KU Leuven

Computer gestuurde regeltechnieken

Practical session

Case study: RIP

Authors:

Willem Melis	r0348639
Maarten Sprengers	r0256246

Professor: Bart De Moor
Assistants: Oscar Mauricio Agudelo
Supinya Piampongsant

2016 - 2017

May 30, 2017

Contents

1 Introduction

The goal of this practical session is to stabilize a rotary inverted pendulum witch is of unstable nature. Further we will investigate the effect of some design parameters on the controller. In the first stadium, Simulink is used for tuning the controller. In a next stadium the practical set-up is used to verify the Simulink model and to gain some practical experience.

The appendix contains a brief description of how to use the matlab files to redo the simulated experiments.

2 Model and open loop analysis

2.1 Model analysis

The continuous state space model where $u = V$ (voltage on the motor) and x=[θ α $\dot{\theta}$ $\dot{\alpha}$] is theoretical derived in the assignment and leads to the following system. The system can only directly measure θ and α, so these determine the matrix C and D.

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 40.7 & -12.2 & 0 \\ 0 & 38.6 & -4.7 & 0 \end{bmatrix} B = \begin{bmatrix} 0 \\ 0 \\ 23.3 \\ 8.3 \end{bmatrix} C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} D = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

2.2 Open loop analysis

The system contains 1 unstable pole at about 5.3 (see figure 1), this obviously means that the system is unstable. The rank of the controllability matrix is 4, the smallest singular value of the controllability matrix is 1.7. So the system is controllable, controllability is necessary if you want to control a system. The observability matrix is of rank 4 which with a smallest singular value of 1. So the system is observable. If a system is observable and controllable than it is also the minimal realisation of the system. This means the system with the minimal numbers of states possible that fits the given behaviour. If the system is controllable then it's stabilizable and if its observable then it's also detectable which means that if there are unstable states, they you can at least detect unstability.

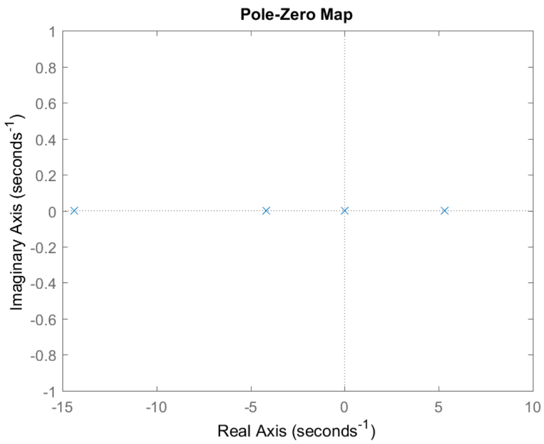


Figure 1: plot of poles and zeros

2.3 Control goals

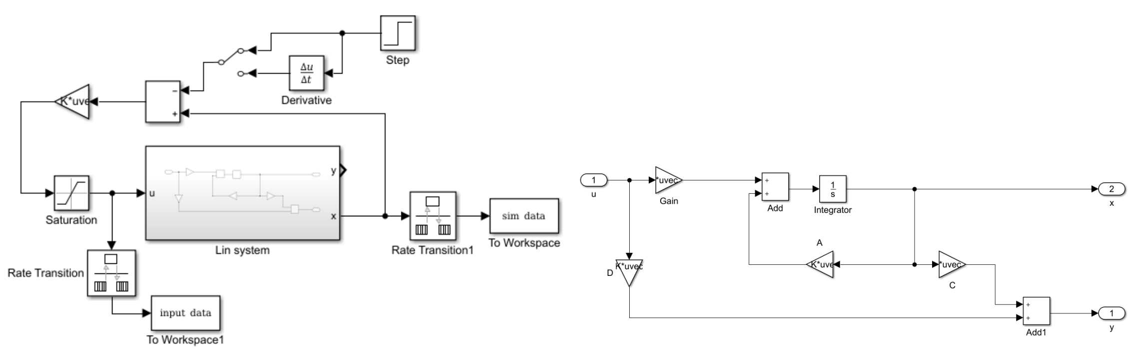
When designing a controller tradeoffs have to be made.If it would be possible to create a fast,robust,.. controller then this would be the best option. However in this report robustness was preferred over speed. However the robustness comes at a price, if robustness is preferred then the controller will lose speed.

So the first priority when designing this controller was to make a stable system. After stabilization is reached, we want the system to be rather robust, it should not be very fast as this will result in a nervous system. We would like that the system can track setpoints up to some step-responses. But our objective is more a robustness system than a fast one.

3 First closed loop

3.1 Simulink

The closed loop diagram is shown in figure 2. The block's that are used are: vector addition, matrix multiplication, saturation block, export data blocks, samplers and in the subsystem some addition, multiplication and integration blocks.



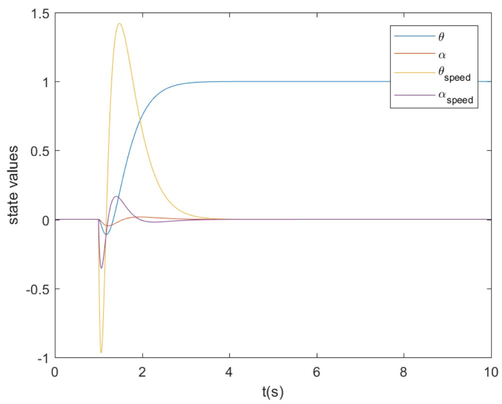
(a) diagram (b) linear system

Figure 2

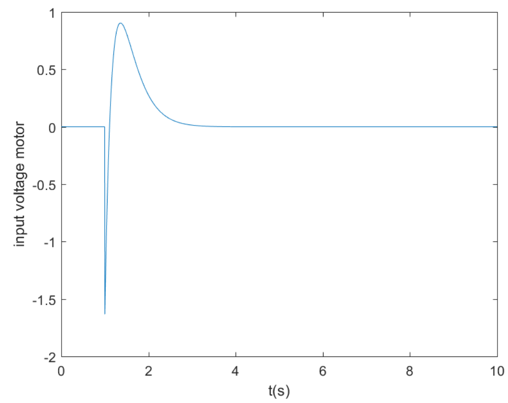
3.2 Determine the values of Q and R

With default Q and R

$$Q = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} R = 1.5$$



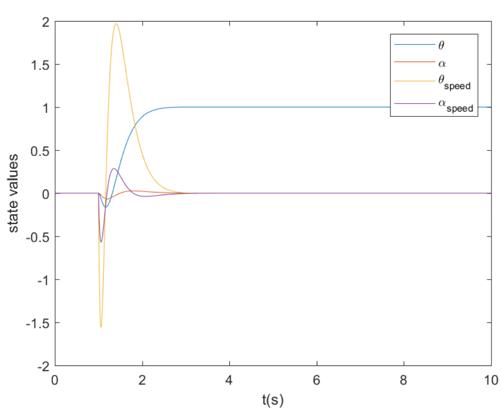
(a) plot of the states in function of time



