



Article

Model-Based Design of an Improved Electric Drive Controller for High-Precision Applications Based on Feedback Linearization Technique

Pierpaolo Dini [†]  and Sergio Saponara ^{*,†} 

Department of Information Engineering, University of Pisa, Lungarno Antonio Pacinotti, 43, 56126 Pisa, Italy; pierpaolo.dini@phd.unipi.it

* Correspondence: sergio.saponara@unipi.it

† Authors equally contributes to this article.

Abstract: This paper presents the design flow of an advanced non-linear control strategy, able to absorb the effects that the main causes of torque oscillations, concerning synchronous electrical drives, cause on the positioning of the end-effector of a manipulator robot. The control technique used requires an exhaustive modelling of the physical phenomena that cause the electromagnetic torque oscillations. In particular, the Cogging and Stribeck effects are taken into account, whose mathematical model is incorporated in the whole system of dynamic equations representing the complex mechatronic system, formed by the mechanics of the robot links and the dynamics of the actuators. Both the modelling procedure of the robot, directly incorporating the dynamics of the actuators and the electrical drive, consisting of the modulation system and inverter, and the systematic procedure necessary to obtain the equations of the components of the control vector are described in detail. Using the Processor-In-the-Loop (PIL) paradigm for a Cortex-A53 based embedded system, the beneficial effect of the proposed advanced control strategy is validated in terms of end-effector position control, in which we compare classic control system with the proposed algorithm, in order to highlight the better performance in precision and in reducing oscillations.

Keywords: model-based design; simulation; mechatronics; dynamic systems; modelling; control theory; brushless motors; electric drives



Citation: Dini, P.; Saponara, S. Model-Based Design of an Improved Electric Drive Controller for High-Precision Applications Based on Feedback Linearization Technique. *Electronics* **2021**, *10*, 2954. <https://doi.org/10.3390/electronics10232954>

Academic Editors: Hassan Haes Alhelou, Amer Al-Hinai and Pierluigi Siano

Received: 9 November 2021

Accepted: 24 November 2021

Published: 28 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In this paper, we propose a non-linear control technique that absorbs the negative effects of the torque ripple delivered by permanent magnet synchronous motor (PMSM), which affect the accuracy of rotor axis position control. As a case study, the position control of the end-effector of a manipulator robot is considered. This allows us to assess the complexity of the development of the control system using our technique, in the context of a fairly complex mechatronic system.

The development of a control technique able to absorb this type of intrinsic non-linearity associated with the use of electric drives based on PMSM is an application of industrial interest. It is certainly well known that PMSM are the most efficient from the point of view of energy and the relationship between inertia and electromagnetic torque delivered. The problems in using PMSM for precision applications are mainly related to the complexity of driving, since an inverter and an efficient modulation algorithm are necessarily required, and to the intrinsic interaction between permanent magnets and stator structure, which overlaps with electromechanical conversion effects, especially at low rotor speeds. In this work, we focus on the development of a control strategy capable of absorbing the main sources of disturbance to the electromagnetic torque.

The contribution of this work is to apply the non-linear control strategy, capable of significantly reducing the effects of electromagnetic torque disturbances, in the context of a

mechatronic system of relatively high complexity, such as a robot manipulator with two degrees of freedom. The validation of the strategy proposed in this work involves a very accurate modelling of all physical and architectural aspects. In particular, a high degree of detail is taken into account in the virtual representation of the electrical drives, in which the modulation and supply aspects through three-phase inverter of the control voltages are modelled. In the modelling of synchronous motors, the model of the main sources of torque ripple is included. In particular, models of the Cogging Torque and Stribeck Torque phenomena are included.

The objective is also to show operationally how to develop and validate efficient control algorithms able to compensate for intrinsic non-linearity effects of the controlled physical process, in the case of strong industrial interest of mechatronic systems of electro-mechanical nature, through accurate modelling and/or mathematical formalisation. The case study is that of a simple manipulator robot, but the procedure we show in this paper is not limited to the case study itself, and could be applied in different industrial applications such as the control of industrial technological processes and assisted driving control of vehicles. Hereafter, the rest of the paper is organized as follows. Section 2 reviews the cogging and Stribeck effects and the related work at the state of art. Section 3 presents the electric drive system modeling. Section 4 deals with advanced control system design. Processor in the loop verification when the proposed technique is implemented on Cortex-A53 processor is reported in Section 5. Conclusions and future works are addressed in Section 6.

2. Review of Cogging and Stribeck Effects and Related Works

2.1. Brief Explanation of the Cogging and Stribeck Effects

In this work, the two most important effects are considered as intrinsic non-linearity representing the torque ripple affecting the performance of the positioning control: Cogging Torque and Stribeck Torque.

Cogging torque arises from the magnetic interaction between the permanent magnets on the rotor surface and the teeth of the stator slots. This interaction results in magnetic forces that alternately attract and repel the two parts in the direction of rotation of the rotor axis [1]. Figure 1 shows how, depending on the relative position between a permanent magnet and a stator tooth, the interaction force changes from being in favour of the direction of rotation (Figure 1a) to being against the direction of rotation (Figure 1b).

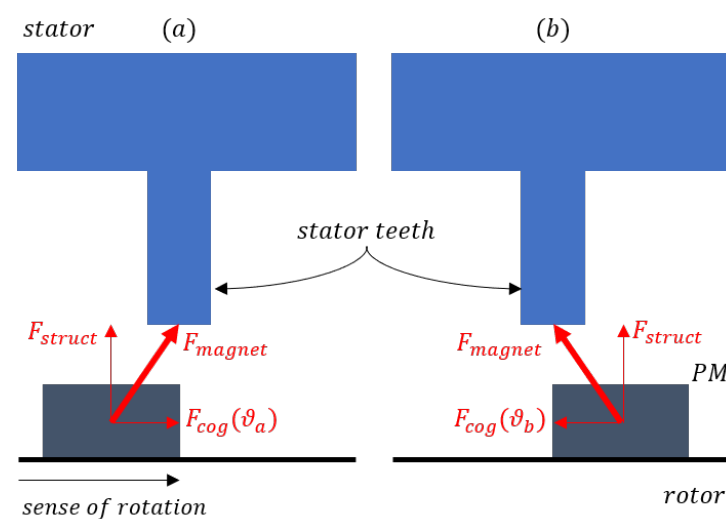


Figure 1. Schematic representation of the magnetic stator-rotor interaction. The interaction force changes from being in favour of the direction of rotation (a) to being against the direction of rotation (b).

The alternation of the phases of attraction and repulsion means that the Cogging Torque can be represented in closed form as an additive disturbance on the electromagnetic torque with the peculiarity of being zero averaged and with a frequency that depends on the internal structure of the machine, in terms of the number of stator teeth and the number of permanent magnets arranged on the rotor [2,3]. In this work, we refer to the formulation given in Equation (1).

$$T_{cog} = \sum_{i=1}^{N_h} T_i \sin\left(i \frac{\theta}{\theta_{cog}} + \alpha_i\right) \quad (1)$$

In the expression of T_{cog} , θ represents the rotation of the rotor axis, N_h the number of harmonics needed to exhaustively represent the phenomenon, T_i and α_i are the modulus and phase coefficients of the i th harmonic and θ_{cog} represents the angular period of the Cogging phenomenon. Furthermore, it has been shown that a limited number of harmonics (in this paper we limit the analysis to four harmonics) provides a sufficiently detailed representation of the phenomenon.

The second phenomenon we take into account when modelling electromagnetic torque ripple is the Stribeck torque. As shown in Figure 2 the Stribeck model, whose expression is given in Equation (2), is a generalisation of the viscous friction force model, which would instead depend only on the velocity (and its sign) [4].

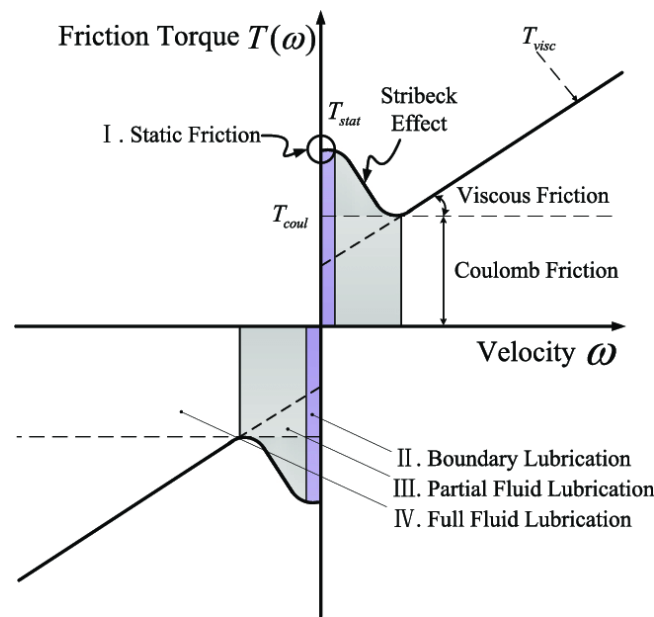


Figure 2. Qualitative representation of the friction phases.

$$T_{str} = \left[T_{coul} + (T_{stat} - T_{coul})e^{-\left(\frac{\omega}{\omega_{str}}\right)^2} + \beta\omega \right] \text{sign}(\omega) \quad (2)$$

In Equation (2), T_{coul} represents the level of torque required to balance the constant friction at each angular velocity, T_{stat} is the peak torque of the static friction to be balanced at nearly zero velocity, β is the coefficient of viscous friction and ω_{str} is the angular velocity at which the actual Stribeck phenomenon is triggered. In essence, the Stribeck phenomenon is the transient that links static friction with the classical model of viscous friction with a model linearly dependent on angular velocity. Note the importance of including Stribeck's model when studying phenomena at relatively low speeds, such as in robotic applications where a certain amount of precision in positioning the end-effector is required.

