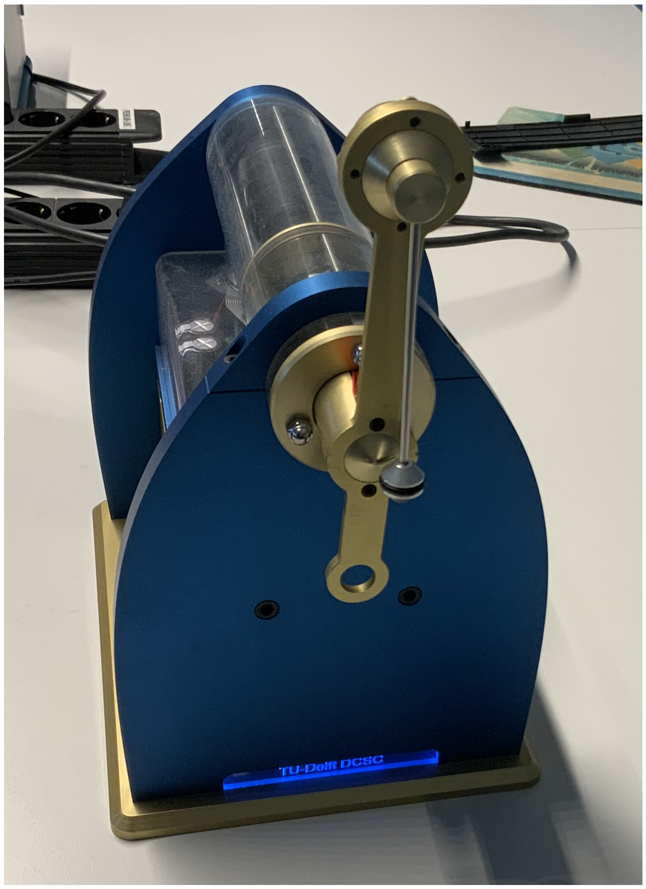
Control System Lab - SC42045

Rotational Pendulum Experiment



Cirillo Matteo
Van der Spijk Tim N.
Laporte Lilian

Technische Universiteit Delft

March 28, 2022

Contents

1 Introduction

This lab assignment follows the guidelines of the course '*SC42045 Control Systems Lab*'. Among three possible setups, a helicopter, a rotational pendulum, and a flexible link, this report focuses on the rotational pendulum. The system consists of an electrical motor, actuating the first arm of a coupled double pendulum. The control objectives for the project are chosen to be stabilization of the system in an unstable equilibrium point, and secondly, input disturbance rejection in an already stable position.

In order to achieve control of the system, a mathematical abstraction must first be devised, in order to represent the system's dynamics.

2 System interface

2.1 Modelling

The model relies on two generalized coordinates, defined as the angles formed between the arms of the pendulums and the vertical axis of the system (see Figure 1). From those, the system's state $\vec{x}(t)$, its non-linear model $\vec{F}$ and its output $\vec{y}$ are respectively defined as:

$$\vec{x}(t) = (\frac{d\theta_1}{dt}(t),\ \frac{d\theta_2}{dt}(t),\ \theta_1(t),\ \theta_2(t),\ T)^T \qquad \frac{d\vec{x}}{dt}(t) = \vec{F}(\vec{x}(t), u(t), t) \qquad \vec{y}(t) = (\theta_1(t),\ \theta_2(t))^T \tag{1}$$

Where u is the input to the motor and T, its torque. (See Appendix A & B for further details)

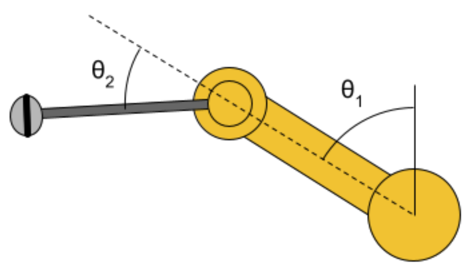


Figure 1: Definition of the generalized coordinates

2.2 Calibration

The aforementioned model is an ideal representation of the plant, stemming from classical mechanics. Some underlying assumptions are made and it does in no way account for all real dynamics. A first notable example of this resides in the continuous time nature of the model against the digital interface of the real plant.

Sampling

Since the entirety of the control design process takes place in continuous time and relies on *Simulink* for AD/DA conversions, it is needed to set a sampling period "h". In order to avoid aliasing artefacts within the operating frequency range, this period should be chosen according to Shannon's sampling theorem. Therefore, as the system's fastest dynamic is the motor's electrical constant $\tau_e \approx 10$ms (see Table 4), h is set to 5ms.

Sensors output transform

As the sensors of the real system give raw voltages, some calibration is needed in order to correctly interpret the acquired data as the angles θ_1 and θ_2. Both sensors behave similarly and can be calibrated in the same fashion. The following therefore only highlights the calibration of sensor θ_1.

Mapping the sensor output to physical positions of the pendulum yields a signal wrapped in the range $[0, 4.9]$ Volts as in Figure 2 below. The map from the wrapped angle to the raw sensor data can therefore be approximated as affine, of the form:

$$v(\theta) = \gamma_0 + \gamma_1\theta \qquad \text{with} \qquad \gamma_0 = v(\theta)\Big|_{\theta=0} \qquad \text{and} \qquad \gamma_1 = \frac{\Delta v}{\Delta\theta} \tag{2}$$

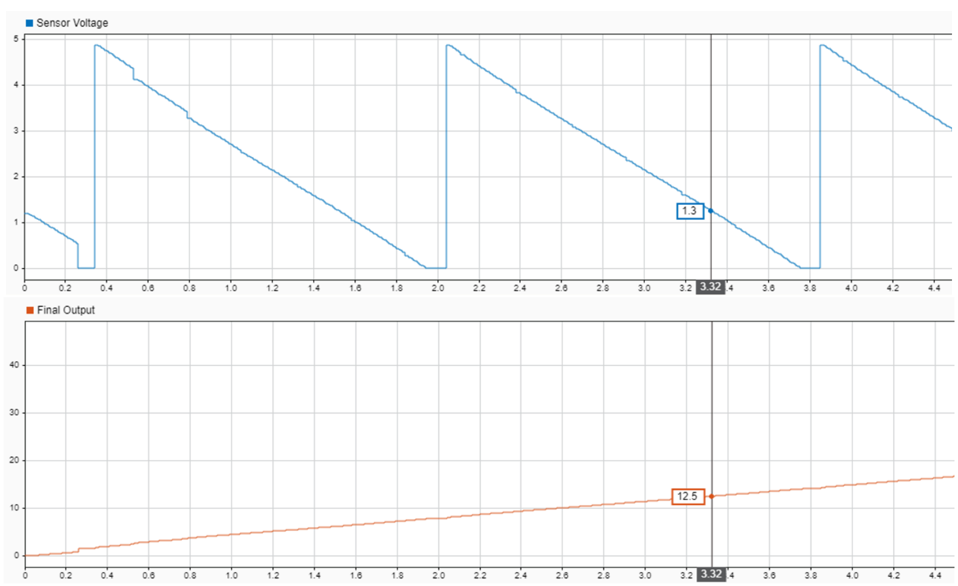


Figure 2: sensor1 data under constant rotation (raw [V], and calibrated [rad], respectively)

The sensor can thus be calibrated by measuring the value of the voltage at only two positions of the angle. First at $\theta = 0$, thus determining the intercept γ_0, and secondly at $\theta = \pi$, to measure the linear variation. With both parameters one can finally invert the relationship and unwrap the signal directly in Simulink to get $\theta(v)$, having thus calibrated the sensor.

Dead-Zones discussion

It should be noted that the sensors both have a dead-zone[1] around $\pi/2$. This is visible on the *reference* signal of Figure 3. It reaches $0.47\pi = 1.48rad$ and stays constant, only to continue after the wrap at $-1.42\pi = -4.48rad$. Those zones thus extend from $[0.47\pi, 0.58\pi]$ on the $[0, 2\pi]$ range, which represents approximately 5%.

Note that attempting to correct this problem by stretching the sensor's output to match the 0 to 2π boundary is not a desirable solution. It certainly yields an attenuated discontinuity, but also spreads an error over the whole range of measurement, (see *stretched* signal in Figure 3) such that $E_{RMS} = 12.7\%$. It is therefore reasonable not to opt for stretching, in order to keep an accurate measurement at the cost of undetermined angles within the deadzones. However, since the controllers later make use of linearized models, it is reasonable to assume that the angles of the controlled plant never come close to the deadzones (located at $\pi/2$ for both links).

