

ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF ELECTRICAL, ELECTRONIC, AND INFORMATION ENGINEERING  
(DEI)



## Project Report

---

# Modeling and Control of a Mechanical System to Reach a Fixed Equilibrium Position Under Design Specifications

---

**Student:** Ouardi Abess

**Matriculation No.:** 0001040657

**Professor:** Lorenzo Marconi

**Course:** Automatic Control T-2

**Academic Year:** 2022/2023

Bologna, Italy

# Contents

<b>1</b>	<b>Introduction and Problem Definition</b>	<b>1</b>
<b>2</b>	<b>System Modeling</b>	<b>3</b>
2.1	Mechanical Model . . . . .	3
2.2	DC Motor Model . . . . .	4
<b>3</b>	<b>Operating Point and Linearization</b>	<b>5</b>
3.1	Equilibrium Point . . . . .	5
3.2	State Variables . . . . .	6
3.3	Nonlinear State Equations . . . . .	6
3.4	Linearization Around the Equilibrium . . . . .	6
3.5	Remarks . . . . .	7
<b>4</b>	<b>Design Specifications</b>	<b>8</b>
4.1	Static Specifications . . . . .	8
4.2	Dynamic Specifications . . . . .	9
4.3	Disturbance Rejection . . . . .	9
4.4	Noise Rejection . . . . .	10
4.5	Summary of Requirements . . . . .	10
<b>5</b>	<b>Controller Design</b>	<b>12</b>
5.1	Need for Lead Compensation . . . . .	12
5.2	Choice of Crossover Frequency . . . . .	12
5.3	First Lead Network – Practical Tuning . . . . .	13
5.4	Second Lead Network – Additional Phase Compensation . . . . .	14
5.5	Summary of Design Choices . . . . .	15
5.6	Root–Locus Considerations . . . . .	15
5.7	Addition of a Corrective Zero . . . . .	16
5.8	Fine Adjustment of the Lead Networks . . . . .	16
5.9	Final Remarks on the Adjusted Design . . . . .	17
<b>6</b>	<b>Filter in Cascade with Controller</b>	<b>18</b>
6.1	Problem Statement . . . . .	18
6.2	Filter Type Evaluation . . . . .	18
6.3	Final Choice: Bessel Filter . . . . .	19

## CONTENTS

---

6.4	Integration with the Regulator . . . . .	20
6.5	Remarks . . . . .	20
<b>7</b>	<b>Simulation and Validation</b>	<b>22</b>
7.1	Full Architecture . . . . .	22
7.2	Nonlinear Plant Subsystem . . . . .	23
7.3	Controller Subsystem . . . . .	24
7.4	Reference Pre-Filtering and Trade-Off . . . . .	25
7.5	Validation and Results . . . . .	26
<b>8</b>	<b>Anti-Windup Circuit</b>	<b>29</b>
8.1	Chosen Anti-Windup Structure: $\Gamma$ -Conditioning . . . . .	29
8.2	With vs Without Anti-Windup: Results . . . . .	32
<b>9</b>	<b>Digital Controller Implementation</b>	<b>34</b>
9.1	Introduction to Digital Control . . . . .	34
9.2	Sampling and Discretization . . . . .	34
9.3	Discretization Procedure in MATLAB . . . . .	36
9.3.1	Sampling choice (numerical) . . . . .	36
9.3.2	Reconstruction/processing delay model . . . . .	36
9.3.3	Controller discretization method (generalized Tustin) . . . . .	36
9.3.4	Numerical validation in MATLAB . . . . .	37
9.4	Anti-Aliasing Filter Design . . . . .	37
9.4.1	Motivation and Role . . . . .	37
9.4.2	Cutoff Frequency Selection . . . . .	38
9.4.3	MATLAB Implementation . . . . .	39
9.5	Simulink Digital Implementation . . . . .	39
9.5.1	Architecture and Sampling Blocks . . . . .	39
9.5.2	Digital Controller Block . . . . .	40
9.5.3	Simulation Results . . . . .	40
<b>10</b>	<b>Conclusions</b>	<b>43</b>

# CHAPTER 1

---

## Introduction and Problem Definition

---

The project focuses on the analysis and control of a nonlinear flexible mechanism with one degree of freedom, developed within the framework of the course *Controlli Automatici T-2*. The mechanism is a six-bar linkage actuated by a DC motor and designed to exhibit nonlinear dynamic behavior due to its flexible components. The main objective is to regulate the angular position of the moving link to a desired equilibrium value while ensuring high precision, robustness, and compliance with given performance specifications.

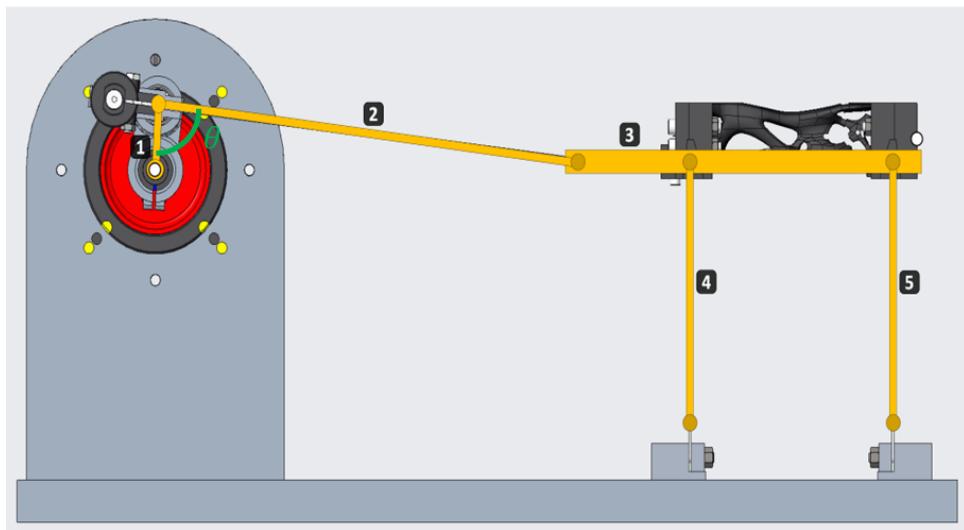


Figure 1.1: Prototype of the flexible six-bar linkage mechanism used for the project. The system consists of rigid and flexible links, resulting in a nonlinear single-degree-of-freedom structure. The goal of control is to bring the link angle  $\theta$  to a specified equilibrium position.

The mechanism is characterized by a periodic potential energy function  $U(\theta)$  and a variable inertia term  $J(\theta)$ , which together define its nonlinear dynamics. These properties make the system an excellent benchmark for studying advanced control design techniques applied to flexible mechanical systems.

## Actuation and Measurement Setup

The mechanism is driven by a DC motor and subjected to external loads acting on the end effector. The overall torque applied to the moving element is expressed as:

$$\tau = C_M - \beta \dot{\theta} + d_C$$

where  $C_M = K_t I_a$  is the motor torque,  $\beta$  is the viscous friction coefficient, and  $d_C$  represents external disturbances.

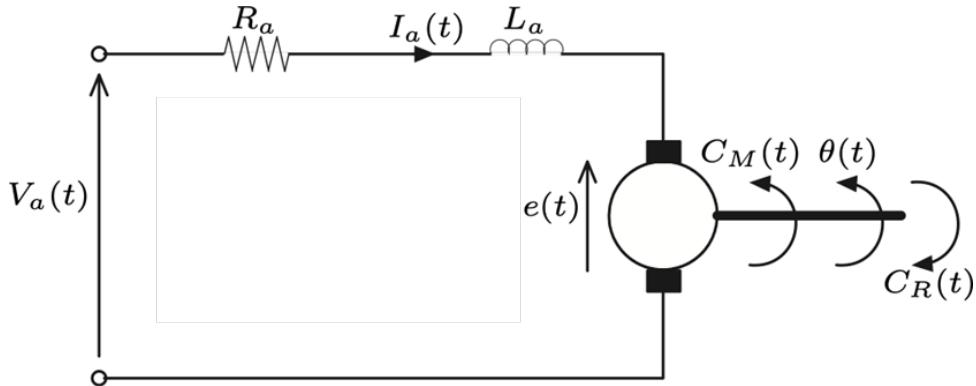


Figure 1.2: Simplified electromechanical scheme of the DC motor driving the mechanism. The control input is the armature voltage  $V_a$ , while the measurable output for feedback is the angular position  $\theta(t)$  obtained from an encoder.

The parameters of the DC motor are summarized as follows:

$$R_a = 29.4 \times 10^{-3} \Omega, \quad L_a = 1.29 \times 10^{-3} \text{ H}, \quad K_t = K_e = 0.0540 \text{ N}\cdot\text{m/A}.$$

The motor torque acts on the mechanical system to produce the desired motion, while the encoder provides the only available measurement for feedback control.

## Project Objective

The final goal of the project is to:

- Model the electromechanical system and identify a stable equilibrium configuration;
- Design and tune a controller capable of bringing the mechanism to this equilibrium while meeting all performance and robustness specifications;
- Validate the complete control strategy through simulations in MATLAB/Simulink, including nonlinear effects, disturbances, and actuator limitations.

## CHAPTER 2

---

# System Modeling

---

This chapter presents the mathematical formulation of the system, starting from the mechanical dynamics of the six-bar linkage and then introducing the DC motor model that provides the actuation torque.

## 2.1 Mechanical Model

The dynamics of the flexible mechanism can be derived using the Lagrange equation:

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}_i} \right) - \frac{\partial T}{\partial q_i} + \frac{\partial U}{\partial q_i} + \frac{\partial R}{\partial q_i} = Q_i$$

where:

- $T$  is the kinetic energy,
- $U$  is the potential energy,
- $R$  is the dissipation function representing viscous losses,
- $q_i$  are the generalized coordinates,
- $Q_i$  are the generalized external forces or torques.

For this single-degree-of-freedom system ( $q = \theta$ ), the kinetic energy depends on the variable inertia  $J(\theta)$ , and the Lagrange equation becomes:

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{\theta}} \right) - \frac{\partial T}{\partial \theta} + \frac{\partial U}{\partial \theta} = Q$$

with

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{\theta}} \right) = J(\theta) \ddot{\theta} + \frac{\partial J}{\partial \theta} \dot{\theta}^2, \quad \frac{\partial T}{\partial \theta} = \frac{1}{2} \frac{\partial J}{\partial \theta} \dot{\theta}^2, \quad Q = K_t I_a - \beta \dot{\theta} + d_C.$$

Substituting the above terms yields the mechanical equation of motion:

$$J(\theta) \ddot{\theta} + \frac{1}{2} \frac{\partial J}{\partial \theta} \dot{\theta}^2 - \frac{\partial U}{\partial \theta} = K_t I_a - \beta \dot{\theta} + d_C.$$

This equation represents a nonlinear second-order system in  $\theta$ , where the nonlinearity arises from the periodic dependence of  $J(\theta)$  and  $U(\theta)$  on the angular position.

## 2.2 DC Motor Model

The DC motor that actuates the mechanism is described by the standard electrical equation:

$$V_a(t) = R_a I_a(t) + L_a \dot{I}_a(t) + K_e \dot{\theta}(t)$$

where:

- $R_a = 29.4 \times 10^{-3} \Omega$  is the armature resistance,
- $L_a = 1.29 \times 10^{-3} \text{ H}$  is the armature inductance,
- $K_t = K_e = 0.0540 \text{ N m/A}$  is the torque and back-EMF constant.

The electrical input is the armature voltage  $V_a(t)$ , and the motor torque applied to the mechanical system is

$$C_M(t) = K_t I_a(t).$$

An encoder mounted on the moving link provides the measured angle  $\theta(t)$ , which serves as the only feedback signal available for control.

## CHAPTER 3

---

# Operating Point and Linearization

---

The analysis and control design are based on the linearized model of the nonlinear system derived in the previous chapter. The first step is the identification of a suitable equilibrium point, followed by the computation of the corresponding state–space representation.

## 3.1 Equilibrium Point

The unforced nonlinear system admits multiple equilibrium configurations due to the periodic nature of its potential and inertia functions. Let  $\theta_s$  denote a stable equilibrium point of the unforced system, satisfying:

$$\theta_s \in [0, \pi].$$

Around this configuration, the control objective is to regulate the motion of the mechanism such that  $\theta(t) \rightarrow \theta_s$  under external disturbances and reference variations.

[Internal Stability] A linear time-invariant system with state matrix  $A$  is:

- Stable if and only if all eigenvalues of  $A$  have real parts less than or equal to zero, and those with zero real part are simple in the minimal polynomial;
- Asymptotically stable if and only if all eigenvalues of  $A$  have strictly negative real parts.

