Haptic Human-Robot Interfaces - LAB 2

Yves Martin - 262946
Luca Rezzonico - 274269

11th May 2021

Table of contents

1 Introduction **2**

2 Friction identification **2**

3 Haptic Paddle Model **2**
3.1 Physics behind the paddle model 2
3.2 Simulink Model of the haptic paddle 3
3.3 Simulation of the haptic paddle model 4
3.4 PID Position Controller 5
3.5 PID Tuning 6

4 Real Haptic Paddle **8**
4.1 Input filter 8
4.2 Regulator setup 9
4.3 Ziegler-Nichols approach 9
4.4 Ziegler-Nichols approach stabilized 11
4.5 Trial and error tuning approach 13

5 Conclusion **14**

1 Introduction

During this practical work the focus is set on controlling the position of the haptic paddle. In the first part a model of the haptic paddle is created and then a PID regulator is tuned to make first attempts to control the position in simulation. After experimenting with the simulation the same regulator is implemented on the real haptic paddle and tuned again, since there might be differences in the behavior of the simulated paddle and the real paddle.

Since there might also be some differences between the two real paddles that were given for the calss, the PID regulators were implemented on each of them in order to campare them.

2 Friction identification

First, the dry friction of our paddles is roughly estimated. In order to do so, the motor torque is gradually increased, until the paddle moves. When the paddle moves, the value of the torque is reduced such that the paddle doesn't move anymore. That way, the paddle switches from static friction to dynamic friction almost every second, and the exact applied momentum at the shift time is recorded.

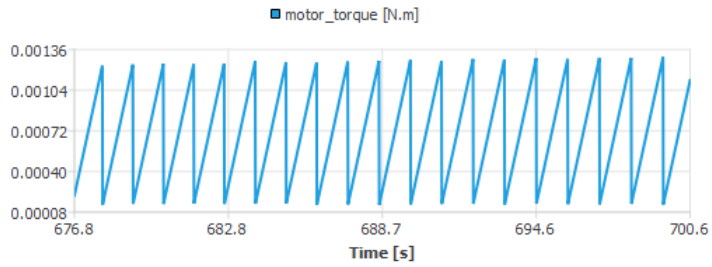


Figure 1: Motor torque is increased until static friction is overcome

In reality the friction torque doesn't depend on the position of the paddle. But when studying the real system, the measurements have to be conducted around a paddle angle of 0° to cancel the effect of gravitation, even if it is not big, because only the motor torque that has to be applied in order to compensate all the other moments can be measured. The experiment was done for an increasing positive motor torque in figure 1, but also for a decreasing negative motor torque. After averaging the values of the momentum when the paddle shifts from static to dynamic friction, i.e the maximum values of the figure 1, the following values are obtained:

	Negative motor torque [N.m]	Positive motor torque [N.m]
Paddle 19	-0.0011	0.0012
Paddle 30	-0.00116	0.00124

Table 1: Motor Torque that needs to be applied in order to overcome the opposed dry friction moment

3 Haptic Paddle Model

Before implementing a PID regulator on the real system, a model of the haptic paddle is implemented using Simulink. In a second step a PID regulator is used to control the paddle's position.

3.1 Physics behind the paddle model

In order to implement a model, the equation of motion of the haptic paddle has to be determined.

$$J_{eq} \cdot \ddot{\phi} = R \cdot T_m - B_{eq} \cdot \dot{\phi} - m_p \cdot g \cdot l \cdot sin(\phi) - r_{p_2} \cdot F_{ext} \tag{1}$$

Where ϕ is the angle, $\dot{\phi}$ the angular velocity and $\ddot{\phi}$ the angular acceleration of the paddle. R is the reduction ratio of the cable transmission and T_m is the motor torque. m_p is the mass of the paddle, g is the gravity and l is the distance between the center of mass and the center of rotation of the paddle. r_{p_2} is the moment arm where the external force F_{ext} is applied. This force can be see as a disturbance applied to the system. The total equivalent inertia and the equivalent viscous friction, computed on the paddle side are respectively:

$$J_{eq} = J_p + R^2(J_{rotor} + J_m) \tag{2}$$

$$B_{eq} = B_p + R^2 B_m \tag{3}$$

Where J_p is the inertia of the paddle, J_{rotor} the inertia of the rotor and J_m the inertia of the screw mounted on the motor shaft. B_p and B_m are the viscous frictions of the ball bearings of the paddle (neglected during simulation) and of the motor respectively.

Note that all these variables are given in SI units to have coherent calculations in the future.

3.2 Simulink Model of the haptic paddle

The equation of motion is implemented in Simulink in figure 2. The equations (2) and (3) can be calculated directly in the Matlab file used to run the simulation, since they don't depend on the time.