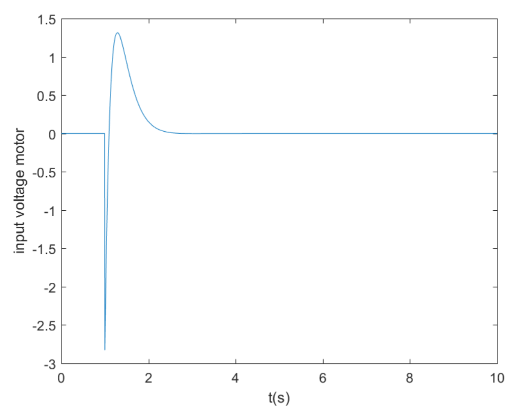
(b) plot of the signal send to the motor in function of time

Figure 3: RIP with default values for Q and R

After evaluating a step response we think that the system can react faster because the limits of the motor aren't reached yet. Therefore we lower $R = 0.5$. In figure 4 you can see that the response is faster and the input-voltage of the motor is higher.



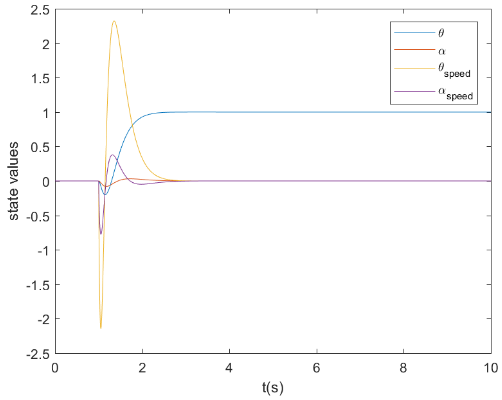
(a) plot of the states in function of time



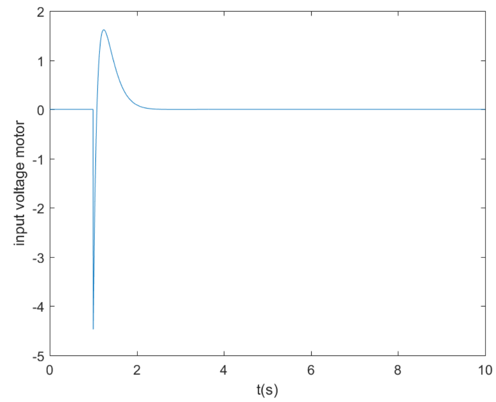
(b) plot of the signal send to the motor in function of time

Figure 4: RIP lower R

When we see the step response we think that by adding some weight to θ that the system becomes still a bit faster. So by changing the weight of θ from 4 to 10 the step response becomes indeed faster. The effect can be seen in figure 5.



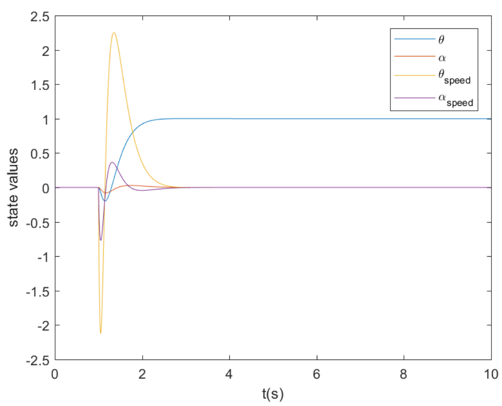
(a) plot of the states in function of time



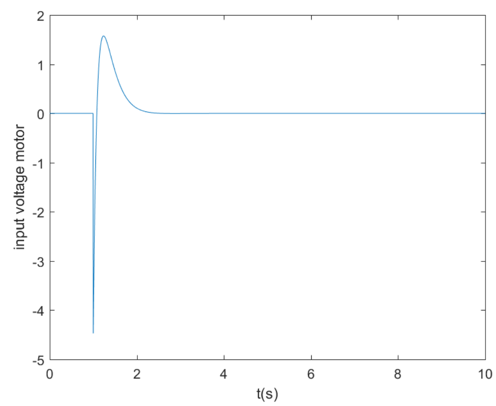
(b) plot of the signal send to the motor in function of time

Figure 5: RIP with increased θ

For increasing the stability we now increase the weight of α from 20 to 60. The effect on the speed is minimal.



