POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Laboratory Report
Inverted Pendulum

Automation and Control Laboratory

Author:

Guido Mancinelli

Pietro Messina

Serena Palmieri

Elena Maria Pesci

Student ID: 10663546 10663556 10660784 10697743

Academic Year: 2023/24

Contents

1 | Modeling

1.1. Introduction

The objective of the laboratory is the control of a QUBE-Servo2 Rotary Inverted Pendulum. The system comprises a PMDC motor, a horizontal rod, a pendulum, and two encoders, which measure the rod and pendulum angles, θ_1 and θ_2.
A hinge attaches the pendulum to the horizontal rod, while the latter is fixed to the motor shaft. The horizontal rod is therefore directly subjected to the DC motor torque, while no torque is directly applied to the pendulum shaft.
The resulting mechanical chain is under-actuated, as it has only one control input and two degrees of freedom.
A schematic picture of the rotatory inverted pendulum with its reference frames and some parameters is shown in Figure 1.1.

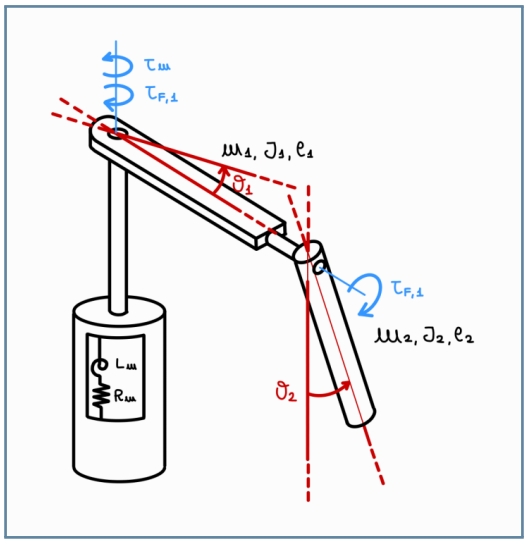


Figure 1.1: Model of the pendulum

In addition to the motor torque and inertias, the system is subject to friction around both the vertical and horizontal axes. Frictional effects have been modeled as viscous, which follow the generalized law $\tau_F = K_F \times \dot{\theta}$.

Furthermore, to model the combined action of the cable of the encoder measuring the pendulum angle and the integrated stabilization mechanism, which serves as a way to set the starting position, a torsional spring acting on θ_1 has been theorized: $\tau_{spring} = k_s \theta_1$.

1.2. Motor equations

The first component that needs to be modeled is the brushed PMDC motor driving the mechanical system. Its equations are the following:

$$\begin{cases} V_{in} = R_m i + L_m \dot{i} + E \\ E = k_m \Omega \\ \tau_m = k_m i \end{cases} \tag{1.1}$$

Where:

- V_{in} is the input voltage
- i is the current flowing in the circuit of the DC motor
- E is the electromotive force (EMF)
- Ω is the angular speed of the shaft of the motor
- τ_m is the torque
- R_m is the motor resistance
- L_m is the motor inductance
- k_m if the torque constant of the motor

Due to the rapidity of the DC motor, which has a time constant with OOM of 10^{-4} (from datasheet values), its dynamical behaviour can be approximated removing transitory-related terms:

$$\tau_m = k_m / R_m V_{in} - k_m^2 / R_m \dot{\theta}_1 \tag{1.2}$$

1.3. Nonlinear equations

After assigning a reference frame to each joint and establishing a set of reference frames using Denavit-Hartenberg conventions, the following inertia, gravitational, Coriolis and stiffness matrices were evaluated:

$$B(q) = \begin{bmatrix} m_1 x_1^2 + J_1 + m_2(l_1^2 + x_2^2 sin^2(\theta_2)) & m_2 l_1 x_2 cos(\theta_2) \\ m_2 l_1 x_2 cos(\theta_2) & m_2 x_2^2 + J_2 \end{bmatrix} \quad g(q) = \begin{bmatrix} 0 \\ m_2 g x_2 sin(\theta_2) \end{bmatrix}$$

$$C(q) = \begin{bmatrix} m_2 x_2^2 sin(\theta_2) cos(\theta_2) \dot{\theta}_2 & m_2 x_2^2 sin(\theta_2) cos(\theta_2) \dot{\theta}_1 - m_2 l_1 x_2 sin(\theta_2) \dot{\theta}_2 \\ -m_2 x_2^2 sin(\theta_2) cos(\theta_2) \dot{\theta}_1 & 0 \end{bmatrix} \quad K_{el} = \begin{bmatrix} k_s \\ 0 \end{bmatrix}$$

This lead to the following system of nonlinear equations:

$$\begin{cases} \left[m_1 x_1^2 + J_1 + m_2(l_1^2 + x_2^2 \sin^2(\theta_2))\right] \ddot{\theta}_1 + m_2 l_1 x_2 \cos(\theta_2) \ddot{\theta}_2 + 2 m_2 x_2^2 \sin(\theta_2) \cos(\theta_2) \dot{\theta}_1 \dot{\theta}_2 \\ \quad - m_2 l_1 x_2 \sin(\theta_2) \dot{\theta}_2^{\,2} + k_s \theta_1 = \tau_m - K_{F,1} \dot{\theta}_1 \\ m_2 l_1 x_2 \cos(\theta_2) \ddot{\theta}_1 + (m_2 x_2^2 + J_2) \ddot{\theta}_2 - m_2 x_2^2 \sin(\theta_2) \cos(\theta_2) \dot{\theta}_1^{\,2} + m_2 g x_2 \sin(\theta_2) = -K_{F,2} \dot{\theta}_2 \end{cases} \tag{1.3}$$

Where:

- τ_m is the previously evaluated motor torque
- m_i, l_i, J_i are the masses, lengths and moments of inertia of the shafts
- x_i are the distances between the centre of rotation of the shafts and their centre of mass
- $K_{F,i}$ are the viscous friction coefficients

1.4. Model linearization

Most of the controllers that have to be implemented need a linear system in order to be derived, so it is mandatory to linearize the model around its equilibria positions. The state vector is $x = (\theta_1 \quad \theta_2 \quad \dot{\theta}_1 \quad \dot{\theta}_2)^T$, and the two equilibrium positions used for the linearization are:

1. The downward stable equilibrium: $x = (0 \quad 0 \quad 0 \quad 0)^T$
2. The upward unstable equilibrium: $x = (0 \quad \pi \quad 0 \quad 0)^T$

1.4.1. Linearization around the stable equilibrium

The model has been linearized around the state vector $x = (0 \quad 0 \quad 0 \quad 0)^T$, obtaining the following equations:

$$\begin{cases} (m_1 x_1^2 + J_1 + m_2 l_1^2)\ddot{\theta}_1 + m_2 l_1 x_2 \ddot{\theta}_2 + (K_{F,1} + \dfrac{k_m^2}{R_m})\dot{\theta}_1 + k_s \theta_1 = \dfrac{k_m}{R_m} V_{in} \\ m_2 l_1 x_2 \ddot{\theta}_1 + (m_2 x_2^2 + J_2)\ddot{\theta}_2 + K_{F,2}\dot{\theta}_2 + m_2 g x_2 \theta_2 = 0 \end{cases} \tag{1.4}$$

In order to rewrite the system in state-space form, the vectors of states x, input u and output y have been identified as:

$$x = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad u = V_{in} \quad y = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \tag{1.5}$$

And solving for the acceleration terms, the system can be rewritten as:

$$\begin{cases} \dot{x} = [A]x + [B]u \\ y = [C]x + [D]u \end{cases} \tag{1.6}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k_s(m_2x_2^2+J_2)}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} & \frac{gl_1m_2^2x_2^2}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} & -\frac{(m_2x_2^2+J_2)(k_m^2/R_m+k_{F,1})}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} & \frac{k_{F,2}l_1m_2x_2}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} \\ \frac{k_sl_1m_2x_2}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} & -\frac{gm_2x_2(m_2l_1^2+m_1x_1^2+J_1)}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} & \frac{l_1m_2x_2(k_m^2/R_m+k_{F,1})}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} & -\frac{k_{F,2}(m_2l_1^2+m_1x_1^2+J_1)}{(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{k_m(m_2x_2^2+J_2)}{R_m\left[(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2\right]} \\ -\frac{k_ml_1m_2x_2}{R_m\left[(m_2l_1^2+m_1x_1^2+J_1)(m_2x_2^2+J_2)-l_1^2m_2^2x_2^2\right]} \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

1.4.2. Linearization around the unstable equilibrium

The model has been linearized around the state vector $x = (0 \quad \pi \quad 0 \quad 0)^T$, obtaining the following equations:

$$\begin{cases} (m_1 x_1^2 + J_1 + m_2 l_1^2)\ddot{\theta}_1 - m_2 l_1 x_2 \ddot{\theta}_2 + (K_{F,1} + \dfrac{k_m^2}{R_m})\dot{\theta}_1 + k_s \theta_1 = \dfrac{k_m}{R_m} V_{in} \\ -m_2 l_1 x_2 \ddot{\theta}_1 + (m_2 x_2^2 + J_2)\ddot{\theta}_2 + K_{F,2}\dot{\theta}_2 - m_2 g x_2 \theta_2 = 0 \end{cases} \tag{1.7}$$

Using the same state, input and output vectors the space-state matrices are now:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{ks(m_2 x_2^2 + J_2)}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} & \frac{g l_1 m_2^2 x_2^2}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} & -\frac{(m_2 x_2^2 + J_2)(km^2/R_m + k_{F,1})}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} & \frac{k_{F,2} l_1 m_2 x_2}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} \\ \frac{ksl_1 m_2 x_2}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} & -\frac{g m_2 x_2 (m_2 l_1^2 + m_1 x_1^2 + J_1)}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} & \frac{l_1 m_2 x_2 (km^2/R_m + k_{F,1})}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} & -\frac{k_{F,2}(m_2 l_1^2 + m_1 x_1^2 + J_1)}{(m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{km(m_2 x_2^2 + J_2)}{R_m((m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2)} \\ \frac{kml_1 m_2 x_2}{R_m((m_2 l_1^2 + m_1 x_1^2 + J_1)(m_2 x_2^2 + J_2) - l_1^2 m_2^2 x_2^2)} \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

2 | Parameters Identification

Before attempting to control the pendulum, it is essential to identify the system parameters to develop a model that accurately reflects reality. Without this step, the simulation would not be consistent with real-world behavior.
To tackle this complex task, this problem was split between identifying motor and mechanical parameters, thanks to the possibility to remove the mechanical attachments of QUBE-Servo2.
The main source consulted in the realization of this chapter is [1].

2.1. Motor parameters

The process of identifying motor parameters relies on the linearity of its equations, that allows the use of techniques such as the frequency response theorem.

In order to be able to tune motor parameters basing on experimental data, which consists in measures of the motor angle θ_1 depending on given voltage inputs V_{in}, the input-output behaviour needs to be modeled.
Starting from 1.2, which represents the electrical characteristic of the motor, and considering the mechanical characteristic in 2.1, the complete motor equation can be modeled as in 2.2:

$$\tau_m = J_m\ddot{\theta_1} + b_m\dot{\theta_1} \tag{2.1}$$

$$J_m\ddot{\theta_1} + b_m\dot{\theta_1} = k_m/R_m V_{IN} - k_m^2/R_m\dot{\theta_1} \tag{2.2}$$

Therefore its transfer function can be written as follows:

$$\frac{\theta_1(s)}{V_{in}(s)} = \frac{k_m}{J_m R_m s^2 + (b_m R_m + k_m^2)s} \tag{2.3}$$

The identification process has been formulated as an optimization problem, aiming to minimize the differences in both the static and dynamical behaviour between real measures and simulation by tuning parameters values.
To avoid overfitting, it was decided to trust datasheet values regarding the motor inertia J_m, which is easier to measure than the other quantities.

The chosen objective function is based on the frequency response theorem, which states that the response of a sinusoidal input is a sinusoid with the same frequency, whose amplitude and phase are the ones of the transfer function evaluated at that frequency:

$$(R_m, k_m, b_m) = argmin \sum_{\omega} (f(\omega) - \hat{f}(\omega, R_m, k_m, b_m))^2 \tag{2.4}$$

Where f can either be gain or phase measures.
In the case at hand, due to the noisy phase measures, it was decided to use the gains as fitting measure. The frequency-dependant gain of the transfer function is obtained as follows:

$$\hat{f}(\omega, R_m, k_m, b_m) = \left| \frac{\theta_1(j\omega)}{V_{in}(j\omega)} \right| = \frac{k_m}{\sqrt{(J_m R_m \omega^2)^2 + ((b_m R_m + k_m^2)\omega)^2}} \tag{2.5}$$

The fitting process was performed using Matlab-integrated `lsqcurvefit()` function, bounding the parameters to have a positive value and using datasheet values as initial guess.

Figure 2.1 shows the result of the fitting process in comparison with the bode transfer function if the parameters of the datasheet are kept.