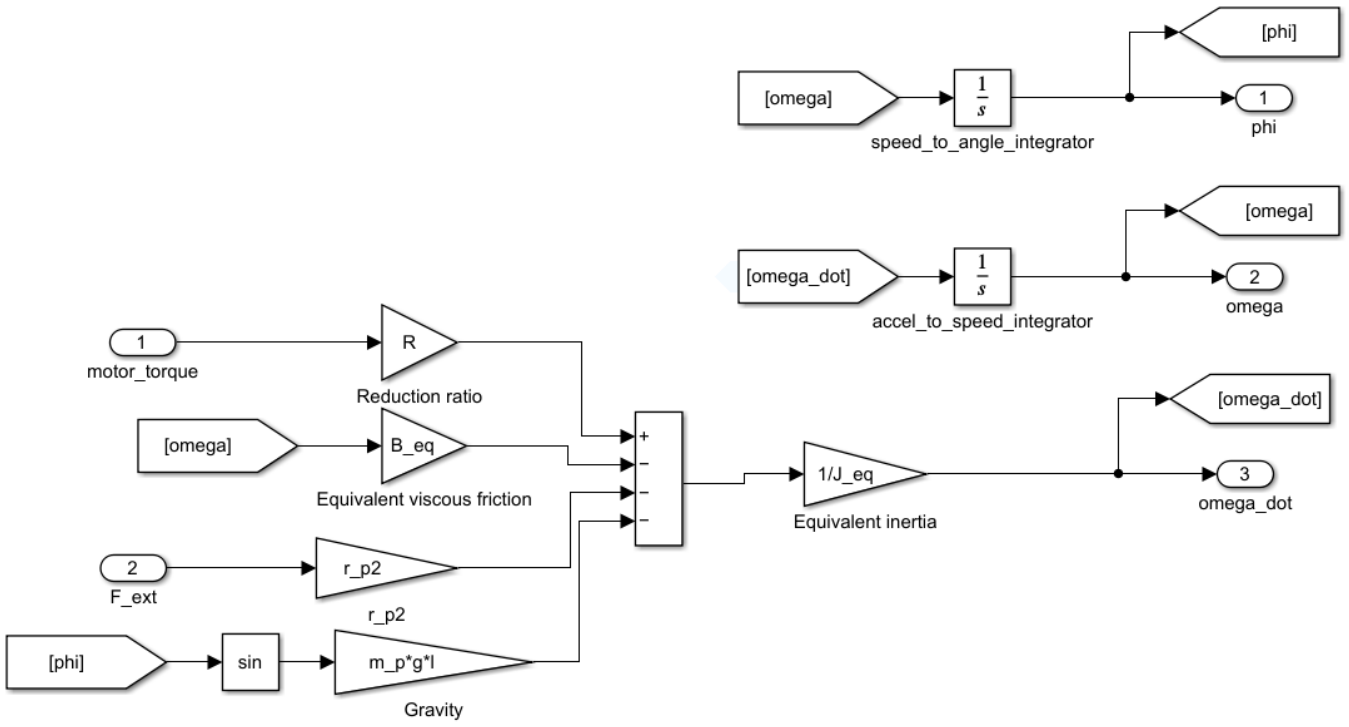


Figure 2: Simulink Model of the Haptic Paddle

This model is put in a subsystem before adding the feedback loop. Note that the units used inside this subsystem are the SI units.

3.3 Simulation of the haptic paddle model

To best see the effect of each component in the equation of motion (1), first no motor torque and no external force disturbance are applied and the paddle's initial angle is set to 10°. The simultion result is visible 3. Note that the radians are converted into degrees in the plot for clarity.

The gravity brings the paddle in oscillation around 0°, due to the non-linear term. The viscous friction attenuates these oscillations in time. The speed of the paddle is highest when the angle crosses 0° and is still when the angle reaches it's peak values. The acceleration in maximal at the extremities of the swing. This motion resembles the motion of a pendulum which is the case of a paddle with no dry friction.

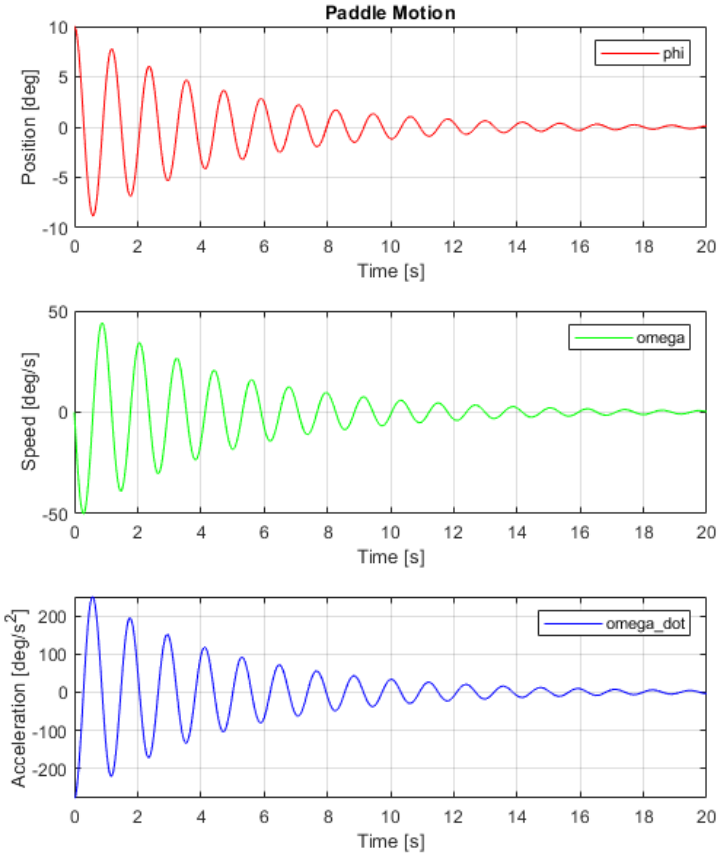


Figure 3: Paddle position, speed and acceleration evolution in time when the paddle is initiated at an angle of 10°

Then in figure 4 a sinusoidal motor torque of amplitude $0.001Nm$ is applied. The paddle turns several times around itself so the position can't be traced back properly since it is out of the range. This behavior is not possible on the real paddle, since the range is delimited by stops.