2.2. Related Work

The authors of [5] propose the reduction of torque ripple by means of an exhaustive analysis of inverter commutations, in electric drives in which a brushless DC motor is used for applications in which it is required to maintain a constant electromagnetic torque at a constant medium speed. Compared to that work, we present several contributions, including a much more exhaustive modelling of the electric motor dynamics, and having considered a complex mechatronic system as a use case, not limiting the analysis to the control of the actuator alone.

The authors in [6] propose the use of jerk estimation to limit the effect of torque ripple during angular acceleration and deceleration of the rotor axis, for position control in robotic applications. The paper lacks a comprehensive description of the synchronous motor model used, and the authors assume that the robot is comparable to an ideal load modelled as a second-order system. In doing so, motions of non-linearity effects due to the robot mechanics are neglected. With respect to this work, we propose a modelling of the mechatronic system with a high degree of detail, so that the quality of the control obtained can be effectively assessed.

In [7], the particularity of the braking torque control is illustrated at first. Then, the speed ranges where the commutation torque ripple can be reduced are derived under the nine common braking PWM strategies. In order to simplify the process of the theory derivation, the corresponding relations between the nine braking PWM strategies and the different pairs of switch states are fully utilized. Finally, a braking control method is proposed to slow down the BLDCM smoothly and stably. Compared to ours, the problem with this work is that the solution found is not generally suitable for position control, and the dynamics of the mechanical load are not considered.

An adaptive nonlinear technique for controlling the angular velocity of a simple BLDC motor for mechatronic applications is proposed in [8]. The algorithm proposed by the authors involves the on-line estimation of some parameters of the electric motor model and the use of a lookup table to approximate the back e.m.f from the angular position measurement and the result of the angular speed control loop. In addition, in this the reduction of the torque ripple is subordinate to the writing of a mathematical model of it. Compared to our proposed work, besides not considering the insertion of the electric drive within the context of complex mechatronic systems, the presence of adaptive algorithms weighs down the computational cost of the algorithm, which may not be adequate for low-cost embedded systems.

The authors in [9] propose a control strategy based on Fuzzy PID logic, for DC motors in a Cartesian Robot, with the aim of absorbing the harmful effects of the Stribeck phenomenon. In contrast to our work, the robot model is represented by a second-order mechanical system and does not take into account the presence of the electric drive. Moreover, the authors' results only show how their strategy is better than a classical PID with regard to the angular positioning of the motor, without showing how the robot's end-effector behaves in the working plane.

In [10], an innovative Adaptive controller is presented that exploits learning paradigm within a novel control architecture, in order to reduce Stribeck effect in a BLDC electric drive. In this work, it is presented the validation based on an exhaustive modelling of electric drive and motor limiting the analysis to only the Stribeck ripple effect without consider a complex mechatronic system architecture, as instead is presented in this paper.

The authors of [11] presents a control algorithm able to suppress the phenomenon of cogging torque in high power brushless motors, while [12] shows an anti-cogging control strategy inserted in the context of a Sensor-less control architecture using an extended Kalman filter. In [13], the authors propose also the validation of the proposed algorithm exploiting the paradigms of the Co-Simulation and Formal methods. The contribution with respect to these articles is to propose an algorithm suitable for absorbing the effects due to more than one source of torque ripple, and to place the control strategy in the context of a complex mechatronic system of industrial interest.

3. System Modelling

3.1. Robot Dynamics

As far as the mechanical part of the mechatronic system is concerned, reference is made to a serial robot with two rotary joints, which essentially represents the structure (neglecting prismatic part) of a SCARA robot [14]. A schematic of the essential structure is shown in Figure 3.

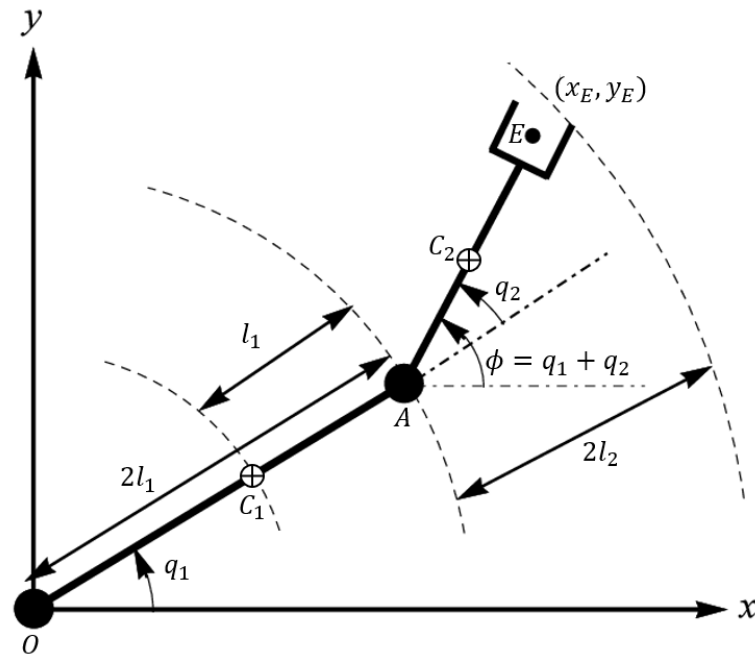


Figure 3. Schematic representation of a 2R planar robot.

Table 1 shows the names and meaning of the parameters needed to define the geometry, kinematics and dynamics of the robot model. In order to include the contribution of the actuator dynamics within the differential equations describing the robot dynamics, the Lagrange method [15] is used. Therefore, it is necessary to write the expressions of the velocities of the centres of mass of the links of the robot, as in Equation (3).

Table 1. Parameters of the dynamic Robot model.

Variables Name	Meaning
m_1, m_2	mass of robot's links
M_1, M_2	mass of the Synchronous motors
I_1, I_2	inertia moments of the Robot's links
J_1, J_2	inertia moments of the actuator rotors
a_1, a_2	semi-lengths of the Robot's links
r_1, r_2	reduction ratios
$q_1, q_2, \dot{q}_1, \dot{q}_2$	relatives angular positions and speed
$\theta_1, \theta_2, \omega_1, \omega_2$	absolutes angular positions and speed

$$\begin{aligned}
 \vec{V}_{C_1} &= \vec{V}_O + \vec{\omega}_1 \wedge O\vec{C}_1 = \dot{q}_1 \vec{u}_z \wedge a_1 [\cos(q_1) \vec{u}_x + \sin(q_1) \vec{u}_y] \\
 \vec{V}_{C_2} &= \vec{V}_A + \vec{\omega}_2 \wedge A\vec{C}_2 = \\
 &= -[2\dot{q}_1 a_1 \sin q_1 + (\dot{q}_1 + \dot{q}_2) a_2 \sin(q_1 + q_2)] \vec{u}_x + \\
 &+ [2\dot{q}_1 a_1 \cos q_1 + (\dot{q}_1 + \dot{q}_2) a_2 \cos(q_1 + q_2)] \vec{u}_y
 \end{aligned} \tag{3}$$

In Equation (3) are reported the expressions of the two mass centers of the robot's links, C_1 and C_2 , exploiting the fundamental law of the kinematics. The link lengths are

denoted by a_1 and a_2 , and are equivalent to the length of the vectors \vec{OC}_1 and \vec{AC}_2 , while \vec{u}_x and \vec{u}_y represent the versors of the Cartesian reference system shown as in Figure 3. We denote by q_1 and q_2 the relative angular positions of the robot links and by ω_1 and ω_2 the absolute velocities of the two links.

Consequently, the kinetic and potential energies associated with each element of the kinematic chain are written down in the Equation (4).

$$\begin{aligned}
 K_{L_1} &= \frac{1}{2} m_1 |\vec{V}_{C_1}|^2 + \frac{1}{2} I_1 |\vec{\omega}_1|^2 = \frac{1}{2} (I_1 + m_1 a_1^2) \dot{q}_1^2 = \frac{1}{2} (I_1 + m_1 a_1^2) \frac{\dot{\theta}_1^2}{r_1^2} \\
 K_{M_1} &= \frac{1}{2} J_1 \dot{\theta}_1^2 \\
 K_{L_2} &= \frac{1}{2} m_2 |\vec{V}_{C_2}|^2 + \frac{1}{2} I_2 |\vec{\omega}_2|^2 = \\
 &= \frac{1}{2} I_2 \left(\frac{\dot{\theta}_1}{r_1} + \frac{\dot{\theta}_2}{r_2} \right)^2 + \frac{1}{2} m_2 \left[4 \frac{\dot{\theta}_1^2}{r_1^2} a_1^2 + \left(\frac{\dot{\theta}_1}{r_1} + \frac{\dot{\theta}_2}{r_2} \right)^2 a_2^2 + 4 \frac{\dot{\theta}_1}{r_1} \left(\frac{\dot{\theta}_1}{r_1} + \frac{\dot{\theta}_2}{r_2} \right) a_1 a_2 \cos \left(\frac{\theta_2}{r_2} \right) \right] \\
 K_{M_2} &= \frac{1}{2} J_2 \dot{\theta}_2^2 + \frac{1}{2} M_2 |\vec{V}_A|^2 = \frac{1}{2} \left(J_2 \dot{\theta}_2^2 + 4 M_2 a_1^2 \frac{\dot{\theta}_1^2}{r_1^2} \right) \\
 U_{L_1} &= -m_1 \vec{OC}_1 \bullet \vec{g} = m_1 a_1 g \sin \left(\frac{\theta_1}{r_1} \right) \\
 U_{M_1} &= 0 \\
 U_{L_2} &= -m_2 \vec{OC}_2 \bullet \vec{g} = m_2 g \left[2a_1 \sin \left(\frac{\theta_1}{r_1} \right) + a_2 \sin \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2} \right) \right] \\
 U_{M_2} &= 2M_2 g a_1 \sin \left(\frac{\theta_1}{r_1} \right)
 \end{aligned} \tag{4}$$