## 3.2 State Variables

To express the system in a standard form, the following state variables are defined:

$$x_1 = \theta, \quad x_2 = \dot{\theta}, \quad x_3 = I_a, \quad u = V_a.$$

The output variable is the measured angular position:

$$y(t) = \theta(t) = x_1(t).$$

## 3.3 Nonlinear State Equations

Based on the mechanical and electrical models, the nonlinear state equations can be written as:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{1}{J(\theta)} \left[ -\frac{\partial U}{\partial \theta} - \beta x_2 + K_t x_3 + d_C(t) \right] \\ \dot{x}_3 = -\frac{R_a}{L_a} x_3 - \frac{K_e}{L_a} x_2 + \frac{1}{L_a} u \end{cases}$$

## 3.4 Linearization Around the Equilibrium

To design the controller, the nonlinear equations are linearized around the equilibrium ( $\theta_s, \dot{\theta} = 0, I_a = I_{a,s}$ ). Let the small-signal variations be:

$$\delta\theta = \theta - \theta_s, \quad \delta\dot{\theta} = \dot{\theta}, \quad \delta I_a = I_a - I_{a,s}, \quad \delta V_a = V_a - V_{a,s}.$$

The resulting linearized model can be expressed in state-space form:

$$\delta\dot{x}(t) = A \delta x(t) + B \delta u(t), \quad \delta y(t) = C \delta x(t)$$

with

$$A = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{1}{J(\theta_s)} \frac{\partial^2 U}{\partial \theta^2} \Big|_{\theta_s} & -\frac{\beta}{J(\theta_s)} & \frac{K_t}{J(\theta_s)} \\ 0 & -\frac{K_e}{L_a} & -\frac{R_a}{L_a} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L_a} \end{bmatrix}, \quad C = [1 \ 0 \ 0].$$

<pre>%evaluation of equilibrium disp('Eigenvalues of the linearized system with theta=0.1443 :'); Eigenvalues of the linearized system with theta=0.1443 : disp(eig(A)); 1.0e+02 * 0.0130 + 0.0000i -0.1105 + 1.4398i -0.1105 - 1.4398i</pre>	<pre>%ensure the equilibrium disp('Eigenvalues of the linearized system with theta=4.8689 :') Eigenvalues of the linearized system with theta=4.8689 : disp(eig(A)) -2.1239 + 0.0000i -10.6357 +99.6891i -10.6357 -99.6891i</pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Punto di equilibrio instabile

Punto di equilibrio stabile

Figure 3.1: Representation of the nonlinear system and its linearization around the equilibrium point  $\theta_s$ .

### 3.5 Remarks

- The matrix  $A$  encapsulates the local dynamics of the system near the equilibrium; its eigenvalues determine the open-loop stability.
- The state variable  $x_1$  corresponds directly to the measurable output  $\theta(t)$ , which will be used for feedback control.
- The linear model provides the foundation for the subsequent controller design and frequency-domain analysis.

## CHAPTER 4

---

# Design Specifications

---

The controller must be designed on the linearized model of the system around the stable equilibrium point  $\theta_s \in [0, \pi]$ . The objective is to ensure precise reference tracking, rejection of external disturbances, and robustness to high-frequency measurement noise, while maintaining stable and well-damped dynamics.

## 4.1 Static Specifications

The control system must guarantee:

- Zero steady-state error in the presence of:
  - a step reference input  $w(t)$  with maximum amplitude of  $15^\circ$  (overall reference  $\theta_s + w(t)$ );
  - a step torque disturbance  $d_C(t)$  of amplitude  $\pm 0.2 \text{ N m}$ .

Since both the reference and the disturbance are constant signals, the closed-loop system must ensure a finite steady-state value of the control input and zero steady-state tracking error:

$$\lim_{t \rightarrow \infty} e(t) = 0.$$

From classical control theory, this condition requires that the open-loop transfer function  $G(s)$  contains at least one pole at the origin, i.e.

$$g = \text{number of poles at the origin in } G(s) \geq 1,$$

so that the system is of type 1 with respect to position tracking. In the Bode diagram, this condition corresponds to an initial slope of approximately  $+20 \text{ dB/dec}$  in the low-frequency region.

## 4.2 Dynamic Specifications

To achieve satisfactory transient performance and robustness, the following conditions must be met:

- Maximum overshoot: less than 5% of the step response.
- Settling time:  $t_s < 0.5$  s (evaluated with the 1% criterion).
- Phase margin:  $\varphi_m > 55^\circ$  to guarantee sufficient robustness.

These requirements are translated into frequency–domain conditions using standard second–order approximations. The desired damping ratio corresponding to a 5% overshoot is:

$$\delta = \frac{-\ln(0.05)}{\sqrt{\pi^2 + \ln^2(0.05)}} = 0.69,$$

and the natural frequency ensuring a 0.5 s settling time is:

$$\omega_n \geq \frac{4.6}{\delta t_s} = 13.33 \text{ rad/s.}$$

Hence, the open–loop crossover frequency must be approximately  $\omega_c \approx 13.3$  rad/s, compatible with the desired closed–loop bandwidth.

## 4.3 Disturbance Rejection

The system must attenuate the tracking error in the presence of a sinusoidal torque disturbance  $d_C(t)$  of amplitude 0.1 N m at low frequency:

$$\omega < 3.14 \text{ rad/s.}$$

The required attenuation on the error signal is at least 10 times:

$$\frac{|E(j\omega)|}{|d_C(j\omega)|} \leq \frac{1}{10}.$$

In decibel form, this corresponds to an attenuation greater than 20 dB at low frequencies, accounting for the dynamics of the plant.

## 4.4 Noise Rejection

For high-frequency measurement noise  $n(t)$  with amplitude up to  $3^\circ$  and frequency:

$$\omega > 10\,000 \text{ rad/s},$$

the control system must achieve an attenuation factor of at least 100 on the tracking error:

$$\frac{|E(j\omega)|}{|n(j\omega)|} \leq \frac{1}{100}.$$

This translates to an attenuation of approximately 40 dB (or 160 dB when considering power gain) at the specified frequencies.

## 4.5 Summary of Requirements

Table 4.1 summarizes all specifications that the designed controller must satisfy.

Table 4.1: Performance and robustness requirements.

Requirement	Target Value
Steady-state error (step input / disturbance)	0
Reference step amplitude	$15^\circ$
Disturbance torque amplitude	$\pm 0.2 \text{ N m}$
Maximum overshoot	< 5%
Settling time (1%)	< 0.5 s
Phase margin	> $55^\circ$
Disturbance attenuation ( $d_C(t)$ )	> $10 \times$ (20 dB) for $\omega < 3.14 \text{ rad/s}$
Noise attenuation ( $n(t)$ )	> $100 \times$ (40 dB) for $\omega > 10^4 \text{ rad/s}$

As a visual verification of these specifications, the open-loop transfer function of the plant was plotted in the frequency domain. Figure 4.1 shows the magnitude and phase Bode diagrams of the system response, together with shaded regions corresponding to the desired performance constraints. The highlighted areas represent, respectively:

- the low-frequency region where high gain is required to ensure zero steady-state error and effective disturbance rejection;
- the mid-frequency region around the crossover, which must provide the target bandwidth and sufficient phase margin for stability and damping;
- the high-frequency region, where strong attenuation is necessary to suppress measurement noise.

It can be observed that the uncompensated plant does not fully meet all design specifications: although the low-frequency gain is adequate, both the phase margin and the noise attenuation are below the required levels. Therefore, a suitable controller must be designed to reshape the open-loop response and guarantee that all performance and robustness criteria are satisfied.

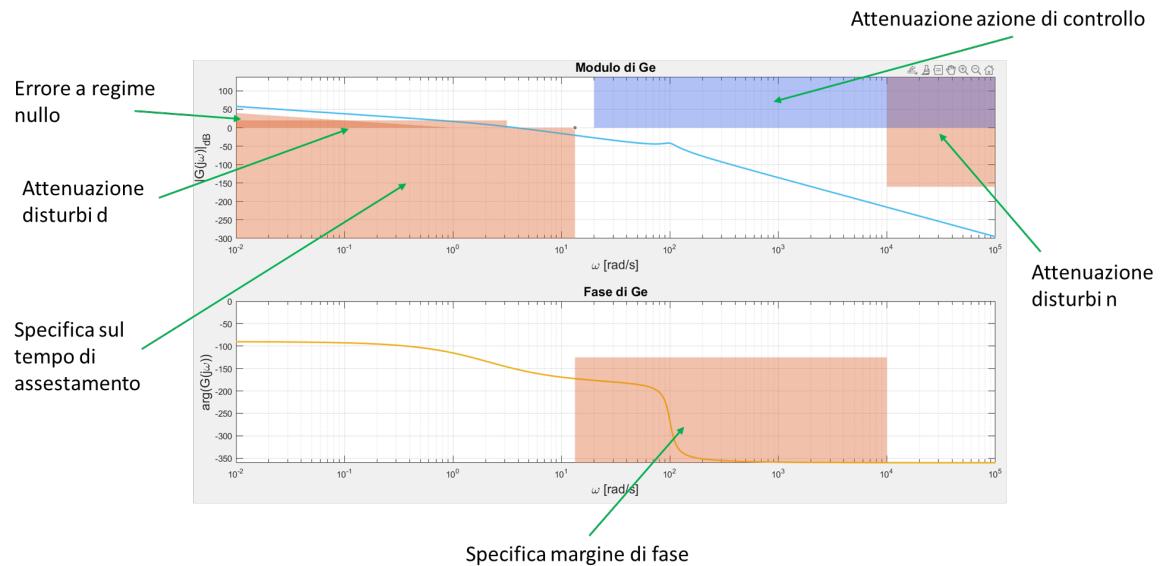


Figure 4.1: Bode diagram of the open-loop transfer function  $G_e(j\omega)$  with shaded regions indicating the required performance zones.

## CHAPTER 5

---

# Controller Design

---

The controller design aims to satisfy the performance and robustness requirements of Chapter 4 through frequency–domain compensation. An inspection of the open-loop Bode diagram of the linearized plant showed that, within the desired crossover region, the system lacks the necessary phase margin. To recover it while maintaining steady-state accuracy, a sequence of lead (phase–advance) networks is introduced.

## 5.1 Need for Lead Compensation

In the frequency range corresponding to the target crossover, the phase of the uncompensated plant never exceeds the minimum required by the specification ( $\varphi_m > 55^\circ$ ). Consequently, additional positive phase must be supplied by one or more compensators of the form:

$$R_i(s) = K_i \frac{1 + \tau_i s}{1 + \alpha_i \tau_i s}, \quad 0 < \alpha_i < 1.$$

Each network contributes a limited amount of phase advance between its zero and pole frequencies. The damping ratio previously derived,  $\delta = 0.69$ , corresponds to an equivalent phase margin of about  $70^\circ$ , thus setting the reference for the compensator design.

## 5.2 Choice of Crossover Frequency

The desired closed-loop speed, defined by the specification  $t_s < 0.5$  s, implies an approximate crossover frequency:

$$\omega_c \approx 13.3 \text{ rad/s.}$$

However, the uncompensated system does not reach the necessary phase margin at this frequency. Lowering  $\omega_c$  would degrade dynamic performance, while increasing it would further reduce the phase margin. Hence,  $\omega_c$  is retained in the same range, and lead networks are added to provide the required phase boost without altering the low-frequency type 1 structure that ensures zero steady-state error.

### 5.3 First Lead Network – Practical Tuning

A first phase-advance network is designed to add roughly  $50^\circ$  of phase at a frequency of:

$$\omega_{c1} = 35 \text{ rad/s.}$$

The desired gain correction of +4 dB yields the following parameters:

$$\alpha_1 = 0.01, \quad \rho_1 = 0.13, \quad \tau_1 = 0.03714.$$

Its transfer function is therefore:

$$R_1(s) = K_1 \frac{1 + \tau_1 s}{1 + \alpha_1 \tau_1 s}.$$

After inserting this network, the gain  $K_1$  was **tuned graphically from the Bode diagram**, adjusting its value until the compensated open-loop magnitude intersected the 0 dB line exactly at  $\omega_{c1}$ . This ensures that the maximum phase contribution occurs where it is most effective.

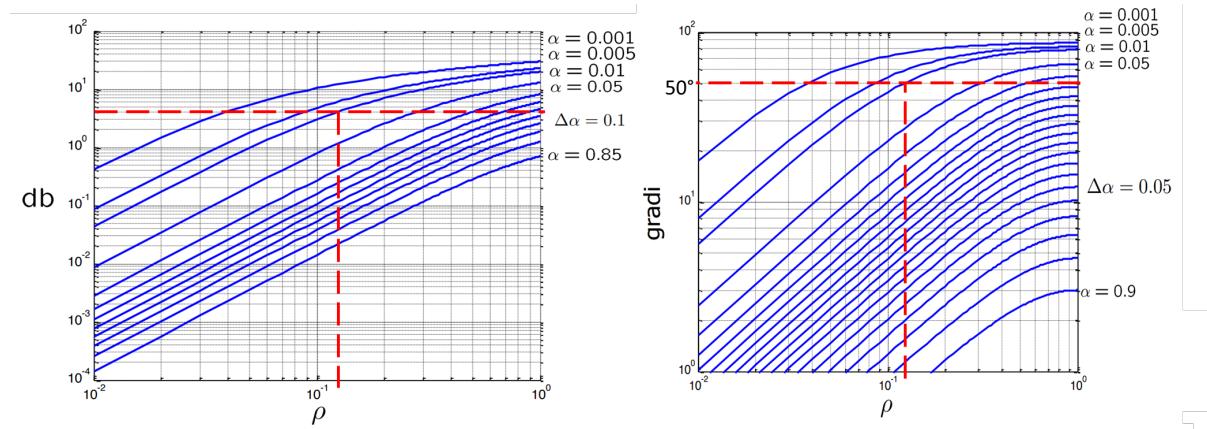


Figure 5.1: First lead compensator: practical tuning based on Bode-diagram adjustment.