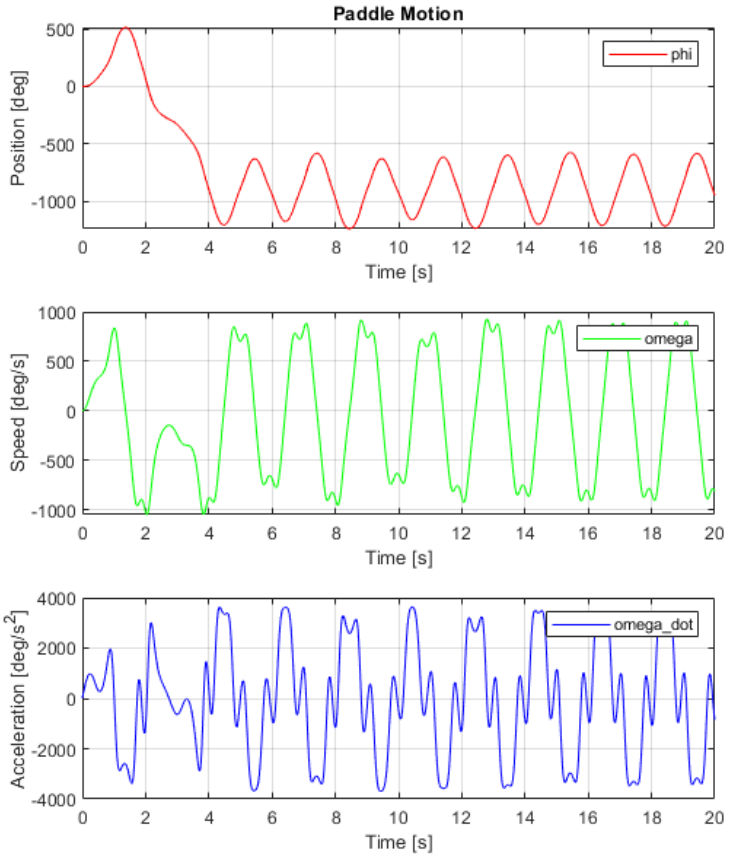


Figure 4: Paddle position, speed and acceleration evolution in time when a sinusoidal motor torque of amplitude $0.001Nm$ is applied

3.4 PID Position Controller

The position is controlled by adding a PID regulator that will measure a position error and compute the necessary torque to give to the motor in order to reduce this error. The self-made PID of figure 5 is then put inside a subsystem.

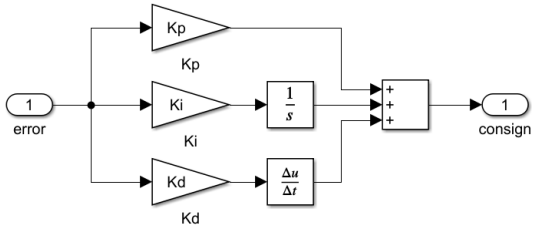


Figure 5: PID controller implemented in Simulink

The final closed-loop system is represented in figure 6. Note that on this level the radians are first converted into degrees for plotting. The converted angle *phi* is then compared to a reference phi_ref, that needs to be tracked by the PID controller.

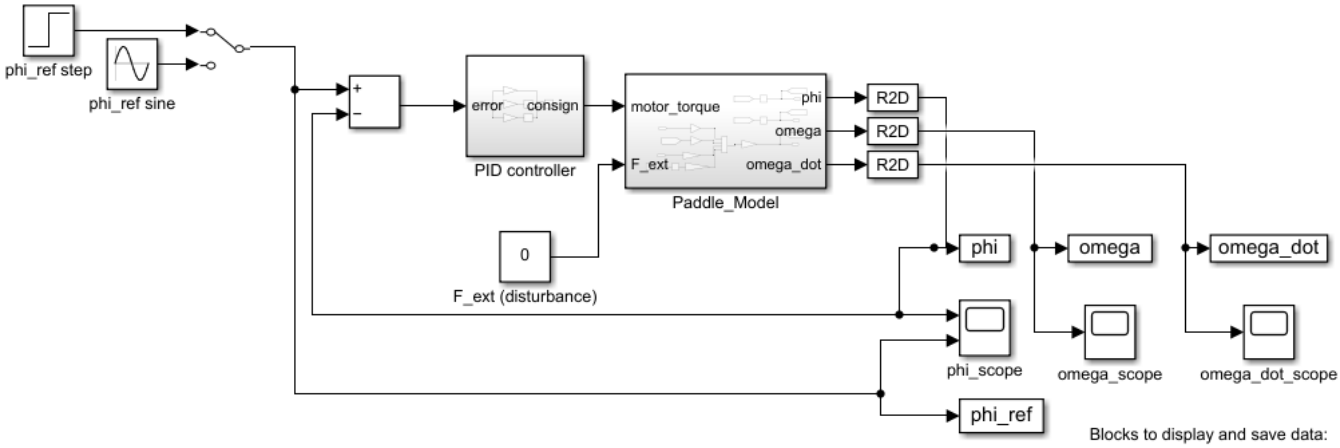


Figure 6: Global Simulink

3.5 PID Tuning

The gains K_p, K_i and K_d have to be tuned such that for a 10° angle step, the response meets the following specifications:

- Maximum overshoot: 25% (2.5°)
- Maximum settling time to reach an error band of ±0.2°: 350 ms
- Maximum rise time: 100 ms

Before implementing the PID controller on the real paddle, the model is first used to tune our PID values. Having a virtual paddle allows to try efficiently many possible PID gains to avoid harming the real paddle. The first method is using the Ziegler-Nichols approach. The gain of the controller is increased until the position oscillates around the reference value. At this point $K_u = 0.005$ and is called ultimate gain. The oscillations are analysed to find the ultimate period $T_u = 0.08$. The PID gains can then be deduced by:

$$K_p = 0.6 \cdot K_u = 0.003 \tag{4}$$

$$K_i = \frac{K_p}{0.5 \cdot T_u} = 0.075 \tag{5}$$

$$K_d = K_p \cdot 0.125 \cdot T_u = 0.00003 \tag{6}$$

With these gains the response of the commanded model to an angle step of 10° is shown in figure 7.

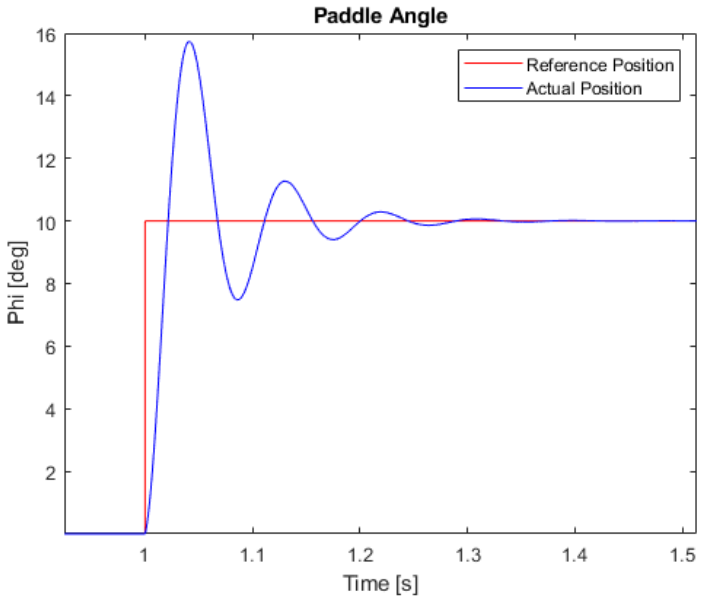


Figure 7: 10° step response of the system commanded by a PID tuned using the Ziegler-Nichols method

The Ziegler-Nichols method didn't result in a step response that meets the specifications, since the overshoot is too high. This is why the gains need to be fine-tuned by hand. The following values were obtained:

Gain	Value
K_p	0.005
K_i	0.0012
K_d	0.00009

Table 2: Fine-tuned PID gains

Figure 8 shows the obtained 10° step response after fine-tuning the PID gains. This response meets the specifications by far, since the overshoot is almost non-existent and the response settles in less than $60ms$.

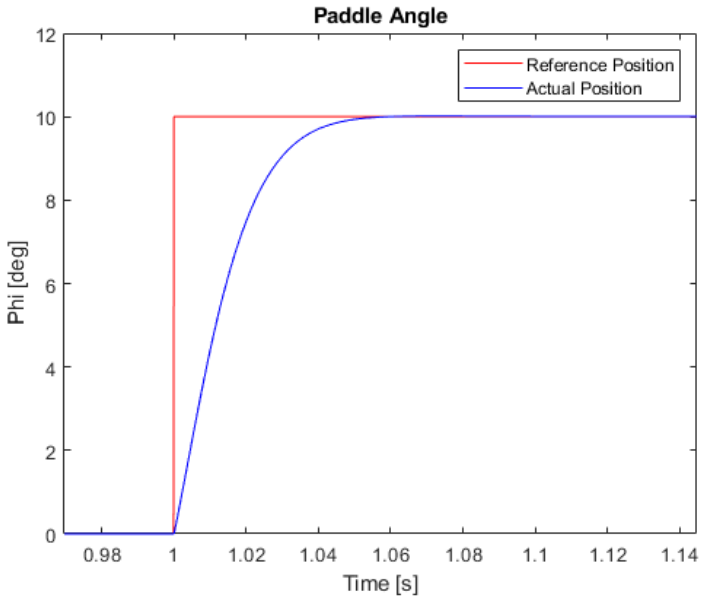


Figure 8: 10° step response of the system commanded by the fine-tuned PID

4 Real Haptic Paddle

After experimenting the effect of the different gains of the position controller in simulation, the real haptic paddle is controlled. The position of the paddle can be given by two sensors with their advantages and disadvantages. The first one used is the encoder on the motor, it has no measurement noise, but only quantization noise. Its drawback is that it is an incremental encoder, the measurement is not absolute, but calculated from the previous one. The second sensor used is the Hall-effect sensor. This sensor is absolute, but it has measurement error, which means even at steady state the measured angle varies. To be able to control the paddle based on this measurement the position has to be filtered first to avoid amplification of the noise.