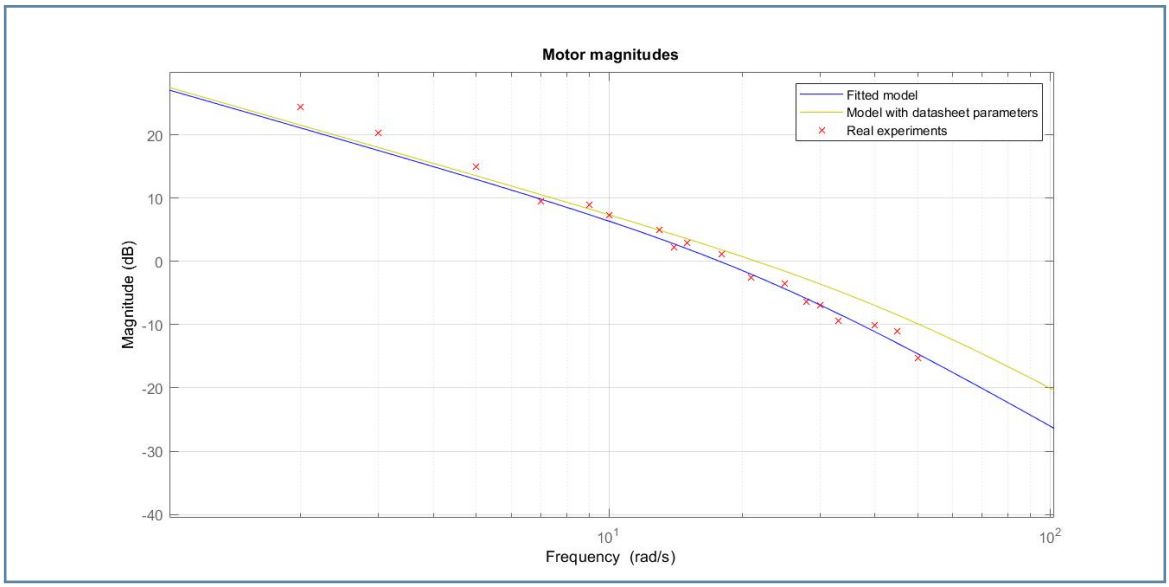


Figure 2.1: Bode diagram of the transfer function with obtained parameters

2.1.1. Validation

To ensure the consistency of the obtained parameters, both magnitude and phase values need to comply with the model ones. Experiments at different input amplitudes and frequencies were carried out, and the result shown in Figure 2.2

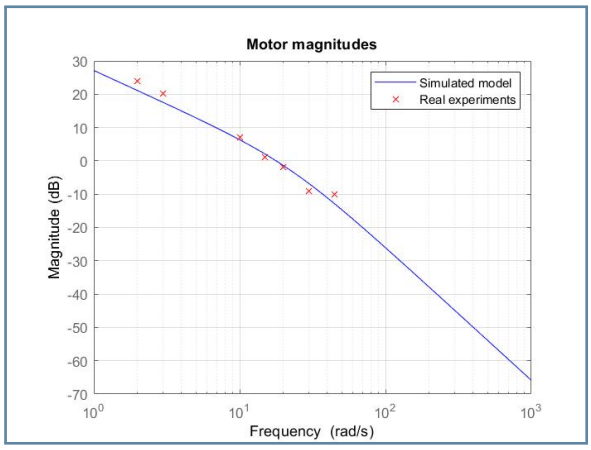


Figure 2.2: Frequency validation

2.2. Mechanical model parameters

The complete mechanical model, as defined in equation 1.3, depends on a vast number of parameters. In order to narrow down the degrees of freedom during the search for the true values, it was decided to trust the QUBE-Servo2 documentation regarding easily measurable quantities, such as masses and lengths.
A spreadsheet of which parameters were taken from the datasheet and which needed estimation (*tbe*) follows in 2.1:

Table 2.1: Mechanical Parameters

Parameter	**Symbol**	**Value**	**Considerations**
Sum of the masses of the horizontal shaft and the encoder	m_1	9.5×10^{-2} kg	/
Mass of the pendulum	m_2	2.45×10^{-2} kg	/
Length of the horizontal shaft	l_1	8.5×10^{-2} m	/
Length of the pendulum	l_2	1.29×10^{-1} m	/
Position of the CoM of the horizontal shaft	x_1	$l_1/2$	The CoM was assumed in the geometrical center of the shaft
Position of the CoM of the pendulum	x_2	$l_2/2$	The CoM was assumed in the geometrical center of the pendulum
Moment of inertia of the combination of encoder and horizontal shaft	$J_{1,eq}$	*tbe*	Neither the total moment of inertia nor the mass distribution of the shaft and encoder were reported in the datasheet
Moment of inertia of the pendulum	J_2	$\frac{1}{12}m_2 l_2^2$	The moment of inertia of a beam rotating around an extremity
Viscous friction coefficient around vertical axis	$K_{F,1}$	*tbe*	The datasheet does not state a value for friction coefficients
Viscous friction coefficient around horizontal axis	$K_{F,2}$	*tbe*	The datasheet does not state a value for friction coefficients

Continued on next page

Table 2.1 – continued from previous page

Parameter	Symbol	Value	Considerations
Elastic coefficient of the supposed torsional spring	k_s	*tbe*	The datasheet does not offer any technical clarification on the integrated stabilization mechanism

As a consequences of these choices, it can be noticed how, apart from $K_{F,2}$, only parameters appearing in the first equation of 1.3 needed to be estimated, and in particular just the ones multiplying θ_1 and its derivatives.
This suggested that an independent estimation of $K_{F,2}$ would allow to simplify the estimation process, as will be discussed.

2.2.1. Pendulum friction estimation

In order to perform an independent estimation of $K_{F,2}$ an isolation experiment was conducted, where the setup involved fixing the horizontal rod and positioning the pendulum at an angle of 10 degrees from the downright vertical position.
The pendulum was then released without imparting any additional forces and allowed to oscillate freely until coming to a complete stop.

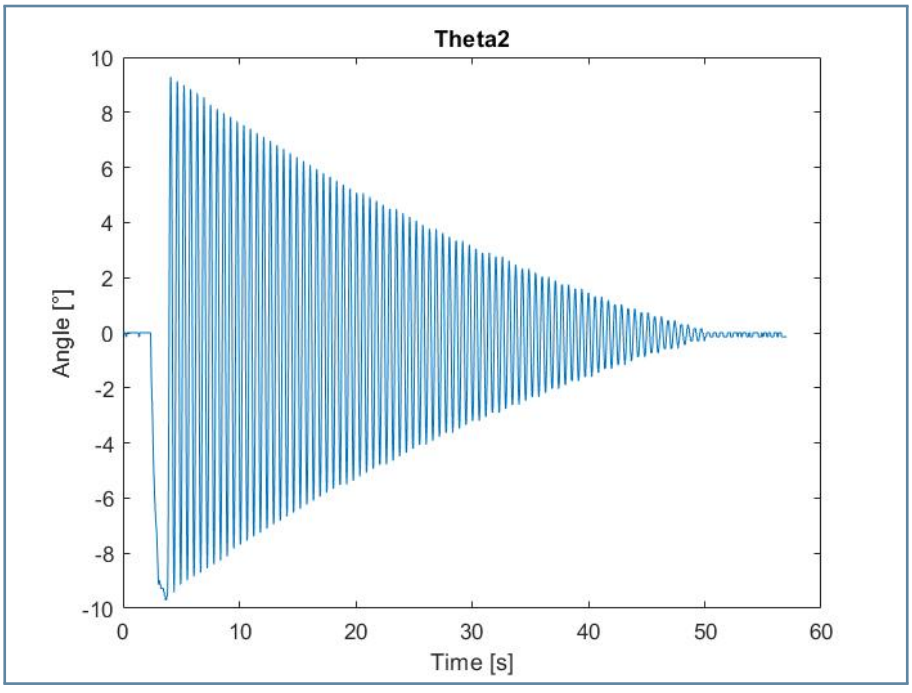


Figure 2.3: Pendulum's free oscillation

The method of logarithmic decrement was utilized to determine the viscous friction coefficient. The logarithmic decrement, denoted as δ, measures the rate at which oscillations decrease in an underdamped system. It is defined as the natural logarithm of the ratio of successive amplitude

peaks:

$$\delta = \frac{1}{n} \ln \left(\frac{x(t)}{x(t+nT)} \right) \tag{2.6}$$

Where:

- $x(t)$ is the amplitude at time t,
- T is the period of oscillation,
- $x(t+nT)$ is the amplitude after n periods.

The damping factor is then computed as:

$$\zeta = \frac{\delta}{\sqrt{4\pi^2 + \delta^2}} \tag{2.7}$$

Finally, the relationship between the viscous friction coefficient and the damping factor depends on the natural frequency ω_n and on the moment of inertia of the pendulum J_2:

$$K_{F,2} = 2\omega_n J_{2,eq} \zeta = 8.6 \cdot 10^{-6} kg \frac{m^2}{s} \tag{2.8}$$

Being:

$$\omega_n = \frac{\omega_0}{\sqrt{1-\zeta^2}} \qquad \omega_0 = \frac{2\pi}{T}$$

2.2.2. Horizontal arm parameters estimation

The selected method to tune the missing parameters was to solve an optimization problem in the time domain, in order to minimize the difference between real measures and the response of the simulated model to the same input. The measure selected to carry out the optimization process is the angle θ_1, as the $\theta_1 - \theta_2$ relation depend only on fixed parameters, making θ_2 redundant.

The kinds of input signals used are steps and sine waves, in order to be able to model both the static and dynamic behaviour of the system. Due to the approximation induced by the optimization problem, input frequencies were to be chosen to obtain a high accuracy in the the significant portion of the frequency diagram.

To get a general idea of the frequency behaviour of the system, the Bode diagram of the linearized model in 1.4.1 with datasheet parameters was traced, and frequencies selected not to miss any meaningful part:

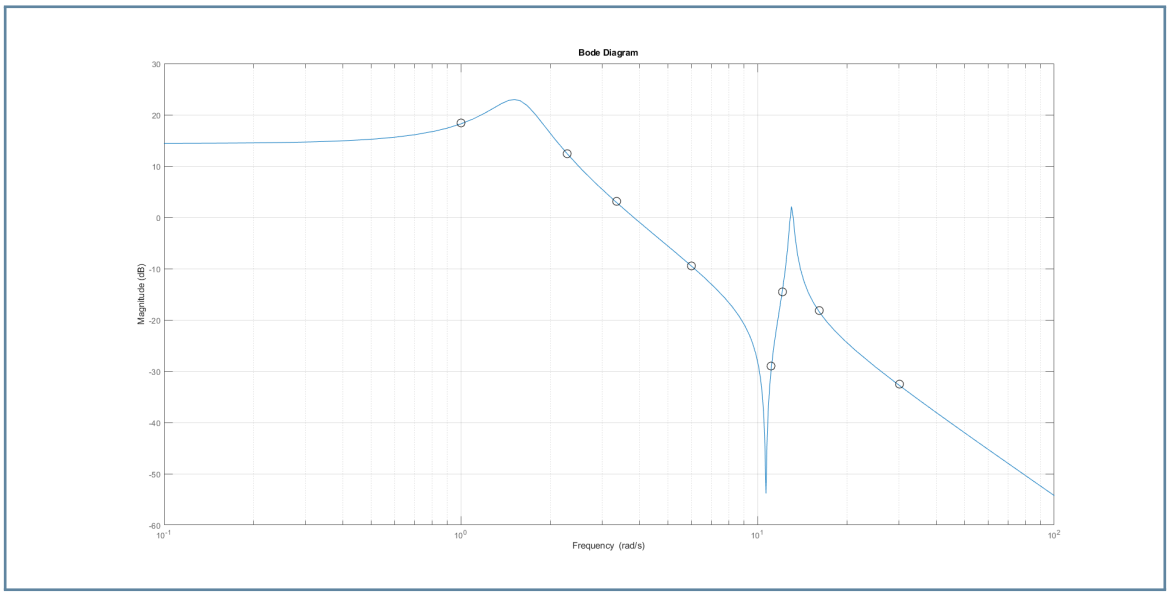


Figure 2.4: Bode diagram of the magnitude. Selected frequencies are marked

In order to simulate the behaviour of the model as function of the missing parameters, at each time sample the system of equations in 1.3 was solved for $\ddot{\theta}_1$ and $\ddot{\theta}_2$ using position and velocities of the previous time sample, and then $\dot{\theta}$ and θ were evaluated by integration over the sample time.