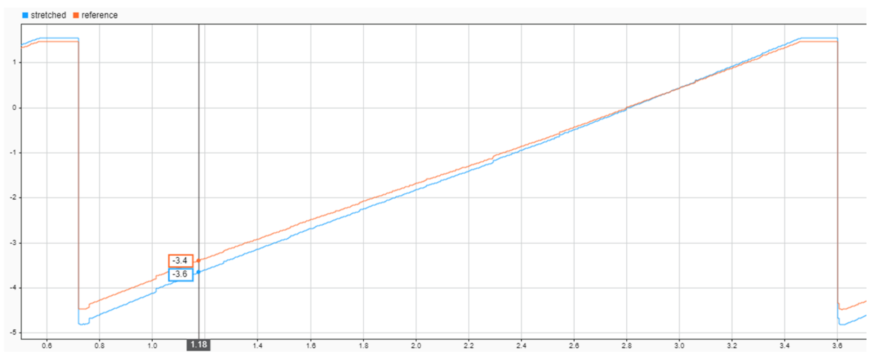


Figure 3: comparison of the stretched (orange) and reference (blue) measurement

3 White Box model

The goal of the White Box modelling is to tune the parameters of the model for it to best represent the real pendulum. This process is notably influenced by unmodelled phenomena. It relies on the Non-Linear Least squares optimization method to compare and fit output data from the model to data from the real pendulum (see Figure 4). Both signals, which result from the same stimuli of the system, are compared with respect to a predefined cost function[2], and the parameters are tuned so that the error is minimized. Note that to avoid over-fitting, the data set is split, and approximately $\frac{2}{3}$ of it is used for optimization whereas the remaining $\frac{1}{3}$ constitutes a test set. The *Matlab* function `goodnessOfFit` returns the Normalized RMS Error between both curves, which is used for validation in the test set.

Finally, as estimating all the parameters at once is too demanding for the algorithm, experiments aiming at reducing the number of degrees of freedom should be used instead.

[1] With the measurand within a dead-zone, the measurement is clamped no matter the sensed value.

[2] In this project, the default sum of squared residuals is used

3.1 Design of experiments

Reducing degrees of freedom in the system implies constraining certain mechanical parts of the plant. The dynamics of the model are then simplified such that they become independent to a subset of parameters. The simplest possible simplification in the plant is the free swinging of the second link alone, detailed below:

Free swinging : By constraining the first link to $\theta_1 = -\pi$, the model simplifies and only depends on physical parameters related to the second link. Signals are thus generated by swinging the second link from different initial angles. The results of the final optimization can be found in the Table 1 opposite.

Signal	Average error in training set	Average error in test set
θ_2	4.12%	11.32%

Table 1: Averaged fitting error over the different data sets for the first experiment

Motor step input : To tune parameters of the motor, the second link must be attached to the first one in order to simplify the load. The simplest way to obtain the motor gain k_m and the motor electrical constant τ_e is to use a step function as input. Errors in the fitting are in Table 2 on the right.

Signal	Average error in training set	Average error in test set
θ_1	7.98%	7.20%

Table 2: Averaged fitting error over the different data sets for the second experiment

First link parameters : Since the first link is directly connected to the motor, we can not swing it freely as with link 2. From now on, the pendulum is unconstrained and subject to different inputs (e.g chirp signal, sine waves...). θ_1 influences θ_2 and both signals must be fitted (see fitting results in Table 3 and fitting example in fig.4).

Signal	Average error in training set	Average error in test set
θ_1	3.8%	9.8%
θ_2	4.9%	11.7%

Table 3: Averaged fitting error over the different data sets for the last experiment

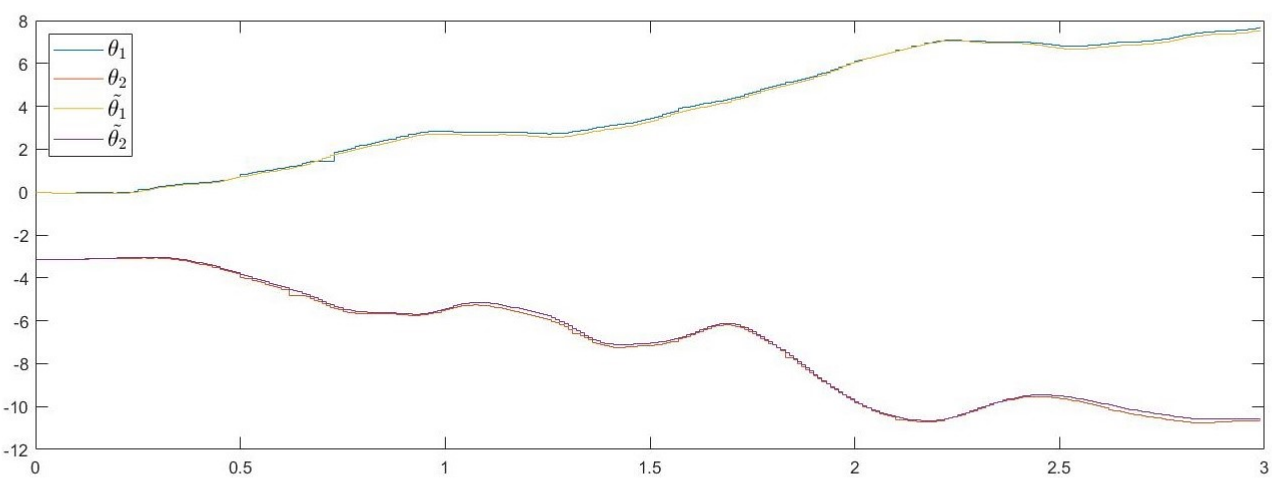


Figure 4: Example of White Box estimation output

3.2 Results

The results obtained above are satisfactory as the error is of at most of around 10%. The averaged estimated parameters are thus kept for subsequent controller design and are listed in Table 4 on the right.

The remaining fitting error can be attributed to two unmodelled dynamics in the system. Firstly static friction in the joint between the two links and secondly, mechanical backlash in the system. The model can in some cases not produce a matching signal with the given parameters, thus generating error. For example, the swing of the second link in experiment 1 is strongly influenced by unmodelled static friction at small oscillations and does exhibit higher error than other experiments.

Finally note that if θ_2 signals have a tendency to be poorer fits than for θ_1, it is also because the second link is the end-effector of the pendulum, and is hence being more affected by backlash.

Parameters	**Estimated**
l1 [m]	0.0897
l2 [m]	0.1
m1 [kg]	0.2
m2 [kg]	0.087972
c1 [m]	0.01
c2 [m]	0.07
I1 [kg.m^2]	0.01076
I2 [kg.m^2]	0.0001
b1 [kg.s^{-1}]	5.33518
b2 [kg.s^{-1}]	0.00004
k$_m$ [N.m]	39.2954
τ_e [s]	0.01

Table 4: Estimated parameters

4 Black Box model

The use of black box modelling is limited as it requires input and corresponding output data to be able to estimate a linear model. Since the system is unstable at most equilibrium points, (see Section 5), no significant amount of measurements can be taken without first stabilizing the system. In the context of this project, Black Box identification is solely pursued as an educational venture and only once the system is already stabilized with both links upwards as described in Section 5.

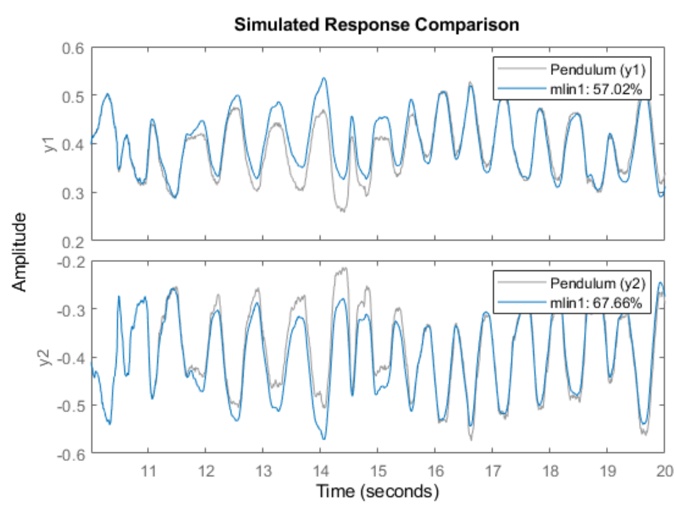


Figure 5: Black box estimation

The Black Box model is computed using the ARX estimator function of *Matlab*. As stated above, the input-output data is generated by recording 10 seconds of the system being stabilized at the unstable equilibrium point. The generated model achieved a goodness of fit of 57.02% for θ_1 and 67.66% for θ_2 (see Figure 5). This would not be considered a sufficient result for use in a control system. However, since this model is not used in practice, it is not optimised any further.

5 Linear control

As previously stated, the project's control objectives are equilibrium point stabilization and input disturbance rejection. This section details the design of adequate controllers and their results.