(a) plot of the states in function of time

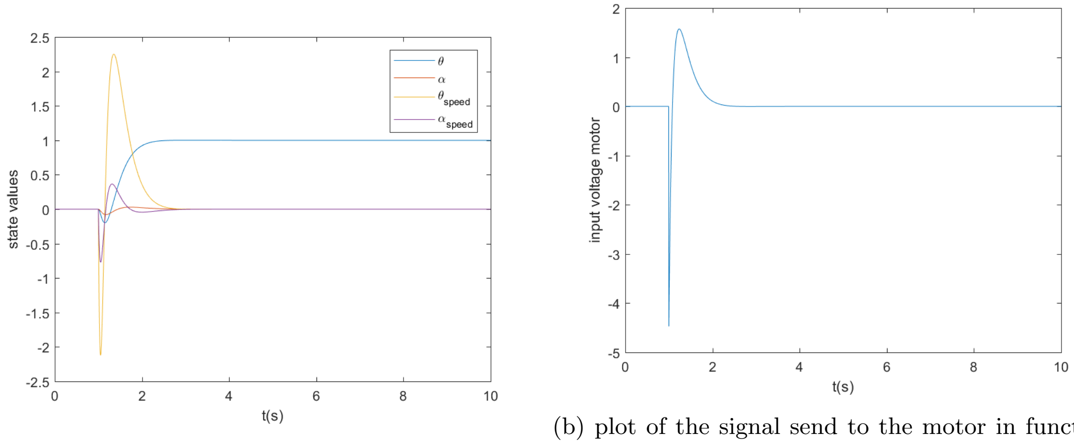


(b) plot of the signal send to the motor in function of time

Figure 6: RIP with increased α

Because we don't like zeros on the diagonal of the weighting matrix we changed from the beginning the angular velocity weights to 0.1. As last step we've set the weight of $\dot{\alpha}$ to 1. The final values for Q and R are:

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 60 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} R = 0.5$$



(a) plot of the states in function of time

(b) plot of the signal send to the motor in function of time

Figure 7: RIP increase $\dot{\alpha}$

3.3 Simulated setpoint change

A step of $\theta = 1$ is performed in figure 7. The motor input stays between the $-5V$ and $+5V$ and there is no overshoot or oscillation when the set-point is reached . You can see that the system is a non-minimal-phase system because there is first a negative response before the system goes to the reference signal.

4 Second closed-loop simulation

4.1 Simulink diagram

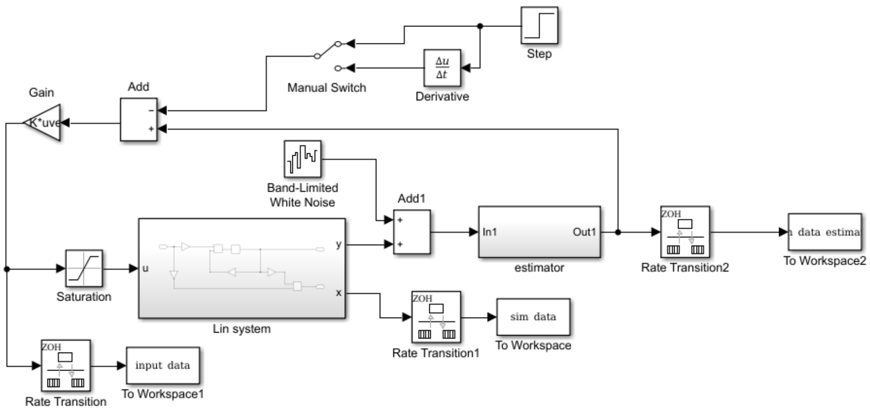
Figure 8a contains the Simulink diagram for the second closed loop system. Just as with the Simulink diagram in the previous section a subsystem was used for the linear system(figure 8c). This time however the internal states where not used by the controller, but as a way to measure how good the estimation of the state is.

The estimator(figure 8b) contains a simple low pass filter on frequency f_c. The filter is followed by a simple discrete finite difference operator. This will result in the measurement of $\dot{\alpha}$ and $\dot{\theta}$. θ and α itself can be measured directly.

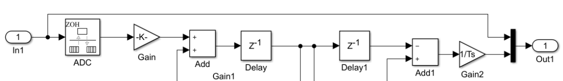
The controller might produce inputs on the motor of over 5 voltage. This obviously cannot be allowed to enter the linear system. That is why the saturation element was used, the physical motors voltage limit is between $\pm 5V$, and so this is also the maximum and minimum voltage used in the saturation element.

The sample frequency was set at $f_s = 200Hz$ which is the same as with the physical system. The estimator contains an ADC which has an f_s of 200Hz and so the controller will only change its outputs at a frequency of 200Hz.

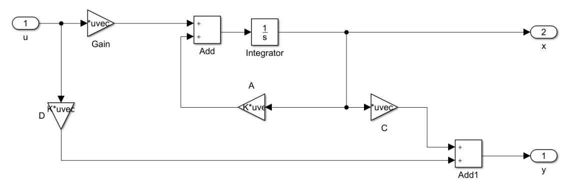
The noise was added after the linear system and before the estimator. This means that the estimated states will contain noise. The power in noise is equal to the variance of the measured noise. This is further explained in the next section. Where the actual measurements are taking place, there it will become clear that the noise is close to Gaussian white noise.



