Delft University of Technology

Control Systems Lab

SC42045

Control Systems Design for Rotational Pendulums

Robert Rooijmans (4362586)
Wouter Meijer (4391659)
Coen van Gruijthuijsen (4353528)

February 16, 2021

Contents

1 Introduction

For the course Control Systems Lab a double pendulum is studied, where only the first link can be actuated using a motor. Firstly, the system will be modelled using a white-box approach. After determining the equations of motion and state-space equations, the parameters of the system are estimated using an error-function and least-squares fit on generated data. To stabilize the double pendulum in the unstable equilibrium where both links point upwards, the system is linearized around this point. After linearizing and designing an observer, two controllers are designed to stabilize the system around this point. Since the physical setups are not available due to the Corona virus, all results were analyzed in Simulink.

2 White-box Model Double Pendulum

In figure 1 a schematic depiction of the double pendulum is shown. Before designing a control system for the double pendulum a white-box model of the system must be determined. This white-box model is constructed by making use of Simulink.

2.1 Dynamics

Before constructing the white-box model of the double pendulum, its dynamics must be studied. The documentation of the pendulum provided the equations of motion, which are described in Eq. 1. Using this equation, the angular acceleration of both the links can be computed at any given state. In this formula matrix **M** is the inertia matrix, matrix **C** contains the centrifugal and Coriolis forces and in matrix **G** the gravitational forces are accounted for. In Appendix A the exact components of these 3 matrices can be found.

$$\ddot{\theta} = \mathbf{M}(\theta)^{-1}\left(\begin{bmatrix} T \\ 0 \end{bmatrix} - \mathbf{C}(\theta,\dot{\theta}) - \mathbf{G}(\theta)\right) \tag{1}$$

with expressions for T and θ as can be seen in 2 and 2.1:

$$T = k_m u - \tau_e \dot{T} \qquad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \tag{2}$$

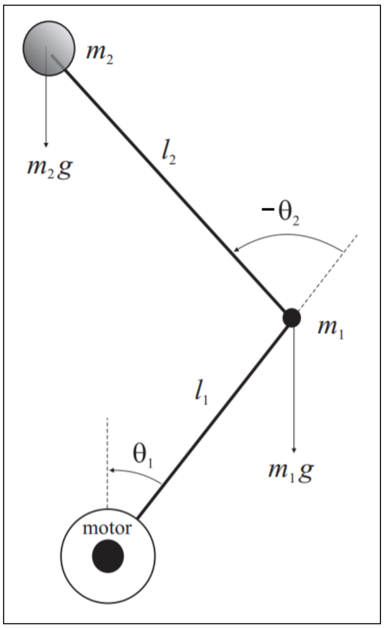

Figure 1: Schematic depiction of the double pendulum

In equation 2, k_m is the motor gain, u is the input signal provided to the motor and τ_e is the motor electric time constant.

2.2 Parameter Estimation

The documents that were delivered with the pendulum contained a set of formulas describing the pendulum completely, however, the parameters used in that formula were unknown. In this section, we describe how we estimated those parameters, what those parameters turned out to be and the Mean Square Error of the model.

2.2.1 Parameter Estimation: Method

We used the MATLAB-Function *lsqnonlin* to estimate the parameters. Because the parameter space is fairly large (13 parameters need to be estimated) it is useful to utilize various setups that isolate a subset of the parameters and make sure the other parameters can not influence the results. Firstly, we have investigated a setup where the first bar is standing still and the second bar is swinging from an arbitrary position. With this method, we could estimate g, m2, c2, I2 and b2. Afterwards, we have investigated 4 setups across 2 variables: a setup where the second bar was fixed to the first bar with a rubber band, i.e. $\theta_2 = \pi$ versus the normal situation and 2 different inputs: A constant input of $u = 0.8$ and a input of $u = sin(t)$. We used the normal situation as a validation of our estimation.