As a prelude, note that there are four trivial equilibrium points in the double pendulum. Both arms upwards (**up-up** position), 1^{st} arm up 2^{nd} arm down (**up-down**), 1^{st} arm down 2^{nd} arm up (**down-up**) and lastly both arms down (**down-down**).

5.1 Choice of Operating points

To motivate a choice of operating point for the two control objectives, it is useful to consider the natural stability of the system, when linearized around the candidate operating points. This can be done symbolically in Matlab using the model derived in Appendix B, with the code displayed in Appendix D.

It results that only the down down position is naturally stable, and is therefore selected as the operating point for Input disturbance rejection. As for equilibrium stabilization, the choice between the three available unstable points is motivated by the relative stability of the linearized system around the respective positions. Indeed, because of static friction, the up down position is almost stable as the first link does not move. The last choice is guided by the authors' preference, in favor of the up up position. Also note that all presented equilibrium points yield both controllable and observable linearized systems. They are therefore suitable candidates for an operating point.

5.2 Equilibrium Stabilization

Controller Layout & Implementation

As Equilibrium Stabilization reduces to pole placement with full state feedback, the chosen layout for the controller is a Linear Quadratic Regulator (to simplify pole placement), coupled with a Luenberg Observer (in order to reconstruct the full state). This in turn is called Output Feedback.

It follows from the mathematics describing this layout (see Appendix E) that controller and observer eigenvalues can both be assigned independently. Therefore, to ensure that the observed state converges quicker than the regulator's actuation, the poles of the LQR regulator are placed prior to the observer ones, which are subsequently set as faster. Values for the Q and R matrix, alongside a refresher of the plant's state and input are higlighted below, in equation 3.

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 & 0 \\ 0 & 0 & 200 & 0 & 0 \\ 0 & 0 & 0 & 110 & 0 \\ 0 & 0 & 0 & 0 & 10 \end{bmatrix} \qquad \vec{x}(t) = \begin{bmatrix} \dot{\theta_1}(t) \\ \dot{\theta_2}(t) \\ \theta_1(t) \\ \theta_2(t) \\ T(t) \end{bmatrix} \qquad R = 350 \qquad u(t) = v_{in}(t) \tag{3}$$

The most important weights here are those of the arms' positions and of the control input. The difference between weights of θ_1 and θ_2 ensures that the contribution to the control input corresponding to the two angles, can not cancel out in such a way that creates an offset stable equilibrium (for example with the second link upwards and the first link inclined, $\alpha\theta_1 + \alpha\theta_2 = 0$). Lastly the weight on the control input is also significant as the system can not afford actuation greater than a threshold ($|u| < 1$). Finally, the observer poles are placed to be proportionally faster than the controller's, with `L = place(A', C', eig(A-B*K)/8).'`.

The exact values displayed above are reached with trial and error to minimize, on one hand, the control input, and on the other, errors in the final results such as steady state oscillations and offsets.

Results

Upon controlling the real plant with the designed controller, a few things can be noted. Notably, that the controller does indeed stabilize the pendulum in the up up position as seen in Figure 6.

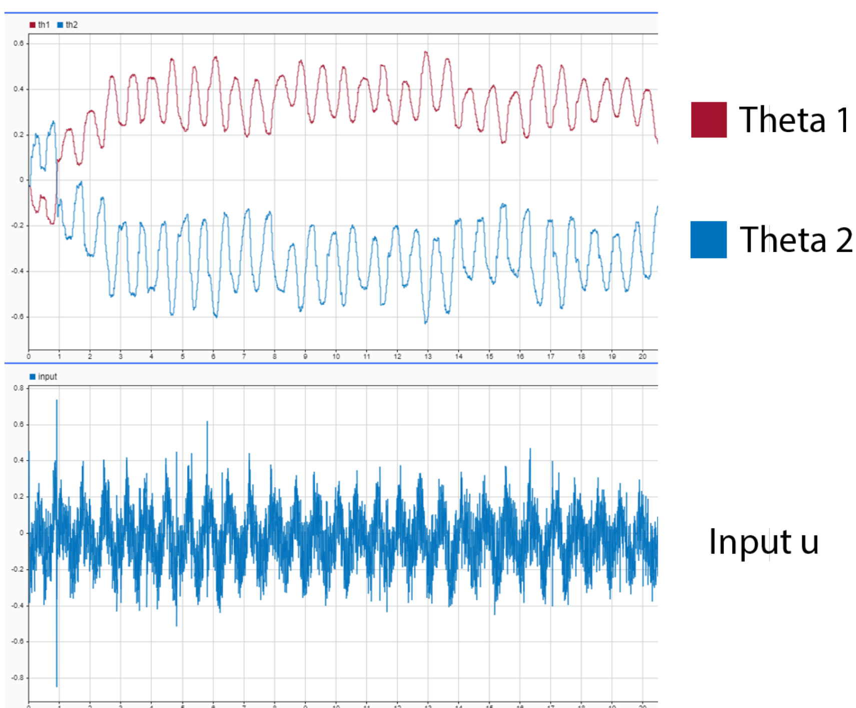


Figure 6: Typical up up stabilization results

The system however does display an offset from the desired $\theta_1 = \theta_2 = 0$ position as well as some smaller oscillations. Both signals average at approximately ± 0.35rad $\approx \pm 20°$ and with oscillations of respectively 0.3rad and 0.4rad average peak to peak amplitude, and both at 3/2Hz. This is once again attributed to unmodelled dynamics in the system, that prevent the controller from displaying ideal behavior. The two main contributing factors are static friction and mechanical backlash. The first causes the second link to be over actuated by the system (thus explaining the oscillations), and the second alienates the effect of a given control input throughout the system as slack builds up. The averaged shift in equilibrium point is thought to be a combination of both factors.

Secondly, it can be noted that the input signal, although not in saturation, appears to be oscillating and fuzzy. It averages at $\bar{u} = 0$, as desired by the equilibrium point, but displays oscillations with the same 1.5Hz frequency as seen in the θ_1, θ_2 signals. This is consistent with the previous conclusions as it highlights a cycle of overshoot (due to unmodelled dynamics) and distorted response which over-steers the system in the opposite direction. This involves both input and outputs and explains why oscillations in both are coherent in frequency.

Lastly, the apparent fuzziness of the control input is due to an equally fuzzy output from the observer. Its pole are indeed very aggressive and result in a nervous system. It should however be noted that whereas possible fixes for the unmodelled dynamics more or less consist in a better modelling, fixing the observer is not trivial. In tweaking its poles, a compromise has to be found between robustness of the observer and the response time of the controller, which has to be fast enough to work.

5.3 Input Disturbance Rejection

To control a MISO system with a PID controller, the inputs have to be combined in a common error signal. There are many ways to achieve this, but all must stabilize the system at $\theta_1 = -\pi$, $\theta_2 = 0$.

Parallel PID controllers: One PID block is used for each θ and the results are added together. This is not optimal as the two controllers are independent and their contributions can cancel out, yielding a slower controller. It is also more difficult to tune in light of the amount of parameters.

Addition of both signals: This solution has the advantage of requiring only one PID controller. A problem however is that since the error signal is $e(t) = \theta_1(t) + \pi + \theta_2(t)$, every position in which this signal vanishes constitutes an unwanted stable equilibrium point from the controller's perspective. In turn this causes the system to be stabilized at a random zero of the error signal and not in the desired operating point.

Weighted addition of signals: To prevent the vanishing of the error signal, a weight is added to one of the signals. This yields $e(t) = \theta_1(t) + \pi + \omega\theta_2(t) \mid \omega \neq 1$ and the only point where the controller can converge to, is the desired natural equilibrium point $\theta_1 = -\pi$ and $\theta_2 = 0$.

Implementation

In light of the discussion above, the controller is implemented through weighted addition of the signals and with with $\omega = 2$ (see Figure 7). It is now necessary to tailor the controller's response to a typical disturbance in the error signal, in order to minimize response time.