(a) main diagram simulink



(b) estimator

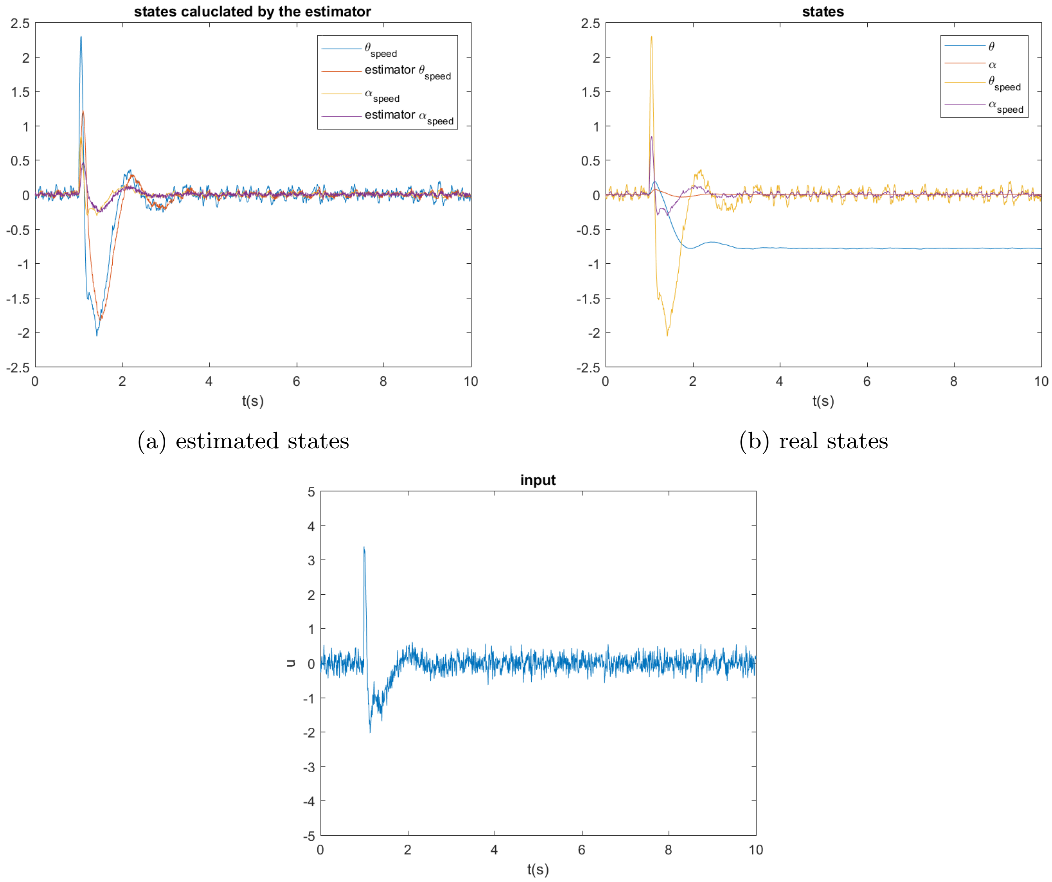


(c) linear system

Figure 8: simulink diagrams

4.2 Simulation with noise

A first approach to the simulation($f_c = 2Hz$) with noise is to try out some parameters and see what happens. A reasonable amount of noise seems to be with a noise of power=10^{-8}. If the noise is much higher then this, then controlling the process becomes very hard. The results of these simulations are displayed in figure 9. As this is not a very scientific way of simulating, no real conclusions can be made from these simulations.



(a) estimated states

(b) real states

(c) input to motor

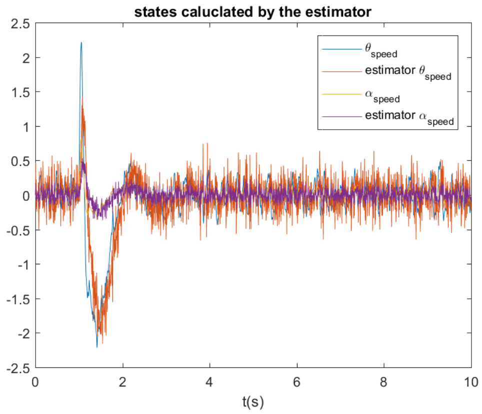
Figure 9: simulation results with a noise with power=10^{-8}

4.3 Simulation with noise estimated from experiments

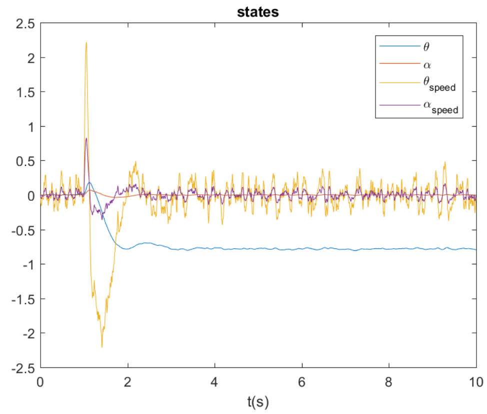
A more scientific way of simulating with noise is to estimate the power of the noise from measurements. As explained in the next section, measurements where taking when the controller and motor where idle. These measurements contain very little to no model behavior and so contains mostly noise. f_c was set to 2Hz and measurements where taken before and after the low pass filter in the estimator.

The variance of these measurements on α and θ where $1.96 \cdot 10^{-7}$ for α and $1.4 \cdot 10^{-6}$ for θ. As the variance is the power in the signal ($power = \frac{u_{eff}^2}{R} = \frac{stdev^2}{1\Omega} =$ variance) there 2 values are taken for the simulation noise.

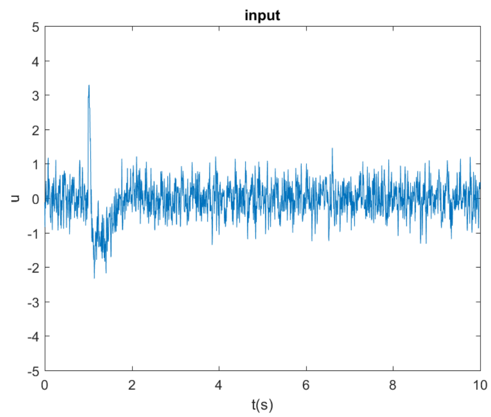
The resulting simulations are displayed in figure 10 and are quiet pleasing. There seems no need to adjust Q and R even further. The initial guess of the noise seems to be quiet accurate even tough it was almost just a gamble. The problem that arises with the estimators is that there is overshoot and a longer settling time.



(a) estimated states



(b) real states

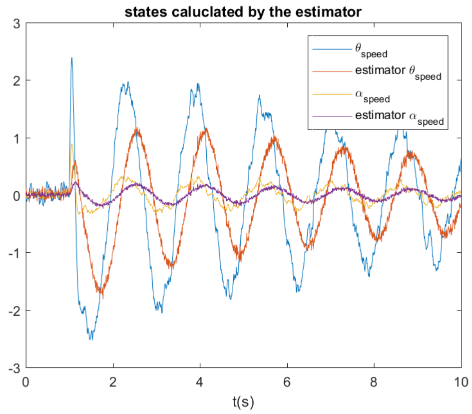


(c) input to motor

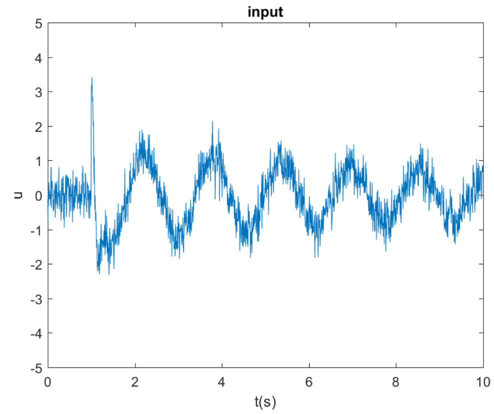
Figure 10: simulation results with noise derived from experiments

4.4 Investigating the role of the cut-off frequency

Most of the noise is high frequency noise while the model dynamics are in the low frequency area. This offers an opportunity to get rid of most of the noise by filtering. A low-pass filter can keep the slow model dynamics and remove the high frequency noise. If the cut-off frequency of this filter is chosen too high then too much noise is allowed to get to the estimator and the controller output will contain a lot of noise (figure 12). If the cut-off frequency is chose too low then part of the model dynamics is filtered out. The controller will not be able to properly control the process as can be seen on figure 11.