The simulation results and real system response to the selected inputs were concatenated to form $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}(J_{1,eq}, K_{F,1}, k_s)$ vectors, fed to the following objective function:

$$(J_{1,eq}, K_{F,1}, k_s) = argmin \sum_n (y_n - \hat{y}_n(J_{1,eq}, K_{F,1}, k_s))^2 \tag{2.9}$$

This approach achieved the results shown in Figure 2.5:

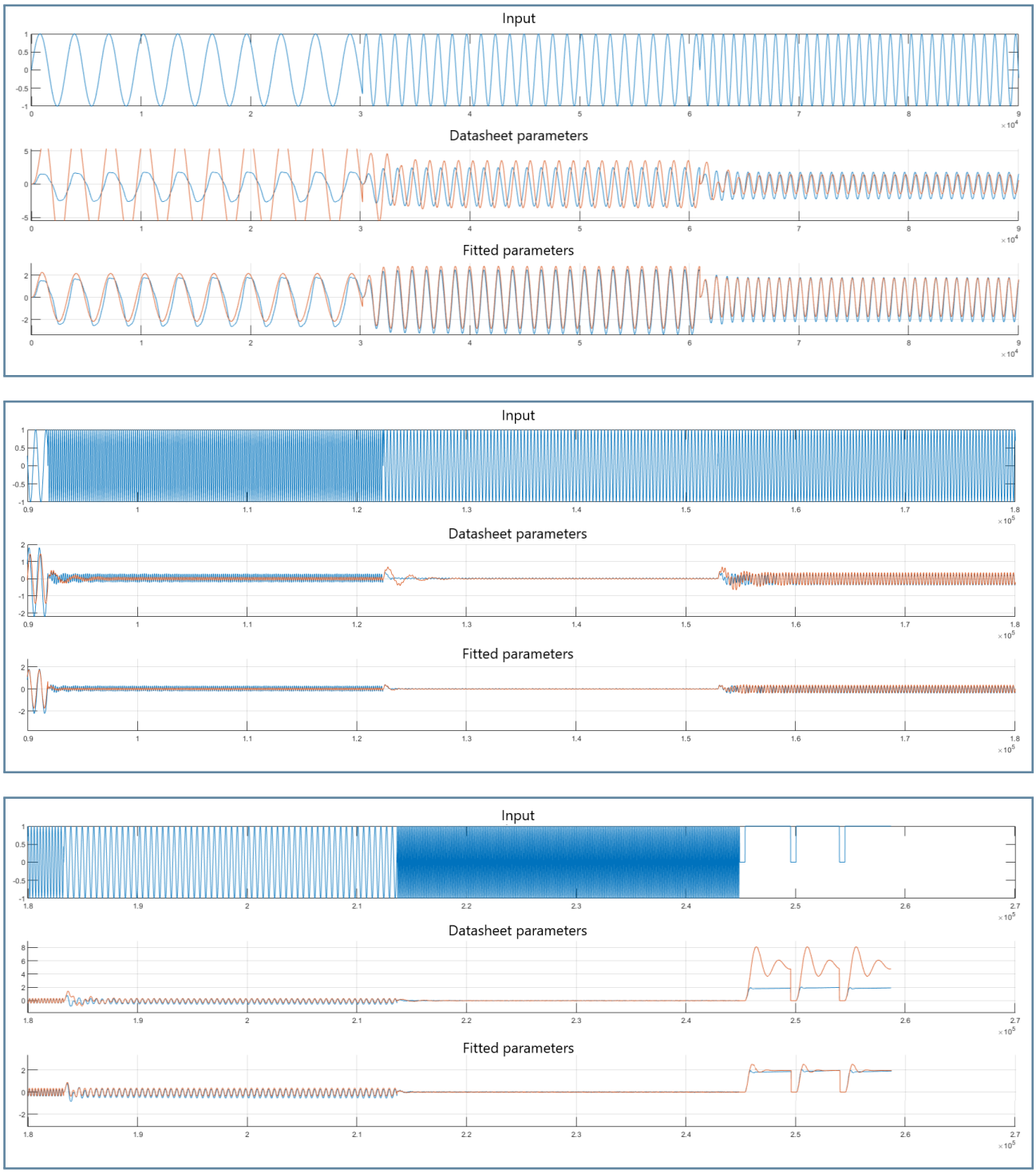


Figure 2.5: In each figure, real and simulation data is shown

2.2.3. Validation

In order to verify the correctness of the obtained results, a validation step is needed. The hypothesized model and the estimated parameters were validated in both frequency and time domain, achieving satisfactory results.

Frequency validation

In order to verify the model behaviour in the frequency domain, experiments consisting in several sinusoidal inputs at various frequencies and amplitudes were performed.
Even if in a nonlinear system domain the frequency response theorem does not hold, and outputs to sinusoidal inputs are not perfect sinusoids, sine waves best fitting them were found and their amplitude and phase graphed in Figure 2.6.
This was done for both real measures and the output of the non-linear model, whose Bode-like graph was traced using the results of hundreds of simulations having sinusoidal inputs at logarithmically equispaced frequencies.

This was performed for θ_1 only, as the evolution of the angular position of the pendulum θ_2 is highly non-linear, hindering the sinusoidal approximation method. Therefore, it was decided to validate it exclusively in the time domain.

This procedure is carried out each time validation is performed.

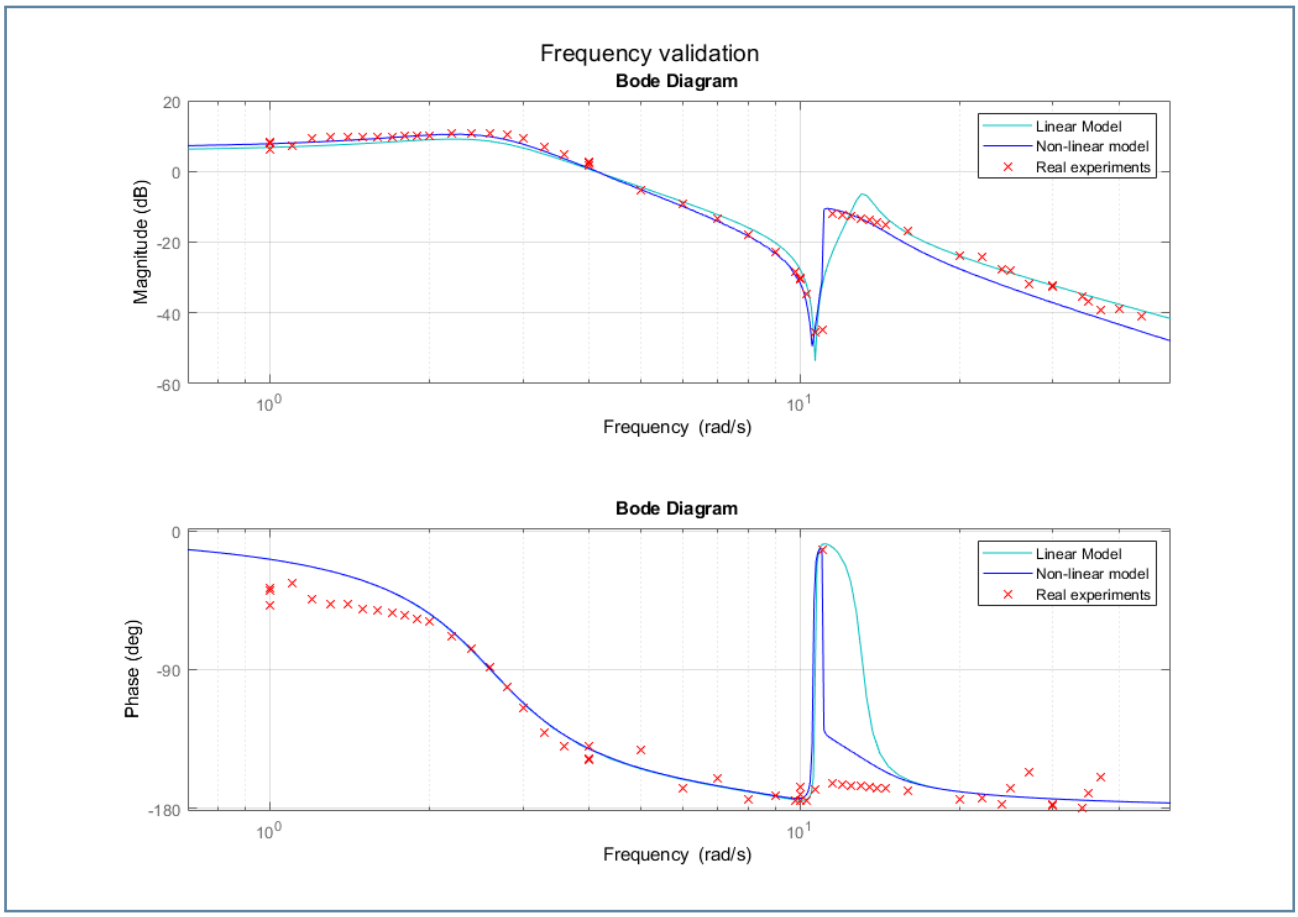


Figure 2.6: Bode diagram of the frequency validation

Time validation

Time validation was carried on with input signals different from the ones used in the fitting step, and more specifically a square wave and a white noise. Henceforth, the normalized root

mean square error (NRMSE) will be employed as a quantitative measure of the error between the real and the simulated behavior. The RMSE is always non-negative, and a value of 0 would theoretically indicate a perfect match between the two responses, each having n samples:

$$RMSE = \sqrt{\frac{\Sigma_{i=1}^{n}(y_{real} - y_{sim})^2}{n}} \tag{2.10}$$

In order to take into account the diverse scales of the signals, the RMSE is normalized by the range of the signal itself, as is most commonly performed in the literature:

$$NRMSE = \frac{RMSE}{y_{max} - y_{min}} \tag{2.11}$$

Time validation yielded the following results:

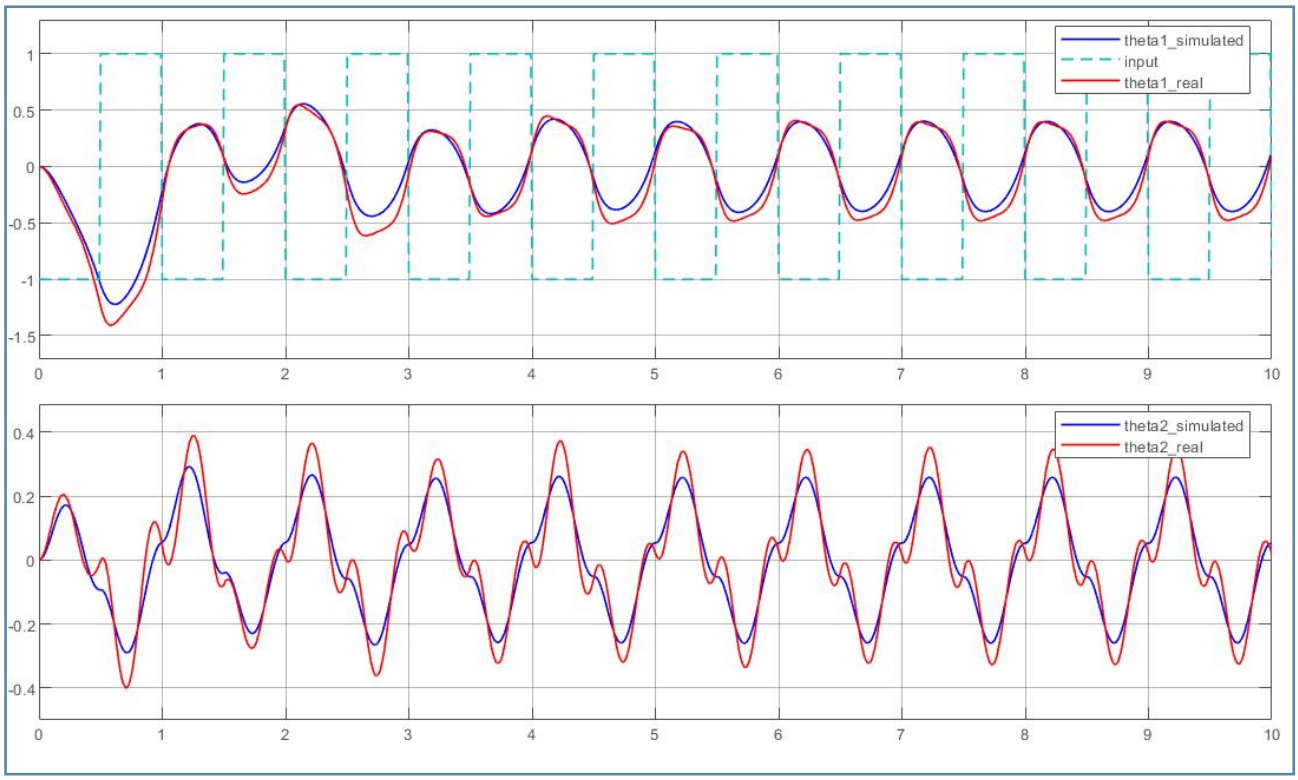


Figure 2.7: System response to a square wave input.
$NRMSE_{\theta 1} = 4.21\%$. $NRMSE_{\theta 2} = 6.64\%$