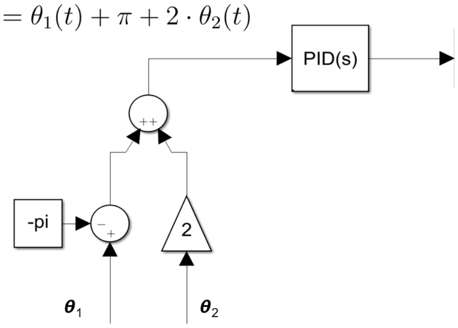


Figure 7: Schematic of the PID controller

The tuning of the parameters in the PID equation (4) below is done manually and online, in an iterative process to assess each step's improvement. It is found that since the desired equilibrium point is naturally stable, integral control is redundant as no steady state error can appear. Note that excluding this term in turn also avoids running into possible windup problems due to calibration inaccuracies.

$$u(t) = Pe(t) + I \int_0^t e(\tau)d\tau + D\frac{de}{dt}(t) \xrightarrow{\mathcal{L}\{...\}(s)} \left.\frac{U(s)}{E(s)}\right|_{e(0)=0} = P + I\frac{1}{s} + Ds \tag{4}$$

The proportional gain was set as high as possible to minimize response time all the while keeping in mind the saturation limit of the control input. Finally the derivative parameter was subsequently set to minimize overshoot. The final parameters are listed below:

$$\begin{cases} P = 1 \\ I = 0 \\ D = -0.4 \end{cases} \tag{5}$$

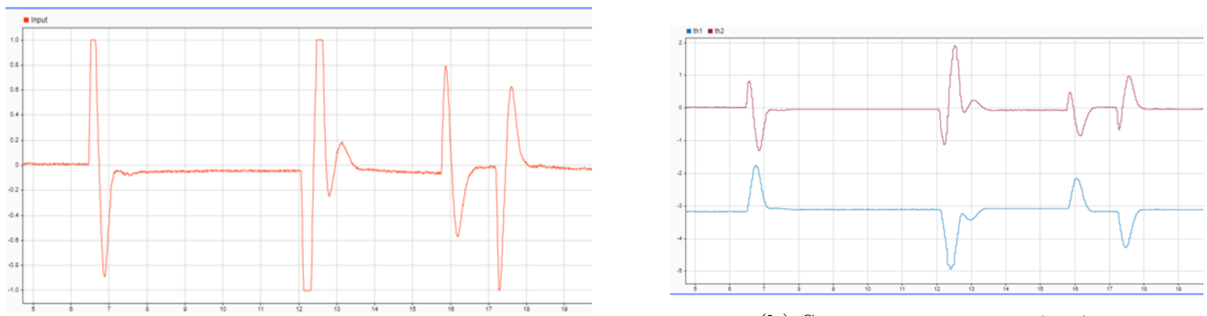
Results

Typical results to input disturbances can be seen in Figure 8. Initial peaks in the θ_2 signals are the initial conditions of such perturbation and the following oscillations are the settling of the controlled system. Table 5 displays the normalized and averaged overshoot[3] and 2% settling time[4] for such input disturbances. The overshoot is almost negligible as it doesn't exceed 5% of the initial disturbance. The settling time can be regarded as the minimal possible response time, since the control input can be seen to saturate on some typical disturbance rejections. In those points, the controller is not be able to deliver any more actuation to the plant.

Signal	**Average normalized overshoot [%]**	**Average normalized settling time [%]**
θ_1	0	5.7
θ_2	2.39	8.4

Table 5: Characteristics of PID response

[3]Normalized with respect to initial disturbance amplitude
[4]Normalized with respect to uncontrolled settling time (with equal disturbance amplitude)



(a) System input to agitation

(b) System output to agitation

Figure 8: PID Results

6 Conclusion

The goals of this Lab are firstly to create a control system that stabilizes a rotational pendulum in an unstable equilibrium point, and secondly, one to reject input disturbances induced to a stable state of the rotational pendulum.

The design of the control system for the former has been achieved by doing pole placement with output feedback implemented by a Linear Quadratic Regulator in combination with a Luenberg observer. The system has been stabilized in equilibrium, albeit with some offset and some smaller oscillations caused by unmodelled dynamics of the system.

With more time, this system could have been improved through the modelling of the main factors contributing to non-idealities in the results of the stabilization. Additionally, some attention could be given to the noisy output of the observer.

The design of the control system for input disturbance rejection has been achieved by implementing a PID controller whose error signal is a weighted sum of the sensor signals. With a minimised settling time and negligible overshoot, it can be concluded that the system does indeed achieve the goal of rejecting input disturbances.

In hindsight, the controller for input disturbance rejection should have been designed as suggested by the material of this course. This means instead of a linear combination input signals, two PID must be cascaded in series to individually regulate the two variables of the SIMO system.

This project has been a first exposure to practical applications of control theory for the whole group and it has enabled a hands on experience to apprehend the challenges of the field. What notably stood out was the importance of modelling and data fitting in the control design process as opposed to the relatively quick implementation of the controllers. Control System Design is a varied field and this project has allowed for a great overview of various typical techniques and working environment.

7 Appendix

A Non-Linear Model

The complete non-linear model used for the project is given by:

$$\frac{d}{dt}\vec{x}(t) = \vec{f}(\vec{x}(t), u(t), t) = \begin{bmatrix} M^{-1}\begin{bmatrix} T \\ 0 \end{bmatrix} - M^{-1}C\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} - M^{-1}G \\ \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ -\frac{1}{\tau_e}T + \frac{k_m}{\tau_e}u(t) \end{bmatrix} \tag{6}$$

with

$$\vec{x}(t) = \begin{bmatrix} \dot{\theta}_1(t) \\ \dot{\theta}_2(t) \\ \theta_1(t) \\ \theta_2(t) \\ T(t) \end{bmatrix} \qquad u(t) = v_{in}(t) \qquad \vec{y}(t) = \begin{bmatrix} \theta_1(t) \\ \theta_2(t) \end{bmatrix} \tag{7}$$

and

$$M(\theta_1(t), \theta_2(t), t) = \begin{bmatrix} P_1 + P_2 + 2P_3\cos(\theta_2(t)) & P_2 + P_3\cos(\theta_2(t)) \\ P_2 + P_3\cos(\theta_2(t)) & P_2 \end{bmatrix}$$

$$C(\dot{\theta}_1(t), \dot{\theta}_2(t), \theta_1(t), \theta_2(t), t) = \begin{bmatrix} b_1 - P_3\dot{\theta}_2(t)\sin(\theta_2(t)) & -P_3(\dot{\theta}_1(t) + \dot{\theta}_2(t))\sin(\theta_2(t)) \\ P_3\dot{\theta}_1(t)\sin(\theta_2(t)) & b_2 \end{bmatrix} \tag{8}$$

$$G(\theta_1(t), \theta_2(t), t) = \begin{bmatrix} -g_1\sin(\theta_1(t)) - g_2\sin(\theta_1(t) + \theta_2(t)) \\ -g_2\sin(\theta_1(t) + \theta_2(t)) \end{bmatrix}$$

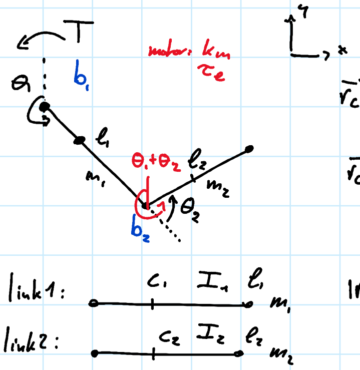
and

$$\begin{gathered} P_1 = m_1c_1^2 + m_2l_1^2 + I_1 \\ P_2 = m_2c_2^3 + I_2 \\ P_3 = m_2l_1c_2 \end{gathered} \qquad \begin{gathered} g_1 = (m_1c_1 + m_2l_1)g \\ g_2 = m_2c_2g \end{gathered} \tag{9}$$

B Double Pendulum Dynamics Derivation

Double Pendulum Dynamics

mardi, 8 mars 2022 19:06



$$\vec{r}_{c1} = \begin{pmatrix}-c_1\sin(\theta_1)\\ c_1\cos(\theta_1)\end{pmatrix} \qquad \dot{\vec{r}}_{c1} = \begin{pmatrix}-c_1\cos(\theta_1)\dot\theta_1\\ -c_1\sin(\theta_1)\dot\theta_1\end{pmatrix}$$

$$\vec{r}_{c2} = \begin{bmatrix}-l_1\sin(\theta_1) - c_2\sin(\theta_1+\theta_2)\\ l_1\cos(\theta_1) + c_2\cos(\theta_1+\theta_2)\end{bmatrix} \qquad \dot{\vec{r}}_{c2} = \begin{bmatrix}-l_1\cos(\theta_1)\dot\theta_1 - c_2\cos(\theta_1+\theta_2)(\dot\theta_1+\dot\theta_2)\\ -l_1\sin(\theta_1)\dot\theta_1 - c_2\sin(\theta_1+\theta_2)(\dot\theta_1+\dot\theta_2)\end{bmatrix}$$