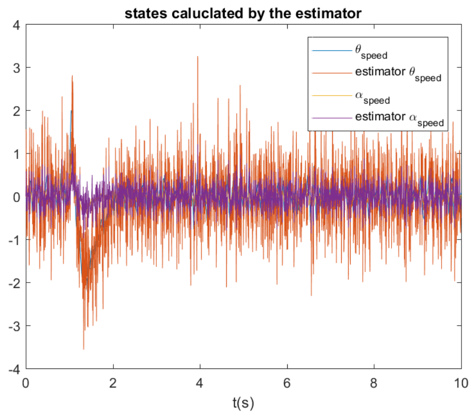


(a) estimated states

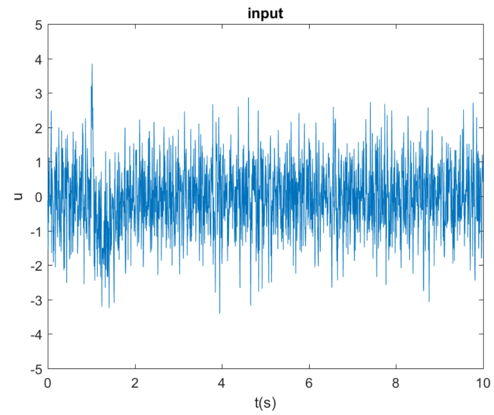


(b) input to motor

Figure 11: simulation results with noise derived from experiments, $f_c = 0.5Hz$



(a) estimated states



(b) input to motor

Figure 12: simulation results with noise derived from experiments, $f_c = 10Hz$

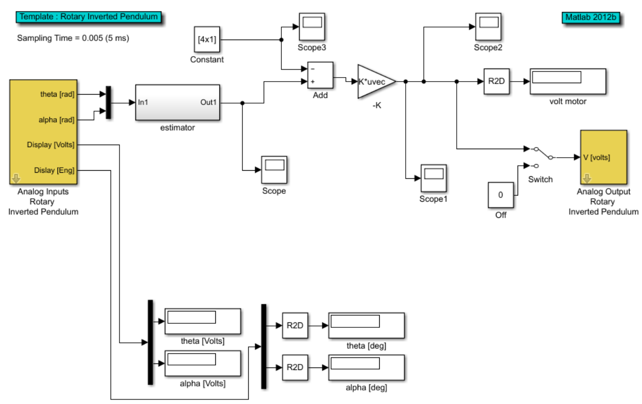
5 Experiments

4 big experiments have been executed.

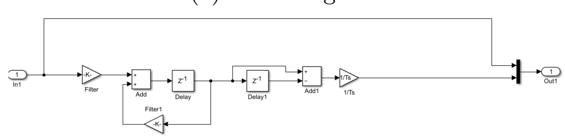
- noise analysis
- tracking, manually changing the set points
- tracking using sine/stair functions
- disturbances

5.1 Simulink

Figure 13 contains the Simulink diagram used for the measurements. The same estimator subsystem as in the previous section was used. With a cut-off frequency of $f_c = 2Hz$, the saturation block is not necessary here. As Simulink itself will limit the output to the motor.



(a) main diagram



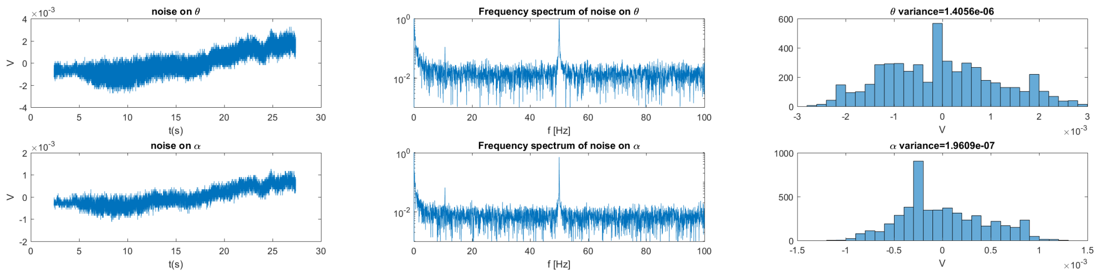
(b) estimator

Figure 13: simulink diagram

5.2 Noise Analysis

θ and α are measured before the low pass filter (zero input), the signal measured is considered noise as displayed in figure 14a. Instead of looking at the noise in function of time its very useful to look at the frequency spectrum and the histogram as is done in figure 14b and figure 14c. From the histogram we can conclude that the noise is Gaussian. In the frequency spectrum you can see that there is a dominant low frequency band and a dominant frequency at 50Hz (net frequency). The rest of the spectrum is more or less flat so Gaussian noise is a good assumption.