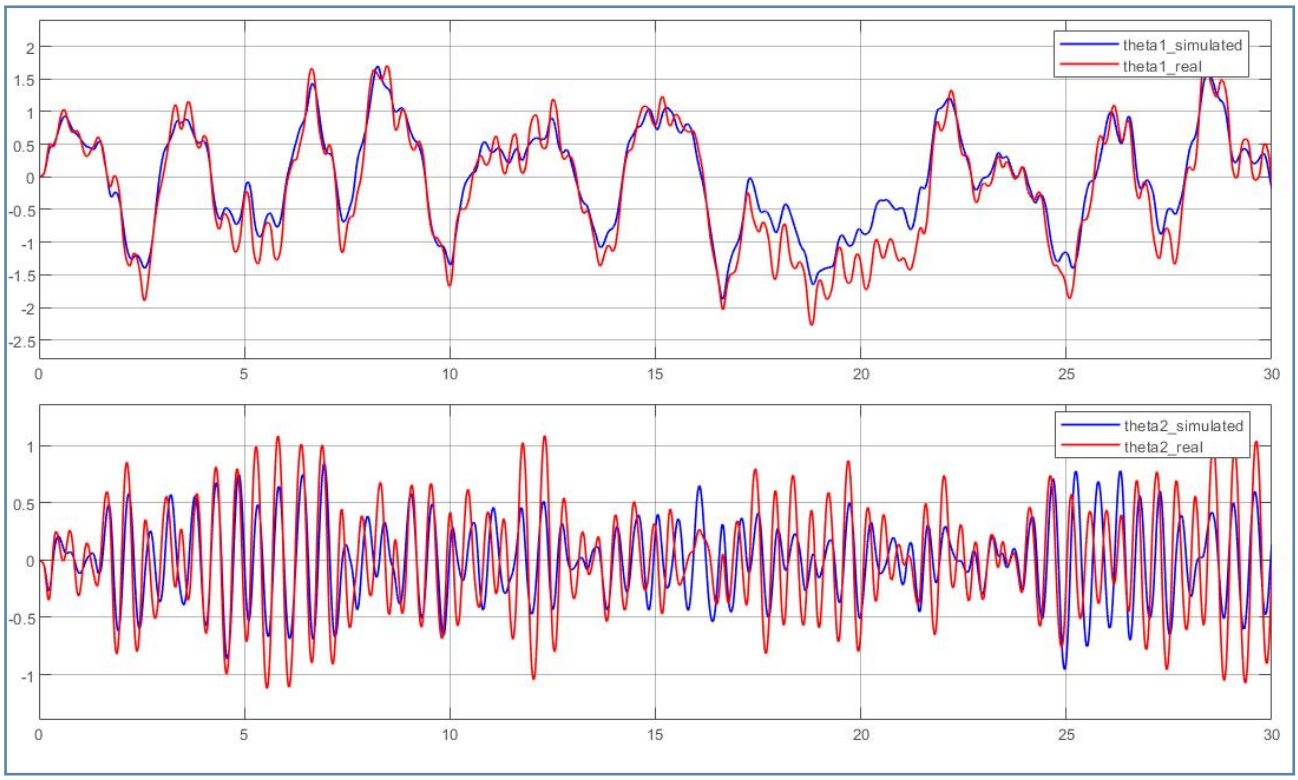


Figure 2.8: System response to a white noise input.
$NRMSE_{\theta 1} = 6.89\%$. $NRMSE_{\theta 2} = 12.92\%$

3 | Control

The main sources consulted in the realization of this chapter are [2], [3], [4].

3.1. Position control of the horizontal arm

The first task is to control the rod position when the pendulum lies near the downward position $\theta_2 = 0$. The system is stable in this configuration.
Implementing a controller allows to achieve a close tracking of the output signal θ_1 and an improved robustness. The controller used is a PID, the choice is based on the possibility of including an anti-wind up of the input signal. The DC motor can, in fact, only receive a voltage contained in the range $[-10V, 10V]$.
The following sections explain how the controller parameters were implemented, beginning with the open-loop analysis of the transfer function.

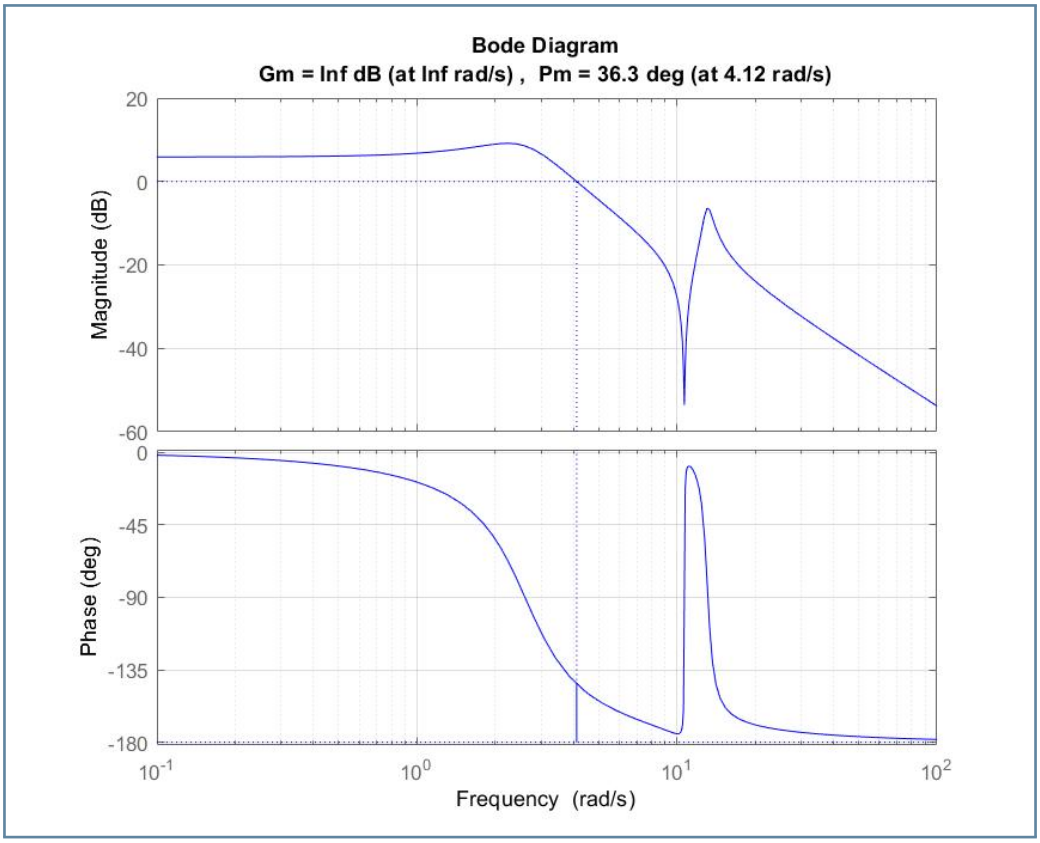


Figure 3.1: Open-loop analysis

The transfer function whose Bode plot is shown in Figure 3.1, referred to as $G(s)$, represents the relation between V_{in} and θ_1.
Its singularities are two pairs of conjugate complex poles at 12.91 *rad/s* and 2.58 *rad/s*, and a pair of conjugate complex zeros at 10.68 *rad/s*. The cutoff frequency is at $\omega_c = 4.12 rad/s$.

A frequency-based approach has been followed for the controller $C(s)$ design. The desired loop function $L(s) = C(s)G(s)$ maintains the zeros of the plant, the pair of poles at 12.91 *rad/s*, and includes a pole in zero: the latter is necessary to achieve satisfactory static performances. The final controller is a PID with an integrator, two zeros at 2.58 *rad/s* that cancel the complex poles of the system, and a pole at $25 rad/s$.

$$L(s) = \frac{120(s^2 + 0.06159s + 114.1}{s(s+25)(s^2 + 1.945s + 166.8}$$

$$C(s) = \frac{7.2622(s^2 + 3.773s + 6.656)}{s(s+25)} = 1.02 + \frac{1.93}{s} + 0.25\frac{s}{0.04s+1}$$

The resulting cut-off frequency ω_c is $3.23 rad/s$, while the phase margin is equal to 80 degrees. ω_c could not be pushed any further at higher frequencies without adding singularities to the controller. A higher gain would cause the resonance of the complex poles at 12.91 *rad/s* to cross 0DB.

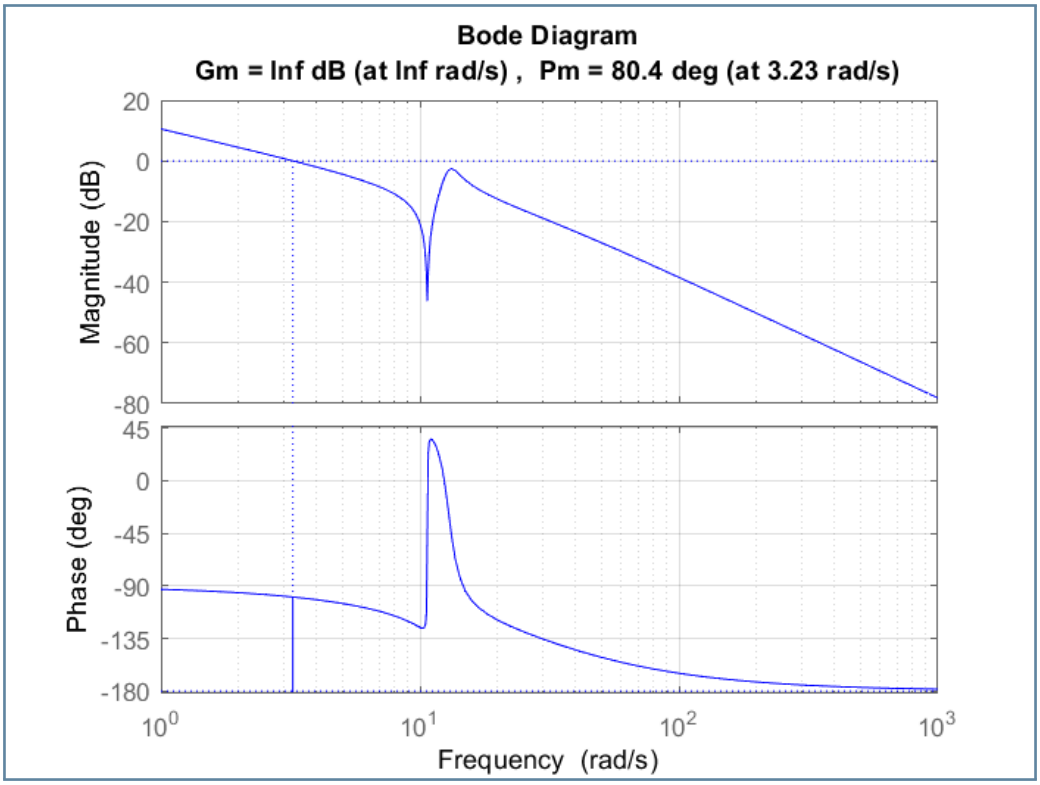


Figure 3.2: Open loop function *L(s)*

In order to obtain a satisfactorily fast system, the cut-off frequency needed to be at least over $3 rad/s$, and this motivated the preference for a PID controller over a PI. Having only one zero

would have meant that the two complex poles at 2.58 rad/s could not have been cancelled.

Results and validation

Figure 3.3 shows the comparison between the simulated system θ_1 and the measured θ_1 when controlled with the PID. The reference is a step with $0.8rad$ amplitude applied at 1 sec. The imprecision between the measure and the simulation is caused by unmodeled static friction acting when the arm moves at very low velocities.

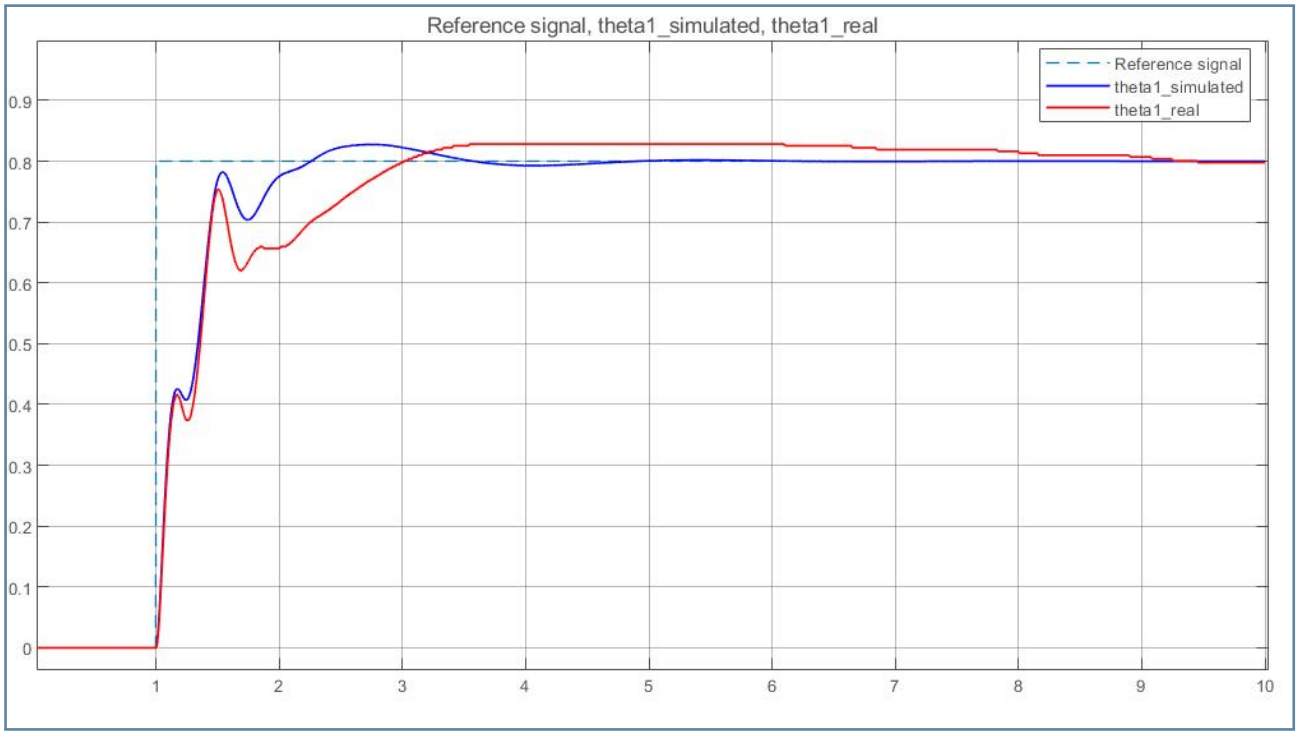


Figure 3.3: Time validation of Horizontal Arm Control

$$NRMSE_{\theta 1} = 4.59\%$$

Figure 3.4 shows the behaviour in frequency domain of the gain and phase of the transfer function from θ_1-reference and θ_1: the simulation of the controlled model resembles the real system precisely.