$$\begin{aligned}|\dot r_{c2}|^2 &= (l_1\cos(\theta_1)\dot\theta_1 + c_2\cos(\theta_1+\theta_2)(\dot\theta_1+\dot\theta_2))^2 \\ &\quad + (l_1\sin(\theta_1)\dot\theta_1 + c_2\sin(\theta_1+\theta_2)(\dot\theta_1+\dot\theta_2))^2 \\ &= l_1^2\cos^2(\theta_1)\dot\theta_1^2 + 2l_1c_2\cos(\theta_1)\cos(\theta_1+\theta_2)\dot\theta_1(\dot\theta_1+\dot\theta_2) + c_2^2\cos^2(\theta_1+\theta_2)(\dot\theta_1+\dot\theta_2)^2 \\ &\quad + l_1^2\sin^2(\theta_1)\dot\theta_1^2 + 2l_1c_2\sin(\theta_1)\sin(\theta_1+\theta_2)\dot\theta_1(\dot\theta_1+\dot\theta_2) + c_2^2\sin^2(\theta_1+\theta_2)(\dot\theta_1+\dot\theta_2)^2 \\ &= l_1^2\dot\theta_1^2 + c_2^2(\dot\theta_1+\dot\theta_2)^2 + 2l_1c_2\dot\theta_1(\dot\theta_1+\dot\theta_2)\underbrace{\left[\cos(\theta_1)\cos(\theta_1+\theta_2) + \sin(\theta_1)\sin(\theta_1+\theta_2)\right]}_{A(\theta_1,\theta_2)}\end{aligned}$$

$$\begin{aligned}A = \cos(\theta_1)\cos(\theta_1+\theta_2) + \sin(\theta_1)\sin(\theta_1+\theta_2) &= \cos(\theta_1)\left[\cos(\theta_1)\cos(\theta_2) - \sin(\theta_1)\sin(\theta_2)\right] + \sin(\theta_1)\left[\sin(\theta_1)\cos(\theta_2) + \cos(\theta_1)\sin(\theta_2)\right] \\ &= \cos^2(\theta_1)\cos(\theta_2) - \cancel{\sin(\theta_1)\sin(\theta_2)\cos(\theta_1)} + \sin^2(\theta_1)\cos(\theta_2) + \cancel{\sin(\theta_1)\sin(\theta_2)\cos(\theta_1)} \\ &= \cos(\theta_2)\end{aligned}$$

Euler Lagrange $\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot q_j}\right) - \frac{\partial \mathcal{L}}{\partial q_j} = 0$

with $\mathcal{L} = \sum_i T_i - \sum_i U_i$

$q_j \in \{q_1, q_2\} = \{\theta_1, \theta_2\}$

$$\begin{aligned}T &= \tfrac{1}{2}m_1 v_{c_1}^2 + \tfrac{1}{2}m_2 v_{c_2}^2 + \tfrac{1}{2}I_1\dot\theta_1^2 + \tfrac{1}{2}I_2(\dot\theta_1+\dot\theta_2)^2 \\ &= \tfrac{1}{2}m_1(\dot x_{c_1}^2 + \dot y_{c_1}^2) + \tfrac{1}{2}m_2(\dot x_{c_2}^2 + \dot y_{c_2}^2) + \tfrac{1}{2}I_1\dot\theta_1^2 + \tfrac{1}{2}I_2(\dot\theta_1+\dot\theta_2)^2 \\ &= \tfrac{1}{2}m_1c_1^2\dot\theta_1^2 + \tfrac{1}{2}m_2l_1^2\dot\theta_1^2 + \tfrac{1}{2}m_2c_2^2(\dot\theta_1+\dot\theta_2)^2 + m_2l_1c_2\dot\theta_1(\dot\theta_1+\dot\theta_2)\cos(\theta_2) + \tfrac{1}{2}I_1\dot\theta_1^2 + \tfrac{1}{2}I_2(\dot\theta_1+\dot\theta_2)^2\end{aligned}$$

$$U = -(\vec F_g\cdot\hat y) - (\vec F_g\cdot\hat x) - (\vec T\cdot\hat\theta_1) = m_1 g y_{c1} + m_2 g y_{c2} - T\theta_1 = m_1 g c_1\cos(\theta_1) + m_2 g\left(l_1\cos(\theta_1) + c_2\cos(\theta_1+\theta_2)\right) - T\theta_1$$

$$\begin{aligned}\mathcal{L}(\theta_1,\theta_2,\dot\theta_1,\dot\theta_2,t) = T - U &= \tfrac{1}{2}m_1c_1^2\dot\theta_1^2 + \tfrac{1}{2}m_2l_1^2\dot\theta_1^2 + \tfrac{1}{2}m_2c_2^2(\dot\theta_1+\dot\theta_2)^2 + m_2l_1c_2\overbrace{\dot\theta_1(\dot\theta_1+\dot\theta_2)}^{\dot\theta_1^2+\dot\theta_1\dot\theta_2}\cos(\theta_2) + \tfrac{1}{2}I_1\dot\theta_1^2 + \tfrac{1}{2}I_2(\dot\theta_1+\dot\theta_2)^2 \\ &\quad - \left(m_1gc_1\cos(\theta_1) + m_2g\left(l_1\cos(\theta_1) + c_2\cos(\theta_1+\theta_2)\right) - T\theta_1\right)\end{aligned}$$

$$\begin{aligned}\mathcal{L} &= \tfrac{1}{2}m_1c_1^2\dot\theta_1^2 + \tfrac{1}{2}m_2l_1^2\dot\theta_1^2 + \tfrac{1}{2}m_2c_2^2(\dot\theta_1+\dot\theta_2)^2 + m_2l_1c_2\dot\theta_1(\dot\theta_1+\dot\theta_2)\cos(\theta_2) + \tfrac{1}{2}I_1\dot\theta_1^2 + \tfrac{1}{2}I_2(\dot\theta_1+\dot\theta_2)^2 \\ &\quad - m_1gc_1\cos(\theta_1) - m_2g\left(l_1\cos(\theta_1) + c_2\cos(\theta_1+\theta_2)\right) + T\theta_1\end{aligned}$$

θ_1:

$$\frac{\partial\mathcal{L}}{\partial\theta_1} = 0 + 0 + 0 + 0 + 0 + 0 + m_1gc_1\sin(\theta_1) + m_2gl_1\sin(\theta_1) + m_2gc_2\sin(\theta_1+\theta_2) + T$$

$$\frac{\partial\mathcal{L}}{\partial\dot\theta_1} = m_1c_1^2\dot\theta_1 + m_2l_1^2\dot\theta_1 + m_2c_2^2(\dot\theta_1+\dot\theta_2) + m_2l_1c_2\cos(\theta_2)(2\dot\theta_1+\dot\theta_2) + I_1\dot\theta_1 + I_2\dot\theta_1 + I_2\dot\theta_2$$

$$\frac{d}{dt}\frac{\partial\mathcal{L}}{\partial\dot\theta_1} = m_1c_1^2\ddot\theta_1 + m_2l_1^2\ddot\theta_1 + m_2c_2^2(\ddot\theta_1+\ddot\theta_2) + m_2l_1c_2\left[\cos(\theta_2)(2\ddot\theta_1+\ddot\theta_2) - \sin(\theta_2)\dot\theta_2(2\dot\theta_1+\dot\theta_2)\right] + I_1\ddot\theta_1 + I_2\ddot\theta_1 + I_2\ddot\theta_2$$

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot\theta_1} = m_1 c_1^2 \ddot\theta_1 + m_2 \ell_1^2 \ddot\theta_1 + m_2 c_2^2 (\ddot\theta_1 + \ddot\theta_2) + m_2 \ell_1 c_2 \left[\cos(\theta_2)(2\ddot\theta_1 + \ddot\theta_2) - \sin(\theta_2)\dot\theta_2(2\dot\theta_1 + \dot\theta_2)\right] + I_1\ddot\theta_1 + I_2\ddot\theta_1 + I_2\ddot\theta_2$$

$$\Rightarrow \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot\theta_1} - \frac{\partial \mathcal{L}}{\partial \theta_1} = 0 \Leftrightarrow (m_1 c_1^2 + m_2 \ell_1^2)\ddot\theta_1 + m_2 c_2^2(\ddot\theta_1 + \ddot\theta_2) + 2 m_2 \ell_1 c_2 \cos(\theta_2)\ddot\theta_1 + m_2 \ell_1 c_2 \cos(\theta_2)\ddot\theta_2 - 2 m_2 \ell_1 c_2 \sin(\theta_2)\dot\theta_1\dot\theta_2 - m_2 \ell_1 c_2 \sin(\theta_2)\dot\theta_2^2 +$$

$$+ I_1\ddot\theta_1 + I_2\ddot\theta_1 + I_2\ddot\theta_2 - (m_1 c_1 + m_2 \ell_1) g \sin(\theta_1) - m_2 c_2 g \sin(\theta_1 + \theta_2) = T$$

$$P_1 = m_1 c_1^2 + m_2 \ell_1^2 + I_1 \qquad g_1 = (m_1 c_1 + m_2 \ell_1) g$$

$$P_2 = m_2 c_2^2 + I_2 \qquad g_2 = m_2 c_2 g$$

$$P_3 = m_2 \ell_1 c_2$$

$$(P_1 + P_2 + 2P_3\cos(\theta_2))\ddot\theta_1 + (P_2 + P_3\cos(\theta_2))\ddot\theta_2 +$$