Setup	MSE θ_1 & θ_2
FIXED+CONSTANT	0.0048 & 0.0000
FIXED+SINUSOIDAL	0.0781 & 0.0000
FREE+CONSTANT	0.0201 & 0.0608
FREE+SINUSOIDAL	0.0447 & 1.0577

Table 1: Mean Square Error of different setups and measurements

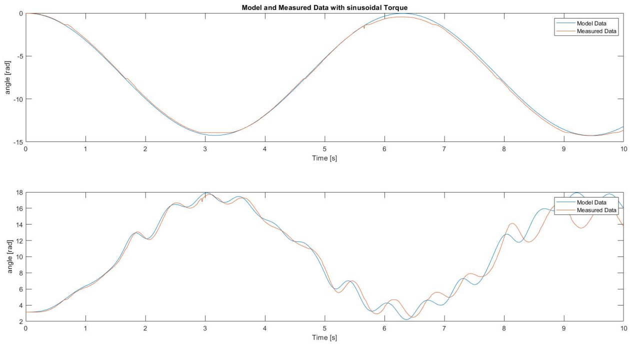

Figure 2: Measured data vs model data of the unbound setup with sinusoidal input

2.2.2 Parameter Estimation: Results

In table 1 the MSE of the model is shown. The mean square error is low, with the exception of the MSE for the last setup of θ_2. We have included a graph of that setup. What is happening there is that small deviations in the model will eventually be hugely amplified. Therefore it is important to always include an observer that can correct those deviations and make sure those deviations stay between acceptable bounds. We will further investigate the observer design in chapter 6. We have also included the parameters that resulted from the LSQnonlin search. One parameter does not influence the model, that is $l2$. A few parameters hit the lower or upper boundary of their search range. This usually indicates that LSQnonlin did not find an optimum. However, upon loosening those bounds, we did not improve our search. The MSE is low enough to prevent it from being a problem, though. Above that, we will show in chapters 4 and 5, Controller design, that our controllers are robust to small deviations in the parameters.

g	l1	l2	m1	m2	c1	c2
9.8150 m/s^2	0.0910 m	0.1000 m	0.0500 kg	0.0435 kg	0.0101 m	0.0777 m
I1	I2	b1	b2	km	Te	
0.0262 kgm^2	2.3000e-05 kgm^2	5.7811 kg/s	2.1048e-05 kg/s	-41.1184 Nm	0.0158 s	

Table 2: The parameters found by lsqnonlin

2.3 Nonlinear Model

Due to recent circumstances we were not able to test the controller on the physical pendulum, therefore we have expanded the Non-Linear Model, depicted in Fig 3, to contain measurement-noise and slightly off parameters. This way, even though we can not see if the controller would indeed control the physical pendulum, we show that the controller is robust enough to control a variety of pendula and it is reasonable to expect that our controller could control the physical pendulum, too. We have chosen for a measurement noise with a power of $10^{-}7$ with a sample rate of 0.01. We did this because we found through visual inspection that the maximum amplitude of measurement noise was in the order of magnitude of 0.01 rad. The noise we apply is in the same order of magnitude of that.

We changed the parameters up to 5%. This is a very large difference and certainly more than the physical model will turn out to be. So, if our controller will be able to handle this, it will most likely be able to control the physical pendulum.