In particular, K_{L_1}, U_{L_1} are the kinetic and potential energies associated with the first link of the robot, K_{M_1}, U_{M_1} are the kinetic and potential energies associated with the motor implementing the first link, K_{L_2}, U_{L_2} are the kinetic and potential energies associated with the second link of the robot while K_{M_2}, U_{M_2} are the kinetic and potential energies of the motor implementing the second link of the robot.

$$\begin{aligned}
 K &= K_{L_1} + K_{M_1} + K_{L_2} + K_{M_2} = A(\theta_2) \dot{\theta}_1^2 + B_0 \dot{\theta}_2^2 + C(\theta_2) \dot{\theta}_1 \dot{\theta}_2 \\
 &= \left[A_0 + A_1 \cos \frac{\theta_2}{r_2} \right] \dot{\theta}_1^2 + B_0 \dot{\theta}_2^2 + \left[C_0 + C_1 \cos \frac{\theta_2}{r_2} \right] \dot{\theta}_1 \dot{\theta}_2 \\
 U &= U_{L_1} + U_{M_1} + U_{L_2} + U_{M_2} = D_1 \sin \frac{\theta_1}{r_1} + D_2 \sin \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2} \right)
 \end{aligned} \tag{5}$$

The expression of the kinetic and potential energy of the whole mechanical system is given in Equation (5). Where A_0, A_1, B_0, C_1, D_1 and D_2 are the coefficients whose expression is given in Equation (6). This step allows to write in a fairly compact form the Lagrangian, which will be used to derive the expressions of the robot dynamics. It should be noted that in this context a 'rigid' relationship has been imposed between the angular variables of the actuators and those of the links of the type $\theta_k = r_k q_k$, assuming an ideal motion transmission system, without loss of generality of the proposed method.

$$\begin{aligned}
A_0 &= \frac{1}{2} \left[\frac{(I_1 + m_1 a_1^2)}{r_1^2} + J_1 + \frac{I_2}{r_1^2} + m_2 \frac{(4a_1^2 + a_2^2)}{r_1^2} \right] \\
A_1 &= \frac{m_2 a_1 a_2}{2r_1^2} \\
B_0 &= \frac{I_2 + m_2 a_2^2}{2r_2^2} + \frac{J_2}{2} \\
C_0 &= \frac{I_2 + m_2 a_2^2}{r_1 r_2} \\
C_1 &= \frac{m_2 a_1 a_2}{2r_1 r_2} \\
D_1 &= (m_1 + 2m_2 + 2M_2) a_1 g \\
D_2 &= m_2 a_2 g
\end{aligned} \tag{6}$$

In Equation (7), the definition of the functional is given, which is elaborated according to the criterion of Lagrange's method mentioned earlier, as given in Equation (8).

$$L(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) \stackrel{\text{def}}{=} K(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) - U(\theta_1, \theta_2) \tag{7}$$

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}_k} - \frac{\partial L}{\partial \theta_k} = Q_k \stackrel{\text{def}}{=} \sum_{i=1}^N \left(\vec{F}_{i,ext} \cdot \frac{\partial \vec{r}_i}{\partial \theta_k} + \vec{\tau}_i \cdot \frac{\partial \vec{\omega}_i}{\partial \dot{\theta}_k} \right) \tag{8}$$

Applying Lagrange's method gives the result given in matrix form in Equation (9). Note that this equation can be put into "Standard" form, as given in Equation (10), where $M(\theta_1, \theta_2)$ represents the inertia matrix, which as can be seen is symmetric, $C(\dot{\theta}_1, \dot{\theta}_2, \theta_1, \theta_2)$ is the matrix of the centrifugal and Coriolis terms, and is already in the form of Christoffel symbols such that $\dot{M} - 2C$ is anti-symmetric, and finally $G(\theta_1, \theta_2)$ is the vector of the gravitational and potential terms. The vector $[\tau_1 \ \tau_2]^T$ represents the vector of link control pairs, the expression of which describes the connection between the electrical and mechanical drive systems of the robot.

$$\begin{bmatrix} 2(A_1 \cos \frac{\theta_2}{r_2} + A_0) & C_1 \cos \frac{\theta_2}{r_2} + C_0 \\ C_1 \cos \frac{\theta_2}{r_2} + C_0 & 2B_0 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -A_1 \frac{\dot{\theta}_2}{r_2} \sin \frac{\theta_2}{r_2} & -(A_1 \frac{\dot{\theta}_1}{r_1} + C_1 \frac{\dot{\theta}_2}{r_2}) \sin \frac{\theta_2}{r_2} \\ A_1 \frac{\dot{\theta}_1}{r_1 r_2} \sin \frac{\theta_2}{r_2} & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} \frac{D_1}{r_1} \cos \frac{\theta_1}{r_1} + \frac{D_2}{r_2} \cos \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2} \right) \\ \frac{D_2}{r_2} \cos \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2} \right) \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{9}$$

$$M(\theta_1, \theta_2) \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + C(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + G(\theta_1, \theta_2) = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{10}$$

3.2. Synchronous Motors Modelling

In this work, without any loss of generality, we consider permanent magnets arranged on the rotor surface, with sinusoidal distribution of the counter electro-motor force [16]. This allows to write the typical equations of an isotropic synchronous machine.

Table 2 shows the names and meaning of the coefficients that characterise the dynamics of the actuators. According to the Unified Theory of Electrical Machines [17], the dynamics of an electrical machine can be described by an equivalent electrical and magnetic circuit. For a synchronous motor with permanent magnets (isotropic), this results in circuit equilibrium in Equation (11).

Table 2. Parameters of the Dynamic Actuators model

Variables Name	Meaning
R_1, R_2	stator resistances
L_1, L_2	stator inductances
λ_1, λ_2	magnetic flux coefficients
p_1, p_2	pole pairs
β_1, β_2	friction coefficients
$T_{i,k}; \alpha_{i,k}$	Cogging Torque Fourier Series coefficients
$T_{coul,k}; T_{stat,k}$	coulomb and static friction torque model
ω_{str}	Stribek rotary speed limit
$\theta_1, \theta_2, \Omega_1, \Omega_2$	absolutes angular position and speed

$$\begin{bmatrix} U_{a,k} \\ U_{b,k} \\ U_{c,k} \end{bmatrix} = R_k \begin{bmatrix} I_{a,k} \\ I_{b,k} \\ I_{c,k} \end{bmatrix} + L_k \frac{d}{dt} \begin{bmatrix} I_{a,k} \\ I_{b,k} \\ I_{c,k} \end{bmatrix} - p_k \frac{d\theta_k}{dt} \lambda_k \begin{bmatrix} \sin(p_k \theta_k) \\ \sin\left(p_k \theta_k - \frac{2\pi}{3}\right) \\ \sin\left(p_k \theta_k - \frac{4\pi}{3}\right) \end{bmatrix} \quad (11)$$

In Equation (12) the mechanical equilibrium for the rotational axis of each electric motor is given. By k , we denote an index related to the k th motor, T_{str} is the Stribek torque according to Equation (2), T_{cog} is the Cogging torque according to Equation (1) and T_{load} is the load torque for the motor.

$$J_k \frac{d^2 \theta_k}{dt^2} + T_{str,k} = T_{em,k} + T_{cog,k} - T_{load,k} \quad (12)$$

$$T_{em,k} = -p_k \lambda_k \left[I_{a,k} \sin(p_k \theta_k) + I_{b,k} \sin\left(p_k \theta_k - \frac{2\pi}{3}\right) + I_{c,k} \sin\left(p_k \theta_k - \frac{4\pi}{3}\right) \right]$$

The connection between the electric drive part and the mechanical part of the robot exploits the hypothesis of motion transmission by means of an ideal gearbox, so that between q_i and θ_i ($\forall i$) there is a congruence relation given by $\theta_i = r_i q_i$, where r_i represents the i th reduction ratio. In this way, the power flow at each robot joint is also fully conserved so that we can write the relation $\theta_i = T_{em,i} \omega_i$, i.e., $\tau_i = r_i T_{em,i}$. Hence, the connection between the robot Equations (9) and (10) and the motor Equations (11) and (12).