The variance of the noise can be computed and then used in the simulation. As the variance is the power in the signal $power = \frac{u_{eff}^2}{R} = \frac{stdev^2}{1\Omega}$= variance

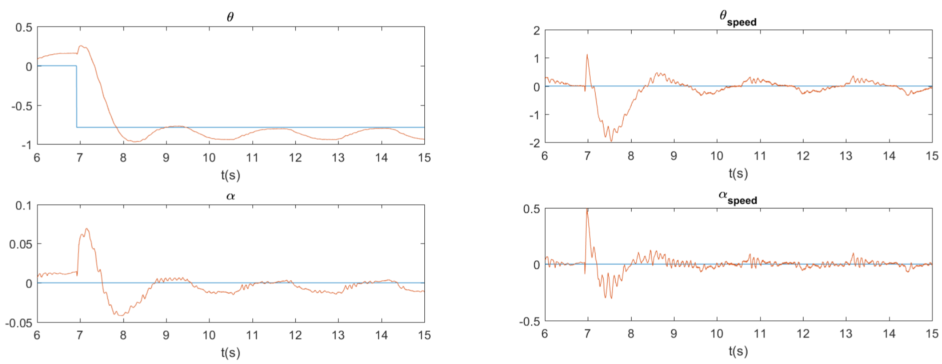


(a) timeplot of noise on θ and α (b) frequency spectrum noise θ α (c) histogram of the noise

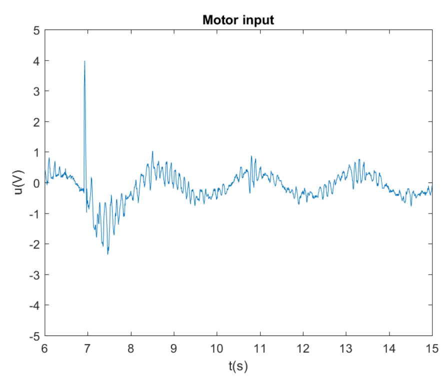
Figure 14: noise measurement of θ and α

5.3 Step response

A step response of $-\frac{\pi}{4}$ is preformed in figure 15. Compared to the step response in figure 9 there's a continuous oscillation around the set-point. Both settling-times are similar, around 1 sec. There is a small steady state error that's probably due to the fact that the linearizion was made around $\theta = 0$. In figure 16 a triangle wave was given as reference input. There is some delay and the overshoot is due to the same problem as the steady state error, combined with the delay.

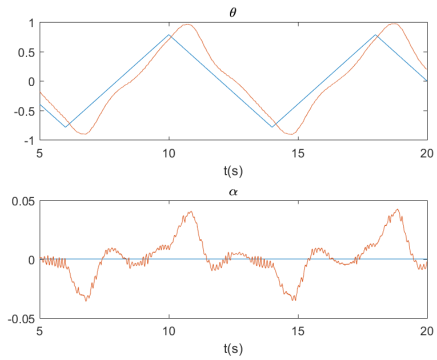


(a) states θ and α (b) states $\dot{\theta}$ and $\dot{\alpha}$

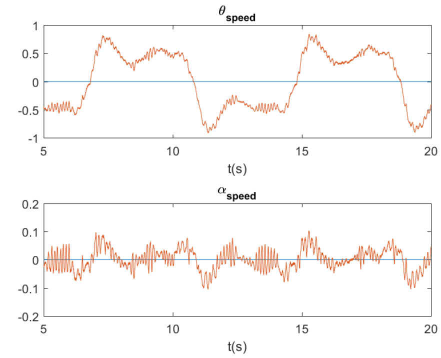


(c) motor input

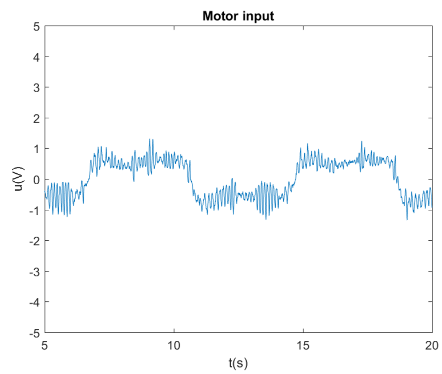
Figure 15: step response



(a) states θ and α



(b) states $\dot{\theta}$ and $\dot{\alpha}$

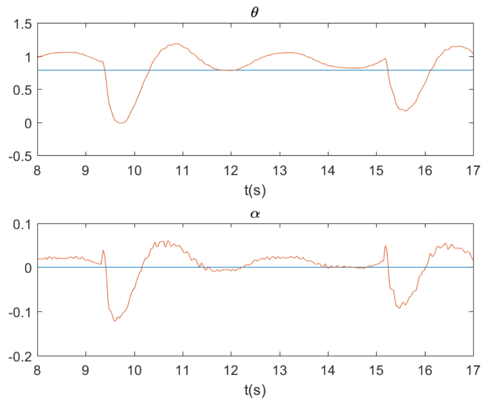


(c) motor input

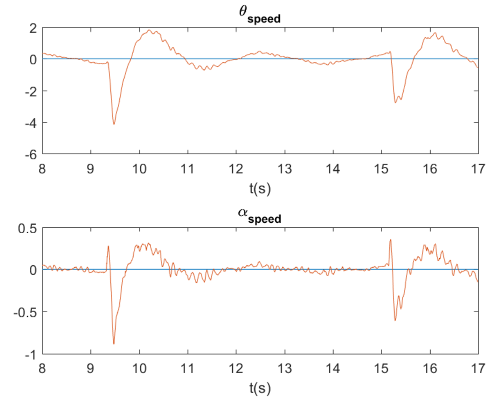
Figure 16: tracking with a triangular wave

5.4 Disturbances

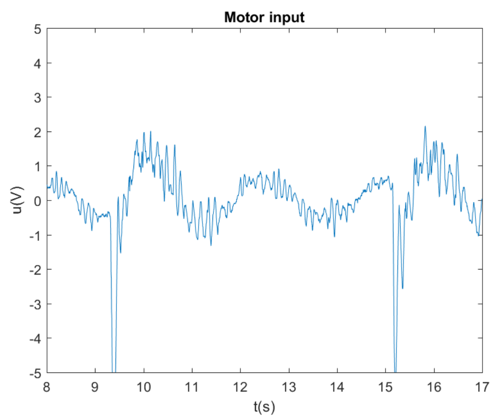
In figure 17 you can see two disturbance rejection test, we think that the inverted pendulum is quite stable, because it reacts fast on the disturbance so is in less than 2 sec in its normal oscillating regime.