## 5.4 Second Lead Network – Additional Phase Compensation

Although the first compensator increased the phase margin, the extended system still did not fully meet the desired value once robustness and bandwidth were considered. A second lead network was therefore introduced to refine the crossover slope and provide an extra  $30^\circ$  of phase at a slightly lower frequency:

$$\omega_{c2} = 25 \text{ rad/s},$$

with a small amplitude correction of +1.7 dB. Its parameters are:

$$\alpha_2 = 0.001, \quad \rho_2 = 0.022, \quad \tau_2 = 0.02783,$$

and the transfer function:

$$R_2(s) = K_2 \frac{1 + \tau_2 s}{1 + \alpha_2 \tau_2 s}.$$

The overall dynamic compensator then becomes:

$$R(s) = K_c R_1(s) R_2(s) = K_c \frac{(1 + \tau_1 s)(1 + \tau_2 s)}{(1 + \alpha_1 \tau_1 s)(1 + \alpha_2 \tau_2 s)},$$

where  $K_c$  is the final proportional gain ensuring unit magnitude at the chosen crossover frequency.

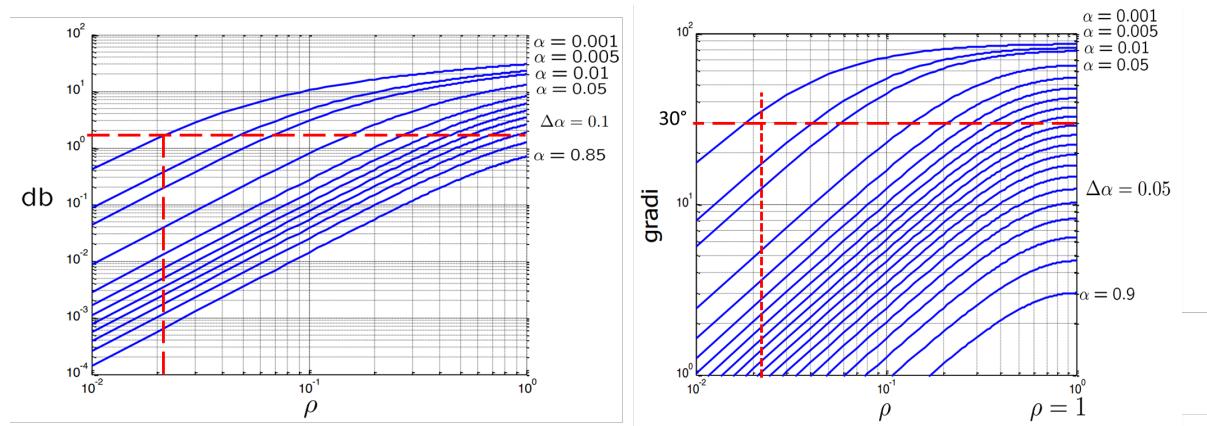


Figure 5.2: Second lead compensator: refinement of phase margin and crossover behavior.

## 5.5 Summary of Design Choices

- Two lead networks are required to achieve the specified phase margin while preserving low-frequency accuracy.
- The first network provides the primary phase boost; the second corrects the residual margin deficit and stabilizes the open-loop slope.
- Gain tuning is performed iteratively from the Bode diagram to ensure the crossover condition and verify that both magnitude and phase targets are met.
- The resulting regulator maintains type 1 behavior and forms the basis of the complete dynamic controller.

After implementing the two lead networks presented in the previous section, the open-loop response satisfied most of the design objectives in terms of bandwidth and steady-state behavior. However, the analysis of the closed-loop performance revealed two remaining issues:

- The specification on the maximum overshoot was not fully met, with a response still slightly under-damped.
- The attenuation of high-frequency noise did not reach the required 160 dB; the achieved attenuation was approximately 100 dB, which was insufficient for the design target.

## 5.6 Root-Locus Considerations

To address the overshoot issue, the evolution of the closed-loop poles was examined through the root-locus diagram. By increasing the frequency at which the lead-network poles act, the real component  $\sigma_a$  of the complex-conjugate poles increases, attracting them toward the real axis and thereby improving the damping ratio. This operation, however, shifts the frequency band of both poles and zeros toward higher values.

Although this shift improves the transient response, it also modifies the Bode phase curve. At higher frequencies, the zeros and poles intervene later, reducing the phase margin at the crossover region and moving the system away from the desired  $\varphi_m = 55^\circ$ . So both poles and zeros of the compensators were scaled by a common factor, preserving their relative separation and shape while improving the damping ratio.

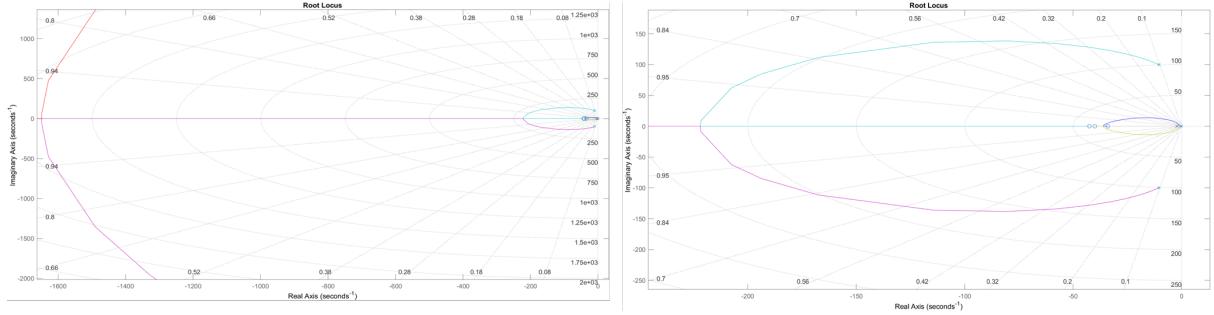


Figure 5.3: Root–locus reasoning: increasing the pole frequencies improves damping but alters the effective phase margin.

## 5.7 Addition of a Corrective Zero

After frequency scaling, the phase margin at the desired crossover frequency was still slightly below the target. To recover the missing phase, an additional zero was introduced in the vicinity of the existing zeros of the lead networks:

$$R_{\text{corr}}(s) = (1 + \tau_z s) R(s)$$

This zero provided a modest increase in phase in the neighborhood of the crossover, without significantly affecting the low–frequency gain or high–frequency attenuation. Its placement was chosen empirically based on the Bode diagram, ensuring that the final open–loop phase exceeded the 55° margin requirement.

## 5.8 Fine Adjustment of the Lead Networks

As the design approached the desired behavior, small deviations between theoretical and practical tuning became noticeable. Practical tuning graphs, typically used as reference in design manuals, introduce tolerances that depend on the chosen templates and the specific dynamics of the plant. To compensate for these discrepancies, the ratio between numerator and denominator of the compensator was slightly modified:

$$R'(s) = K_c \frac{(1 + \tau'_1 s)(1 + \tau'_2 s)}{(1 + \alpha'_1 \tau'_1 s)(1 + \alpha'_2 \tau'_2 s)},$$

where the parameters  $\tau'_i$  and  $\alpha'_i$  differ slightly from their nominal values. This adjustment allowed for minor corrections in phase and magnitude, aligning the practical results with the ideal theoretical regulator.

## 5.9 Final Remarks on the Adjusted Design

- The scaling of poles and zeros to higher frequencies improved the damping ratio but required compensatory actions to maintain phase margin.
- The additional zero was crucial for restoring the phase margin to approximately  $55^\circ$  at the desired crossover frequency.
- Small parametric adjustments were justified by the intrinsic tolerances of practical tuning procedures and by the minor differences among standard Bode templates.
- After these refinements, the regulator provided the desired transient performance, while remaining compatible with the next design stages involving filtering and anti-windup protection.

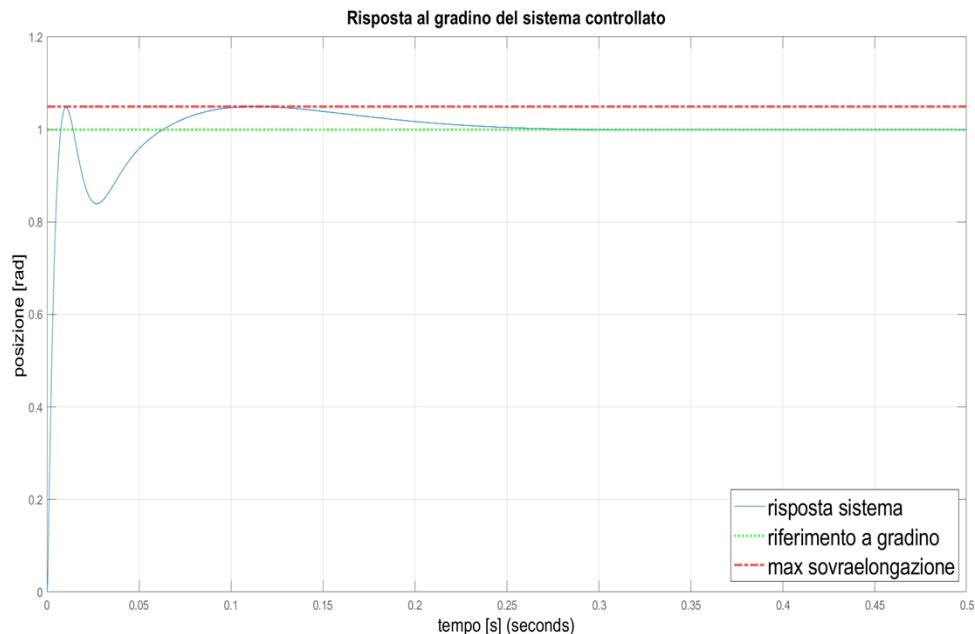


Figure 5.4: Step response behaviour

# CHAPTER 6

---

## Filter in Cascade with Controller

---

Following the regulator tuning, the closed-loop system still failed to satisfy the high-frequency noise attenuation requirement of 160 dB. Although the lead-compensated regulator provided good phase and transient performance, its magnitude roll-off was not steep enough to suppress the high-frequency measurement noise  $n(t)$  defined in the specifications. To meet this condition, a low-pass filter was designed and inserted in cascade with the controller.

### 6.1 Problem Statement

From the Bode analysis, the open-loop system guaranteed an attenuation of approximately 100 dB at the noise frequencies, still 60 dB below the target. Therefore, an additional filtering stage was required to achieve a continuous, monotonic attenuation beyond the desired cutoff frequency, while preserving the phase margin of the compensated regulator.

### 6.2 Filter Type Evaluation

Several classical low-pass filters were evaluated in simulation:

#### Chebyshev Type II and Elliptic

These filters exhibited a notch-like behaviour, characterized by sharp local attenuations followed by passband recovery. Such non-monotonic attenuation is undesirable in this application, as it introduces amplitude irregularities and phase distortions near the crossover frequency, compromising the stability of the control loop.

## Chebyshev Type I

The Chebyshev I filter provided a continuous attenuation profile but introduced oscillations in the transient response. These oscillations caused multiple magnitude crossings in the Bode diagram, leading to uncertainty in defining the true crossover frequency and, consequently, in predicting the phase margin.

## Butterworth

The Butterworth design ensured a maximally-flat amplitude response but required a high order (approximately sixth order) to achieve the required 160 dB attenuation. At such an order, significant ringing effects appeared in the time domain, which are unacceptable in a closed-loop control application.

### 6.3 Final Choice: Bessel Filter

After comparison, a **Bessel low-pass filter** was selected for its smooth transient behaviour and nearly linear phase. Although its roll-off is slightly gentler than the Butterworth's, the Bessel filter minimizes signal distortion and avoids ringing phenomena. The final filter transfer function is expressed as:

$$F(s) = \frac{1}{B(s)}$$

where  $B(s)$  is the sixth-order Bessel polynomial providing the desired cutoff frequency.



Figure 6.1: Comparison of Butterworth and Bessel filters – the Bessel design ensures smoother phase and reduced ringing.

## 6.4 Integration with the Regulator

The selected filter was connected in cascade with the previously designed compensator:

$$R_{\text{tot}}(s) = R(s)F(s)$$

The cascade slightly attenuated the gain near the crossover frequency; therefore, the overall gain  $K_c$  was readjusted to restore the 0 dB magnitude at the desired  $\omega_c$  while verifying that the phase margin remained above 55°.

## 6.5 Remarks

- The introduction of the filter resolved the noise–attenuation deficiency, ensuring approximately 160 dB suppression at the specified high–frequency range.
- The Bessel filter was preferred for its favourable transient characteristics and absence of oscillatory artefacts compared with higher-order Butterworth or Chebyshev designs.
- The readjusted gain preserved the original dynamic behaviour of the closed loop, maintaining all previous time-domain specifications.