4. Control System Design

4.1. State Space Representation

As a first step in the control vector design, the dynamic equations of the system are reorganised into state vector form. Furthermore, it is convenient to apply the coordinate transformations given in Equation (13) to the electrical equilibrium equations of the motors.

$$\vec{X}_{\alpha\beta,k} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \vec{X}_{abc,k} = B \vec{X}_{abc,k} \quad \vec{X}_{dq,k} = \begin{bmatrix} \cos \theta_k & \sin \theta_k \\ -\sin \theta_k & \cos \theta_k \end{bmatrix} \vec{X}_{\alpha\beta,k} = P(\theta_k) \vec{X}_{\alpha\beta,k} \quad (13)$$

In Equation (13), the matrix B is called Blondell's matrix and allows to reduce the number of equations so as to describe a three-phase motor as if it were single-phase, while the matrix $P(\theta_k)$ is called Park's matrix, and is needed to make the single-phase representation equivalent to a DC motor, eliminating the dependence on θ_k in the expression of the electromagnetic torque.

$$\begin{bmatrix} U_{d,k} \\ U_{q,k} \end{bmatrix} = R_k \begin{bmatrix} I_{d,k} \\ I_{q,k} \end{bmatrix} + L_k \frac{d}{dt} \begin{bmatrix} I_{d,k} \\ I_{q,k} \end{bmatrix} + p_k \frac{d\theta_k}{dt} \begin{bmatrix} -L_k I_{q,k} \\ \lambda_k + L_k I_{d,k} \end{bmatrix}; \quad T_{em,k} = \frac{3}{2} p_k \lambda_k I_{q,k} \quad (14)$$

Equation (14) represents Park's model of electrical equilibrium, in which it is noted that the new expression of $T_{em,k}$ is independent of angular position. The state form obtained

is represented in Equations (15) and (16), where it is highlighted that the control variables of an electro-mechanical system are the supply voltages of the motors.

$$\frac{d\vec{x}}{dt} = \vec{f}(\vec{x}) + G(\vec{x})\vec{u} = \vec{f} + \sum_{k=1}^{n_u} \vec{g}(\vec{x})u_k$$

$$\vec{f} = \begin{bmatrix} f_{id,1} \\ f_{iq,1} \\ f_{id,2} \\ f_{iq,2} \\ f_{\theta_1} \\ f_{\theta_2} \\ \vec{f}_\omega \end{bmatrix} = \begin{bmatrix} \frac{1}{L_1}(-R_1 I_{d,1} + p_1 \omega_1 L_1 I_{q,1}) \\ \frac{1}{L_1}(-R_1 I_{q,1} - p_1 \omega_1 (\lambda_1 + L_1 I_{d,1})) \\ \frac{1}{L_2}(-R_2 I_{d,2} + p_2 \omega_2 L_2 I_{q,2}) \\ \frac{1}{L_2}(-R_2 I_{q,2} - p_2 \omega_2 (\lambda_2 + L_2 I_{d,2})) \\ \omega_1 \\ \omega_2 \\ \vec{f}_\omega \end{bmatrix}; \quad G = \begin{bmatrix} \frac{1}{L_1} & 0 & 0 & 0 \\ 0 & \frac{1}{L_1} & 0 & 0 \\ 0 & 0 & \frac{1}{L_2} & 0 \\ 0 & 0 & 0 & \frac{1}{L_2} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (15)$$

$$\vec{f}_\omega = \begin{bmatrix} 2(A_1 \cos \frac{\theta_2}{r_2} + A_0) & C_1 \cos \frac{\theta_2}{r_2} + C_0 \\ C_1 \cos \frac{\theta_2}{r_2} + C_0 & 2B_0 \end{bmatrix}^{-1} \left\{ \begin{bmatrix} T_{em,1} + T_{cog,1} - T_{str,1} \\ T_{em,2} + T_{cog,2} - T_{str,2} \end{bmatrix} - \right.$$

$$\left. \begin{bmatrix} -A_1 \frac{\omega_2}{r_2} \sin \frac{\theta_2}{r_2} & -\left(A_1 \frac{\omega_1}{r_1} + C_1 \frac{\omega_2}{r_2}\right) \sin \frac{\theta_2}{r_2} \\ A_1 \frac{\omega_1}{r_1 r_2} \sin \frac{\theta_2}{r_2} & 0 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} + \begin{bmatrix} \frac{D_1}{r_1} \cos \frac{\theta_1}{r_1} + \frac{D_2}{r_2} \cos \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2}\right) \\ \frac{D_2}{r_2} \cos \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2}\right) \end{bmatrix} \right\} \quad (16)$$

The state vector is $\vec{x} = [I_{d,1}, I_{q,1}, I_{d,2}, I_{q,2}, \theta_1, \theta_2, \omega_1, \omega_2]$ while the control vector is given by $\vec{u} = [U_{d,1}, U_{q,1}, U_{d,2}, U_{q,2}]$. It is important to note that proceeding with modelling as shown above provides a more accurate and less abstract level of description of the physical process than is often done in the development of control algorithms for robotic applications. Typically, the development of control algorithms, such as computed torque [18] or back-stepping [19], assumes that the control vector is represented by the actuation torques at the robot joints, inherently assuming that the electrical drive is capable of delivering the required torques. This is not only wrong from the point of view of describing the physical system, making the mathematical model too abstract and untrue in representing the link between the system's inputs and outputs, but it also does not allow strategies to be developed to reduce the non-linearities of the implementation system, as they are directly neglected. This is also an important contribution of the work we have presented, as it makes it possible to develop a deterministic control strategy without having to compensate for deficiencies in mathematical modelling through computationally burdensome tools, such as the use of predictive, adaptive or machine learning control paradigms.

4.2. Application of the Feedback Linearization Technique

As already mentioned, the control technique selected is Feedback linearization (input-output) [20]. This non-linear control technique allows the contribution given by the a priori knowledge through the mathematical model of the total torque ripple, the sum of the Cogging and Stribeck phenomena, to be inserted directly into the control vector.

$$\begin{aligned} h_1(\vec{x}) &= 2a_1 \cos \frac{\theta_1}{r_1} + 2a_2 \cos \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2}\right) = h_1(\theta_1, \theta_2) \\ h_2(\vec{x}) &= 2a_1 \sin \frac{\theta_1}{r_1} + 2a_2 \sin \left(\frac{\theta_1}{r_1} + \frac{\theta_2}{r_2}\right) = h_2(\theta_1, \theta_2) \\ h_3(\vec{x}) &= i_{d,1} \\ h_4(\vec{x}) &= i_{d,2} \end{aligned} \quad (17)$$

In Equation (17) the expressions of the controlled outputs are given, where h_1 and h_2 represent the expressions of the Cartesian coordinates of the robot end-effector, while h_3 and h_4 are the direct axis currents in the Park model of the motors. This choice allows to control the position of the end-effector, which is in fact a function of the angular positions of the motors, and the current of the motors so as to guarantee a certain efficiency in the delivery of the electromagnetic torque, emulating a field-oriented control (FOC) [21].

$$\begin{aligned} \frac{dh_i}{dt} &= \frac{\partial h_i}{\partial x} \frac{\partial x}{\partial t} = \frac{\partial h_i}{\partial x} \left(\vec{f} + \sum_{k=1}^{n_u} \vec{g}_k u_k \right) = \frac{\partial h_i}{\partial x} \vec{f} = L_f h_i \\ &\dots \\ \frac{d^{\mu_i} h_i}{dt^{\mu_i}} &= \frac{d}{dt} \left(L_f^{\mu_i-1} h_i \right) = \frac{\partial}{\partial x} \left(L_f^{\mu_i-1} h_i \right) \left(\vec{f} + \sum_{k=1}^{n_u} \vec{g}_k u_k \right) = L_f^{\mu_i} h_i + \sum_{k=1}^{n_u} L_{g_k} L_f^{\mu_i-1} h_i u_k \end{aligned} \quad (18)$$

The operative procedure for applying Feedback linearization involves deriving the control outputs as many times as necessary to obtain expressions in which the contributions of the control variables appear. In Equation (18) the expression of the derivative μ_i th for the i th control output is given in general form. We denote the relative degree for the output i th by μ_i , with the condition that $\sum \mu_i \leq n_x$.

$$\begin{bmatrix} L_f^{\mu_1} h_1 \\ L_f^{\mu_2} h_2 \\ \dots \\ L_f^{\mu_l} h_l \end{bmatrix} + \begin{bmatrix} L_{g_1} L_f^{\mu_1-1} h_1 & \dots & L_{g_m} L_f^{\mu_1-1} h_1 \\ L_{g_1} L_f^{\mu_2-1} h_2 & \dots & L_{g_m} L_f^{\mu_2-1} h_2 \\ \dots & \dots & \dots \\ L_{g_1} L_f^{\mu_l-1} h_l & \dots & L_{g_m} L_f^{\mu_l-1} h_l \end{bmatrix} \vec{u} = \vec{v} = \vec{\Gamma}(\vec{x}) + \Sigma(\vec{x}) \vec{u} \quad (19)$$

$$\vec{u} = \Sigma^+(\vec{x}) \left(\vec{v} - \vec{\Gamma}(\vec{x}) \right) + (I - \Sigma^+(\vec{x}) \Sigma(\vec{x})) \lambda$$

Proceeding with the derivation of each control output, in general we obtain the matrix expression given in Equation (19), where with $L_{g_i} L_f h_j(\vec{x}) = \frac{\partial}{\partial \vec{x}} \left(\frac{\partial h_j}{\partial \vec{x}} \vec{f} \right) \vec{g}_i$ is the Lie derivative of the function h_j in the directions of the vector fields \vec{f} and \vec{g}_i respectively.

$$\begin{aligned} \dot{\vec{z}} &= \begin{bmatrix} A_1 & 0 & 0 & \dots & 0 \\ 0 & A_2 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & A_l \end{bmatrix} \vec{z} + \begin{bmatrix} B_1 & 0 & \dots & 0 \\ 0 & B_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & B_l \end{bmatrix} \vec{v} \\ \vec{z} &= \begin{bmatrix} \vec{z}_1 \\ \vec{z}_2 \\ \dots \\ \vec{z}_l \end{bmatrix}; \quad \vec{z}_i = \begin{bmatrix} h_i \\ L_f h_i \\ \dots \\ L_f^{\mu_i-1} h_i \end{bmatrix}; \quad A_i = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}; \quad B_i = \begin{bmatrix} 0 \\ 0 \\ \dots \\ 0 \\ 1 \end{bmatrix} \end{aligned} \quad (20)$$