$$-2P_3\sin(\theta_2)\dot\theta_1\dot\theta_2 - P_3\sin(\theta_2)\dot\theta_2^2 - g_1\sin(\theta_1) - g_2\sin(\theta_1+\theta_2) = T$$

$$\theta_1: \begin{bmatrix} P_1 + P_2 + 2P_3\cos(\theta_2) & P_2 + P_3\cos(\theta_2)\end{bmatrix}\begin{bmatrix}\ddot\theta_1 \\ \ddot\theta_2\end{bmatrix} - \begin{bmatrix} P_3\dot\theta_2\sin(\theta_2) & P_3(\dot\theta_1+\dot\theta_2)\sin(\theta_2)\end{bmatrix}\begin{bmatrix}\dot\theta_1 \\ \dot\theta_2\end{bmatrix} - g_1\sin(\theta_1) - g_2\sin(\theta_1+\theta_2) = T$$

$$\theta_2: \frac{\partial \mathcal{L}}{\partial \theta_2} = -m_2 \ell_1 c_2 \dot\theta_1(\dot\theta_1 + \dot\theta_2)\sin(\theta_2) + m_2 g c_2 \sin(\theta_1 + \theta_2)$$

$$\frac{\partial \mathcal{L}}{\partial \dot\theta_2} = m_2 c_2^2(\dot\theta_1 + \dot\theta_2) + m_2 \ell_1 c_2 \dot\theta_1\cos(\theta_2) + I_2(\dot\theta_1 + \dot\theta_2)$$

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot\theta_2} = m_2 c_2^2(\ddot\theta_1 + \ddot\theta_2) + m_2 \ell_1 c_2\left[\ddot\theta_1\cos(\theta_2) - \dot\theta_1\dot\theta_2\sin(\theta_2)\right] + I_2\ddot\theta_1 + I_2\ddot\theta_2$$

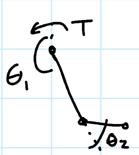
$$\Rightarrow \frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot\theta_2} - \frac{\partial \mathcal{L}}{\partial \theta_2} = 0 \Leftrightarrow (m_2 c_2^2 + m_2 \ell_1 c_2\cos(\theta_2) + I_2)\ddot\theta_1 + (m_2 c_2^2 + I_2)\ddot\theta_2 - \cancel{m_2 \ell_1 c_2 \sin(\theta_2)\dot\theta_1\dot\theta_2}$$

$$+ m_2 \ell_1 c_2 \sin(\theta_2)\dot\theta_1^2 + \cancel{m_2 \ell_1 c_2 \sin(\theta_2)\dot\theta_1\dot\theta_2} - m_2 g c_2 \sin(\theta_1+\theta_2) = 0$$

$$\Leftrightarrow (P_2 + P_3\cos(\theta_2))\ddot\theta_1 + P_2\ddot\theta_2 + P_3\sin(\theta_2)\dot\theta^2 - g_2\sin(\theta_1+\theta_2) = 0$$

$$\theta_2: \begin{bmatrix} P_2 + P_3\cos(\theta_2) & P_2\end{bmatrix}\begin{bmatrix}\ddot\theta_1 \\ \ddot\theta_2\end{bmatrix} + \begin{bmatrix} P_3\dot\theta_1\sin(\theta_2) & 0\end{bmatrix}\begin{bmatrix}\dot\theta_1 \\ \dot\theta_2\end{bmatrix} - g_2\sin(\theta_1+\theta_2) = 0$$

Frictionless Model:



let $P_1 = m_1 c_1^2 + m_2 \ell_1^2 + I_1 \qquad g_1 = (m_1 c_1 + m_2 \ell_1) g$

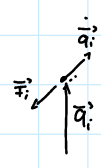
$P_2 = m_2 c_2^2 + I_2 \qquad g_2 = m_2 c_2 g$

$P_3 = m_2 \ell_1 c_2$

$$\begin{bmatrix} P_1 + P_2 + 2P_3\cos(\theta_2) & P_2 + P_3\cos(\theta_2) \\ P_2 + P_3\cos(\theta_2) & P_2\end{bmatrix}\begin{bmatrix}\ddot\theta_1 \\ \ddot\theta_2\end{bmatrix} + \begin{bmatrix} -P_3\dot\theta_2\sin(\theta_2) & -P_3(\dot\theta_1+\dot\theta_2)\sin(\theta_2) \\ P_3\dot\theta_1\sin(\theta_2) & 0\end{bmatrix}\begin{bmatrix}\dot\theta_1 \\ \dot\theta_2\end{bmatrix} - \begin{bmatrix} g_1\sin(\theta_1) + g_2\sin(\theta_1+\theta_2) \\ g_2\sin(\theta_1+\theta_2)\end{bmatrix} = \begin{bmatrix} T \\ 0\end{bmatrix}$$

With Friction $\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot q_r}\right) - \frac{\partial \mathcal{L}}{\partial q_r} - |F_d| = 0$

viscous forces along an axis $\vec{F_i} = -b_i \dot{\vec{q_i}}$



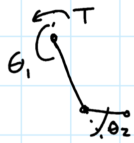
we here have $F_1 = -b_1\dot\theta_1 \quad F_2 = -b_2\dot\theta_2$

i.e. simply add $\begin{bmatrix} b_1 & 0 \end{bmatrix}\begin{bmatrix}\dot\theta_1\end{bmatrix}$ to our model:

we here have $F_1 = -b_1\dot{\theta}_1 \quad F_2 = -b_2\dot{\theta}_2$

i.e. simply add $\begin{bmatrix} b_1 & 0 \\ 0 & b_2 \end{bmatrix}\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$ to our model.

Final Model:



let $P_1 = m_1c_1^2 + m_2\ell_1^2 + I_1 \qquad g_1 = (m_1c_1 + m_2\ell_1)g$

$P_2 = m_2c_2^2 + I_2 \qquad g_2 = m_2c_2 g$

$P_3 = m_2\ell_1 c_2$

$$\boxed{\underbrace{\begin{bmatrix} P_1+P_2+2P_3\cos(\theta_2) & P_2+P_3\cos(\theta_2) \\ P_2+P_3\cos(\theta_2) & P_2 \end{bmatrix}}_{M(\theta_1,\theta_2)}\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \underbrace{\begin{bmatrix} b_1-P_3\dot{\theta}_2\sin(\theta_2) & -P_3(\dot{\theta}_1+\dot{\theta}_2)\sin(\theta_2) \\ P_3\dot{\theta}_1\sin(\theta_2) & b_2 \end{bmatrix}}_{C(\theta_1,\theta_2,\dot{\theta}_1,\dot{\theta}_2)}\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} - \underbrace{\begin{bmatrix} g_1\sin(\theta_1)+g_2\sin(\theta_1+\theta_2) \\ g_2\sin(\theta_1+\theta_2) \end{bmatrix}}_{G(\theta_1,\theta_2)} = \begin{bmatrix} T \\ 0 \end{bmatrix}}$$

We add a motor; with k_m, τ_e s.t. $\boxed{\dot{T} = -\frac{1}{\tau_e}T + \frac{k_m}{\tau_e}v_{in}}$

C Detailed White Box model

Initial Position	$E_{RMS}(\theta_2)$ [%]
$\frac{\pi}{4}$ 1	7.1286
$\frac{\pi}{4}$ 2	0.90117
$\frac{\pi}{4}$ 3	1.4394
$\frac{-3\pi}{4}$ 1	5.3113
$\frac{-5\pi}{6}$ 1	37.671
$\frac{-5\pi}{6}$ 2	1.2822
$\frac{10\pi}{12}$ 1	7.5637
$\frac{3\pi}{4}$ 1	2.5456

Initial Position	$E_{RMS}(\theta_2)$ [%]
$\frac{\pi}{4}$ 4	5.4421
$\frac{-3\pi}{4}$ 2	1.8759
$\frac{10\pi}{12}$ 2	20.964
$\frac{3\pi}{4}$ 2	0.5316

Table 6: Error on Training and Test set for the estimation of parameters of the second link

Training Signals	$E_{RMS}(\theta_1)$ [%]
Step input 1	0.21469
Step input 2	6.6914

Testing Signals	$E_{RMS}(\theta_1)$ [%]
Step input 3	0.13807
Step input 4	6.9112

Table 7: Error on Training and Test set for the estimation of parameters of the motor

Training Signals	$E_{RMS}(\theta_1)$ [%]	$E_{RMS}(\theta_2)$ [%]
Waveform 1	1.3345	1.4849
Waveform 2	2.0586	7.3501
Waveform 4	0.16801	0.16691
Chirp signal 1	0.72967	0.63162
Chirp signal 3	0.73032	6.6476
Chirp signal 4	0.63883	0.16727
Sine ramp 2	2.8851	4.8995
Sine ramp 3	24.532	20.999
Sine lin 1	0.87102	45.618