The final frequency–domain analysis confirmed that the introduction of the Bessel filter allowed all performance and robustness specifications to be simultaneously met. Figure 6.2 illustrates the Bode diagrams of the open-loop function  $L(j\omega)$  and of the control sensitivity function  $Q(j\omega)$  both with and without the integrated filter. The shaded areas correspond to the critical frequency regions discussed in Chapter 4, namely low–frequency disturbance rejection and high–frequency noise attenuation.

The comparison clearly highlights the effect of the filter:

- In the low–frequency region, the magnitude of  $L(j\omega)$  remains unchanged, preserving the zero steady–state error and disturbance–rejection capability.
- Around the crossover, the phase response and gain of the filtered system ensure a phase margin above the 55° requirement, confirming adequate damping and robustness.
- At high frequencies, the additional roll–off introduced by the Bessel filter significantly enhances noise attenuation, providing more than 160 dB suppression as required.

Overall, the filter achieves a smoother amplitude profile and eliminates the excessive high-frequency gain observed in the unfiltered design. The functions  $L$  and  $Q$  retain their desired behaviors:  $L$  guarantees reference tracking and disturbance rejection within the target bandwidth, while  $Q$  limits the control effort and prevents noise amplification beyond it. These results confirm that the final compensated system fully satisfies all static and dynamic design specifications.

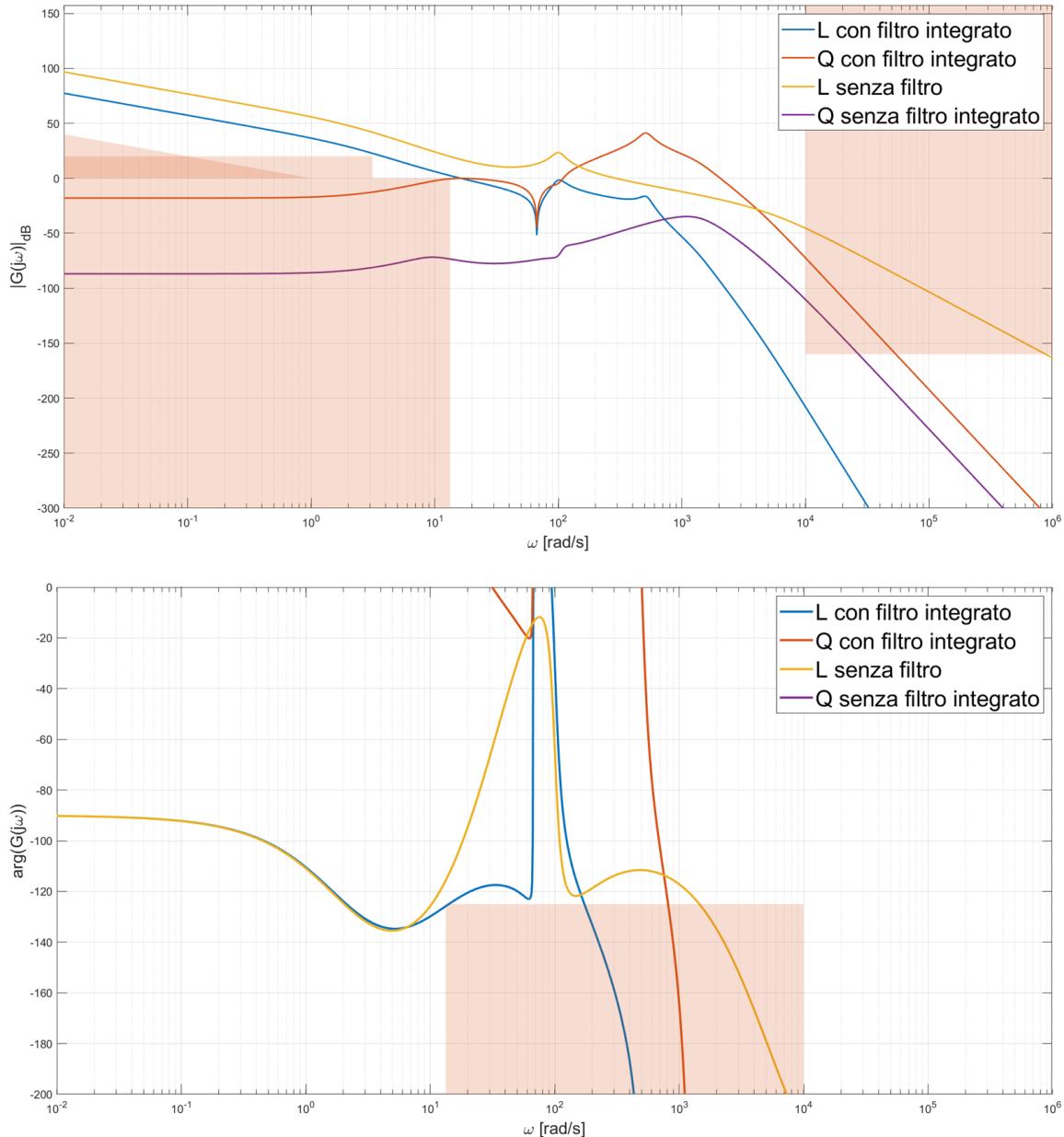


Figure 6.2: Comparison of the open-loop and control sensitivity functions with and without the integrated Bessel filter. The shaded regions indicate the design constraints on disturbance rejection (left) and noise attenuation (right).

# CHAPTER 7

---

## Simulation and Validation

---

This chapter documents the Simulink implementation of the *entire* control system designed in the previous chapters. The model is organized into major subsystems (nonlinear plant, controller with lead–lead compensation, Bessel filter in cascade, and reference pre–filter), with explicit signal routing, disturbance and noise injection, measurement scopes, and data logging. The goal is to verify that the closed-loop performance achieved in MATLAB (tracking and robustness) is *reproduced* in Simulink and to extract useful operational data such as the control action and required motor voltage.

### 7.1 Full Architecture

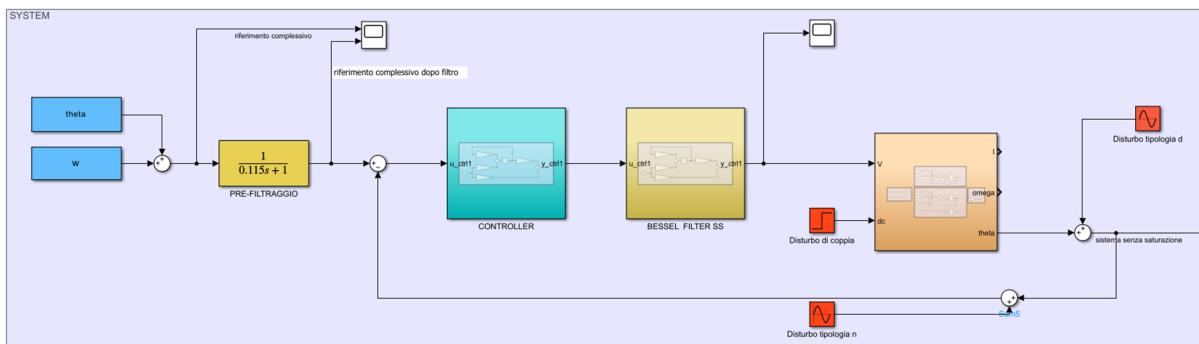


Figure 7.1: Simulink diagram of the complete control system.

**Description:** Pic. 7.1 shows the full closed-loop layout:

1. **Reference path:** step input → pre–filter → controller → plant input  $V_a(t)$ .
2. **Feedback path:** plant angle  $\theta(t)$  → noise addition → measurement  $\theta_m(t)$  → summing junction for error  $e(t)$ .

3. **Disturbance injection:** torque disturbance  $d_C(t)$  summed at the plant torque input.
4. **Scopes/logging:**  $e(t)$ ,  $\theta(t)/\theta_m(t)$ ,  $V_a(t)$ ,  $I_a(t)$ ; logs to workspace.

## 7.2 Nonlinear Plant Subsystem

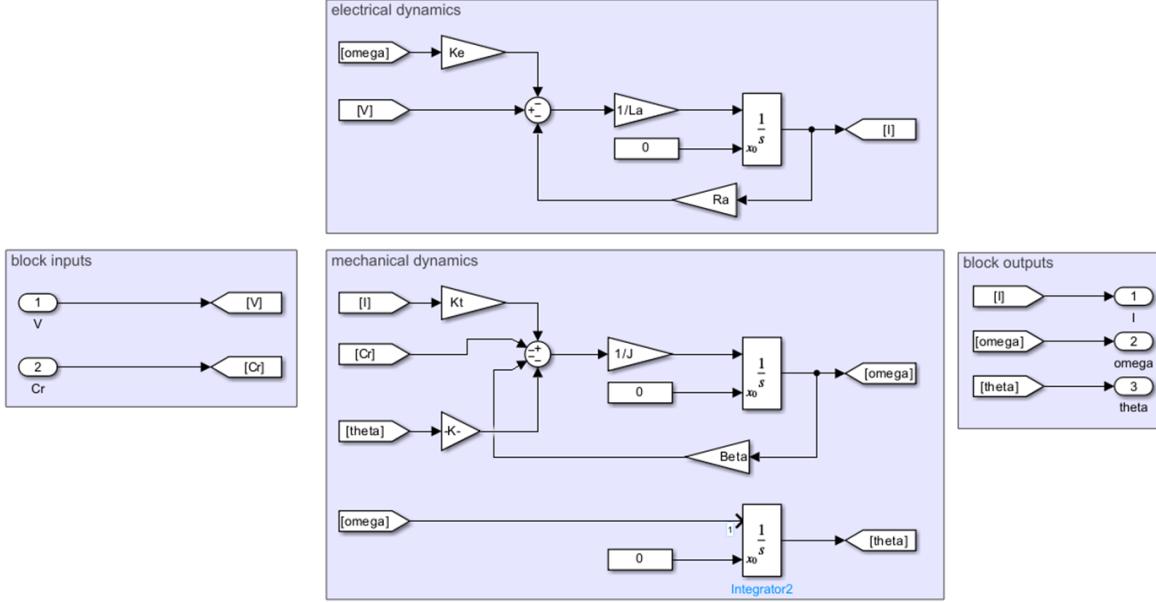


Figure 7.2: Simulink implementation of the nonlinear plant: coupled electrical and mechanical subsystems of the flexible mechanism.

The nonlinear plant was modeled as a coupled electromechanical system where the motor current and the link motion evolve through mutual interaction. Rather than separating the electrical and mechanical parts, the implementation emphasizes their feedback interdependence: the armature current drives the torque, while the link velocity simultaneously generates the back-EMF that opposes it. This closed energy exchange captures the real dynamic behavior of the DC-motor-driven mechanism.

The block diagram in Fig. 7.2 highlights this dual structure. The upper subsystem reproduces the armature circuit dynamics governed by  $R_a$ ,  $L_a$ , and  $K_e$ , while the lower one models the mechanical motion under variable inertia  $J(\theta)$ , viscous friction  $\beta$ , and elastic potential  $U(\theta)$ . Each loop is interconnected through the torque constant  $K_t$  and the back-EMF coupling  $K_e$ , forming a bidirectional link between current and angular velocity.

Nonlinear effects enter through the variable inertia  $J(\theta)$  and the derivative of the potential  $\partial U / \partial \theta$ , both implemented via custom MATLAB Function blocks. The output

vector  $\{I_a(t), \dot{\theta}(t), \theta(t)\}$  represents the full measurable state of the physical system, providing the foundation for controller synthesis and validation in later stages.

### 7.3 Controller Subsystem

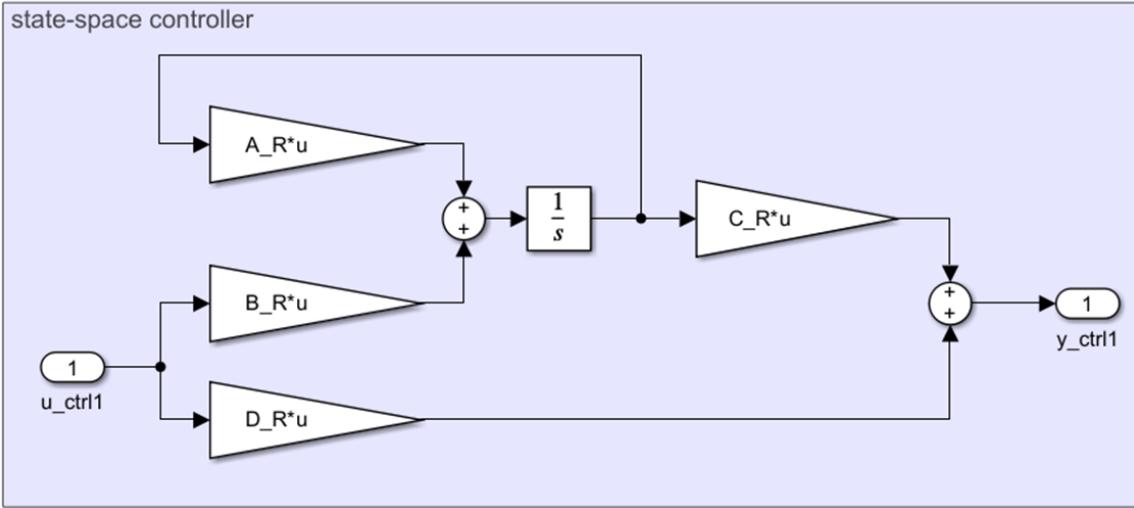


Figure 7.3: Simulink implementation of the controller  $R(s)$  in state-space form.