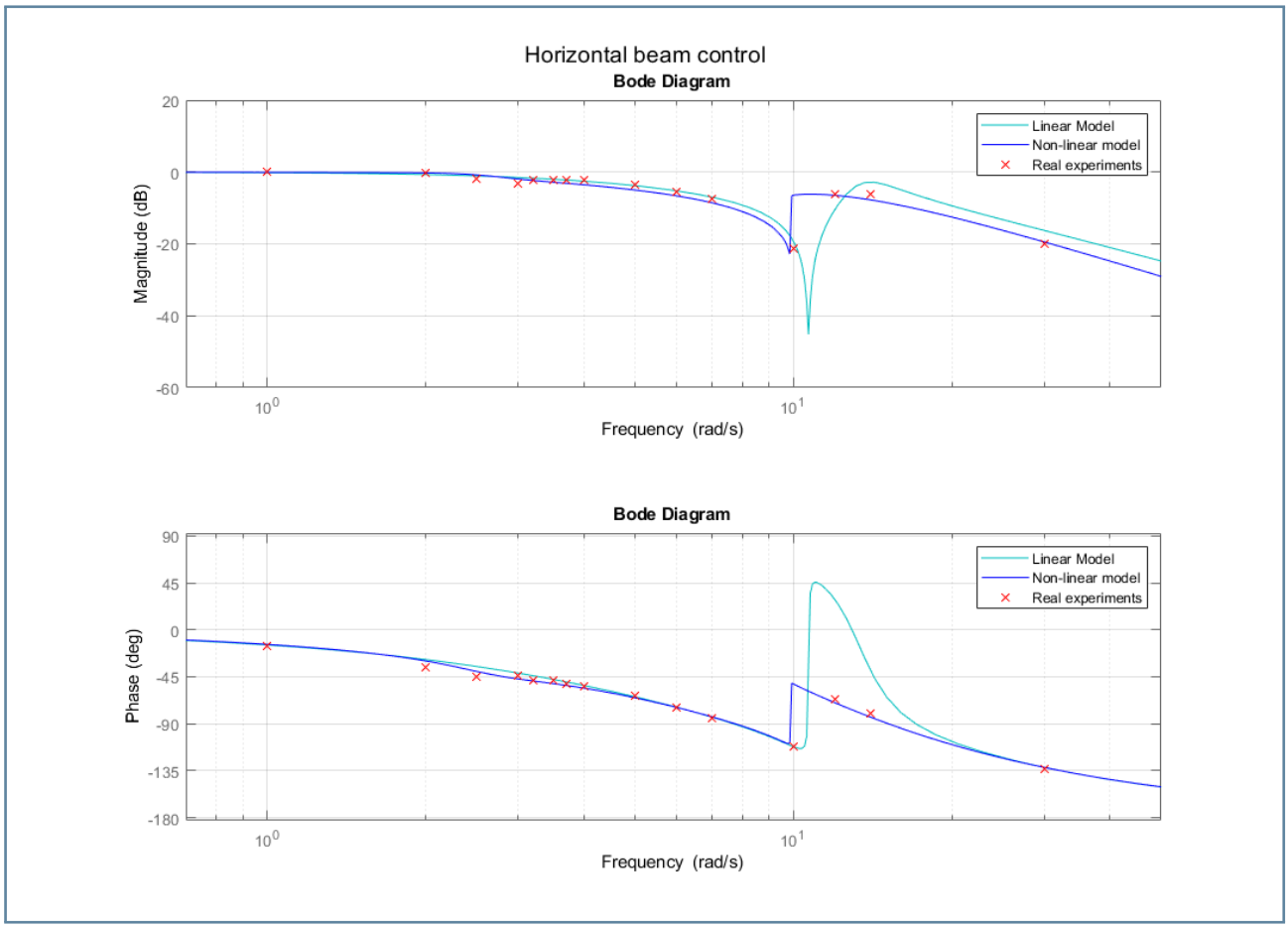


Figure 3.4: Frequency validation of Horizontal Arm Control

3.2. Stabilization of the pendulum in the upright position and position control of the horizontal arm

The stabilization of the Inverted Pendulum in the upward unstable equilibrium $x = (0 \quad \pi \quad 0 \quad 0)^T$ requires the implementation of a closed-loop control. The two techniques compared are pole placement and linear quadratic control. Both consist of the computation of the gain of a state feedback.
In the real system, only the states θ_1 and θ_2 are measured. The other two states, $\dot{\theta}_1$ and $\dot{\theta}_2$, in this first analysis are obtained by applying to the angles signals in Simulink the sequence of a derivative block and a filter to reduce the spike effects of the encoder quantization.

Enlarged System

Only one output can benefit from the action of an integrator because the system is driven by just one input signal, the voltage applied to the motor. Furthermore, the plant cannot present a derivative action on the chosen output.
Respecting the cited constraints, the system has been enlarged to include in the states the error between θ_1 and its reference, to ensure static performance. The matrices of the system with the integrator on θ_1 are denoted with the subscript i.

$$C_{\text{int}} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$

$$A_i = \begin{bmatrix} A & \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ -C_{\text{int}} & 0 \end{bmatrix} \quad B_i = \begin{bmatrix} B \\ 0 \end{bmatrix}$$

The state feedback control law assumes the following structure, where $v(t)$ is the integral of the error between θ_1 and its reference:

$$u(t) = -K \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} = - \begin{bmatrix} K_x & K_v \end{bmatrix} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix}$$

The choice of including the integrator in the study of stability comes from the necessity of respecting the physical constraints on the range of θ_1, which can approximately vary in the range of $[-\frac{3}{4}\pi, \frac{3}{4}\pi]$. In this initial phase the reference of θ_1 is a constant equal to 0.

3.2.1. Pole Placement

The design of the Pole Placement controller can be performed on the enlarged system, since the pair (A_i, B_i) is reachable. The selection of the poles of the closed-loop system is based on two main criteria:

1. Ensuring a cut-off frequency of the transfer function between θ_1 and its reference greater than 1 rad/s
2. Maintaining the control input inside the range $[-10, 10]V$ to avoid saturation.

The chosen poles of the enlarged closed-loop are $[-8 - 8.1 - 8.2 - 8.5 - 9]$, which are the eigenvalues of $A_i - B_i K_{pp}$, with $K_{pp} = [-11.06, 54.26, -2.83, 4.80, -17.59]$.

The Bode diagram of the loop function L, obtained using the linear approximation of the system around the upper equilibrium, follows in Figure 3.5:

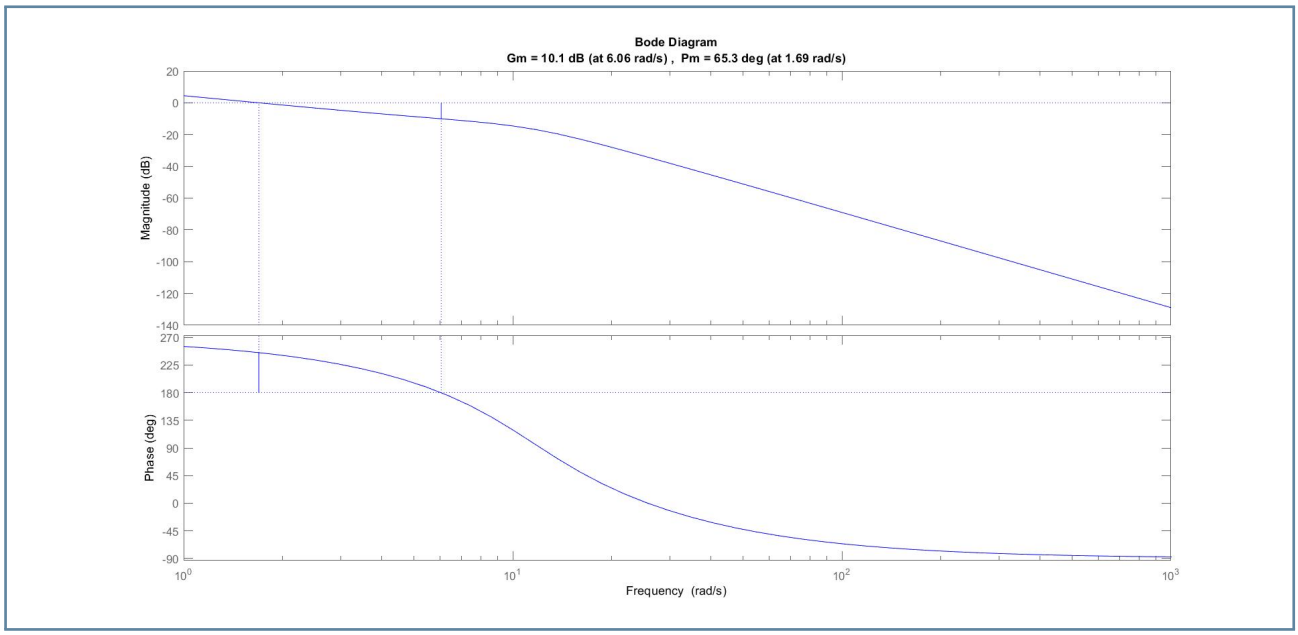


Figure 3.5: Bode of PP loop function

Results and validation

The experimental results prove that the control is able to stabilize the system in the upward position with no oscillations, even when voltage disturbances are presents. The reference tracking works with a settling time around 1 second.

Figure 3.6 shows the frequency validation of the closed loop from the θ_1-reference to θ_1. The bode diagram of the response of nonlinear models was built as explained in Section 2.2.3.

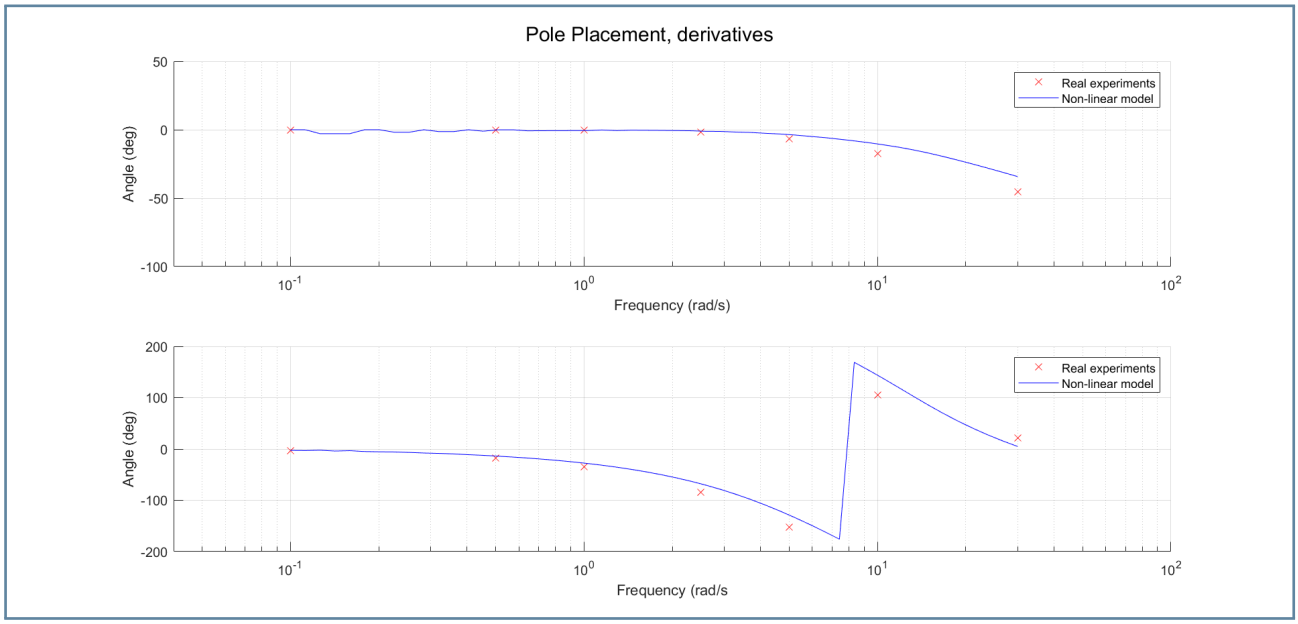
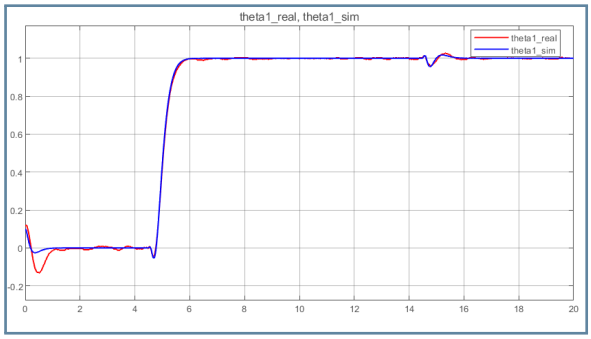