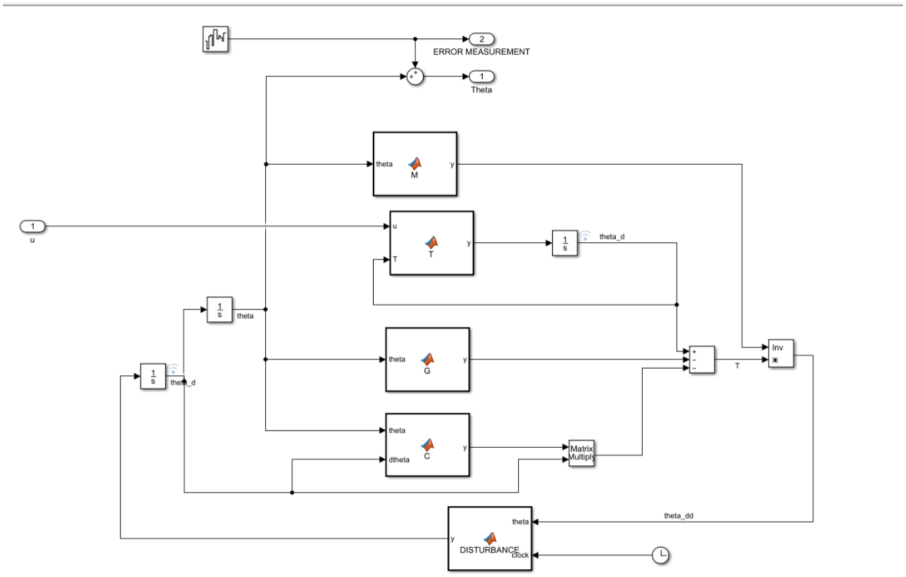

Figure 3: Simulink nonlinear model

2.4 Linearization

To control this non-linear system around an equilibrium point the system is linearized. We have considered the state as described in eq 3.

$$X = \begin{bmatrix} T & \theta_1 & \theta_2 & \dot{\theta}_1 & \dot{\theta}_2 \end{bmatrix}^T \tag{3}$$

The system is linearized around the unstable equilibrium $\theta = [0, 0]$, meaning the links of the double pendulum both point upwards. The system is linearized using the properties shown in equations 4 and 5 and by setting terms with more than two factors of θ or derivatives of θ equal to zero.

$$\sin\theta \approx \theta \tag{4}$$

$$\cos\theta \approx 1 \tag{5}$$

After linearization the system is written in state-space equations, using state-vector x, as can be seen in equations 6 and 7:

$$\dot{x} = Ax + Bu \tag{6}$$

$$y = Cx + Du \tag{7}$$

This gives the following linearized state-space model around $\theta = [0, 0]$ as can be seen in equation 8 and 9. This model can now be used for controller design. It can also be discretized with the c2d-function of MATLAB, which will be used in chapter 5. Both routes are viable.

$$\begin{bmatrix} \dot{T} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} = \begin{bmatrix} -50.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 \\ 49.73 & 2.564 & -1.777 & -419.8 & 0.002305 \\ -109.5 & 124.5 & 134.1 & 924.2 & -0.09722 \end{bmatrix} \begin{bmatrix} T \\ \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -2999 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u \tag{8}$$

$$\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} T \\ \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \tag{9}$$

3 Black-box model

A different approach towards modeling a system is the black-box approach. In black-box modeling only the relationship between input and output is studied, without considering the internal dynamics of the analyzed system.

3.1 Black-box model: signal design

To estimate the black-box of this system around an equilibrium point an appropriate signal must be designed. This signal must contain a wide enough frequency band, so the model can be estimated accurately in the frequency domain. Since the system can be assumed to be linear close to equilibrium points in this case, the amplitude can be taken as a negative or positive constant when estimating the black-box model around an equilibrium point. The input signal used in this research was a random digital signal with zero-mean. The initial configuration of the double pendulum was $\theta = [\pi, 0]$, so with both links pointing downwards. Data is generated using signals that were designed as described above. Since only data from the earlier mentioned configuration is generated, the system is linearized around this point. Unfortunately, it is not possible to use black-box methods around unstable equilibria. This means the control objective of stabilizing the system in the up-up configuration is not possible using black-box methods.

3.2 Black-box model: ARMAX

Using an auto-regressive moving average model with exogenous inputs, also known as ARMAX, a black-box estimation of the system is determined. ARMAX makes use of formula 10 to estimate the relationship between input u and output y accounting for noise e:

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{C(q)}{A(q)}e(k) \tag{10}$$

Appropriate orders have to be chosen for the polynomials A, B and C. The input-output delay for this setup is set to zero for both outputs. If the order of A, B and C are chosen too low, the system might be too complex to describe and the results would be insufficient. If the order of the polynomials is too high over-fitting might occur. So the orders should preferably be as low as possible, while still being able to accurately describe the input-output relationship. Using educated guessing and trial and error, the orders that are chosen for A, B and C are shown in equations 11, 12 and 13. To see if the model is not only able to describe its own data, a validation data set is used. This data is not used in estimating the black-box model, but is later used as a validation method after the model has been estimated.

$$O_a = \begin{bmatrix} 1 & 1 \\ 4 & 4 \end{bmatrix} \tag{11}$$

$$O_b = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \tag{12}$$

$$O_c = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \tag{13}$$

In figure 4 the results of the black-box estimation are shown. The model "*sys1*" is estimated using the data set "*data1*". As expected the simulated response compares very well to the data set. To validate the black-box estimation, validation data set "*data2*" is used. The black-box model is able to describe *data2* in both θ_1 and θ_2 with an accuracy of around 80%, ensuring that the estimated black-box model is reasonably accurate around the equilibrium point.