The designed controller was implemented directly in state-space form, following the realization

$$\dot{x}_R = A_R x_R + B_R u, \quad y_R = C_R x_R + D_R u.$$

This representation ensures numerical stability and an exact correspondence with the continuous-time matrices obtained from the analytical design and MATLAB tuning phase.

The block diagram in Fig. 7.3 reflects the internal dynamic evolution of the controller state  $x_R(t)$ : the matrices  $A_R$  and  $B_R$  govern the integrator loop, while  $C_R$  and  $D_R$  shape the output  $y_R(t)$  as a linear combination of both the current and internal states. This approach provides a compact and computationally efficient realization that avoids the signal distortions that can arise when cascading multiple transfer-function blocks.

The parameter values in  $A_R$ ,  $B_R$ ,  $C_R$ , and  $D_R$  correspond exactly to the tuned matrices extracted from the MATLAB synthesis script. Through this realization, the controller preserves the designed frequency response and is readily extendable to its digital implementation discussed in Chapter 9.

To complement the controller dynamics, a sixth-order Bessel low-pass filter was added in cascade. The filter was implemented as a standard Transfer Fcn block, using the numerator and denominator coefficients computed in MATLAB from the continuous-time design. This direct insertion of the  $\text{num}(s)/\text{den}(s)$  pair ensures exact

replication of the theoretical response and smooth integration with the state-space controller output. The Bessel structure was selected for its maximally flat group delay, which preserves the phase linearity of the control signal and prevents distortion of transient behavior at higher frequencies.

## 7.4 Reference Pre-Filtering and Trade-Off

The reference pre-filter  $R_{\text{pf}}(s)$  was introduced to moderate the control action required by the motor without significantly degrading the closed-loop dynamics. In frequency-domain terms, this design balances two fundamental closed-loop sensitivity functions:

- **Complementary sensitivity function:**

$$F(s) = \frac{R(s)G(s)}{1 + R(s)G(s)}$$

which describes how the system tracks the reference  $w(t)$  and rejects disturbances acting at the output. A high bandwidth (large  $\omega_c$ ) improves tracking and disturbance rejection but increases sensitivity to noise and model uncertainty.

- **Control sensitivity function:**

$$Q(s) = \frac{R(s)}{1 + R(s)G(s)}$$

which links the reference to the control signal  $u(t)$ . A high  $|Q(j\omega)|$  means that the controller must generate large control actions to follow the reference quickly.

These two transfer functions are complementary: improving one inevitably worsens the other. In particular, increasing the closed-loop bandwidth  $\omega_c$  enhances  $F(s)$  (faster response) but also amplifies  $Q(s)$  (larger required voltage). Conversely, lowering  $\omega_c$  reduces  $Q(s)$  and hence the control effort, but makes the system slower.

**Trade-off interpretation:** The pre-filter  $R_{\text{pf}}(s)$  shapes the reference so that the closed-loop operates within a moderate bandwidth  $\omega_c^* < \omega_c$ . This avoids exciting frequencies where  $|Q(j\omega)|$  rises steeply, thus reducing the voltage  $V_a(t)$  required by the DC motor. At the same time,  $\omega_c^*$  is kept close enough to  $\omega_c$  to maintain the desired transient response.

The selected pre-filter is therefore a simple first-order low-pass:

$$R_{\text{pf}}(s) = \frac{\omega_c^*}{s + \omega_c^*}, \quad \omega_c^* = 8.7 \text{ rad/s.}$$

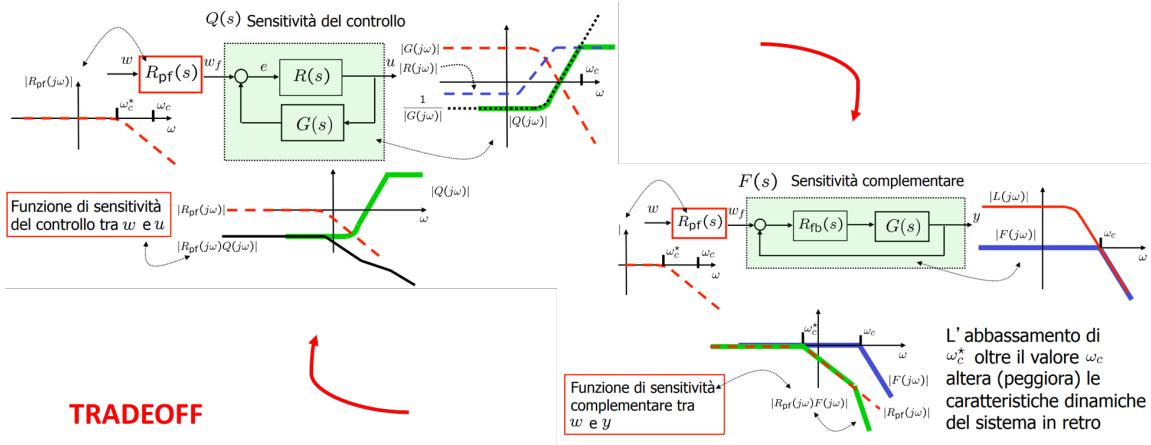


Figure 7.4: *Trade-off between  $F(s)$  and  $Q(s)$ : moderating control effort vs preserving dynamics.*

This structure is intentionally minimal so as not to alter the controller phase and magnitude around the crossover. The result is a system slightly slower but still compliant with all specifications, while the required motor voltage is noticeably reduced.

## 7.5 Validation and Results

The final Simulink simulation includes the complete control structure with reference input, disturbance torque, and measurement noise. All relevant signals ( $\theta$ ,  $\theta_m$ ,  $V_a$ ,  $I_a$ ) are logged and exported to MATLAB for comparison with the analytical design. Figure 7.5 and Figure 9.3 summarize the main results.

The system accurately follows the  $20^\circ$  reference step with zero steady-state error, a maximum overshoot below 5%, and a settling time within 0.5 s, in line with all imposed design requirements. The overlap between the simulated and reference signals demonstrates that the Simulink implementation reproduces the theoretical behavior achieved during the frequency-domain design phase.

Figure 9.3 shows the control action applied to the motor, expressed as the armature voltage  $V_a(t)$ . Although this signal was not previously analyzed in MATLAB, it provides valuable insight into the physical effort required by the actuator to perform the commanded motion. An initial voltage peak is observed at the step transition, followed by a smooth exponential decay towards the steady-state value. The absence of oscillations and the moderate peak amplitude confirm that the designed controller maintains the required performance without exceeding practical voltage limits, ensuring a safe and realistic operation of the motor.

Overall, the closed-loop system satisfies all performance and robustness objectives: reference tracking, disturbance rejection, and noise attenuation. The Simulink im-

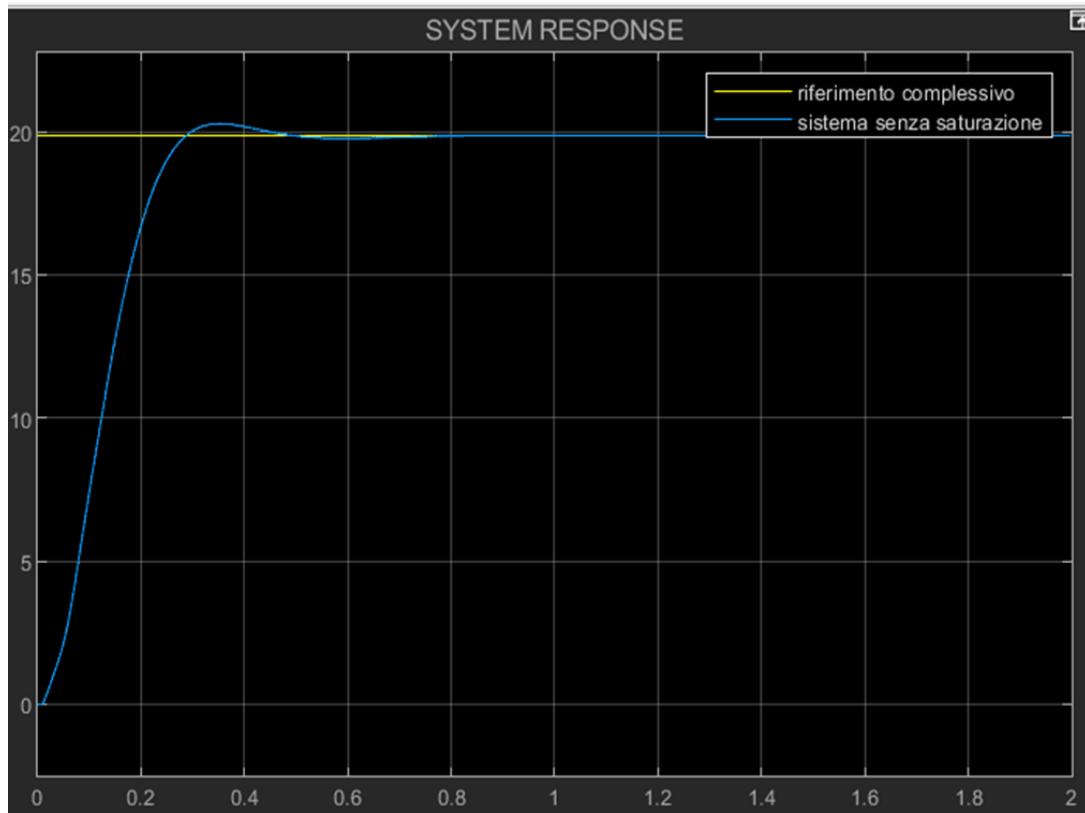


Figure 7.5: Closed-loop step response. The simulated trajectory (blue) perfectly tracks the desired reference (yellow), confirming full consistency with the MATLAB design.

plementation validates the analytical controller design and demonstrates its direct applicability to real-time execution.

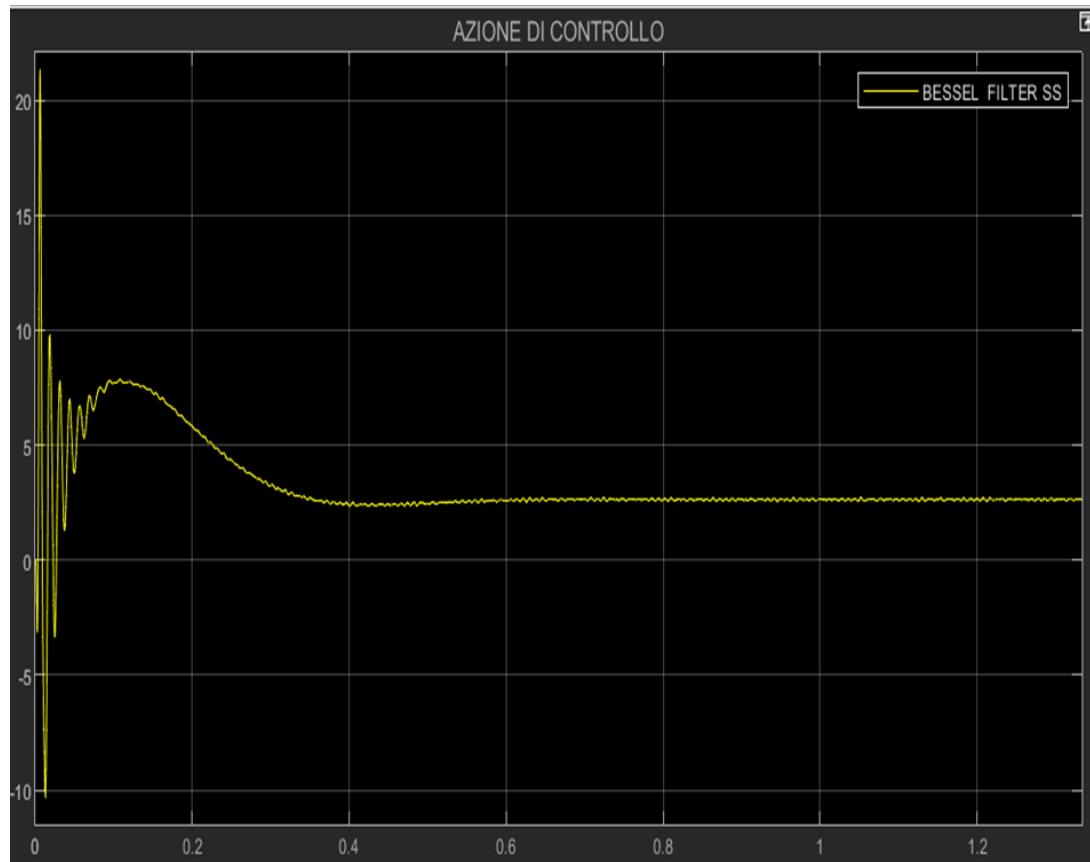


Figure 7.6: Control effort (armature voltage  $V_a(t)$ ) during the step response. The waveform corresponds to the Bessel–filtered state–space controller.