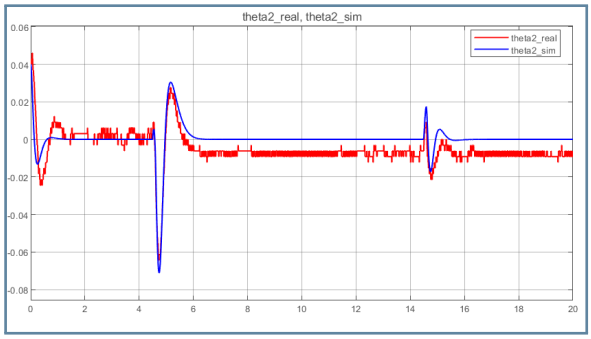
Figure 3.6: Frequency behaviour comparison

The time validation of the Pole Placement control is shown in Figure 3.7. The reference applied is a step at time 4.5 seconds. Every ten seconds,a small voltage disturbance is also applied. Noise in derivative signals is due to encoder discretization, and is especially visible in $\dot{\theta}_2$ due to θ_2 varying in the order of magnitude of the minimum encoder step.



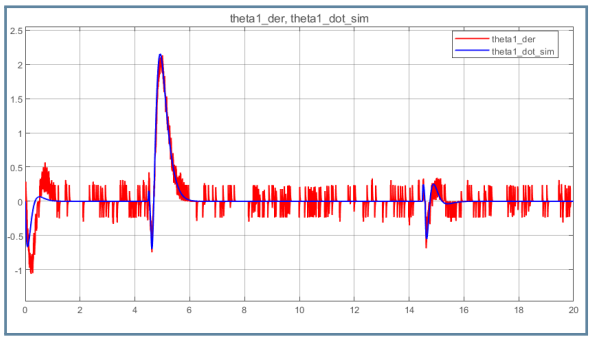
(a) θ_1

$NRMSE_{\theta 1} = 1.52\%$

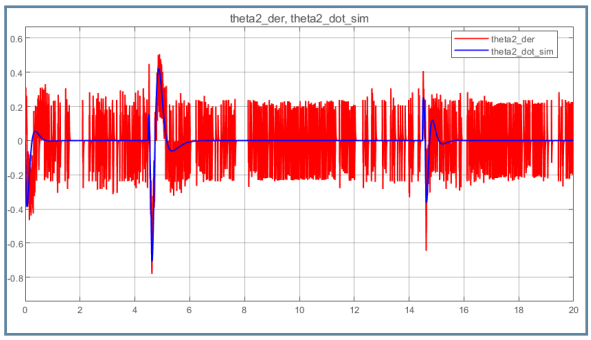


(b) θ_2

$NRMSE_{\theta 2} = 7.18\%$



(c) $\dot{\theta_1}$



(d) $\dot{\theta_2}$

Figure 3.7: Time validation of Pole Placement control

3.2.2. Linear Quadratic Control

The design parameters of the LQ control with infinite horizon are the matrices Q and R, which respectively weigh the states and input of the system in the performance index J to minimize. Both have to be symmetrical and $Q \geq 0$, $R > 0$.

$$J = \int_0^\infty x'(\tau)Qx(\tau) + u'(\tau)Ru(\tau)\, d\tau$$

The conditions to ensure an asymptotically stable closed-loop using Linear Quadratic Control are:

1. the reachability of the pair (A_i, B_i)
2. the observability of the pair (A_i, C_q), where $C_q \quad s.t. \quad Q = C_q' C_q$.

The chosen matrices are $R = 0.1$ and $C_q = [0, 10, 0.001, 0.001, 10]$.

The weight applied to each state through C_q is calibrated evaluating the necessity to strictly control them to zero:

- The coefficients associated with θ_2 and the error of θ_1 wrt reference are the highest (10), in order to ensure stability and reference tracking
- The term of C_q related to θ_1 is zero because its behaviour is regulated through the integrator.
- Terms referring to velocities have small values, both equal to 0.001.

R has been fixed to 0.1 to ensure a non-saturating input signal. The obtained gain of the state feedback is $K_{lq} = [-16.45, 80.02, -4.16, 6.57, 31.62]$.

Figure 3.8 shows the Bode graph of the loop function L, obtained using the linear model of the system:

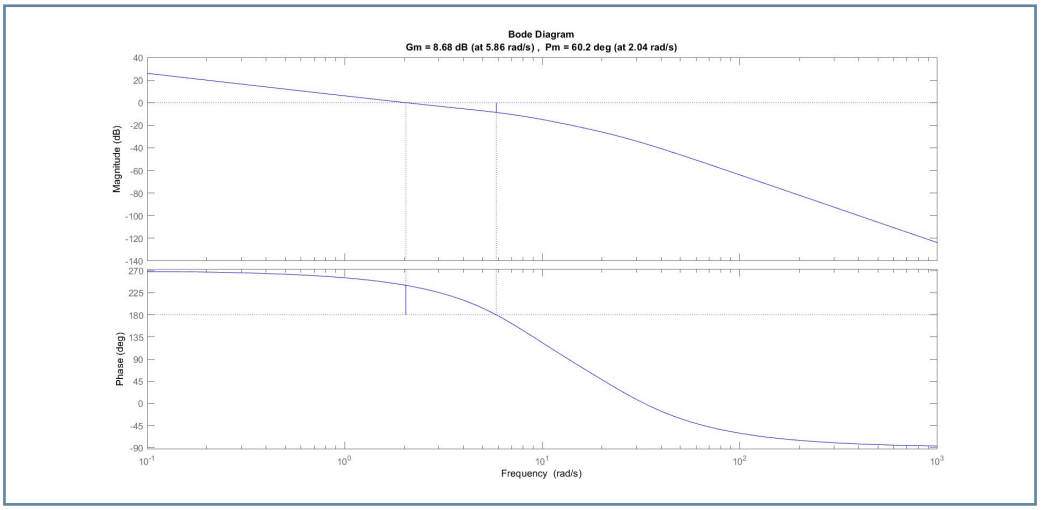


Figure 3.8: Bode of LQ loop function

Results and validation

The experimental results plotted below testify the stabilization in the upward position and the precision in reference tracking of θ_1.
Figure 3.9 shows the frequency validation of the closed loop from the θ_1-reference to θ_1. The bode diagram of the response of nonlinear models was built as explained in Section 2.2.3.

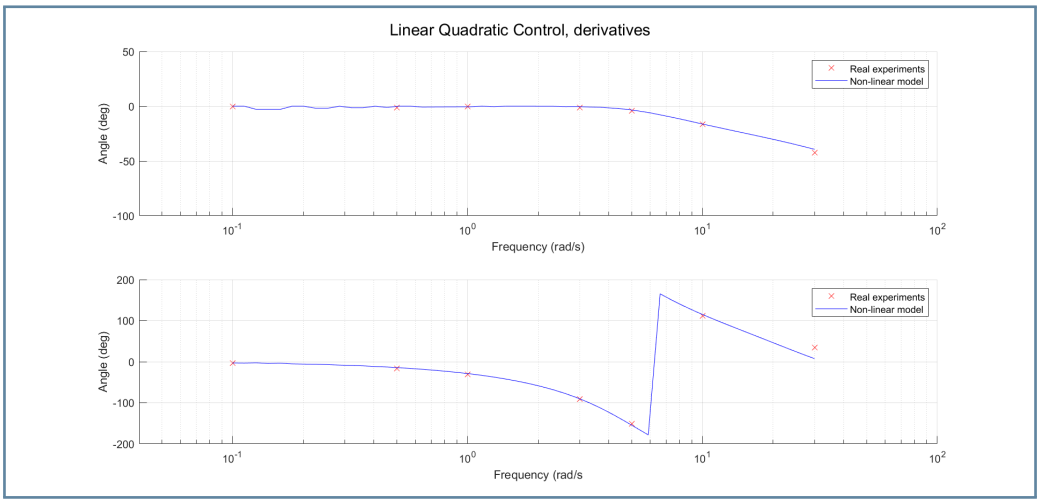
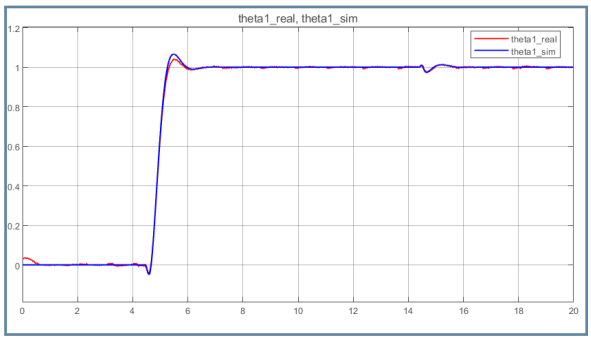


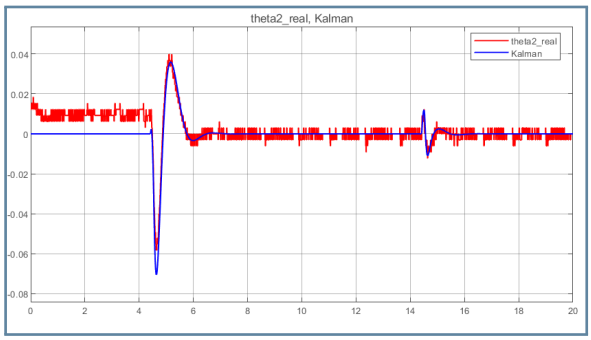
Figure 3.9: Frequency behaviour comparison

The time validation of the Pole Placement control is shown in Figure 3.10. The reference applied is a step at time 4.5 seconds. Every ten seconds a small voltage disturbance is also applied. The cause of noisy measures is discussed in Section 3.2.1



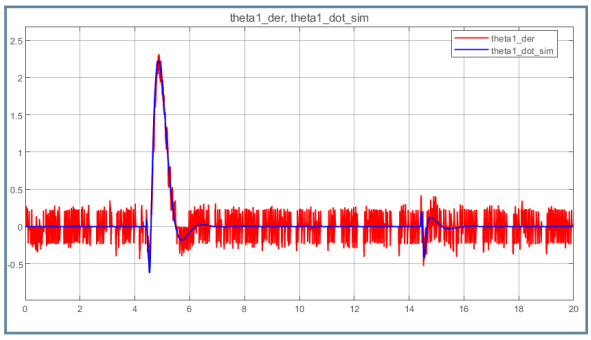
(a) θ_1

$NRMSE_{\theta 1} = 0.67\%$

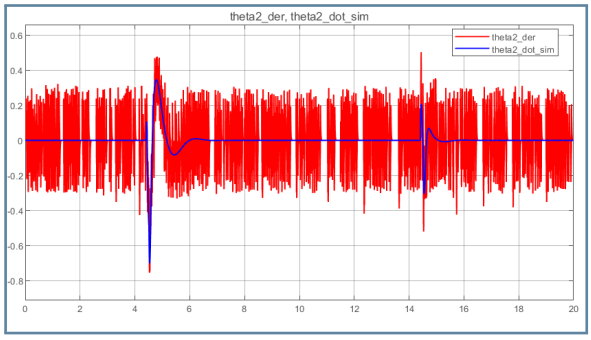


(b) θ_2

$NRMSE_{\theta 2} = 5.34\%$



(c) $\dot{\theta_1}$



(d) $\dot{\theta_2}$

Figure 3.10: Time validation of Linear Quadratic Control

Observers

The pair (A, C) is observable, therefore an alternative to the use of derivatives of the signals is the implementation of a state observer. Two different techniques have been compared: Luenberger Observer and Kalman Filter. The performances obtained by the two estimators are comparable, their implementation is illustrated in the sections below.