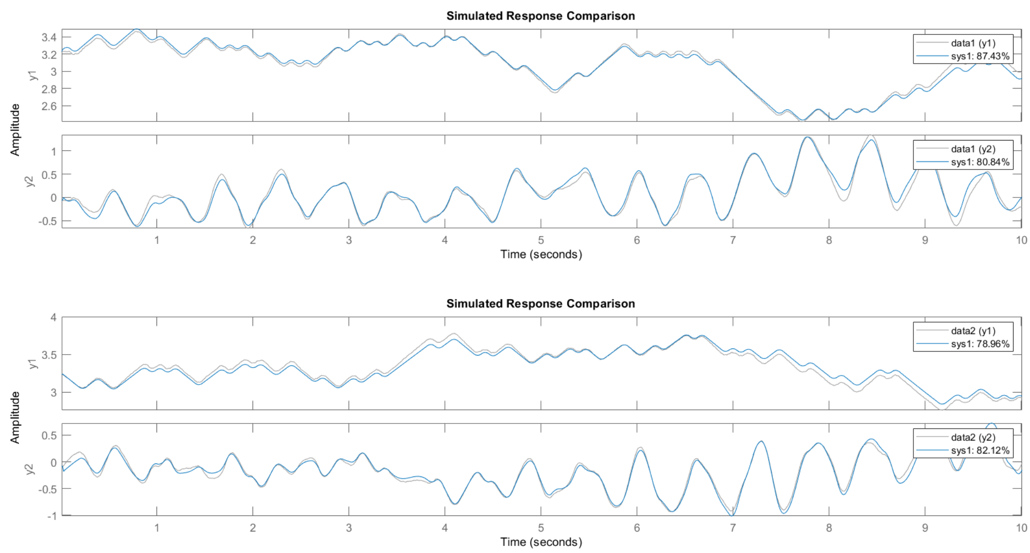

Figure 4: Results of the black-box estimation of the double pendulum

Controller	natural frequency (ω_n)	damping factor (ζ)	Settling time (s)	Input power
Controller 1	4	0.6	1.0244	0.0145
Controller 2	4	0.65	0.5766	0.0148
Controller 3	3.75	0.7	0.6580	0.0137
Controller 4	3	1.1	2.9913	0.0124

Table 3: Settings and Results of 4 Pole-placement controllers.

4 Pole Placement Controller

In this (sub)section, we will only discuss the Feedback-controller, the reference-gain will be discussed in a later section. We designed a few controllers with the place-command. Those controllers had to adhere to the following: 1. The desired input of the controller should not be higher than 1, the maximum the pendulum can accept, for a prolonged period of time. 2. The Controller should track the reference as fast as possible. We have chosen to design our controller with the pole-placement technique in continuous-time. As discussed in the chapter 2, we have chosen for the following state: $X = [T\ \theta_1\ \theta_2\ \dot{\theta_1}\ \dot{\theta_2}]^T$ Because the state consists of 5 elements, the controller needs to have 5 poles. All of these poles needs to be negative to keep stability. 2 poles were chosen close to the imaginary axis and the others far away, to make the system as close to a second order system as possible. in Eq. 14. the formula for choosing the dominant poles is shown, i.e. the poles close to the imaginary axis. We have used this formula and in the following paragraphs we will explain which and why which parameters we have chosen. The other parameters were chosen arbitrarily and were chosen as $P = [-50, -60, -100]$.

$$s_{1,2} = -\omega_n * (\zeta \pm \sqrt{\zeta^2 - 1}) \quad (14)$$

The damp function (provided by MATLAB) of the uncontrolled system told that there was a steady-gain, two poles with a natural frequency around 11 rad/sec and 2 poles with even higher frequency. A natural frequency around 3 to 5 rad/s is therefore a good idea. If the natural frequency is chosen higher the system performs better, (it stabilizes faster and will follow the reference closer), but the stability might not be guaranteed. Because we do not know exactly what we are controlling (because we change the parameters randomly), it is important that we check that extensively.