(a) states θ and α



(b) states $\dot{\theta}$ and $\dot{\alpha}$

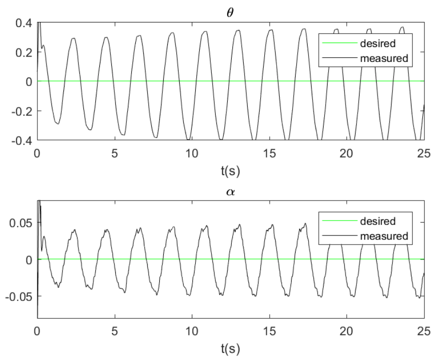


(c) motor input

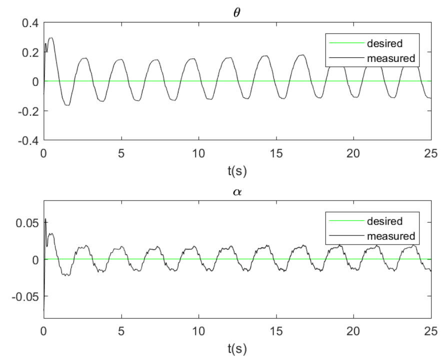
Figure 17: disturbance rejection test

5.5 The role of the cut-off frequency of the low pass filter

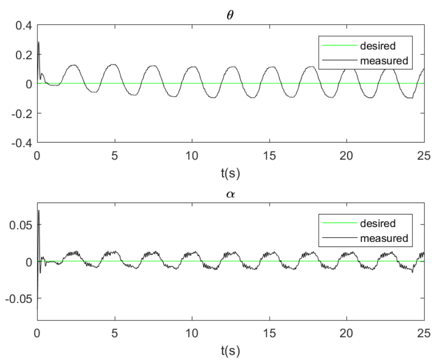
When the cut-off frequency is to low, the inverted pendulum oscillates with a bigger amplitude, see figure 18. Witch makes sense because the controller gets 'less' information and therefore can't react as fast. But when the filter is set to high (above 3Hz) the controller react very heavy on the noise contained in the estimated states (see figure 19). The motor starts to click a lot, see figure 20. The ideal cut-off frequency is between 1-3 Hz. Therefore we keep 2 Hz for our experiments, because it is also a choice of what you prefer: a little bit faster reaction and a bit less amplitude in the oscillation or no clicking of the motor. It is clear that a cut-off frequency of above 3 Hz is highly unwanted for the equipment.



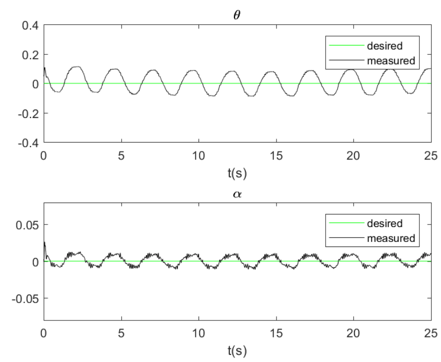
(a) $f_c = 0.5Hz$



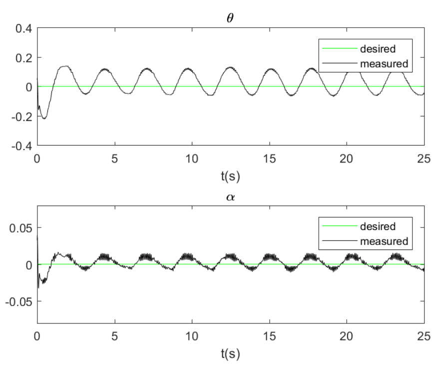
(b) $f_c = 1Hz$



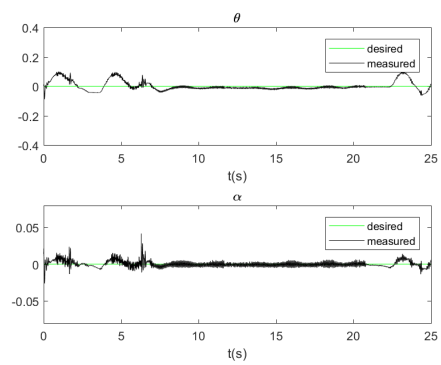
(c) $f_c = 2Hz$



(d) $f_c = 3Hz$

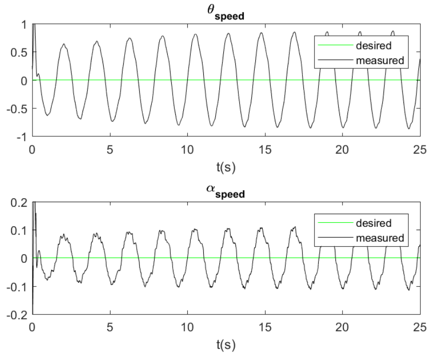


(e) $f_c = 4Hz$

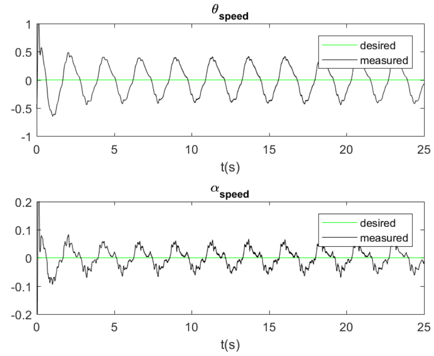


(f) $f_c = 8Hz$

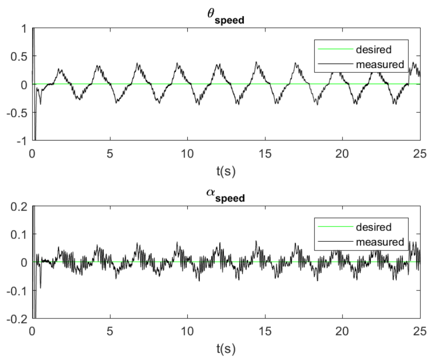
Figure 18: measured θ and α states for different cut off frequencies