The Feedback linearization procedure results in a change of coordinates in which the dynamics of the new state vector is linear, as shown in Equation (20). Note that the state representation in addition to being linear represents a chain of integrators, consequently the pair (A_i, B_i) turns out to be completely reachable/controllable $\forall i$.

$$\vec{z} = \begin{bmatrix} \vec{z} \\ \vec{\sigma}(\vec{x}) \end{bmatrix} \text{ with } \vec{\sigma}(\vec{x}) \text{ s.t. } \frac{\partial \vec{z}}{\partial \vec{x}} \neq 0 \quad (21)$$

In general, a change of basis with reduction of the cardinality of the state space is obtained. Formally, this requires that certain mathematical conditions are fulfilled, which define the criterion of zero dynamics [22]. The first condition is given in Equation (21), where we impose that the Jacobian of the base change is formally lawful (invertible). This means that the completion $\vec{\sigma}(\vec{x}) \in \mathbb{R}^{n_x - \sum \mu_i}$ will have to be constructed appropriately to satisfy the condition.

$$\dot{\vec{\sigma}}(\vec{x}) = \vec{\gamma}(\vec{z}, \vec{\sigma}) + \vec{\xi}(\vec{z}, \vec{\sigma}) v \quad (22)$$

The completion vector $\vec{\sigma}(\vec{x})$ will in general have a non-linear state representation, dependent on both the new and old state variables and also on the new control vector, as given in Equation (22).

$$L_{g_k} \sigma_i = 0 \quad \forall i, k \rightarrow \dot{\vec{\sigma}}(\vec{x}) = \vec{\gamma}(\vec{z}, \vec{\sigma}) \rightarrow \dot{\vec{\sigma}}(\vec{x}) = \vec{\gamma}(0, \vec{\sigma}) \text{ A.S.} \quad (23)$$

The second condition, given in Equation (23), serves to eliminate the contribution of the new control vector v from the dynamics of the vector $\vec{\sigma}$. In this way it is possible to implement the zero-dynamics criterion, which guarantees us that even in the case of a reduction of the cardinality of the state space, the vector v constructed by means of a Feedback linearization technique is able to asymptotically stabilize even the state variables that serve only to make the base change formally lawful.

$$\begin{aligned} \frac{dh_3}{dt} &= \frac{dI_{d,1}}{dt} \rightarrow U_{d,1} \rightarrow \mu_3 = 1 \\ \frac{dh_4}{dt} &= \frac{dI_{d,2}}{dt} \rightarrow U_{d,2} \rightarrow \mu_4 = 1 \\ \frac{dh_i}{dt} &= \frac{\partial h_i}{\partial \theta_1} \frac{d\theta_1}{dt} + \frac{\partial h_i}{\partial \theta_2} \frac{d\theta_2}{dt} = \phi_1(\theta_1, \theta_2)\omega_1 + \phi_2(\theta_1, \theta_2)\omega_2 = \Phi(\theta_1, \theta_2, \omega_1, \omega_2) \\ \frac{d^2 h_i}{dt^2} &= \frac{d\Phi}{dt} = \frac{\partial \Phi}{\partial \theta_1} \omega_1 + \frac{\partial \Phi}{\partial \theta_2} \omega_2 + \frac{\partial \Phi}{\partial \omega_1} \dot{\omega}_1 + \frac{\partial \Phi}{\partial \omega_2} \dot{\omega}_2 = \Psi(\theta_1, \theta_2, \omega_1, \omega_2, I_{q,1}, I_{q,2}) \\ \frac{d^3 h_i}{dt^3} &= \frac{d\Psi}{dt} = \frac{\partial \Psi}{\partial \theta_1} \omega_1 + \frac{\partial \Psi}{\partial \theta_2} \omega_2 + \frac{\partial \Psi}{\partial \omega_1} \dot{\omega}_1 + \frac{\partial \Psi}{\partial \omega_2} \dot{\omega}_2 + \frac{\partial \Psi}{\partial I_{q,1}} \dot{I}_{q,1} + \frac{\partial \Psi}{\partial I_{q,2}} \dot{I}_{q,2} \rightarrow U_{q,i} \rightarrow \mu_i = 3, i = 1, 2 \end{aligned} \quad (24)$$

As shown in the set of Equation (24), in the case study of our interest, as a result of the choice of control outputs shown above, a “complete” Feedback linearization is obtained in which there is no change in the cardinality of the state space.

$$\begin{aligned} \frac{d^3 h_1}{dt^3} &= \alpha_1(\vec{x}) + \beta_{11}(\vec{x})U_{q,1} + \beta_{12}(\vec{x})U_{q,2} = v_1 \\ \frac{d^3 h_2}{dt^3} &= \alpha_2(\vec{x}) + \beta_{21}(\vec{x})U_{q,1} + \beta_{22}(\vec{x})U_{q,2} = v_2 \\ \frac{dh_3}{dt} &= \frac{dI_{d,1}}{dt} = \alpha_3(\vec{x}) + \frac{1}{L_1}U_{d,1} = v_3 \\ \frac{dh_4}{dt} &= \frac{dI_{d,2}}{dt} = \alpha_4(\vec{x}) + \frac{1}{L_2}U_{d,2} = v_4 \end{aligned} \quad (25)$$

It is operationally verified by the essential steps given in Equation (25) that the control vector \vec{u} can be written as in Equation (26).

$$\vec{u} = \Sigma^{-1}(\vec{v} - \vec{\Gamma}) = \begin{bmatrix} 0 & 0 & \beta_{11} & \beta_{12} \\ 0 & 0 & \beta_{21} & \beta_{22} \\ \frac{1}{L_1} & 0 & 0 & 0 \\ 0 & \frac{1}{L_2} & 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 - \alpha_1 \\ v_2 - \alpha_2 \\ v_3 - \alpha_3 \\ v_4 - \alpha_4 \end{bmatrix} = \begin{bmatrix} U_{d,1} \\ U_{d,2} \\ U_{q,1} \\ U_{q,2} \end{bmatrix} \quad (26)$$

5. Processor-In-the-Loop Simulations

For the validation of the control system described above, the results obtained through a Processor-In-the-Loop (PIL) simulation are reported, in which the control algorithm is implemented on an Embedded platform, a Raspberry Pi 3 model B (RPI3B), with a Broadcom BCM2837 chipset integrating Cortex-A53 @ 1.2 GHz processor, while the model of the drive, synchronous motors and robot mechanics is implemented in the Simulink environment (MATLAB 2018b version). The interface between the embedded system and the simulation environment uses the Raspberry Pi Support from Simulink [23], provided by MathWorks. Figure 4 shows the functional block diagram of the system composed of both the electric drive and the robot dynamics, where it is highlighted that the proposed control system works on the dynamic variables concerning the electric drive and that the trajectory generation system in the robot operating space works on the robot variables. In addition, Figure 4 highlights the division between blocks implemented on the embedded system and functional blocks implemented in the Simulink environment.

Therefore, in order to ensure that the motor control is such that it moves the robot end-effector as desired, it is necessary to introduce a conversion block based on inverse kinematics. In order not to deal with numerical problems and numerical ill-conditioning, the algebraic approach is adopted as given in the set of Equation (27).

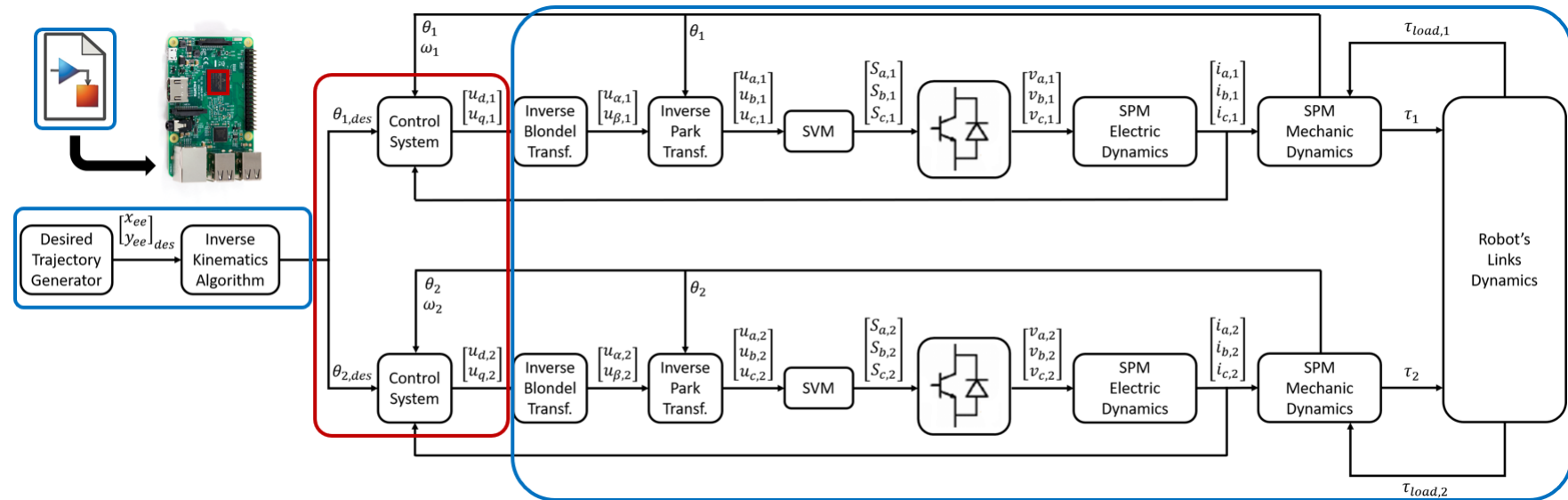


Figure 4. High level Description of the System Architecture.

$$\begin{aligned}
 C_2 &= \frac{x^2 + y^2 - a_1^2 - a_2^2}{2a_1a_2} \\
 S_2 &= \sqrt{1 - C_2^2} \\
 q_2 &= \arctan\left(\frac{S_2}{C_2}\right) \\
 q_1 &= \arctan\left(\frac{y}{x}\right) - \arctan\left(\frac{a_2 \sin q_2}{a_1 + a_2 \cos q_2}\right)
 \end{aligned} \tag{27}$$

Using these equations, we calculate the angular position references for the electric motors that actuate the robot joints, from the references defined for the Cartesian variables of the robot. For applications where trajectories are very complicated, in addition to the “conversion” between Cartesian and angular variables, inverse kinematics should be integrated using geometric methods [24]. The control we have developed is compared with the classic FOC control realised with Cascade architecture, which is the algorithm most commonly implemented in motor controllers such as RoboteQ [25] and Curtis Instruments [26].

