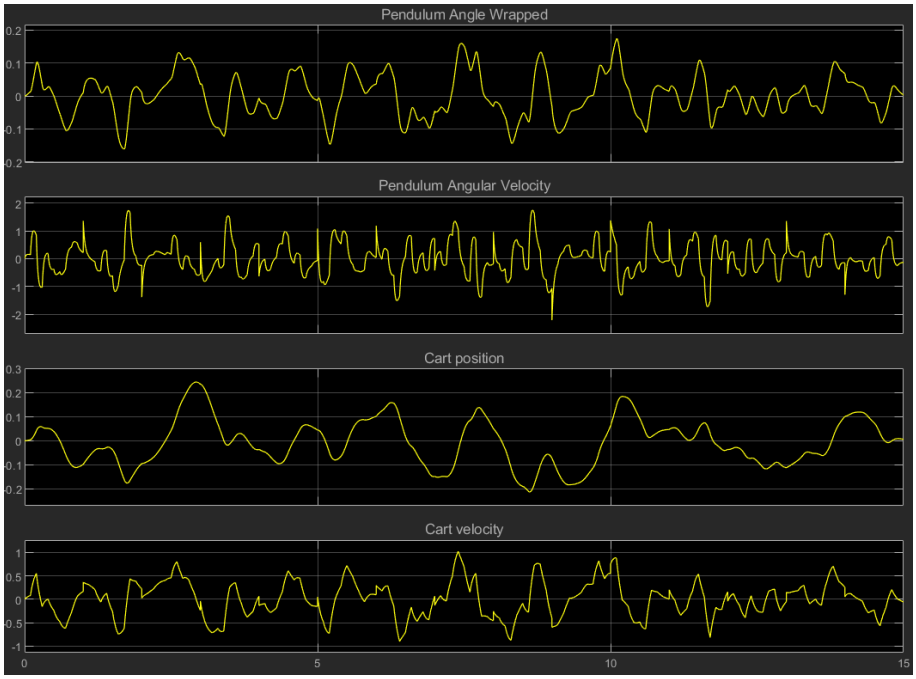


Figure 12: Parallel Control System Response

5.2.2 Cascade PID Control

Similarly, a cascade PID controller system is made using the previously defined Pendulum-Cart Subsystem as the Plant, and a few Simulink components to allow controlling of the system.

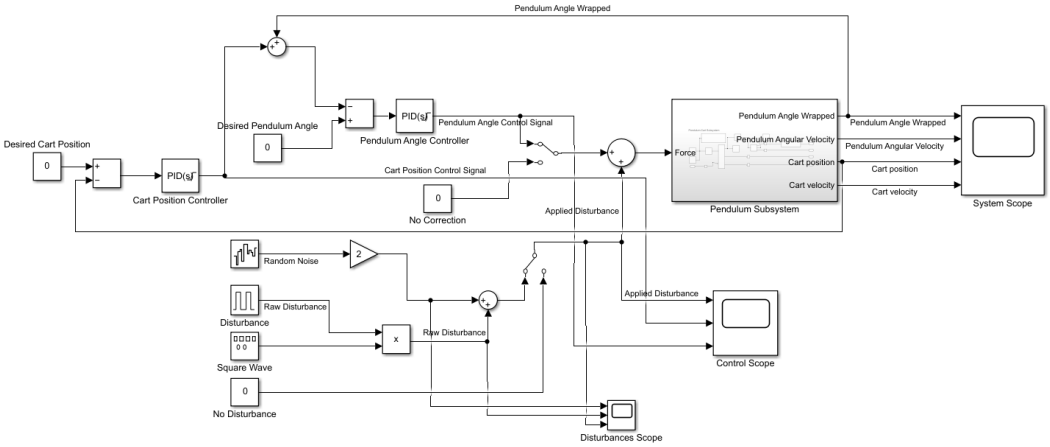


Figure 13: Simscape/Simulink Cascade Control System

The same disturbance is applied as on the parallel system, seen below.