Choosing a damping factor is somewhat more difficult. It involves trial and error and check what looks good. This has to be combined with fine tuning the natural frequency. However, for most applications, a damping factor around 0.7 is a good idea. Therefore, we have tested various values for these parameters. We ran those tests multiple times for each value pair, because each test the randomized parameters were different and could lead to different results. An example of such a test is shown in table 3 and fig 5.

4.1 stabilize around unstable equilibrium

We made a test where the controller had to stabilize the pendulum around it's unstable equilibrium, i.e. $\theta = [0\ 0]^T$. The starting position was for all controllers $\theta_{start} = [0.3\ \ -0.1]^T$. We have analyzed 4 promising controllers, and we checked them on the power of the input signal and the settling time (which we have defined as the last time either one of the (absolute) angles was above 0.05 rad). The results are displayed in table 3. According to this table and to fig 5, which shows a simulation of each controller, Controller 3 is the best controller of this set. Therefore, we will compare all future controllers discussed in this paper with this controller.

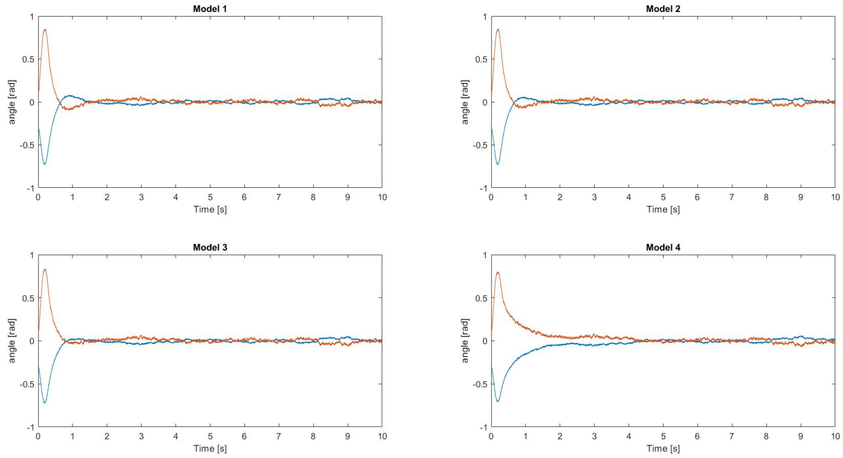

Figure 5: Simulation of 4 controllers

4.2 Pole placement controller: reference tracking

Using the pole placement controller, the pendulum is made to track an external reference signal. While maintaining the second link in the upwards pointing position, the first link is made to track the signal $0.3 \sin t$. To design this reference tracking a reference with a gain K_r is added to the control input, leading to a control input that is formulated as follows:

$$u(k) = -K_{pole}\hat{x}(k) + K_r r(k) \tag{15}$$

In designing this reference gain, the magnitude is equal to the inverse of the DC gain of the closed loop system. In figure 6 the results of this can be seen. The yellow line is the reference signal that θ_1 is supposed to follow. As can be seen this control objective is met, although with a slight delay. Since θ_2 is defined relative to θ_1, the angle of θ_2 is negative and approximately equal in magnitude compared to θ_1. This means the second link is pointing upwards and the control objective is achieved.