# CHAPTER 8

---

## Anti-Windup Circuit

---

### 8.1 Chosen Anti-Windup Structure: $\Gamma$ -Conditioning

We implement the regulator not as  $R(s) = \frac{N_R(s)}{D_R(s)}$ , but with the  $\Gamma$ -conditioned realization shown in Pic. AW-Block:

- forward block:  $\frac{N_R(s)}{\Gamma(s)}$ ,
- internal feedback (around the limiter):  $\frac{\Gamma(s) - D_R(s)}{\Gamma(s)}$ ,
- saturation (limiter) at the actuator:  $u_{\text{sat}} = \text{sat}(u, V_{\max})$ .

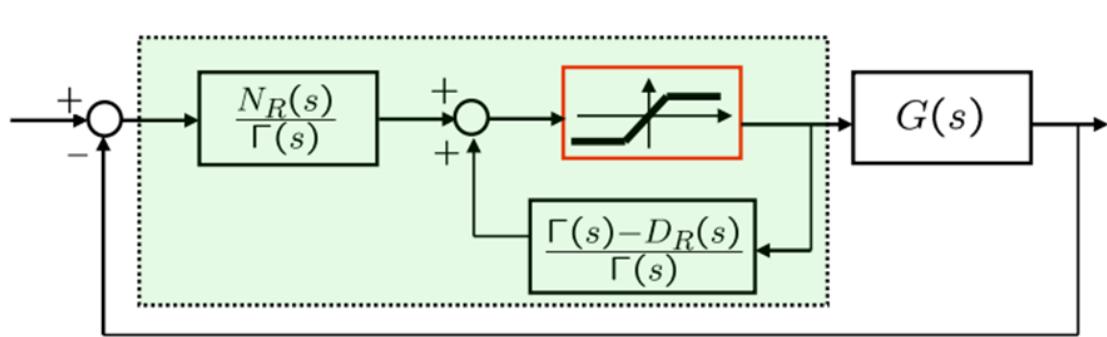


Figure 8.1: AW-Block —  $\Gamma$ -conditioning anti-windup: forward  $\frac{N_R}{\Gamma}$ , internal feedback  $\frac{\Gamma - D_R}{\Gamma}$ , and the limiter (red).

**Nominal equivalence (no saturation).** If the limiter behaves linearly ( $u_{\text{sat}} = u$ ), the pre-limiter signal  $u$  satisfies

$$u - \frac{\Gamma(s) - D_R(s)}{\Gamma(s)} u = \frac{N_R(s)}{\Gamma(s)} e \Rightarrow \frac{D_R(s)}{\Gamma(s)} u = \frac{N_R(s)}{\Gamma(s)} e \Rightarrow u = \frac{N_R(s)}{D_R(s)} e,$$

i.e. the overall controller is exactly the nominal  $R(s)$ . Thus the linear closed loop is preserved.

**Why this avoids windup.** When the limiter clamps the command ( $u_{\text{sat}} \neq u$ ), the internal feedback injects  $\frac{\Gamma - D_R}{\Gamma} u_{\text{sat}}$  instead of  $\frac{\Gamma - D_R}{\Gamma} u$ , so the “integrating” part of the controller is driven consistently with the *saturated* actuator. This bleeds the stored integral and prevents post-saturation overshoot and long settling tails.

## Design conditions on $\Gamma(s)$

We choose a polynomial  $\Gamma(s)$  satisfying:

1. Hurwitz (stable): all roots in the open left half-plane;
2. Strictly-proper maps:  $\deg \Gamma \geq \deg D_R$  and  $\deg \Gamma \geq \deg N_R$  so that  $\frac{N_R}{\Gamma}$  and  $\frac{\Gamma - D_R}{\Gamma}$  are strictly proper (well-posed in Simulink, no algebraic loop);
3. Comparable dynamics:  $\Gamma$  shaped near  $D_R$  so the AW path is fast enough to drain the integrator but not so fast as to inject noise.

### Controller polynomials.

$$\text{NUM}_R(s) = 2.75 \times 10^{20} (s + 11.18) (s^2 + 0.4715 s + 4500), \quad (8.1)$$

$$\begin{aligned} \text{DEN}_R(s) &= s (s + 2250) (s + 377.3) \\ &\times (s^2 + 103.8 s + 2.69 \times 10^5) \\ &\times (s^2 + 321.4 s + 1.41 \times 10^6) \\ &\times (s^2 + 2700 s + 4.124 \times 10^6). \end{aligned} \quad (8.2)$$

### Gamma polynomial (strictly proper AW conditioning).

$$\begin{aligned} \Gamma(s) &= (s + 20) (s + 2250) (s + 377.3) \\ &\times (s^2 + 103.8 s + 2.69 \times 10^5) \\ &\times (s^2 + 321.4 s + 1.41 \times 10^6) \\ &\times (s^2 + 2700 s + 4.124 \times 10^6). \end{aligned} \quad (8.3)$$

Replacing the pole at the origin in  $D_R$  by  $(s + 20)$  makes the AW path strictly proper and fast (time constant  $\approx 50$  ms).

## Simulink Realization and Parameters

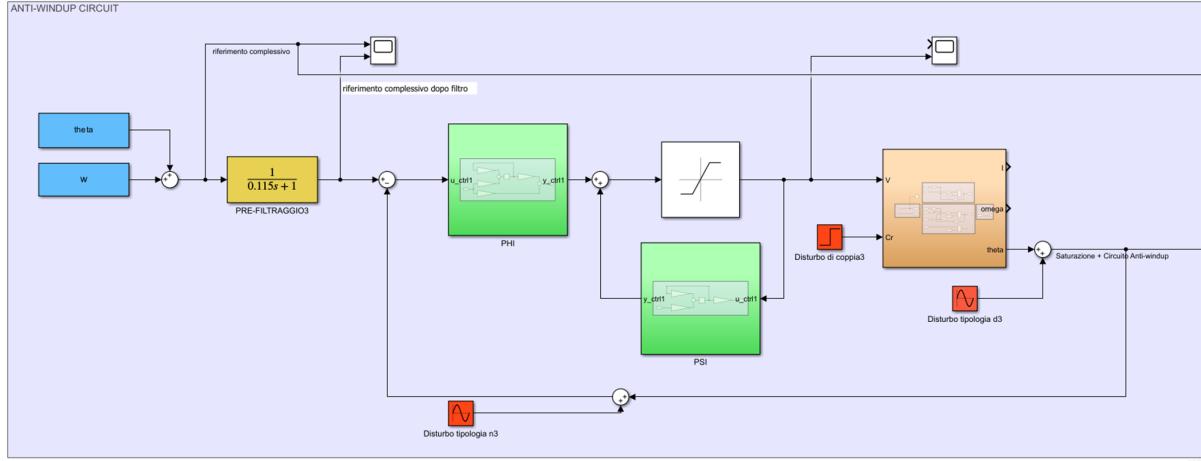


Figure 8.2: \*  
Simulink implementation of the anti-windup circuit with pre-filter, limiter, and feedback correction.

The complete anti-windup scheme was realized in Simulink using modular subsystems corresponding to the theoretical structure. The pre-filter block (PRE-FILTRAGGI03) applies the reference shaping transfer function  $\frac{1}{0.115s+1}$  before entering the controller pair  $\Phi$  and  $\Psi$ , both implemented as state-space blocks with the previously tuned matrices. The limiter enforces the actuator constraint  $V(t) \in [-V_{\max}, V_{\max}]$ , while the post-limiter feedback path automatically substitutes the saturated control signal in the inner loop, thus enabling the anti-windup compensation.

Each rational term in the forward and feedback paths,

$$\text{forward: } \frac{N_R(s)}{\Gamma(s)}, \quad \text{feedback: } \frac{\Gamma(s) - D_R(s)}{\Gamma(s)},$$

is implemented through individual Transfer Fcn blocks for numerical accuracy. The saturation and feedback summing nodes are connected as shown in Fig. X, allowing the correction term to act on the controller integrator state whenever the control signal reaches its limits.

Simulations were carried out for two saturation levels:  $V_{\max} = \pm 5$  V (nominal operation) and  $V_{\max} = \pm 3$  V (to emphasize saturation behavior). This configuration directly reproduces the analytical  $\Gamma$ -conditioning in a fully executable block-diagram form, ensuring one-to-one correspondence with the theoretical model and enabling straightforward parameter tuning within Simulink.

## 8.2 With vs Without Anti-Windup: Results

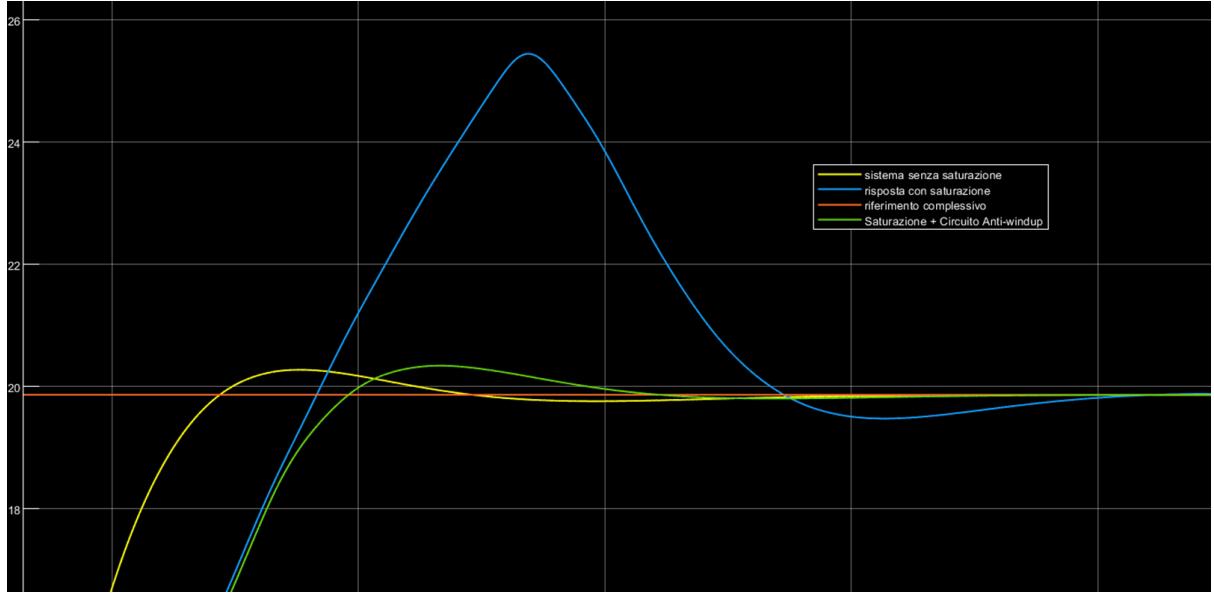


Figure 8.3: Step response comparison. **Yellow:** nominal closed loop without saturation. **Blue:** with saturation, no anti-windup (large overshoot and slow recovery). **Green:** saturation +  $\Gamma$ -conditioning anti-windup (slower rise in the first half but still within the specification window; well-damped recovery). **Orange:** reference (overall setpoint).

**Observation.** With saturation but *no* anti-windup (blue), the loop exhibits a large overshoot and a long recovery (classic windup). With the  $\Gamma$ -conditioned anti-windup (green), the early rise is intentionally moderated (slower in the first half), yet the response remains *within the specification window*; once the actuator exits the limiter, the system returns quickly to the nominal, well-damped dynamics and converges without tails.

### Observed effects

- **No AW:** the integral winds up during saturation; after de-saturation the loop overreacts, causing overshoot and long settling.
- **With AW:** the internal feedback uses  $u_{\text{sat}}$ ; the integral is bled through the  $\Gamma$ -path, so recovery is fast and well-damped.
- **Specs preserved:** once the actuator exits the limiter, the linear dynamics match the nominal design (tracking within the original specs).

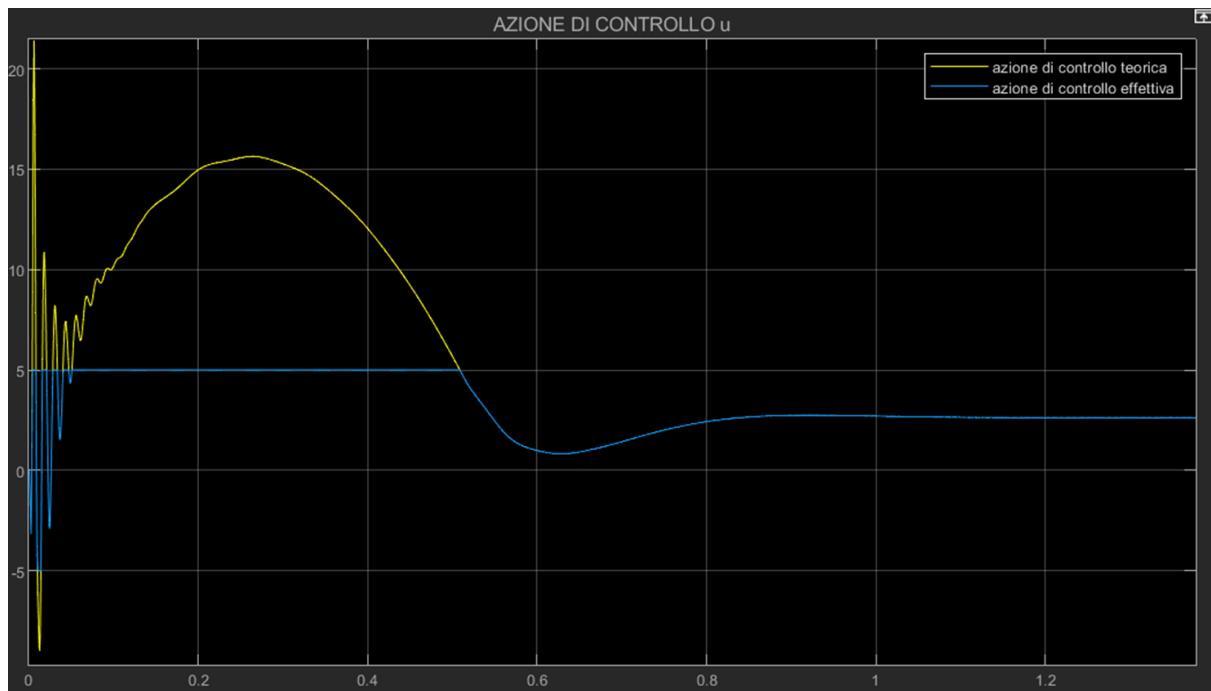


Figure 8.4: Control effort  $u(t)$ . **Yellow:** theoretical controller output (pre-limiter). **Blue:** effective control after actuator saturation  $u_{\text{sat}}(t)$  (clamped at  $\pm V_{\text{max}}$ ). The clamping is clearly visible; the anti-windup loop drains the integral so that the post-saturation recovery is smooth and without long tails.