Testing Signals	$E_{RMS}(\theta_1)$ [%]	$E_{RMS}(\theta_2)$ [%]
Waveform 3	0.45933	0.5184
Chirp signal 2	0.98477	2.7409
Sine ramp 1	14.993	37.059
Sine lin 2	3.3646	6.6995

Table 8: Error on Training and Test set for the estimation of parameters of the first link

Parameters	**Theoretical**	**Estimated**
l1 [m]	0.1	0.0897
l2 [m]	0.1	0.1
m1 [kg]	0.125	0.2
m2 [kg]	0.05	0.087972
c1 [m]	-0.04	0.01
c2 [m]	0.06	0.07
I1 [kg.m^2]	0.074	0.01076
I2 [kg.m^2]	0.00012	0.0001
b1 [kg.s^{-1}]	4.8	5.33518
b2 [kg.s^{-1}]	0.0002	0.00004
k_m [N.m]	50	39.2954
τ_e [s]	0.03	0.01

Table 9: Final averaged parameters

D Matlab Symbolic Linearization

```
close all;
clear;

%% MODEL LINEARIZATION
syms P1 P2 P3 g1 g2 b1 b2
syms th1 th2 dth1 dth2 ddth1 ddth2
syms T te km vin

M = [P1+P2+2*P3*cos(th2) P2+P3*cos(th2);
     P2+P3*cos(th2) P2];
C = [b1-P3*dth2*sin(th2) -P3*(dth1+dth2)*sin(th2);
     P3*dth1*sin(th2) b2];
G = [-g1*sin(th1)-g2*sin(th1+th2);
     -g2*sin(th1+th2)];
iM = inv(M);

%symbolic state of the system
state = [dth1;dth2;th1;th2;T];
%non-linear system function
f = [iM*[T;0]-iM*C*[dth1; dth2]-iM*G;
     dth1; dth2; -T/te+km*vin/te];

%% LINEARIZATION POINT

%POSITIO   |    x_eq         |   u_eq
%up up      [0;0;0;0;0]            0
%down up    [0;0;-pi;-pi;0]        0
%up down    [0;0;0;-pi;0]          0
%down down  [0;0;-pi;0;0]          0

x_eq = [0;0;-pi;-pi;0];
u_eq = 0;

%Linearized symbolic state space model
lA = [diff(f, dth1) diff(f, dth2) diff(f, th1) diff(f, th2) diff(f, T)];
lB = diff(f, vin);

clear M C G iM
%% Parameters initialization

g = 9.81;
l1 = 0.0897;
l2 = 0.1;
m1 = 0.2;
m2 = 0.087972;
c1 = 0.01;
c2 = 0.07;
I1 = 0.01076;
I2 = 0.0001;
```

```
b1 = 5.33518;
b2 = 4e-05;
km = 39.2954;
te = 0.01;

%% Evaluation of the Continuous Time Linear System

%---Model paramters substitution
P1 = m1*c1^2 + m2*l1^2 + I1;
P2 = m2*c2^2 + I2;
P3 = m2*l1*c2;
g1 = (m1*c1+m2*l1)*g;
g2 = m2*c2*g;

% Assign physical parameters to linear system
nA = subs(lA);
nB = subs(lB);

% Substitute the linearization point
nA0 = subs(nA, state, x_eq);
nA0 = subs(nA0, vin, u_eq);
nB0 = subs(nB, state, x_eq);
nB0 = subs(nB0, vin, u_eq);

% Final CT Linearized system
A = eval(nA0);
B = eval(nB0);
C = [0 0 1 0 0;
      0 0 0 1 0];
D = 0;

clear lA lB nA nB nA0 nB0 f
clear ddth1 ddth2 dth1 dth2 th1 th2
clear vin T state P1 P2 P3 g1 g2

%% Continuous Time Linearized State Space Model
csys = ss(A, B, C, D);
csys.InputName = 'vin';
csys.OutputName = ['th1'; 'th2'];
csys.StateName = ["dth1";"dth2";"th1";"th2";"T"];
csys.Name = 'RotPend';
%% CONTROLLABILITY

fprintf("\n\nControlability : \n");
disp(rank(ctrb(csys)));
fprintf("\n\nStability : \n");
disp(eig(A));
```

E Output feedback maths

LQR maths

vendredi, 4 mars 2022 17:19

linearization around $\vec{x}_{eq} = (0,0,0,0)^T$ $u_{eq} = 0$!

$\dot{\vec{x}} = A\vec{x} + B\vec{u}$ from $\dot{\vec{x}} = \vec{f}(\vec{x}, u) = \vec{f}((\dot{\theta}_1, \dot{\theta}_2, \theta_1, \theta_2, T), V_{in})$

$$A = \frac{\partial \vec{f}}{\partial \vec{x}}\Big|_{\substack{\vec{x}=\vec{x}_{eq}=\vec{0} \\ u=u_{eq}=0}} \qquad B = \frac{\partial \vec{f}}{\partial u}\Big|_{\substack{\vec{x}=x_{eq}=0 \\ u=u_{eq}=0}} \qquad C = \frac{\partial \vec{g}}{\partial \vec{x}} = \frac{\partial \vec{\theta}}{\partial \vec{x}} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$D = \frac{\partial \vec{g}}{\partial u} = [0]$$

$$\begin{matrix} \dot{\vec{x}} = A\vec{x} + B\vec{u} \\ \vec{y} = C\vec{x} + D\vec{u} \end{matrix} \Rightarrow \begin{cases} s\mathcal{L}\{\vec{x}\} - \vec{x}_0 = A\mathcal{L}\{\vec{x}\} + \mathcal{L}\{B\vec{u}\} \\ \mathcal{L}\{\vec{y}\} = C\mathcal{L}\{\vec{x}\} + D\mathcal{L}\{\vec{u}\} \end{cases}$$

$$\Rightarrow \begin{cases} (sI - A)\mathcal{L}\{\vec{x}\} = \mathcal{L}\{B\vec{u}\} + \vec{x}_0 \\ \mathcal{L}\{\vec{y}\} = C\mathcal{L}\{\vec{x}\} + D\mathcal{L}\{\vec{u}\} \end{cases}$$

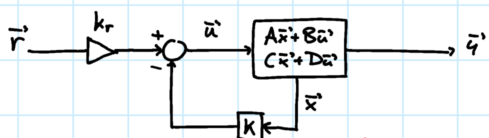
$$\text{TF: } \frac{\mathcal{L}\{out\}}{\mathcal{L}\{in\}}\Big|_{C.I.=0} = \frac{\mathcal{L}\{\vec{y}\}}{\mathcal{L}\{\vec{u}\}}\Big|_{\vec{x}_0=\vec{0}} = \frac{C(sI-A)^{-1}B\mathcal{L}\{\vec{u}\} + \vec{0} + D\mathcal{L}\{\vec{u}\}}{\mathcal{L}\{\vec{u}\}}$$

$$\Rightarrow G(s) = C(sI-A)^{-1}B + D$$

if $s \in eig(A)$, $sI - A$ becomes singular and therefore a pole of the transfer function

$\rightarrow$ system is stable if $eig(A) \in \mathbb{R}_{\geq 0}$

<u>full state feedback</u>



$\rightsquigarrow \vec{u} = -K\vec{x} + k_r\vec{r}$

$$\begin{cases} \dot{\vec{x}} = A\vec{x} + B(-K\vec{x} + k_r\vec{r}) = \overbrace{(A - BK)}^{A'}\vec{x} + \overbrace{Bk_r}^{B'}\vec{r} \\ \vec{y} = C\vec{x} + D(-K\vec{x} + k_r\vec{r}) = \underbrace{\overbrace{(C - DK)}^{C'}\vec{x} + \overbrace{Dk_r}^{D'}\vec{r}}_{\text{new LTI system with } \vec{r} \text{ as input!}} \end{cases}$$

$$\text{whose TF} = G'(s) = \frac{\mathcal{L}\{\vec{y}\}}{\mathcal{L}\{\vec{r}\}} = C'(sI - A')^{-1}B' + D'$$
$$= (C-DK)(sI - (A-BK))^{-1}Bk_r + Dk_r$$

$\rightarrow$ poles are $eig(A - BK)$

$\rightarrow$ use K to stabilize the system by placing eigenvalues of $A-BK$!