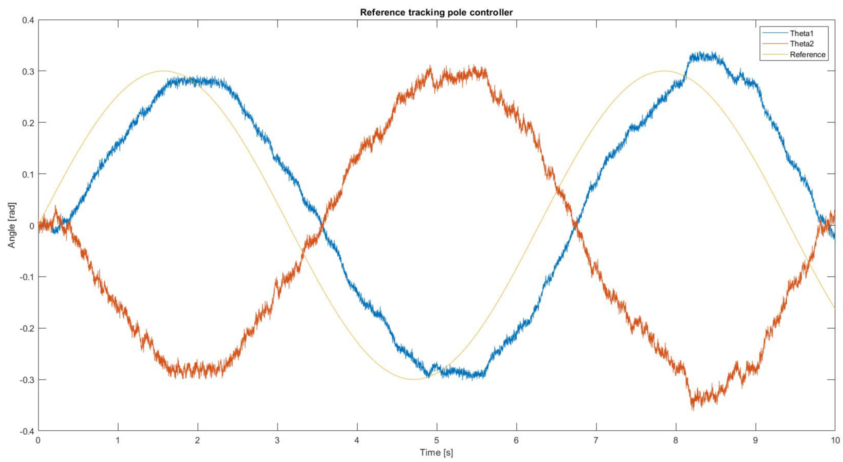

Figure 6: Results of reference tracking by making use of the pole placement controller

4.3 Pole placement controller: stabilizing after disturbance

An additional control objective is stabilizing the double pendulum in upright position after a disturbance. To simulate a disturbance a function is implemented that gives the second link a push when 1 second has passed in the simulation for the duration of 0.1 second. This push is equal to an angular acceleration of 25 rad/s^2 on top of the angular acceleration due to its configuration and control actions. The results of this simulation can be seen in figure 7. As can be seen the push on the second link immediately causes the first link to respond. Although the system gets pushed more than $0.5rad$ away from the equilibrium point where the system is linearized, the double pendulum stabilizes reasonably fast and reaches the equilibrium point again after some overshoot. The settling time is reached around 4 seconds after disturbance.

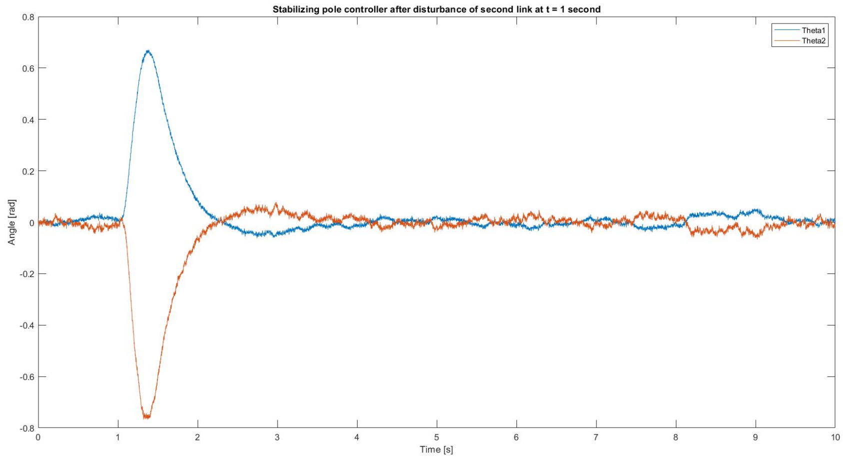

Figure 7: Results of stabilizing after disturbance by making use of the pole placement controller

5 LQR Controller

The LQR controller is a type of controller which in the end controls the system in a similar manner as does pole placement. However here the the poles are arrived at through first formulating an optimization problem and then solving it.

$$J = \int_0^\infty (x^T Qx + u^T Ru)dt \tag{16}$$

This formulation has 2 tuning parameters: Q and R. The Q matrix corresponds to the importance assigned to actuator speed. The R terms corresponds to the penalty assigned to actuator effort aka energy consumption. This formulation adds more intuition to the tuning process, because now the control engineer can selectively translate his control objectives to relative weights in Q and R. In our case the following points were considered. The pendulum setup runs on wired power so total energy consumption is not a constraint, however peak actuator effort is a constraint. The motor input should be between -1 and 1 to avoid saturation. The choice has been made to choose Q so that speed is maximised while keeping R so that |u| does not exceed the threshold. Practically this amounted to a Q matrix where a large penalty (100) is given to errors in the angles and a smaller penalty (1) is given to errors in the torque and angular velocities. The penalty(2000) to actuator effort in R was made sufficiently large such that the input u indeed never grew larger than 1.