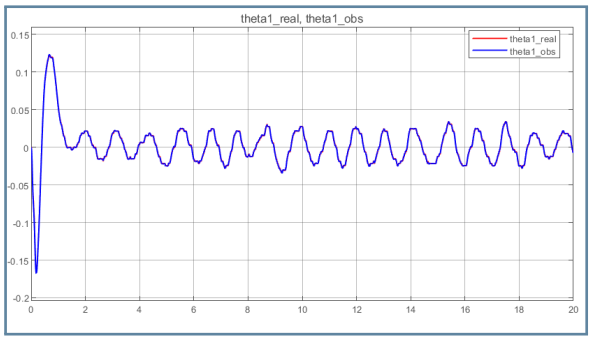
3.2.3. Pole Placement and Luenberger Observer

The gain L of the estimator is found using the place command in Matlab, where the chosen poles $[-60, -65, -58, -63]$ are way faster than the dynamic of the controlled system. Since the separation principle holds, it's possible to design the observer and the controller independently.

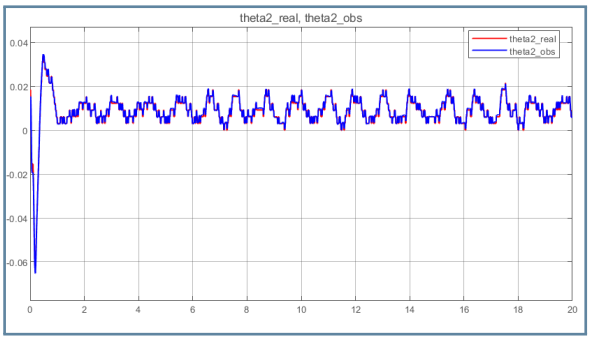
$$\dot{\hat{x}}(t) = Ax(t) + Bu(t) + L[y(t) - C\hat{x}(t) - Du(t)]$$

Results and validation

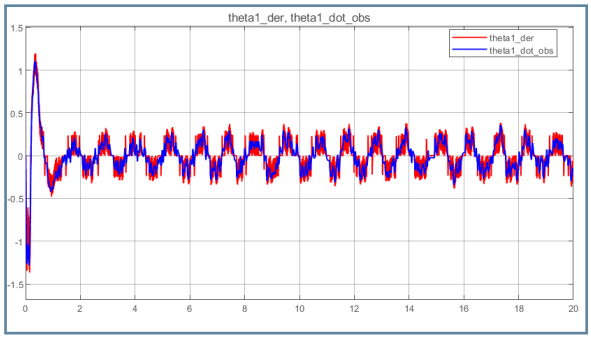
In Figure 3.11, the time validation of the four measured states θ_1, θ_2, $\dot{\theta}_1$, $\dot{\theta}_2$, and their estimate is graphed. The observer is a reliable replica of the state.
The reference chosen for θ_1 is 0. Note that the oscillations of θ_1 have amplitude of 1 degree. The cause of noisy measures is discussed in Section 3.2.1



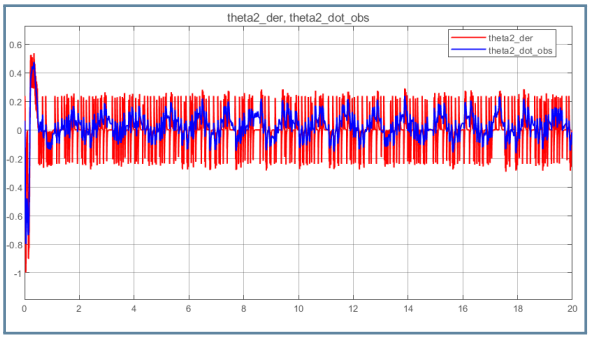
(a) θ_1
$NRMSE_{\theta 1} = 0.29\%$



(b) θ_2
$NRMSE_{\theta 1} = 0.94\%$



(c) $\dot{\theta}_1$



(d) $\dot{\theta}_2$

Figure 3.11: Luenberger Observer

Figure 3.12 shows the frequency validation of the closed loop from the θ_1-reference to θ_1. The

bode diagram of the response of nonlinear models was built as explained in Section 2.2.3.

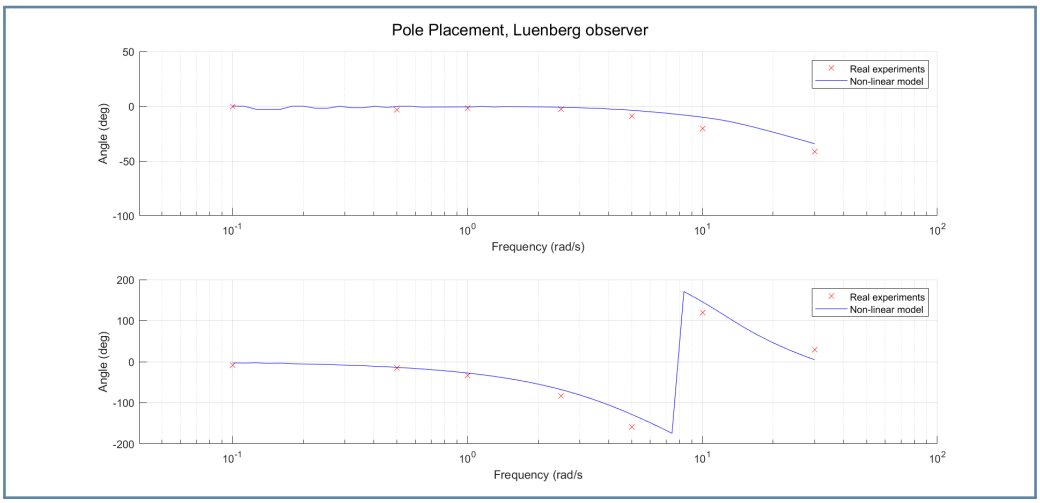


Figure 3.12: Frequency behaviour comparison

3.2.4. Linear Quadratic Control and Kalman Filter

The Kalman Filter has the same structure as the Luenberger Observer, with gain $L = PC^T R^{-1}$ (where P is the solution of the Riccati equation and R is the covariance of the measurement noise). The system is modeled as:

$$\dot{x}(t) = Ax(t) + Bu(t) + v_x(t)$$

$$y(t) = Cx(t) + v_y(t)$$

where v_x and v_y are white noises with zero mean. v_x represents the modeling error, v_y embodies the measurement error.

The first two rows of its covariance matrix Q are set to zero because no modeling error can be associated with the derivatives of the angles.
The remaining coefficients have been estimated from the comparison between the measured output and the linear simulation of the system when the input signal is a square wave, as shown in Figure 2.7. During laboratory testing, the necessity of having a faster estimator has dictated the choice of multiplying the estimated Q by 1 order of magnitude.

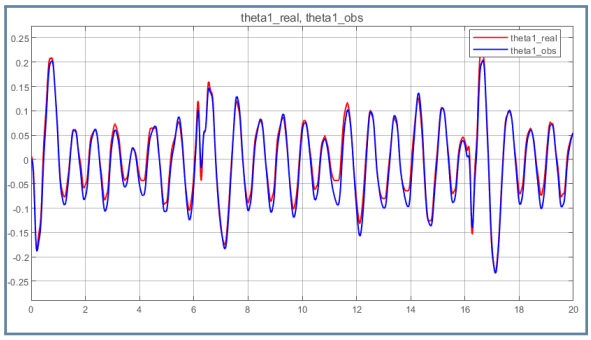
R has been set as a diagonal matrix with values equal to 0.002, the sensibility of the encoders.
The matrices were chosen such that: $Q \geq 0$, $R > 0$.
The cross-covariance matrix between the modeling noises and output noises has been set equal to zero.

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2.6 & -2 \\ 0 & 0 & -2 & 10.4 \end{bmatrix} \quad R = \begin{bmatrix} 0.002 & 0 \\ 0 & 0.002 \end{bmatrix}$$

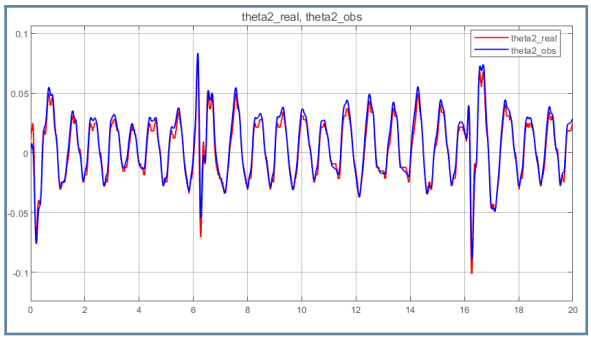
The stability of the overall system is ensured by the separation principle and the correct design of the Kalman Filter and LQ controller, but the robustness qualities of LQ are not automatically maintained.
They could be preserved using the Loop Transfer Recovery procedure, but this technique cannot be applied to the considered system, because the conditions to have the same number of inputs and outputs and invariant zeros with negative real parts are not satisfied.

Results and validation

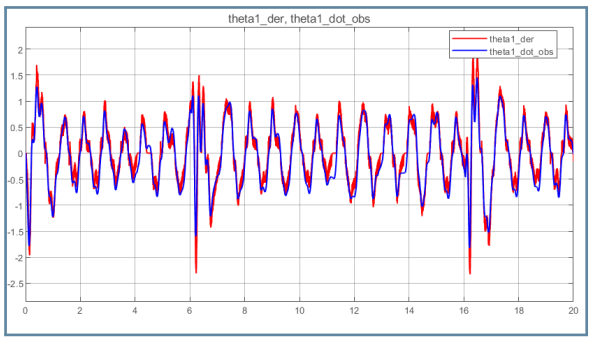
In the four graphs 3.13 are plotted the measured states θ_1, θ_2, $\dot{\theta}_1$, $\dot{\theta}_2$, and their estimate. The filter is able to produce a good esteem of the state.
The reference chosen for θ_1 is 0, and there is a small voltage disturbance applied at 6 seconds and 16 seconds. Note that the oscillations of θ_1 have amplitude of 10 degrees, caused by the combination of the disturbance with the use of the estimated state.



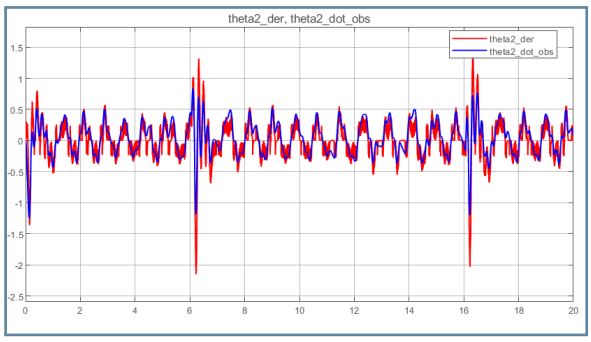
(a) θ_1
$NRMSE_{\theta 1} = 3.32\%$



(b) θ_2
$NRMSE_{\theta 2} = 2.88\%$



(c) $\dot{\theta}_1$



(d) $\dot{\theta}_2$

Figure 3.13: Kalman Filter

Figure 3.14 shows the frequency validation of the closed loop from the θ_1-reference to θ_1. The bode diagram of the response of nonlinear models was built as explained in Section 2.2.3.

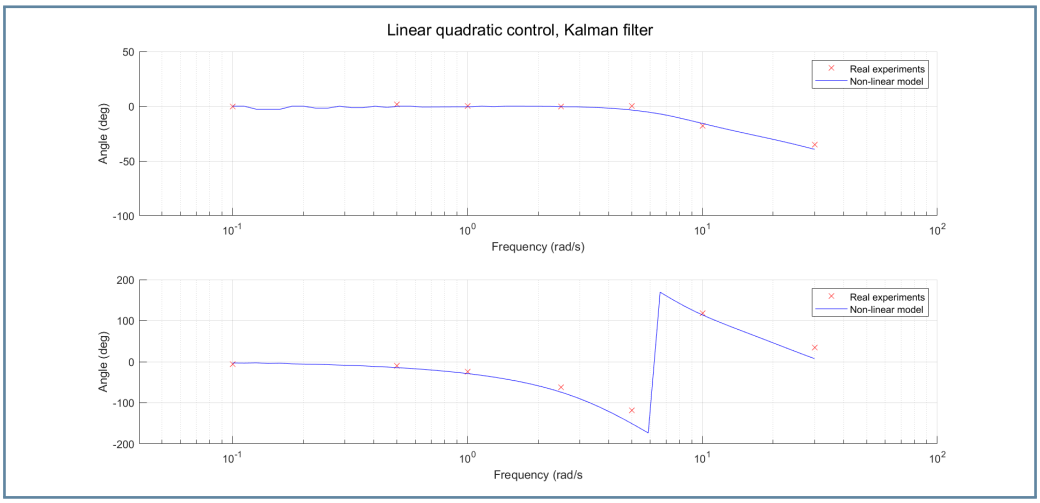


Figure 3.14: Frequency behaviour comparison

4 | Swing-up of the vertical arm

The last task of the laboratory is to realize a control able to swing up the Inverted Pendulum from the downward stable equilibrium to the upward unstable equilibrium. The two techniques used are able to bring the system into the neighborhood of the upper equilibrium; when the threshold of 160 degrees is reached, the LQ controller takes over and stabilizes the pendulum.