Figure 5 represents the comparison between the Cascade control and our proposed non-linear control, regarding the tracking of the position reference of the first actuator of the robot.

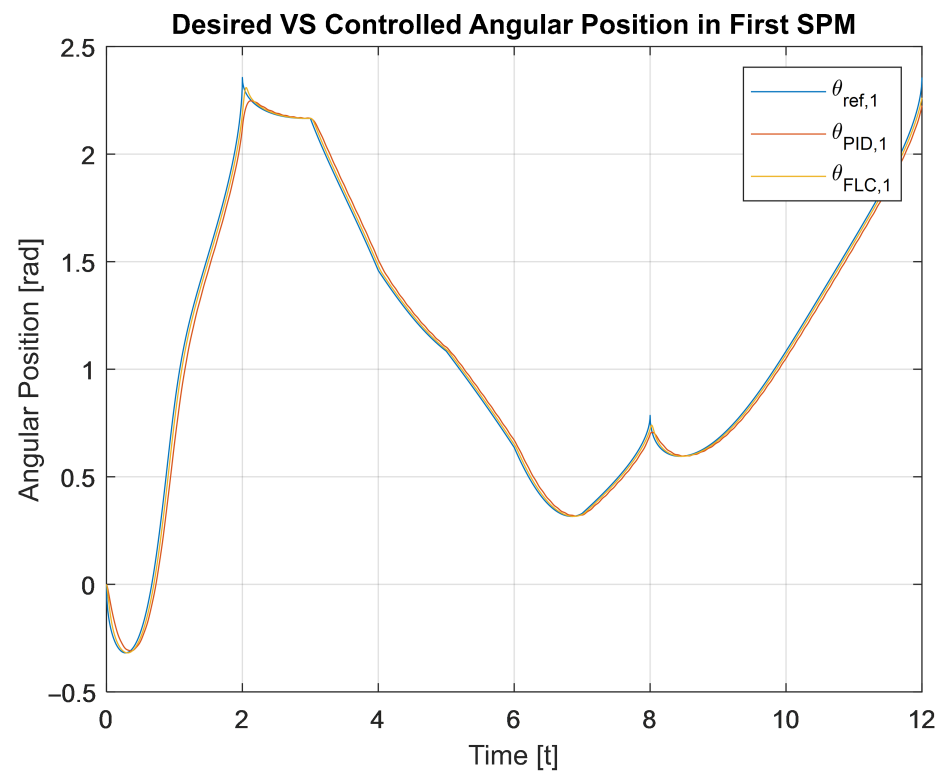


Figure 5. First joint trajectory tracking result.

Figure 6 shows in detail the behaviour of the first actuator, in the application cases of Cascade control and the proposed control, in the “critical” (cusp-shaped) situations. As can be seen, the behaviour of the motor position follows the trajectory more efficiently and also reduces, as expected, the oscillating effect due to the sources of torque ripple. Note that Figures 5 and 6 depict the trend of the same variable. The first one over the whole simulation time interval, while the second one depicts the portion of interest at the cusp management points in the reference signal.

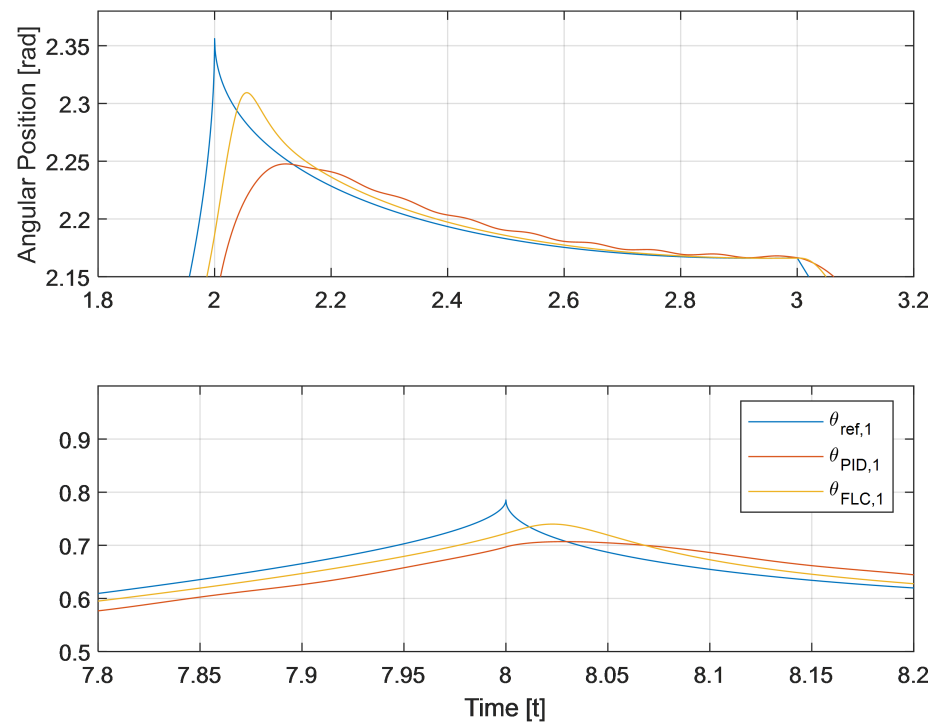


Figure 6. Zoom in the “cuspid” point managing.

Figure 7 shows the result of the position control of the second robot joint, in terms of rotation of the second robot motor. Again, as shown in more detail in Figure 8, our proposed control is more precise and absorbs almost totally the oscillations due to the torque ripple.

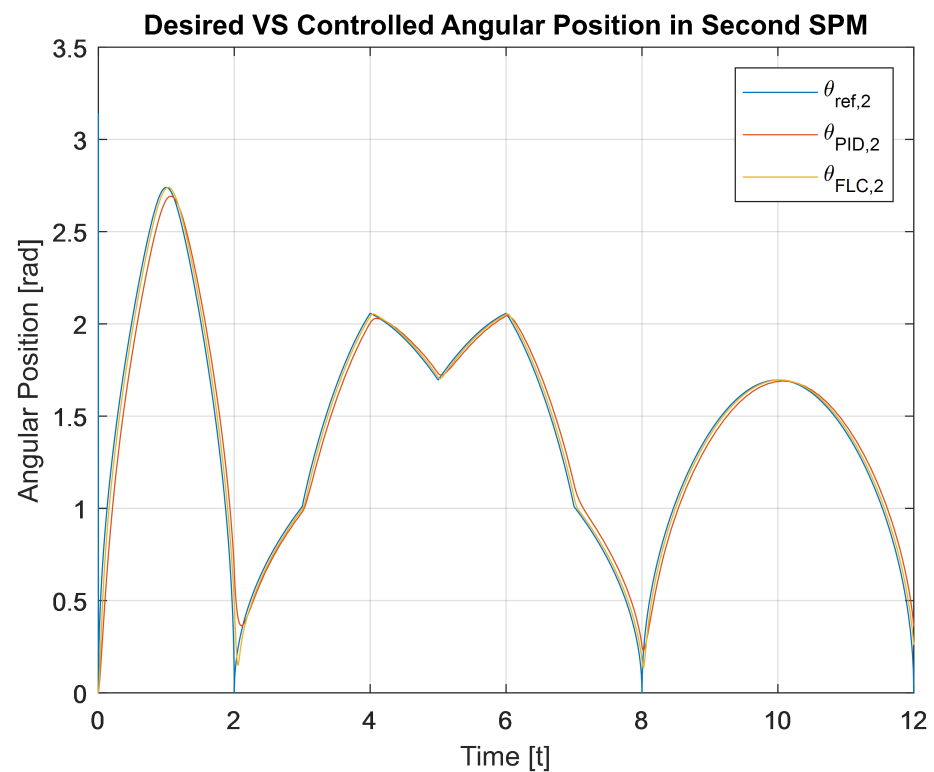


Figure 7. Second joint trajectory tracking result.

Note that Figures 7 and 8 depict the trend of the same variable. The first one over the whole simulation time interval, while the second one depicts the portion of interest at the cusp management points in the reference signal.