5.1 LQR controller: reference tracking

For the LQR controller the same control objectives are tested to compare the two controllers. The reference tracking works in the same method as explained in chapter 4.2, but now with the LQR controller instead of the pole placement controller. The results of this can be seen in figure 8. Again the control objective is met, although with a slight delay making the performance of the LQR controller very similar to the performance of the pole placement controller.

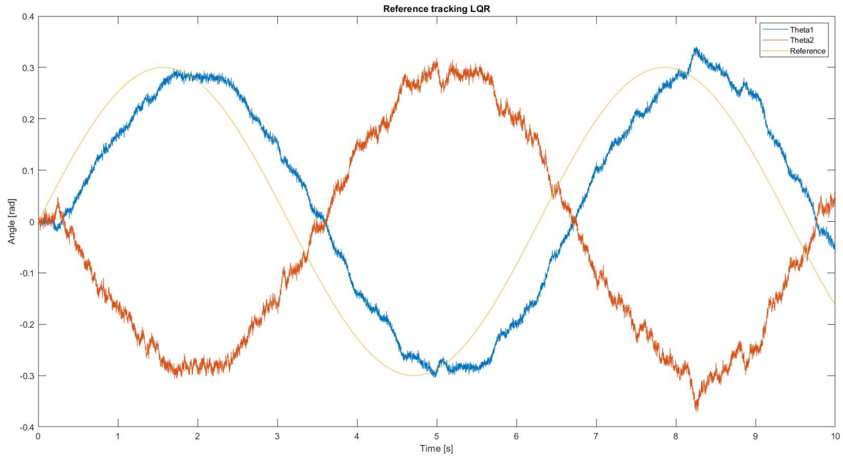

Figure 8: Results of reference tracking by making use of the LQR controller

5.2 LQR controller: stabilizing after disturbance

Again the control objective of stabilizing after disturbance is simulated in a similar way as described in 4.3, only now with the LQR controller. As can be seen the controller is better able to resist the disturbance than the pole placement controller. The angle of the second link is lower after disturbance and the controller achieves the system to be in equilibrium with a lower settling time and with less overshoot.

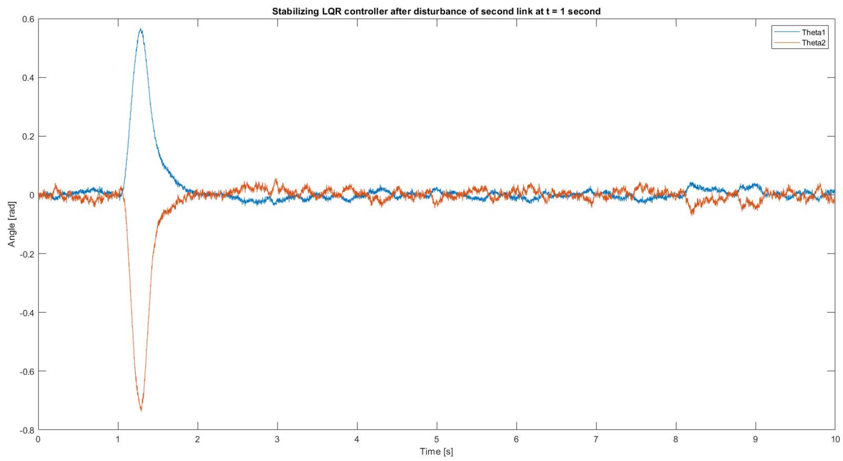

Figure 9: Results of stabilizing after disturbance by making use of the LQR controller

6 Observer design

The internal linear model in our controller slowly deviates from the physical model. This is due to the linearization and small deviations in the parameters. Therefore, it is important to update the state of the model by an observer. The observer has to adhere to the following:
1. The observer has to make sure the model follows the double pendulum reasonably enough, which is the most important requirement. If this requirement is not reached, the system will not be stable. Another way of phrasing this requirement would be that the observer has to be much (around 1 Order of Magnitude) faster than the controller or the system.
2. The observer has to be insensitive to measurement-noise, so some sort of filter is required. If the observer is overly receptive to noise, the controller will give unnecessary inputs and therefore results will be sub-optimal.
In this part, we will use the same test setup as in section 4. The controller we use is Controller 3 from that section. In Fig. 10 the results are plotted. The measurements are in the upper panel and the angles according