4.1 Input filter

To reduce the noise amplified by the derivative part of the PID controller, it is important to have a smooth input signal. The function $hapt\_LowPassFilter()$ was used with a cut-off frequency of 50Hz to filter the position given by the hall-effect sensor, in order to reduce the measurement noise. This filter was not implemented in the experiments when using the encoder, but could have been used to reduce quatization noise too. The results are compared later in this section. The low-pass filter basically implements the following formula:

$$y(t) = \alpha \cdot x(t) + (1 - \alpha) \cdot x(t - 1)$$

$$\alpha = \frac{dt}{\tau + dt}$$

$$\tau = \frac{1}{2\pi \cdot \omega_{cut-off}}$$

where $x(t)$ is the actual position, $x(t-1)$ is the previously filtered position, $y(t)$ is the actual filtered position, α is called the smoothing factor, τ is the rise time, i.e the time to reach 63% of the steady-state value and dt is the sampling period.

The PID implemented is used as a position controller, so the filtering should be done on the position. The velocity is calculated from the position, so filtering the velocity has no effect on the quality of the control. But if needed a filtered version of the speed can be used. An important remark concerns the cut-off frequency of the filter, it should not be to low, because the system could get unstable due to the induced delay. This remark is even more important when the gains chosen get higher. Basically, the higher the gain the shorter the delay margin before the system gets unstable.

4.2 Regulator setup

The regulator is disabled by default when the board is starting. the default position used by the controller is the position measured by the encoder. The sensor used to control the paddle position is by default the encoder, but can be changed to the Hall-effect sensor from the user interface. Once the position sensor is selected, the position regulator can be switched on and the target angle (for example 10°) can be changed in the user interface. The controller will then move the paddle to the selected target angle and even after an external disturbance is applied, the position converges back to the target angle.

4.3 Ziegler-Nichols approach

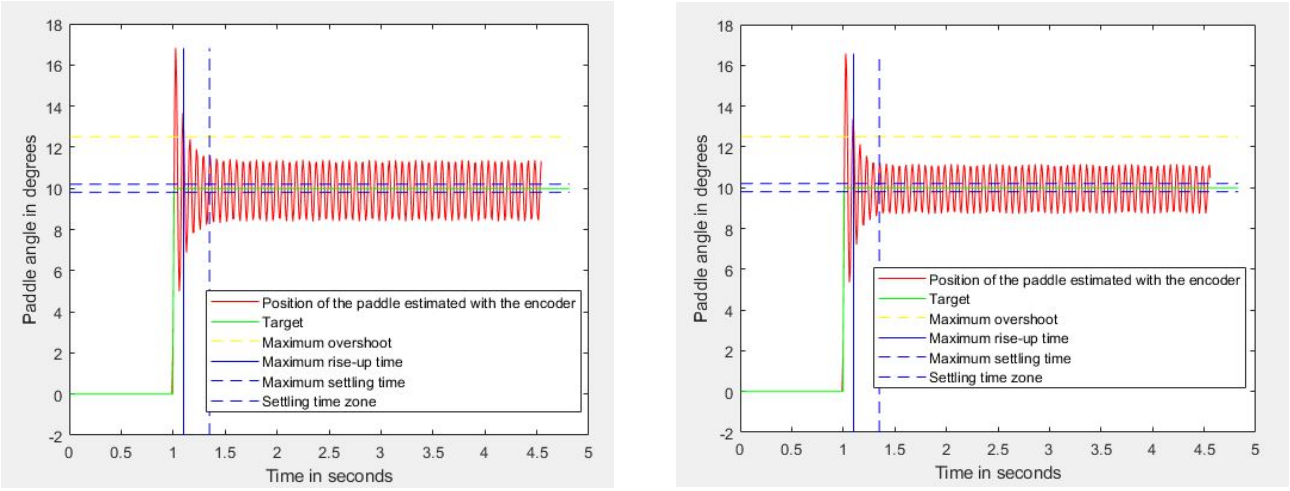
By tuning of the PID encoder-based position controller using the Ziegler-Nichols method described in section 3.5 during simulation, the following gains were obtained:

Gain	Value
K_p	0.042
K_i	3.6522
K_d	0.00012

Table 3: PID gains for the position controller on the encoder angle obtained with the Ziegler-Nichols method

To make the plot easier to read, the specifications of section 3.5 were included into the graph. A first vertical blue line shows the maximal rise time the system is allowed to take and is placed 100ms after the step input. The second vertical line is dashed in blue and indicates the maximal settling time the system is allowed to take to stabilise to a value within a small interval around the reference position. This small interval is delimited by the two horizontal dashed blue lines. The horizontal dashed yellow line represents the maximum overshoot the specifications tolerate for the paddle. Finally the green trace is the reference position the paddle controller has to follow and the red one is its actual position.