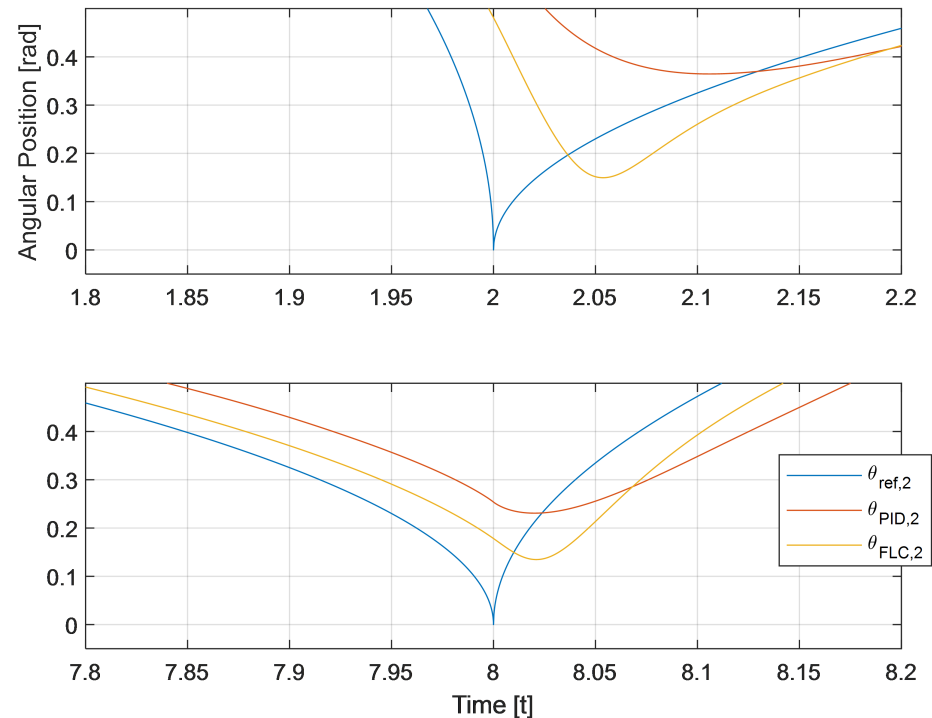


Figure 8. Zoom in a “cuspid” point managing.

It should be noted that the proposed control does not calculate the supply voltages in such a way as to obtain a desired position for the individual actuator, but instead calculates a supply voltage for each motor such that the mechanical structure moves as desired. In fact, the inverse kinematics block is used precisely to convert the reference signals expressed in terms of a time series into the Cartesian coordinates of the robot’s end-effector. Figures 9 and 10 show the result obtained with the proposed control technique, regarding the tracking of the reference signal on the X-coordinate of the robot’s end-effector. Note that Figures 9 and 10 depict the trend of the X component of the end-effector position. The first one over the whole simulation time interval, while the second one depicts the portion of interest at management of the steady-state condition in the reference signal. As could be expected, from the results related to the angular position control of the single actuator, our proposed control largely absorbs the oscillations on the positioning. Figure 10 shows in detail the phases in which it is required to maintain a constant position and it can be seen that in the Cascade control case (red line) there are ripples with respect to the reference (blue line). Our proposed control (yellow line) has almost no error in the phases in which the reference is constant in time, and is however more precise in the linear phases of the reference.

The same applies to the control of the Y-coordinate of the robot’s end-effector. As can be seen in Figures 11 and 12, in the steady-state phase the control we designed is more efficient and more precise. Note that Figures 11 and 12 depict the trend of the Y component of the end-effector position. The first one over the whole simulation time interval, while the second one depicts the portion of interest at management of the steady-state condition and the two linear transients, in the reference signal.

The trajectory in the required robot workspace is computed in terms of (X_{des}, Y_{des}) such that the end-effector traces on the XY plane a profile reminding of spectacle frames. For this proof of concept, the slopes of the reference signals were calculated manually to

ensure that the simulation had a duration of twelve seconds but of course the simulations can be parameterized to track any profile in the Cartesian plane.

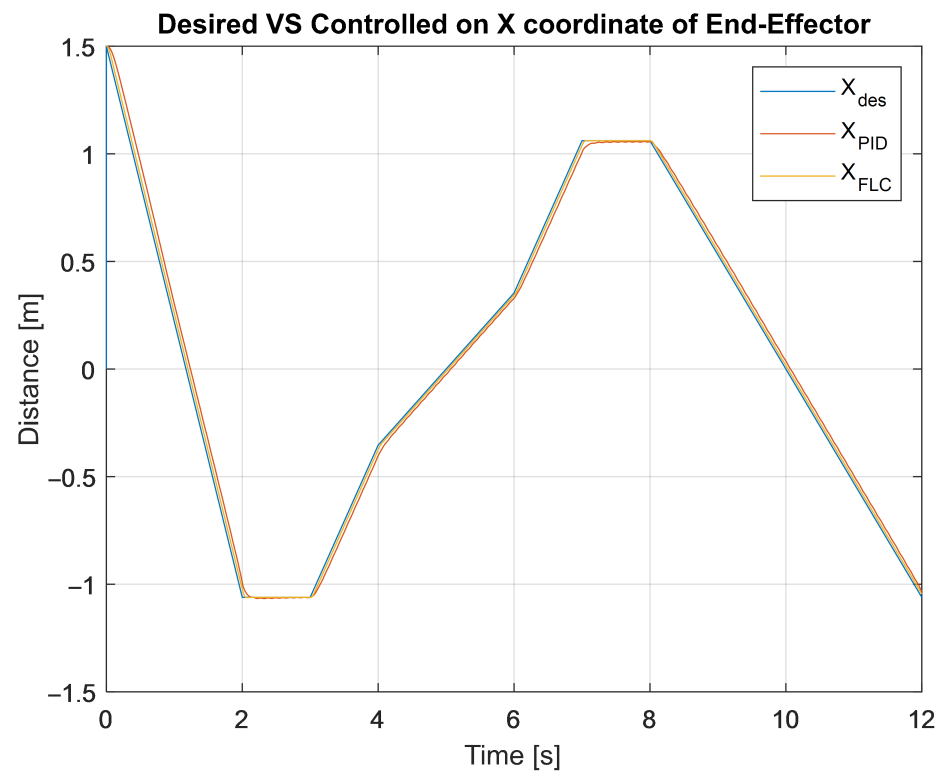


Figure 9. Trajectory tracking result for X component of end-effector position.

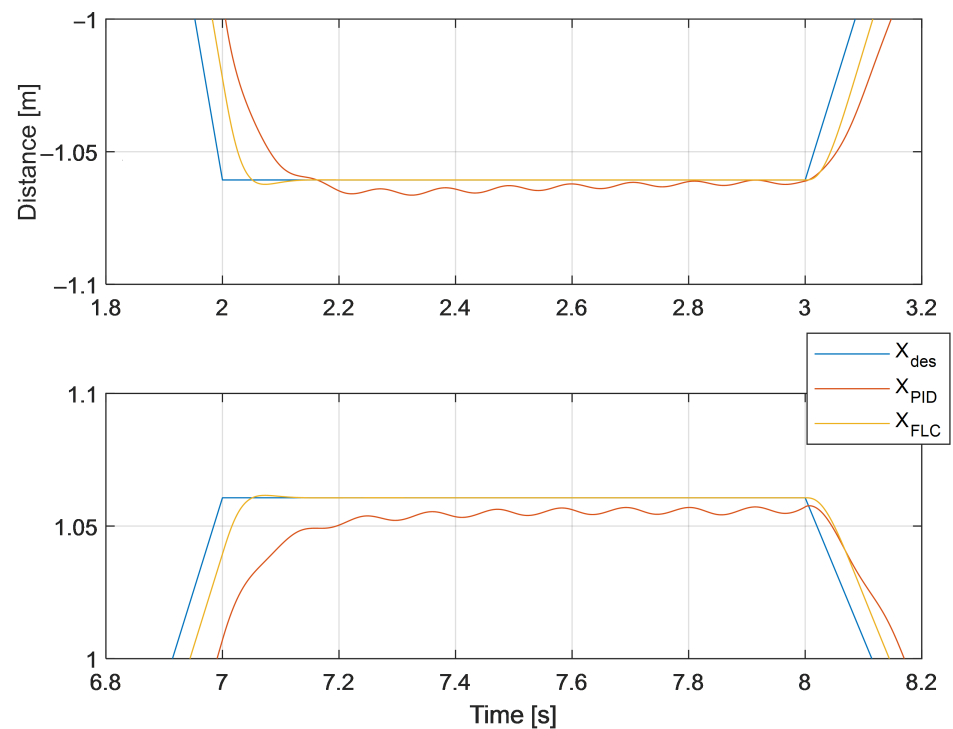


Figure 10. Zoom in a “steady-state” transitory.

A comparison between the control with classical architecture and the proposed control algorithm is shown in Figure 13. The robot is made to start from a position “far” from

the required profile, in order to emulate the transient of reaching the working plane. In Figure 14 are shown in detail how the edge is managed in the case of using the classical control and the proposed control. It is also highlighted that in the upper straight phase of this desired profile, the classical control is not able to compensate in any way the oscillations in the positioning of the robot's end-effector, due to the torque ripple.