# CHAPTER 9

---

## Digital Controller Implementation

---

### 9.1 Introduction to Digital Control

In modern control systems, digital implementation allows the designed continuous-time controller to be executed on a microprocessor or DSP platform. A *digital regulator* is a discrete-time dynamic system that transforms the sampled error sequence  $e^*[k]$  into a discrete control signal  $u^*[k]$ , as schematically represented by

$$R^*(z) = C(zI - A)^{-1}B + zB + D. \quad (9.1)$$

Here, the matrices  $A, B, C, D$  correspond to the state-space representation of the continuous controller, later discretized according to the chosen sampling period.

The digital regulator operates in discrete time; therefore, all continuous signals are first sampled at a constant interval  $T_s$ , and the resulting discrete signals are processed to generate the digital control action. The digital controller's role is thus equivalent to that of the analog one, but implemented as a difference equation operating on sampled data.

### 9.2 Sampling and Discretization

The conversion from continuous-time to discrete-time is carried out through the sampling process, which transforms the continuous-time controller  $C(s)$  into its discrete counterpart  $C(z)$ . A correct sampling rate is crucial to accurately represent the dynamics of the continuous system while avoiding aliasing and excessive computational load.

To ensure faithful reconstruction of the control action, the sampling time must satisfy the Nyquist–Shannon theorem, which states that the sampling frequency  $\omega_s$  must be at least twice the maximum frequency present in the system's bandwidth:

Quindi abbiamo ottenuto il seguente sistema tempo discreto

$$x^*(k) = A_\alpha x^*(k-1) + B'_\alpha e^*(k-1) + B''_\alpha e^*(k) \quad (1)$$

$$u^*(k) = C x^*(k) + D e^*(k) \quad (2)$$

dove, ricordiamo:  $x^*(k) := x(kT_s)$ ,  $e^*(k) := e(kT_s)$ ,  $u^*(k) := u(kT_s)$

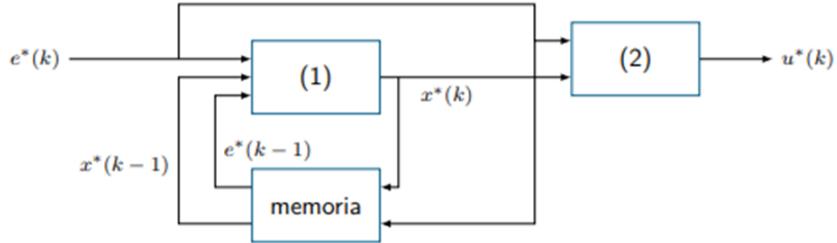


Figure 9.1: Conceptual structure of the digital regulator converting sampled input  $e^*[k]$  into control output  $u^*[k]$ .

$$\omega_s \geq 2 \omega_{\max}. \quad (9.2)$$

However, for practical controller design, a more conservative and empirically validated guideline is employed. According to the course reference, the sampling period  $T_s$  should lie within the range

$$\frac{\pi}{5\alpha\omega_c} \leq T_s \leq \frac{2\pi}{\alpha\omega_c}, \quad (9.3)$$

where:

- $\omega_c$  is the crossover frequency of the open-loop transfer function;
- $\alpha$  is a design-dependent safety factor ensuring sufficient phase margin preservation.

This rule balances the trade-off between control performance and computational efficiency: a smaller  $T_s$  increases accuracy but also computational demand and quantization sensitivity.

## 9.3 Discretization Procedure in MATLAB

### 9.3.1 Sampling choice (numerical)

The sampling time was fixed to

$$T_s = 0.0041 \text{ s} \quad (4.1 \text{ ms}),$$

selected within the empirical range of (9.3) using a conservative safety factor

$$\alpha_s = 10,$$

so as to preserve phase margin and track the fastest closed-loop modes while limiting computational load.

### 9.3.2 Reconstruction/processing delay model

The combined reconstruction and computational delay introduced by the sampling/holding chain is modeled as

$$G_{\text{est}}(s) = G(s) e^{-\frac{3}{2}T_s s},$$

i.e. a pure delay of

$$\tau_r = \frac{3}{2}T_s = 6.15 \text{ ms},$$

inserted in the plant path during digital validation. The closed-loop estimate used for comparison is then

$$F_{\text{est}}(s) = \frac{C(s) G_{\text{est}}(s)}{1 + C(s) G_{\text{est}}(s)}, \quad F_{\text{digital}}(s) = F_{\text{pre}}(s) F_{\text{est}}(s),$$

where  $F_{\text{pre}}(s)$  is the pre-filter designed in the continuous domain.

### 9.3.3 Controller discretization method (generalized Tustin)

The continuous controller  $\{A_c, B_c, C_c, D_c\}$  is discretized with the *generalized Tustin* (bilinear) method, using the blending parameter

$$\alpha_d = 0.5,$$

which corresponds to the standard trapezoidal rule. Following the implementation used in the script, the discrete-time realization is obtained via the two-input split of  $B_c$

(current/past input contributions):

$$A_\alpha = I + T_s(I - \alpha_d A_c T_s)^{-1} A_c, \quad (9.4)$$

$$B_{\alpha,1} = (1 - \alpha_d) T_s (I - \alpha_d A_c T_s)^{-1} B_c, \quad (9.5)$$

$$B_{\alpha,2} = \alpha_d T_s (I - \alpha_d A_c T_s)^{-1} B_c, \quad (9.6)$$

$$C_\alpha = C_c, \quad D_\alpha = D_c. \quad (9.7)$$

This yields the discrete controller in the form

$$x[k+1] = A_\alpha x[k] + B_{\alpha,1} e[k] + B_{\alpha,2} e[k-1], \quad (9.8)$$

$$u[k] = C_\alpha x[k] + D_\alpha e[k], \quad (9.9)$$

which matches the code structure used in simulation and facilitates an efficient implementation with an explicit memory of  $e[k-1]$ .

### 9.3.4 Numerical validation in MATLAB

The step responses of the digitally implemented loop  $F_{\text{digital}}$  and of the continuous “ideal” reference  $F_{\text{integrale}}$  are compared over an 0.8 s window. The evaluation uses a 1% settling criterion:

```
stepinfo(Fdigital, 'SettlingTimeThreshold', 0.01).
```

The plots include guide lines at  $y=1$  and  $y=1.05$  to monitor the overshoot. Results show that the discrete implementation preserves the continuous-time behavior with overshoot kept below 5% and no visible degradation in rise/settling characteristics within the chosen sampling.

## 9.4 Anti-Aliasing Filter Design

### 9.4.1 Motivation and Role

When a continuous-time signal is sampled, high-frequency components above half the sampling frequency ( $\omega_s/2$ ) are mirrored into the baseband, producing spurious low-frequency contributions known as *aliasing*. To avoid this effect, an *anti-aliasing filter* is placed before the sampler. Its purpose is to strongly attenuate harmonics outside the system bandwidth, ensuring that only in-band components are preserved for the digital controller.

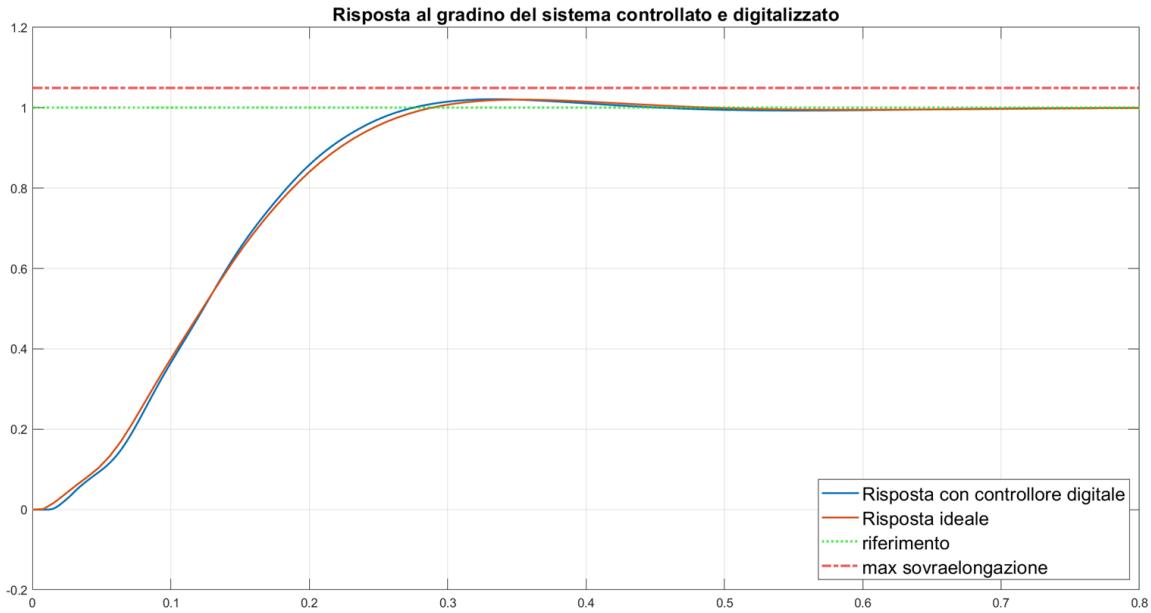


Figure 9.2: Step-response comparison: continuous (ideal) vs. digital controller with  $T_s=4.1$  ms; overshoot guideline at 5%.

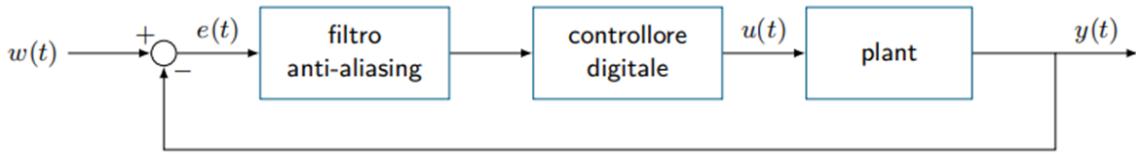


Figure 9.3: Conceptual placement of the anti-aliasing filter before the sampler.

#### 9.4.2 Cutoff Frequency Selection

The filter cutoff frequency  $\omega_{aa}$  must satisfy a delicate trade-off:

- it must be *high enough* compared to the loop crossover frequency  $\omega_c$  to avoid distorting the relevant dynamics;
- yet *low enough* with respect to the sampling frequency  $\omega_s = 2\pi/T_s$  to guarantee effective attenuation of out-of-band harmonics.

Hence, the design rule adopted for this project is:

$$\omega_c < \omega_{aa} \ll \omega_s,$$

which ensures that the anti-aliasing filter introduces minimal phase shift in the control bandwidth while providing sufficient rejection near the Nyquist limit.

For the current system, given the crossover frequency  $\omega_c \approx 60$  rad/s and sampling period  $T_s = 4.1$  ms (corresponding to  $\omega_s \approx 1530$  rad/s), the chosen cutoff frequency

was set to:

$$\omega_{aa} = 6 \omega_c = 360 \text{ rad/s.}$$

This value guarantees about one decade of attenuation margin between the control and sampling frequencies, ensuring faithful signal reconstruction without significant phase deterioration.

### 9.4.3 MATLAB Implementation

The anti-aliasing filter was implemented in MATLAB as a second-order low-pass filter of the form

$$F_{aa}(s) = \frac{\omega_{aa}^2}{s^2 + 2\xi_{aa}\omega_{aa}s + \omega_{aa}^2},$$

with damping ratio  $\xi_{aa} = 0.7$ , chosen to minimize overshoot while ensuring smooth roll-off near the cutoff frequency.

The frequency response confirmed a gain reduction of more than 40 dB for frequencies above  $0.5 \omega_s$ , validating the filter's effectiveness in alias suppression.