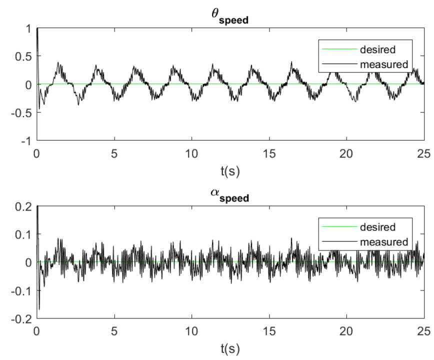
(a) $f_c = 0.5Hz$



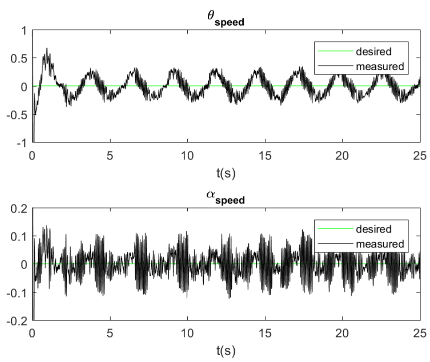
(b) $f_c = 1Hz$



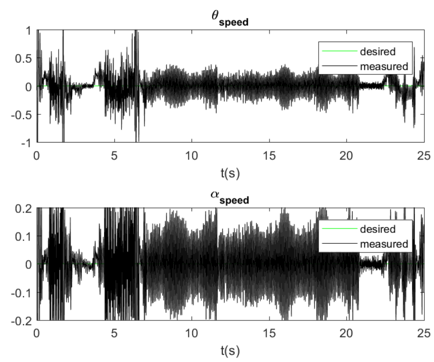
(c) $f_c = 2Hz$



(d) $f_c = 3Hz$

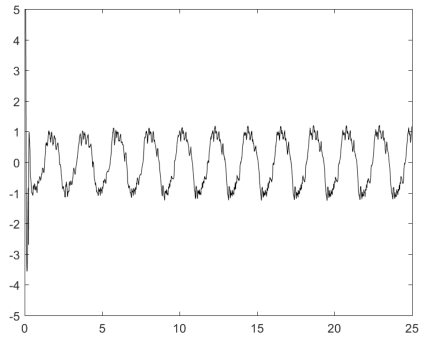


(e) $f_c = 4Hz$

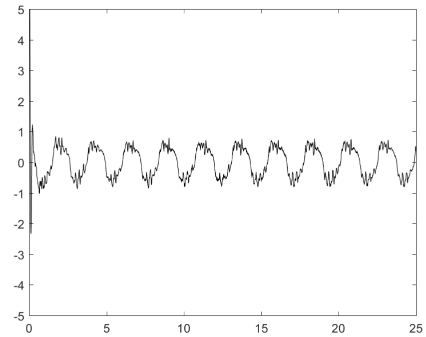


(f) $f_c = 8Hz$

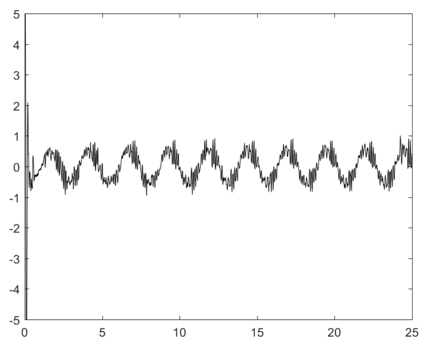
Figure 19: calculated $\dot{\theta}$ and $\dot{\alpha}$ states for different cut off frequencies



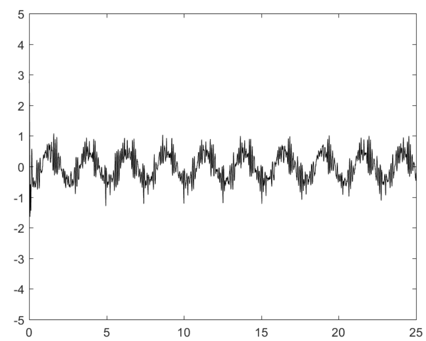
(a) $f_c = 0.5Hz$



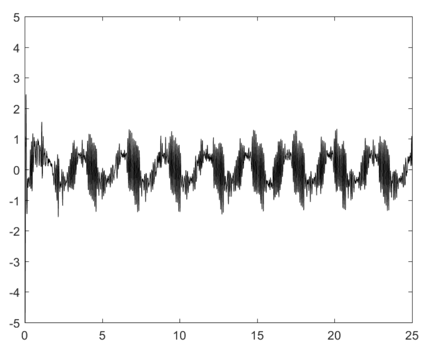
(b) $f_c = 1Hz$



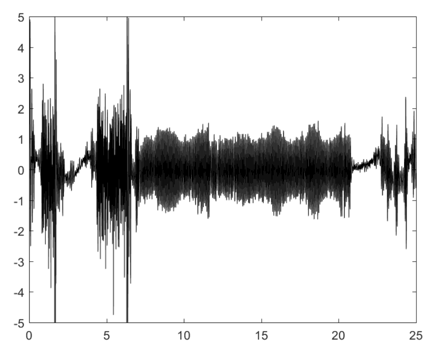
(c) $f_c = 2Hz$



(d) $f_c = 3Hz$



(e) $f_c = 4Hz$



(f) $f_c = 8Hz$

Figure 20: motor input for different cut off frequencies

6 Conclusions

The main conclusion is that controlling a system is making choices. You have to specify what is the desired controller and keep in mind that the ideal controller probably don't exist. A second conclusion is that velocity states can be estimated with position states and therefore some sensors can be eliminated in price sensitive applications. But you have to always keep in mind that the states will be contaminated with noise and it is also a lesson to design a set-up with a high signal-to-noise ratio, low noise sensitivity.

A Appendix: Matlab Code

A.1 Relevant files

The Matlab files usefull towards the user are:

1. part1_lin_model.m
2. part2_designController.m
3. part3_designController.m
4. part4_plots.m
5. part5_plots.m
6. run_all.m

A.2 Brief description

part1_lin_model.m: analysis of the linear model
part2_designController.m: find the proper Q and R
part3_designController.m: simulations with noise

In order to run the next 3 files the measurement data should be available in a folder named "measurement_data". This can be downloaded from the following dropbox link: https://www.dropbox.com/s/k93pgb8j81l2r44/measure

part4_plots.m: plots of experiments tracking and disturbance rejection
part5_plots.m: noise experiments and experiments with different f_c's
run_all: generates all the figures and saves them in the folder structure /report/img