The resulting trajectory using the encoder position measurements and Ziegler-Nichols approach to tune the gains of table 3 is visible in figure 9 for both paddles. Both paddles oscillate around the reference position after the step is applied. It seemed like the derivative gain given by this method was very low and from the plot it gets clear.



(a) Paddle 30

(b) Paddle 19

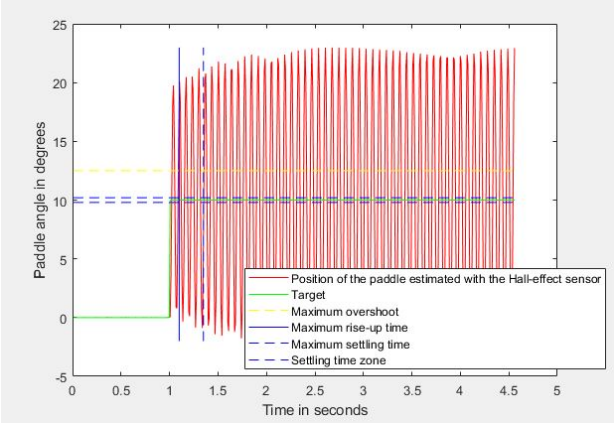
Figure 9: 10° step response of the real haptic paddle with a PID position controller on the position given by the encoder and tuned with the Ziegler-Nichols method

Again the Ziegler-Nichols method is used, but this time to tune the PID position regulator based on measurements from the Hall-effect sensor. The following gains were obtained:

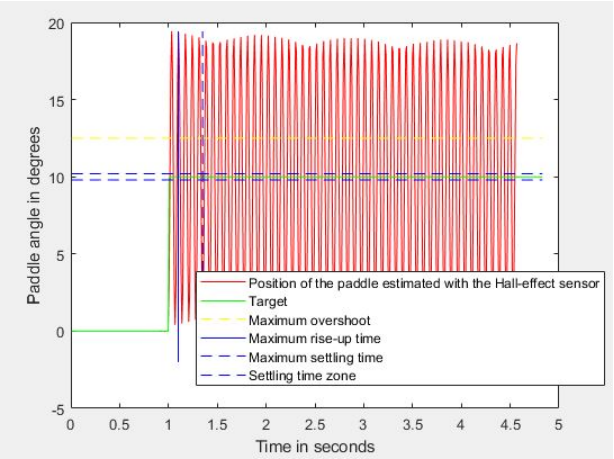
Gain	Value
K_p	0.0081
K_i	0.4035
K_d	0.00004035

Table 4: PID gains for the position controller on the hall angle obtained with the Ziegler-Nichols method

These gains are approximately a factor 10 lower than those obtained using the position measurement of the encoder on the motor. This can be explained because the noisy system is more likely to get unstable. Even if a low-pass filter was used to smoothen the position contaminated by measurement noise, the gains of the controller can't be as high as the ones used for the control of the position given by the encoder. In figure 10 the same step of 10° is applied for both paddles and the oscillations around the reference position are even bigger than the ones seen when using the encoder position. This higher variance might be explained by the additional measurement noise present for the Hall-sensor. But it is hard to say, since the gains differ much from one sensor to the other. But the derivative gain is again very low compared to the others.



(a) Paddle 30



(b) Paddle 19

Figure 10: 10° step response of the real haptic paddle with a PID position controller on the position given by the Hall-effect sensor and tuned with the Ziegler-Nichols method

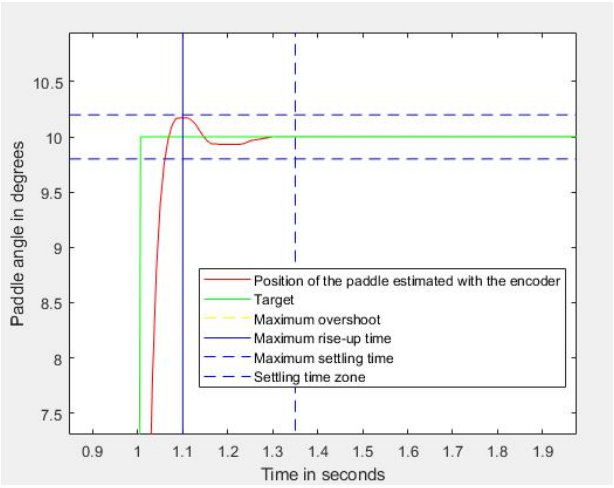
4.4 Ziegler-Nichols approach stabilized

As seen before the Ziegler-Nichols method used to calculate the gains of the position controller result in a constantly oscillating response. To improve the stability of the system, the component K_d was multiplied by a factor 10, to better damp these oscillations. The gains applied to the position controller using the encoder angle are thus:

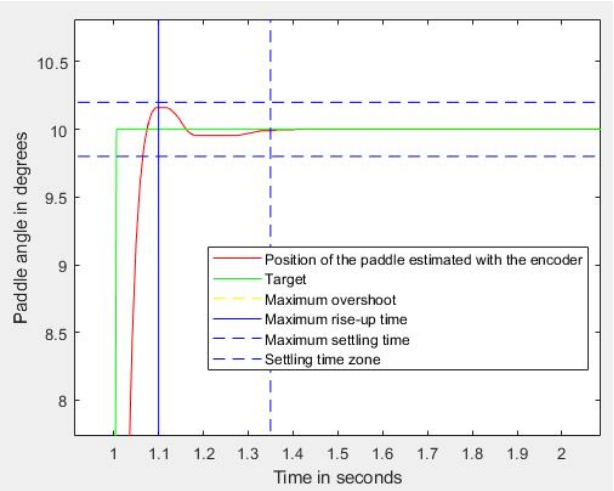
Gain	Value
K_p	0.042
K_i	3.6522
K_d	0.0012

Table 5: PID gains for the position controller on the encoder angle obtained with the Ziegler-Nichols method and multiplying K_d by 10

Figure 11 zooms on the response of the two haptic paddles to the same 10° step as before, using the encoder position measurement, and the result is much better. By changing the damping properties of the paddle, the system now meets the requirements. A little detail can be observed after the paddles overshoot the reference, they seem to stabilize below the reference for approximately 100ms, this behavior is due to static friction. The error difference is not amplified enough by the gains to overcome the static friction. But eventually the integrator brings the paddle back to the target.



(a) Paddle 30



(b) Paddle 19

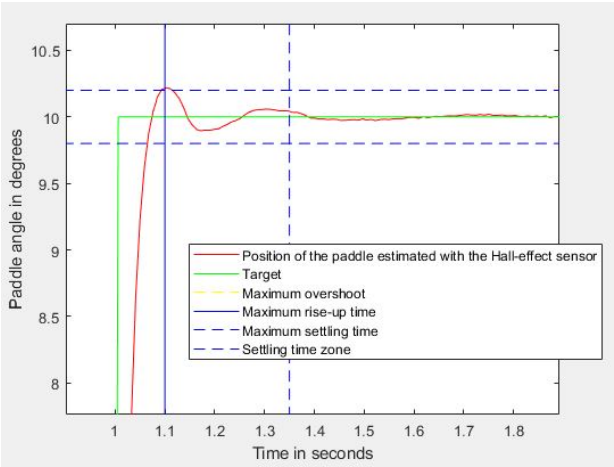
Figure 11: 10° step response of the real haptic paddle with a PID position controller on the position given by the encoder, tuned with the Ziegler-Nichols method and K_d multiplied by 10

For the same reason a factor 10 was applied to the K_d gain of the position controller using the Hall-effect sensor measurements of the angle, and thus the values used are:

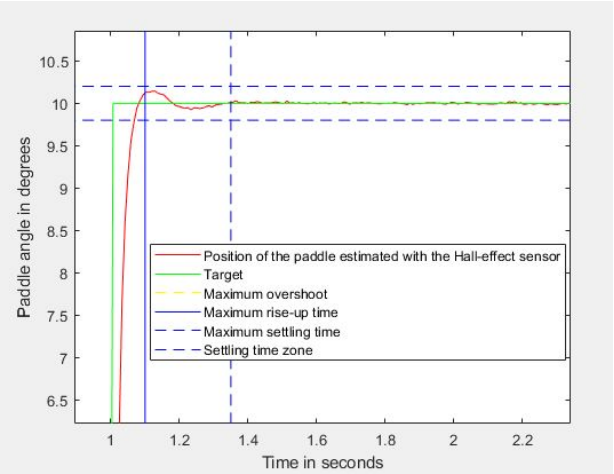
Gain	Value
K_p	0.0081
K_i	0.4035
K_d	0.0004035

Table 6: PID gains for the position controller on the hall angle obtained with the Ziegler-Nichols method and multiplying K_d by 10

Figure 12 show the details of the step response of the paddles, using the Hall-effect sensor position measurements, and the result is satisfying too. The system also meets the requirements for the PID controller on the hall angle.



(a) Paddle 30



(b) Paddle 19

Figure 12: 10° step response of the real haptic paddle with a PID position controller on the position given by the Hall-effect sensor, tuned with the Ziegler-Nichols method and K_d multiplied by 10

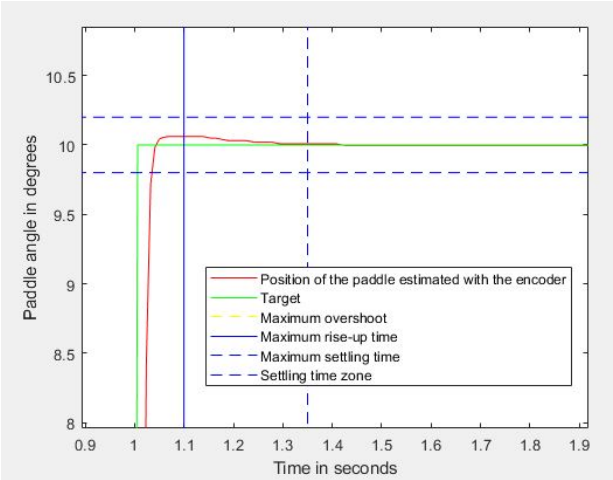
4.5 Trial and error tuning approach

As seen previously the Ziegler-Nichols tuning method doesn't result in an optimal solution. It gives a good idea for the order of the gains that have to be chosen, but not more. For this reason, trial and error was applied to find a better solution to control the position of the paddle. The final choice for the tuning parameters is the following:

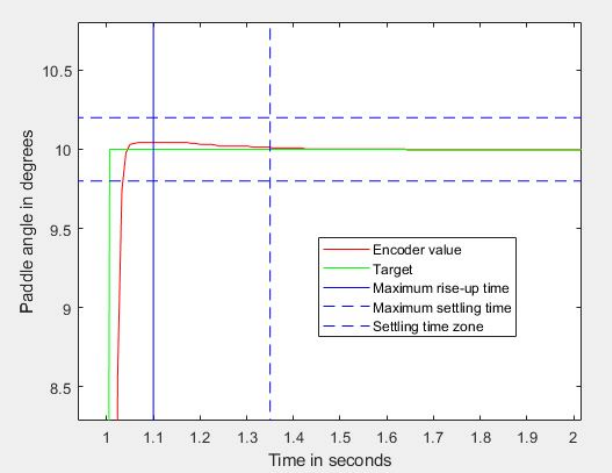
Gain	Value
K_p	0.3
K_i	3
K_d	0.003

Table 7: PID gains for the position controller on the encoder angle obtained with the trial and error method

The associated step response of the paddles controlled by the encoder-based PID regulator is showing in details in figure 13. For paddle 19 there is again the static friction that stops the system right over the target angle, but the integral part of the controller is well tuned and deletes this error afterwards. The controller tuned by experimenting by hand and adapting the gains to improve the behavior, satisfies all specifications by far.



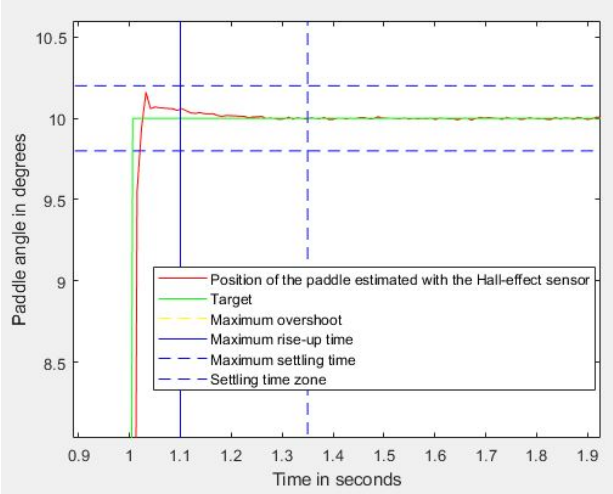
(a) Paddle 30



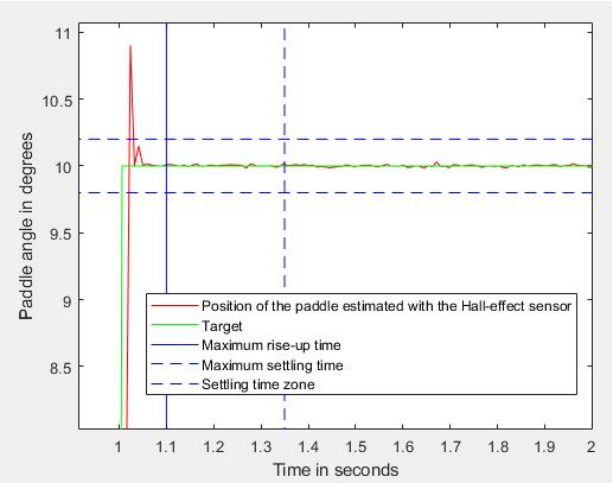
(b) Paddle 19

Figure 13: 10° step response of the real haptic paddle with a PID position controller on the position given by the encoder, tuned with the trial and error method

Now to compare both sensors, the PID gains were kept the same for the Hall-effect sensor based position controller as in table 7 for the encoder-based PID regulator. The specifications are also met when using the hall sensor angle, even if the gains are kept the same. This is because the controller tuned in table 7 is very stiff. Little position variations around the steady state are visible, which is typical when using the absolute sensor.



(a) Paddle 30



(b) Paddle 19

Figure 14: 10° step response of the real haptic paddle with a PID position controller on the position given by the Hall-effect sensor, tuned with the trial and error method

5 Conclusion

The main reason why the simulated model is so different from the real haptic paddle, is that the dry friction of the real paddle is not model in the simulation. The real system doesn't return to an angle of 0° when it is initiated at any angle, because the dry static friction moment is not overcome by the torque induced by the gravity, but in simulation

the model stabilizes again at 0°. To have a good model dry friction need to be included.

An other important remark, when using the low-pass filter to attenuate the high frequency variations in the Hall-effect sensor position measurements, has to be made. A special care has to be taken not to chose the cut-off frequency too low. It was chosen to be 50Hz when experimenting with stiff controller gains, because otherwise the delay induced by the filtering could make the system unstable.