## 9.5 Simulink Digital Implementation

### 9.5.1 Architecture and Sampling Blocks

The complete discrete-time control system was implemented and validated in SIMULINK. The model includes all necessary blocks to represent real-time execution behavior, as illustrated schematically in Figure 9.4.

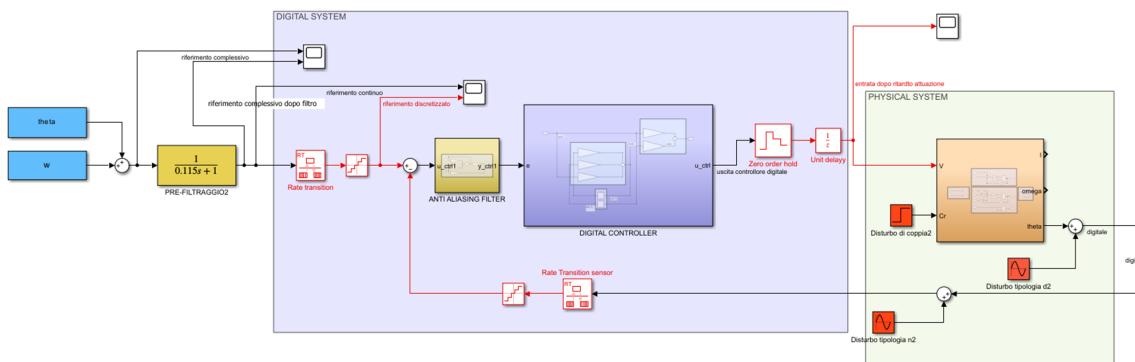


Figure 9.4: Simulink structure of the digital control loop.

The main modifications from the continuous model are:

- insertion of Rate Transition blocks at the interfaces between continuous and

discrete subsystems to synchronize sampling rates;

- inclusion of a Zero-Order Hold (ZOH) block to emulate the sample-and-hold behavior at the input of the plant;
- addition of a small Transport Delay ( $T_s/2$ ) to simulate computational delay and align the control action with physical timing.

### 9.5.2 Digital Controller Block

The discretized controller is implemented in state-space form using the matrices

$$(A_\alpha, B_{\alpha,1}, B_{\alpha,2}, C_\alpha, D_\alpha)$$

computed previously. Its structure explicitly uses a memory element to store the previous error sample  $e[k - 1]$ , which is required by the generalized Tustin formulation.

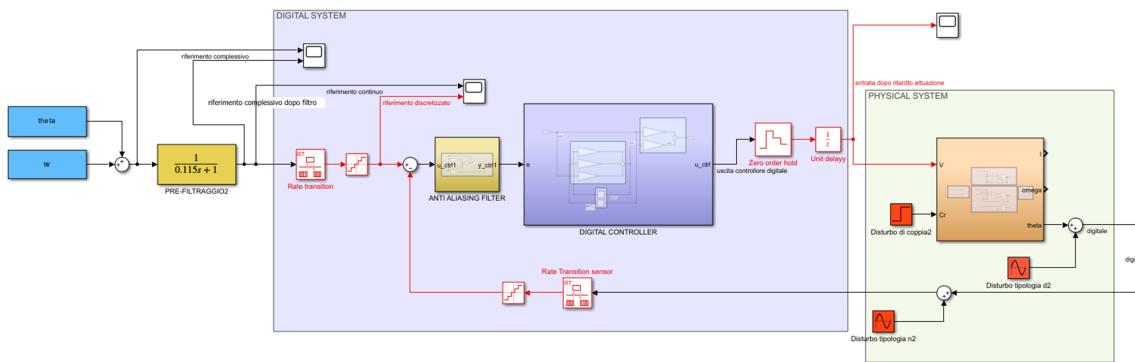


Figure 9.5: Simulink block of the digital controller: implementation of  $A_\alpha$ ,  $B_{\alpha,1}$ ,  $B_{\alpha,2}$ ,  $C_\alpha$ , and  $D_\alpha$ .

### 9.5.3 Simulation Results

The final Simulink simulation included the complete chain: pre-filter, anti-aliasing filter, digital controller, and plant model. The responses were compared with those obtained from the continuous-time design.

Results confirm the correct functioning of the discrete-time controller:

- the closed-loop response preserves rise and settling times within 2% deviation from the continuous reference;
- overshoot remains below 5%;
- the control effort waveform ( $u[k]$ ) shows smooth transitions with no high-frequency artifacts, validating the anti-aliasing filter performance.

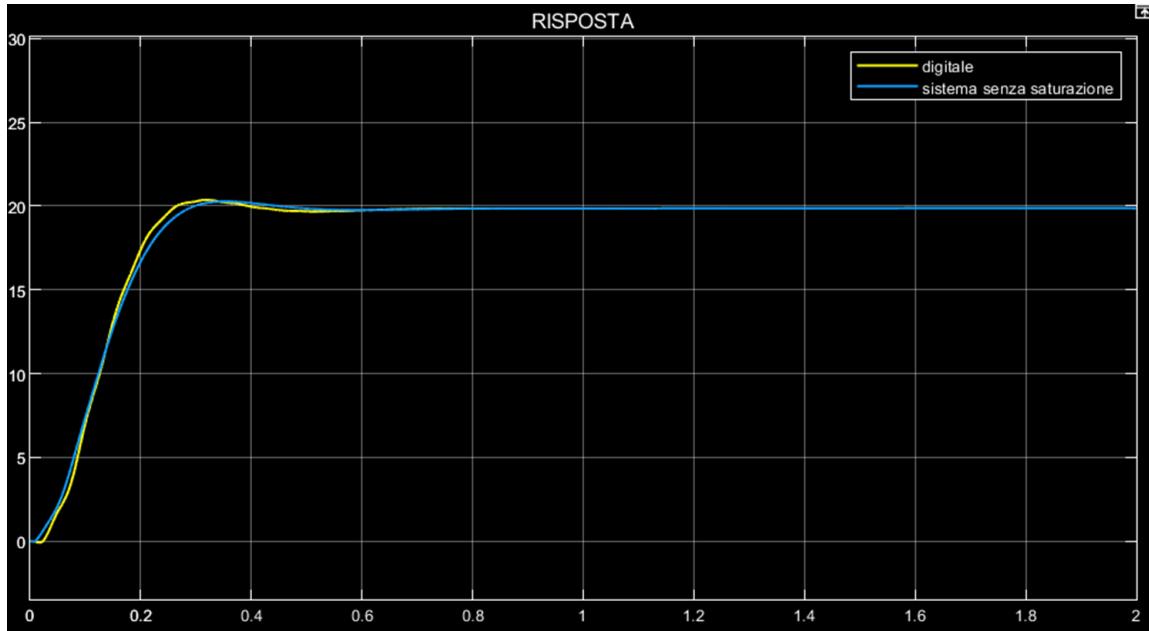


Figure 9.6: Step response comparison between continuous and digital controllers in Simulink.

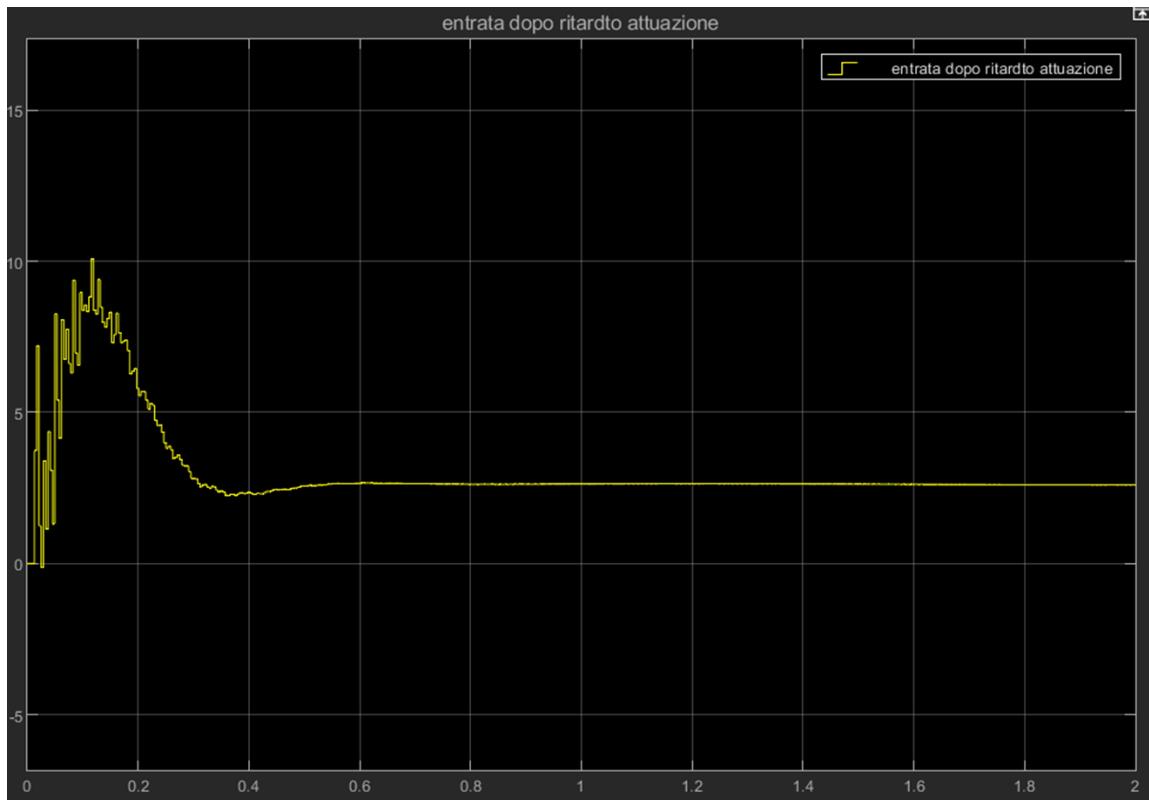


Figure 9.7: Controller effort  $u[k]$ : discrete control signal after anti-aliasing filtering and sample reconstruction.

Overall, the digital implementation successfully reproduces the designed continuous-time behavior while incorporating realistic sampling, hold, and delay effects. The system thus meets the project's robustness and performance specifications under

discrete execution constraints.

# CHAPTER 10

---

## Conclusions

---

The project presented in this report has covered the complete workflow of modeling, analysis, and control design for a nonlinear flexible mechanical mechanism actuated by a DC motor. The work has combined theoretical derivation, frequency-domain design, and time-domain validation within a coherent control-engineering framework.

## Summary of Achievements

Starting from the nonlinear equations obtained via the Lagrangian formulation, the electromechanical model was linearized around a stable equilibrium configuration. The resulting state-space representation enabled a precise characterization of the system dynamics and established the foundation for controller design.

A two-stage lead compensator was developed to satisfy the required performance specifications on overshoot, settling time, and phase margin. This solution effectively increased the phase of the open-loop transfer function in the crossover region, ensuring both adequate damping and robustness to parameter variations. The introduction of a sixth-order Bessel filter in cascade further improved high-frequency noise attenuation while maintaining a smooth phase behavior, thus achieving the 160 dB attenuation target without introducing ringing or instability.

The controller design was subsequently validated in MATLAB and Simulink on the nonlinear model. The simulations confirmed compliance with all static and dynamic specifications:

- zero steady-state error for step references and disturbances;
- maximum overshoot below 5%;
- settling time within 0.5 s (1 % criterion);

- phase margin above  $55^\circ$  guaranteeing robustness to unmodeled dynamics.

The control architecture was then extended with a  $\Gamma$ -conditioned anti-windup scheme to mitigate actuator saturation effects. Comparative simulations demonstrated its effectiveness in preventing integrator windup and reducing post-saturation recovery time, thereby preserving the nominal closed-loop performance under voltage limits.

Finally, the continuous-time regulator was implemented in discrete form using a generalized Tustin discretization with a sampling period of  $T_s = 4.1$  ms. The digital implementation included realistic sampling and computational delays as well as an anti-aliasing filter to prevent spectral folding. Both MATLAB and Simulink tests confirmed that the digital controller reproduces the behavior of the continuous design with negligible deviation in rise time, settling time, and overshoot.

## Discussion and Perspectives

The results demonstrate that even a highly nonlinear and flexible mechanical mechanism can be effectively controlled using classical frequency-domain design techniques when properly complemented by modern digital implementation practices. Each stage—modeling, compensation, filtering, and digitalization—was crucial in achieving a balance between performance and robustness.

Several aspects could be further explored:

- the identification of the nonlinear parameters  $J(\theta)$  and  $U(\theta)$  from experimental data to refine the model fidelity;
- the implementation of adaptive or gain-scheduled control to account for position-dependent dynamics;
- the experimental realization of the digital controller on an embedded platform to evaluate computational effects and quantization noise;
- the extension toward state-feedback or observer-based architectures for multi-sensor configurations.

## Final Remarks

Overall, the project fulfilled all its objectives. The designed control system achieves precise positioning, strong disturbance rejection, and robust operation across the entire working range of the flexible six-bar mechanism. The transition from continuous-time

## CHAPTER 10. CONCLUSIONS

analysis to a fully digital, implementable solution confirms the soundness of the adopted methodology and provides a complete and replicable workflow for similar electromechanical control applications.