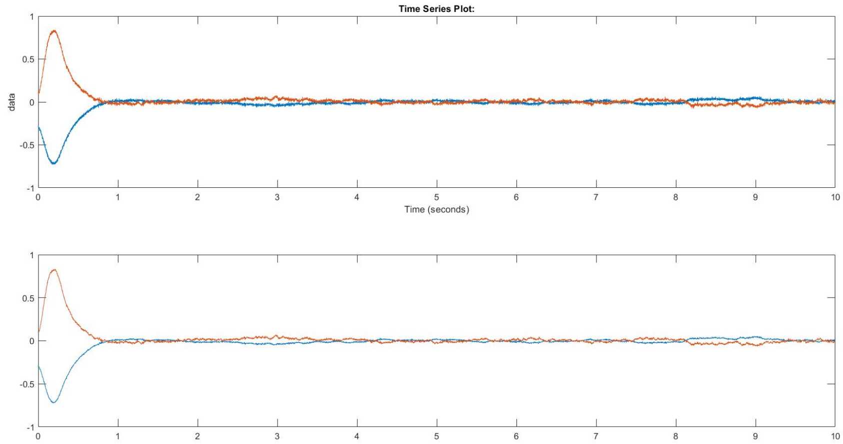

Figure 10: Comparison between measurement and the model

to the model are in the lower panel, so that those can be easily compared. The pictures show a stable model, so it follows the measurements closely enough. Although less clearly, The observer acts as a filter too: the highest frequencies are filtered out.

7 Discussion and Conclusion

Firstly, our design process has revealed that between pole placement and LQR the achievable performance is very similar. Motivations for either method should mainly take into account their distinctions in the design process. Especially for increasingly complex systems LQR allows much more selective and more intuitive tuning of the controller than pole placement does.

Secondly, when comparing blackbox and whitebox estimation methods it has become apparent that while blackbox requires far less domain knowledge to model it is limited to modeling and thus controlling configurations where system identification can be performed. This severely limits its applications.

Thirdly, the physical system will always deviate from what has been modeled in unexpected and interesting ways. We are sad that the Corona crisis has hindered us in testing the controllers we designed in the real world. However to account for the inevitable challenges with the physical setup we added variance to our nonlinear model which takes the place of the physical setup in our testing efforts.

Finally, for the final control objective of 'swing up' we would need to further expand our controller with a preliminary stage where we bring our pendulum from the down-down configuration to the linearised working space of the up-up configuration.

A Appendix

$$M = \left[\begin{array}{cc} m_1\,{c_1}^2 + m_2\,{c_2}^2 + 2\,m_2\,\cos\left(\theta_2\right)\,c_2\,l_1 + m_2\,{l_1}^2 + I_1 + I_2, & m_2\,{c_2}^2 + l_1\,m_2\,\cos\left(\theta_2\right)\,c_2 + I_2 \\ m_2\,{c_2}^2 + l_1\,m_2\,\cos\left(\theta_2\right)\,c_2 + I_2, & m_2\,{c_2}^2 + I_2 \end{array}\right]$$

$$C = \left[\begin{array}{cc} b_1 - c_2\,l_1\,m_2\,\dot{\theta}_2\,\sin\left(\theta_2\right), & -c_2\,l_1\,m_2\,\sin\left(\theta_2\right)\,\left(\dot{\theta}_1 + \dot{\theta}_2\right) \\ c_2\,l_1\,m_2\,\dot{\theta}_1\,\sin\left(\theta_2\right), & b_2 \end{array}\right]$$

$$G = \left[\begin{array}{c} -g\,\sin\left(\theta_1\right)\,\left(c_1\,m_1 + l_1\,m_2\right) - c_2\,g\,m_2\,\sin\left(\theta_1 + \theta_2\right) \\ -c_2\,g\,m_2\,\sin\left(\theta_1 + \theta_2\right) \end{array}\right]$$