<u>observer</u>

$\tilde{\vec{x}}$ = estimated state = $(\dot{\tilde{\theta}}_1, \dot{\tilde{\theta}}_2, \tilde{\theta}_1, \tilde{\theta}_2, \tilde{T})^T$

observer

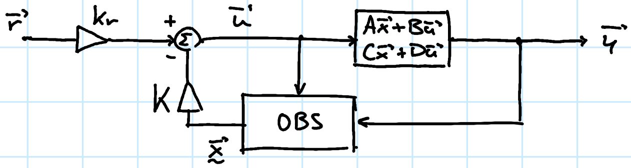
$\underset{\sim}{\vec{x}}$ = estimated state = $(\underset{\sim}{\dot{\theta}_1}, \underset{\sim}{\dot{\theta}_2}, \underset{\sim}{\theta_1}, \underset{\sim}{\theta_2}, \underset{\sim}{I})^T$

$\vec{y}, \vec{u}$ = available output / input signals

$\rightarrow \dot{\underset{\sim}{\vec{x}}} = F\underset{\sim}{\vec{x}} + G\vec{u} + H\vec{y}$ choose F, G, H s.t.

$$\dot{\underset{\sim}{\vec{x}}} = A\underset{\sim}{\vec{x}} + B\vec{u} + L(\underbrace{\vec{y} - (C\underset{\sim}{\vec{x}} + D\vec{u})}_{\text{estimation error}})$$

and we build the system:



we have:

does it work with observer feedback instead of state feedback?

$$\begin{cases} \dot{\vec{x}} = A\vec{x} + B\vec{u} \\ \vec{y} = C\vec{x} + D\vec{u} \\ \vec{u} = -K\underset{\sim}{\vec{x}} + k_r\vec{r} \\ \dot{\underset{\sim}{\vec{x}}} = A\underset{\sim}{\vec{x}} + B\vec{u} + L(\vec{y} - (C\underset{\sim}{\vec{x}} + D\vec{u})) \end{cases}$$

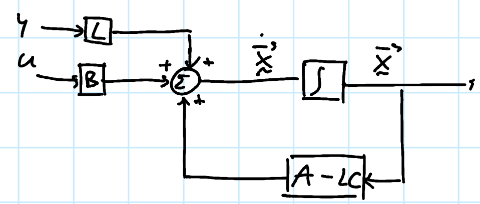
base plant model (first two equations)

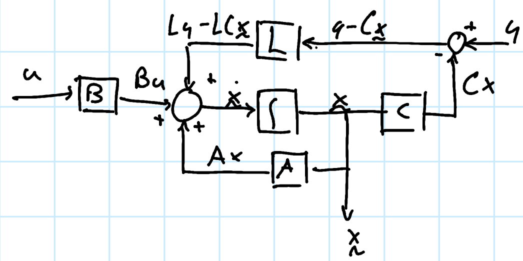
OBS feedback

OBS

OBS $\dot{\underset{\sim}{\vec{x}}} = A\underset{\sim}{\vec{x}} + Bu + L(\vec{y} - C\underset{\sim}{\vec{x}})$

$\dot{\underset{\sim}{\vec{x}}} = (A - LC)\underset{\sim}{\vec{x}} + Bu + L\vec{y}$





by 1,3:

$$\dot{\vec{x}} = A\vec{x} - BK\underset{\sim}{\vec{x}} + Bk_r\vec{r} \qquad (*)$$

define the observer error : $\vec{e} = \vec{x} - \underset{\sim}{\vec{x}}$

$$\begin{aligned} \dot{\vec{e}} &= \dot{\vec{x}} - \dot{\underset{\sim}{\vec{x}}} = A\vec{x} + \cancel{B\vec{u}} - A\underset{\sim}{\vec{x}} - \cancel{B\vec{u}} - L(\vec{y} - C\underset{\sim}{\vec{x}} - D\vec{u}) \\ &= A(\vec{x} - \underset{\sim}{\vec{x}}) - L(\vec{y} - C\underset{\sim}{\vec{x}} - D\vec{u}) \\ &= A\vec{e} - L(C\vec{x} + \cancel{D\vec{u}} - C\underset{\sim}{\vec{x}} - \cancel{D\vec{u}}) \\ &= A\vec{e} - LC\underbrace{(\vec{x} - \underset{\sim}{\vec{x}})}_{\vec{e}} = (A - LC)\vec{e} \end{aligned}$$

$$\Rightarrow \dot{\vec{e}} = (A - LC)\vec{e} \qquad ②$$

and (*) becomes $\dot{\vec{x}} = A\vec{x} - BK(\vec{x} - \vec{e}) + Bk_r\vec{r}$

$$= (A - BK)\vec{x} + BK\vec{e} + Bk_r\vec{r} \qquad ①$$

so we have

$$\frac{d}{dt}\begin{bmatrix} \vec{x} \\ \vec{e} \end{bmatrix} = \begin{cases} (A - BK)\vec{x} + BK\vec{e} + Bk_r\vec{r} \\ (A - LC)\vec{e} \end{cases}$$

and finally $\vec{y} = C\vec{x} + D\vec{u} = C\vec{x} + D(-K\underset{\sim}{\vec{x}} + k_r\vec{r})$

$$\begin{aligned} &= C\vec{x} - DK(\vec{x} - \vec{e}) + Dk_r\vec{r} \\ &= (C - DK)\vec{x} + DK\vec{e} + Dk_r\vec{r} \end{aligned}$$

Final SS model

$$\frac{d}{dt}\begin{bmatrix} \vec{x} \\ \vec{e} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix}\begin{bmatrix} \vec{x} \\ \vec{e} \end{bmatrix} + \begin{bmatrix} Bk_r \\ 0 \end{bmatrix}\vec{r}$$

$$\frac{d}{dt}\begin{bmatrix}\vec{x}\\ \vec{e}\end{bmatrix} = \begin{bmatrix}A-BK & BK\\ 0 & A-LC\end{bmatrix}\begin{bmatrix}\vec{x}\\ \vec{e}\end{bmatrix} + \begin{bmatrix}Bk_r\\ 0\end{bmatrix}\vec{r}$$

$$\vec{y} = \begin{bmatrix}C-DK & DK\end{bmatrix}\begin{bmatrix}\vec{x}\\ \vec{e}\end{bmatrix} + \begin{bmatrix}Dk_r\\ 0\end{bmatrix}\vec{r}$$

a linear system with state $\begin{bmatrix}\vec{x}\\ \vec{e}\end{bmatrix} = \begin{bmatrix}\vec{x}\\ \vec{x}-\hat{x}\end{bmatrix}$

input $\vec{r}$ and output $\vec{y}$

→ we want to stabilize that system to $\begin{bmatrix}0\\0\end{bmatrix}$ i.e. observer error $=0$!

→ therefore the transfer function:

$$G(s) = C^*(sI - A^*)^{-1}B^* + D^*$$

$$= \begin{bmatrix}C-DK & DK\end{bmatrix}\left(sI - \begin{bmatrix}A-BK & BK\\ 0 & A-LC\end{bmatrix}\right)^{-1}\begin{bmatrix}Bk_r\\ 0\end{bmatrix} + \begin{bmatrix}Dk_r\\ 0\end{bmatrix}$$

→ poles are eigenvalues of $A^* = \begin{bmatrix}A-BK & BK\\ 0 & A-LC\end{bmatrix}$

recall $M^{-1} = \frac{\operatorname{adj}(M)}{\det(M)}$ → poles are zeros of determinant

→ zeros of $\det(sI-A)$ = eigenvalues

but $\det(sI - A^*) = \underbrace{\det(sI-(A-BK))\det(sI-(A-LC))}_{}- \cancel{\det(BK)}\det(0)$

the 2 characteristic polynomials are INDEPENDENT → use K to place system eigenvalues and then L to place faster observer eigenvalues!

OUR CASE

$$\begin{cases}\dot{\vec{x}} = A\vec{x} + B\vec{u}\\ \vec{y} = C\vec{x} + D\vec{u}\\ \vec{u} = -K\hat{x} + k_r\vec{r}\\ \dot{\hat{x}} = A\hat{x} + B\vec{u} + L(\vec{y} - (C\hat{x} + D\vec{u}))\end{cases} \longrightarrow \begin{cases}\dot{\vec{x}} = A\vec{x} + Bu\\ \vec{y} = C\vec{x}\\ \vec{u} = -K\hat{x}\\ \dot{\hat{x}} = (A-LC)\hat{x} + Bu + L\vec{y}\end{cases}$$

$\vec{x} = (\dot{\theta}_1, \dot{\theta}_2, \theta_1, \theta_2, T)^T$

$\vec{y} = \begin{bmatrix}\theta_1\\ \theta_2\end{bmatrix}$

use K = place(A, B, eigs1)
and L = place(A', C', eigs2).'

the state space is simply a model for the dynamics we're trying to control but we can't access the full state inside, since it is a physical plant and not a simulated one.

the observer and controller make use of that model to output something we can inded access since those are computed. The estimated state and the control input.