In order for the swing up control to start, a constant 0.4V tension is fed to the system for 0.2s, forcing a first little oscillation of the pendulum. The direction of the first oscillation must be carefully imposed, so that during the swing-up the pendulum rotates clockwise and the encoder records correctly the turn.

Both methods use the filtered derivatives of the angular positions to obtain $\dot{\theta}_1$ and $\dot{\theta}_2$.

4.1. Bang Bang Controller

The principle behind a bang-bang controller is to abruptly switch between two different states. In this instance, the control law alternates the direction of the applied tension based on the sign of the pendulum velocity. More specifically, the force is applied in the opposite direction with respect to the direction in which the pendulum is oscillating. By doing so, it's ensured that its momentum is sufficiently increasing, until an adequate amplitude is reached for the LQ controller to take over.

It's important to mention, however, that a slight but significant modification has been introduced to the traditional bang-bang logic. Instead of switching between two fixed opposite voltage values, a simulink clock block has been employed in order to continuously increase the absolute value of the input force. This ensures a more rapid swing up while also limiting the position of the horizontal rod.

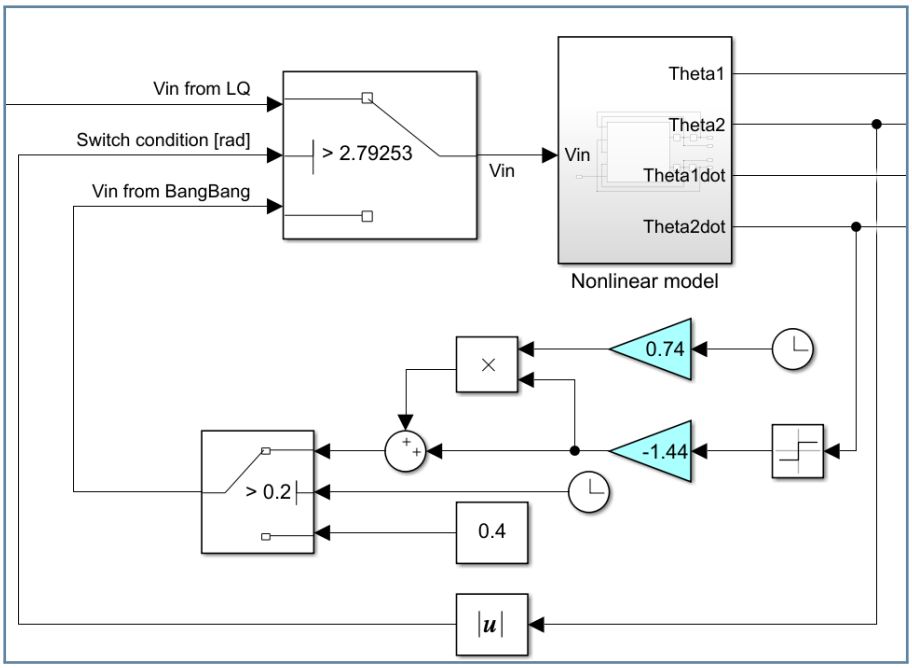


Figure 4.1: Simulink swing-up scheme

The initial voltage value and the rate at which it is increased have been carefully selected in order to minimize the time required for the swing up, hence the number of oscillations, while also limiting θ_1, thus ensuring to avoid that the movement of the rod would be obstructed by the physical limitations of the rotary servo system itself.

Results and validation

Examining the results achieved by both the real system and the simulation, it is evident that the controller successfully achieves the upright position for θ_2, as shown in Figure 4.2. Although experimental and simulated results appear not to be perfectly identical, they ensure a good representation of the actual system.
Ultimately, the swing up is achieved in 1.8 seconds at the third oscillation.

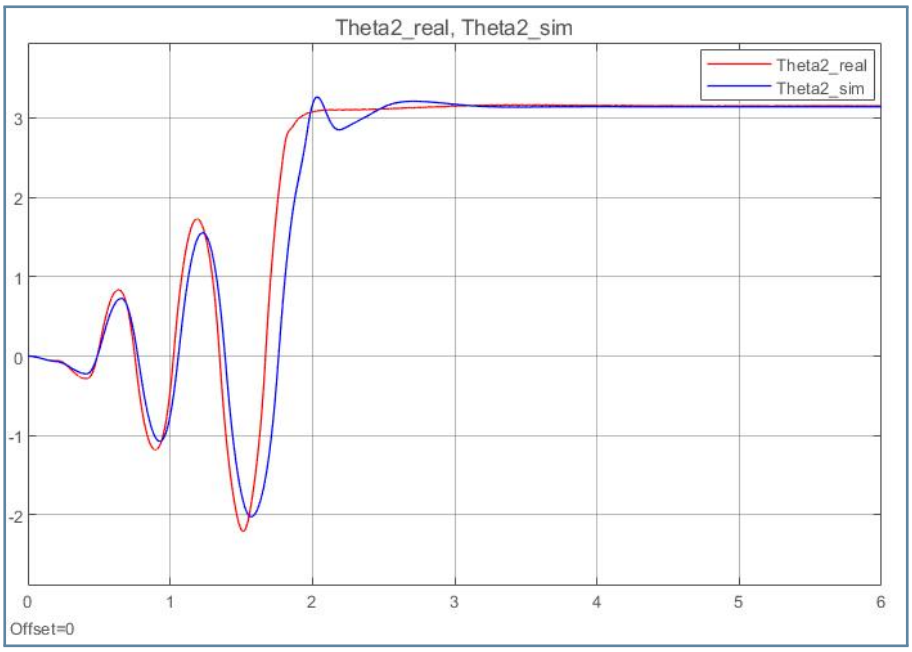


Figure 4.2: Swing Up comparison.
$NRMSE_{\theta 2} = 7.38\%$

4.2. Lyapunov based Controller

The founding idea of this control technique is to formulate a Lyapunov function of the closed loop system that has the point of minimum in the upward equilibrium. The function is defined using the energy of the pendulum.

The energy E of a pendulum hinged at one extreme can be expressed as the sum of kinetic and potential energy. The potential gravitational energy is fixed to zero in the downward equilibrium.

$$E = \frac{1}{2}(J_2 + m_2 x_2^2)\dot{\theta_2}^2 + m_2 g(x_2 - x_2 cos(\theta_2)) \tag{4.1}$$

The control aim is to reach the energy of the upward position. The quantity E_{up} includes a scaling factor of 1.1 due to the presence of friction.

$$E_{up} = 1.1 \times m_2 g(x_2 + x_2) \tag{4.2}$$

The Lyapunov function V is expressed as $V = \frac{1}{2}(E - E_{up})^2$.
V is always greater than zero, beside when the energy of the upward equilibrium is reached and the pendulum is able to swing up.
To properly define V as a Lyapunov function is necessary that its derivative $\dot{V}$ is lower than zero. This property is essential to ensure that the states evolve towards the minimum of V.
The condition $\dot{V} < 0$ is respected through the choice of $\ddot{\theta_2}$:

$$\dot{V} = (E - E_{up})\dot{\theta_2}((I_2 + m_2 x_2^2)\ddot{\theta_2} + m_2 g x_2 sin(\theta_2)) \tag{4.3}$$

$$\ddot{\theta_2} = -\frac{m_2 g x_2 sin(\theta_2)}{I_2 + m_2 x_2^2} - const(E - E_{up}) sign(\dot{\theta_2}) \tag{4.4}$$

The first term of the acceleration of the pendulum $\ddot{\theta_2}$ sets $\dot{V}$ to zero, the second term $-const(E - E_{up})sign(\dot{\theta_2}))$ is designed to ensure a negative $\dot{V}$ with a module proportional to the square of the distance from the desired energy E_{up}.
The parameter *const*, equal to 900, has been carefully tuned to ensure that the oscillation of the horizontal rod would be smaller than the physical range of θ_1.

The control signal applied to the system is the voltage of the DC motor, therefore the chosen acceleration $\ddot{\theta_2}$ must be obtained through the voltage. The relation between the two comes from solving the first equation in 1.3 for $\ddot{\theta_1}$ and substituting it second. The result is used inside a Matlab function that takes as input the state and outputs the required V_{in}.

Results and validation

In Figure 4.3 it's possible to observe the comparison between the results achieved by the real system and the simulation in terms of θ_2: the controller brings the system successfully to the upward position, and does so almost exactly as the simulation predicts.
The swing up is accomplished in 2.7 seconds at the third oscillation.

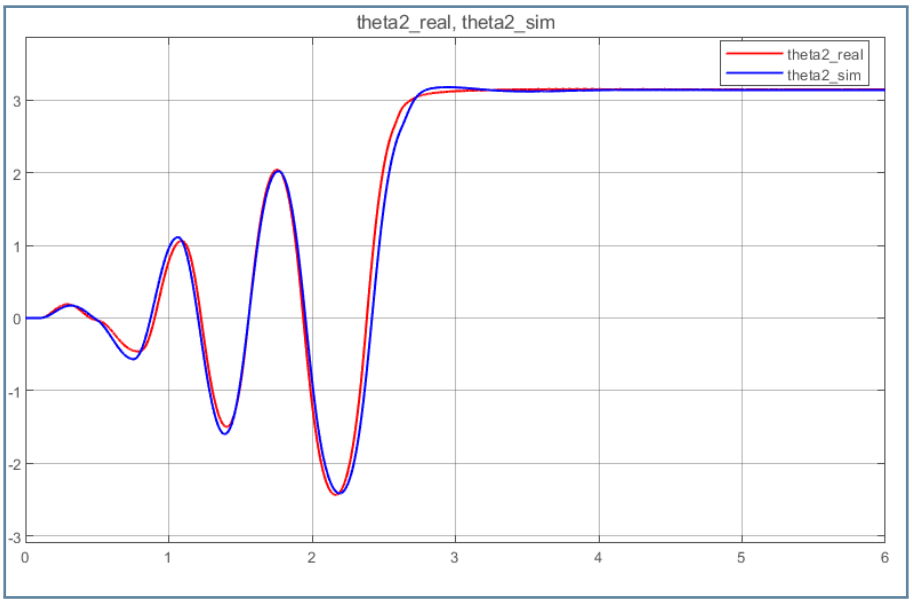


Figure 4.3: Swing Up comparison.
$NRMSE_{\theta 2} = 3.80\%$

5 | Conclusions

Significant attention was devoted to modeling the rotary inverted pendulum setup and identifying the unknown parameters that most accurately reflect reality. This step was crucial, as precise simulation is essential for developing functional controllers. The results obtained are shown in the validation section of the open loop system 2.2.3.

In the context of controlling the position of the horizontal rod when the pendulum is in the downward position, it has been determined that the use of a PID controller is the most appropriate. An important aspect to emphasize is the impact of static friction on the system's performance. Through a series of experiments, it has been observed that when operating at low voltages, the force exerted on the horizontal rod encounters difficulty in overcoming static friction. This phenomenon is illustrated in Figure 3.3, where the settling time of the real experiment exceeds that of the simulated one.
The decision not to include this friction in the model was made because the situation where the motor has to deal with such a problem only occurs in this specific task. In the stabilization of the pendulum in the upward position, the motor will no longer encounter the need to apply low voltages to track a reference, as stabilization is a more demanding task that necessitates greater energy from the motor.

Two different techniques have been experimented for stabilization control: Pole Placement and Linear Quadratic (LQ) control. While both methods are effective, upon analyzing the closed-loop frequency behavior illustrated in Figure 3.6 for Pole Placement and Figure 3.9 for LQ, it becomes evident that the gains in the LQ system ensure a higher experimental cut-off frequency compared to Pole Placement. Consequently, LQ control can be considered a slightly better option.

Both Pole Placement and LQ controllers are developed from a system represented in a linear state space, requiring the measurement of all states. However, the given setup allows for measuring only positions, not velocities. The adopted solutions are to either derive the position data or use an observer.
Deriving the positions results in good system performance, even though the velocities become highly disturbed. This disturbance occurs because the position is measured by an encoder, causing the signal to appear as a series of steps rather than a smooth function. Using an observer avoids this problem, but it results in poorer overall performance.

The final task is to swing up the pendulum until it reaches the upright position. In chapter 4, two techniques achieving excellent results are presented.

The bang-bang controller is fast but not very reliable, as its performance slightly differs from the simulation. The Lyapunov-based controller, on the other hand, offers a more consistent and reliable approach. In fact, even if the pendulum is disturbed to the point of falling down, the controller allows the system to reach the upright position again.

6 | Bibliography

[1] Quanser Inc. 2016. User Manual - QUBE-Servo2 Experiment - Set Up and Configuration.

[2] Paolo Bolzern, Riccardo Scattolini, Nicola Schiavoni. 2023. Advanced and Multivariable Control. Società Editrice Esculapio s.r.l.

[3] Luca Magni, Riccardo Scattolini. 2008. Fondamenti di Controlli Automatici. McGraw-Hill.

[4] MathWorks. MathWorks - Makers of MATLAB and Simulink. `https://it.mathworks.com/?s_tid=gn_logo`