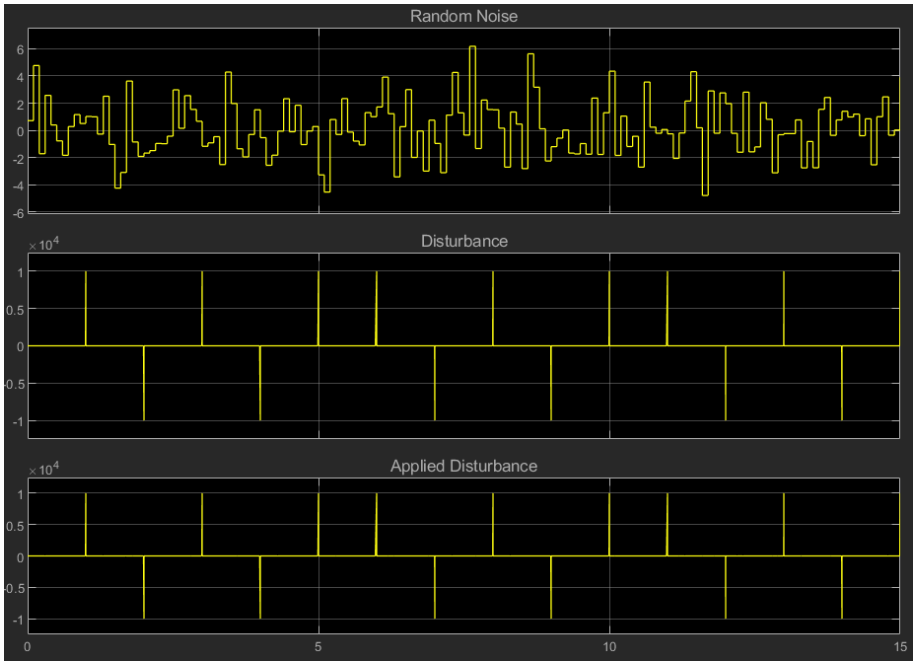


Figure 14: Disturbance Graphs

Similarly to tuning the PID controllers of the parallel system, the controllers are tuned independently in the cascade system. The controllers have built in anti-windup of type back-calculation with a coefficient of 1, and saturation with the bounds -10 and 10. It may be worth noting, that the pendulum angle is wrapped to be between $-\pi$ and π [rad]. The simulated control system's behaviour can be seen below.

Applying the aforementioned disturbances to the defined control system, the pendulum angle always eventually converges to 0 [rad] with a maximum divergence of magnitude $2 \cdot 10^{-1}$ as seen in the figure below.

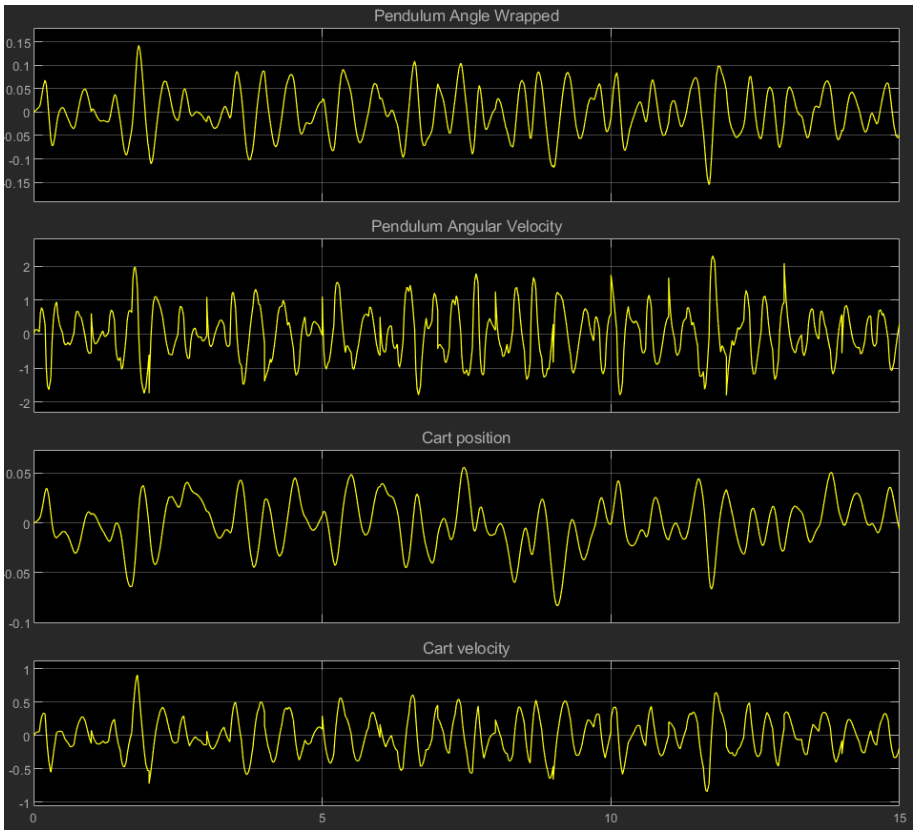


Figure 15: Cascade Control System Response

6 Discussion

7 Future Work

8 Conclusion

9 References

[1] Control Tutorials for MATLAB and Simulink. *Inverted Pendulum: Simscape Modeling*. URL: https://bit.ly/3r3Glm7.

10 Appendix

Appendix A