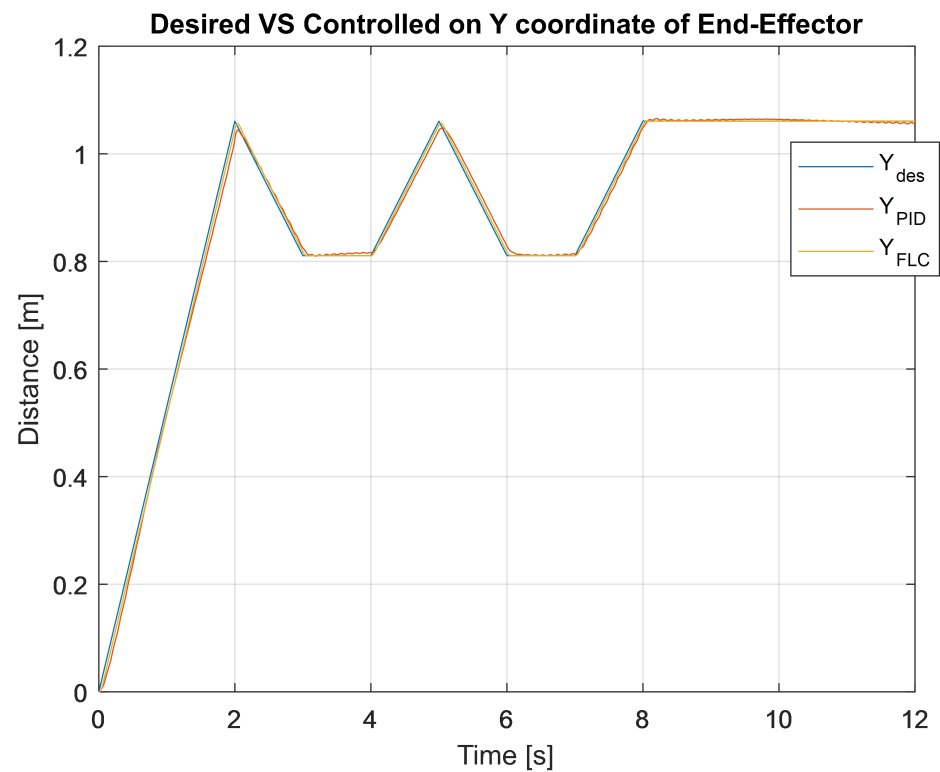


Figure 11. Trajectory tracking result for Y component of end-effector position.

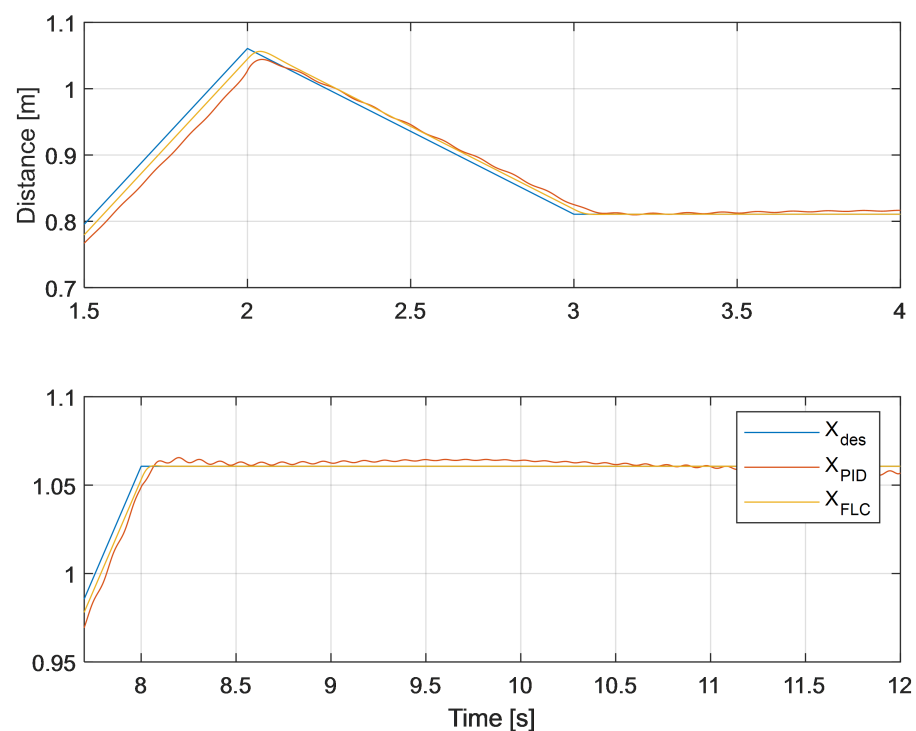


Figure 12. Zoom in a “steady-state” transitory.

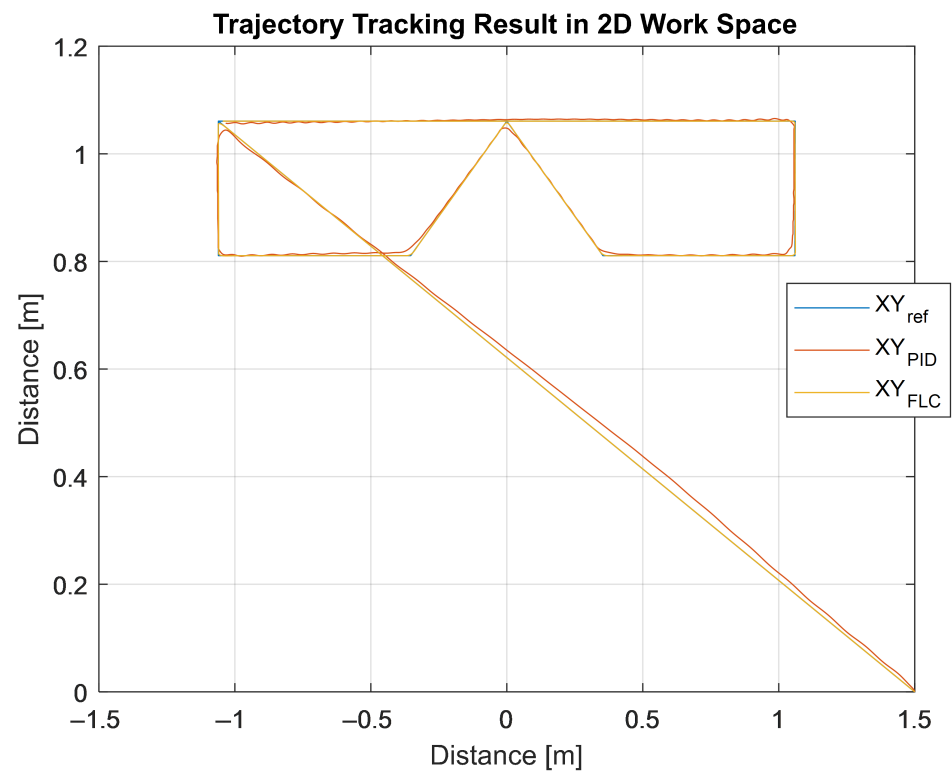


Figure 13. Obtained trace on end-effector work plane.

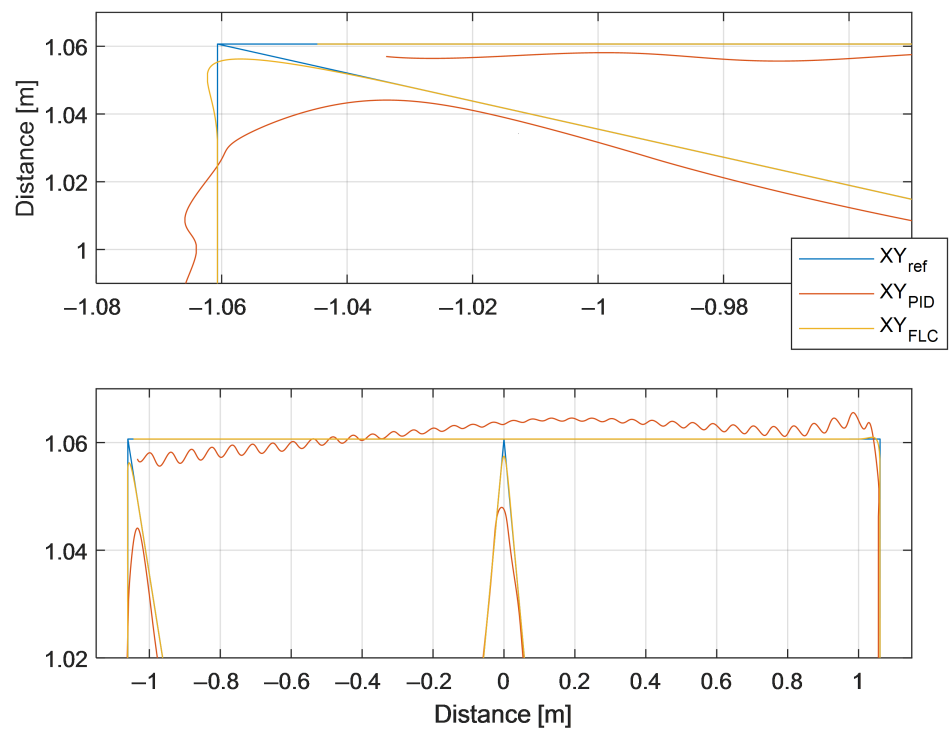


Figure 14. Zoom of previous figure.

6. Conclusions and Future Work

This work shows how to develop a control algorithm that is able to compensate for the intrinsic non-linearities of a complex mechatronic system. The approach introduced envisages a rather accurate mathematical modelling, both regarding the mechanical system, represented in this case by the two links of the robot, and the actuation system composed of the electric drive and the PMSM. The operational procedure to efficiently derive the expression (in closed form) of the control vector is shown. The validation of the control strategy described in the Processor-In-the-Loop context is presented, exploiting SW tools typical of the Model-Based Design approach, such as the automatic generation of low-level code and the HW support provided by MathWorks for the implementation on a Cortex-A53 based embedded platform. The results obtained are promising and the proposed procedure can be generalised in different mechatronic applications such as assisted driving of vehicles or industrial processes. The algorithmic solution we have developed improves the performance of electric drives for manipulator robots and is certainly of great industrial interest, for example in smart manufacturing applications [27], where technological innovation based on advanced research is gaining ground, thanks to the foresight of many industrial partners on the international scene.

Author Contributions: P.D. and S.S. contributed equally. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by MIUR through project Dipartimenti di Eccellenza Crosslab.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gieras, J.F. *Electrical Machines: Fundamentals of Electromechanical Energy Conversion*; CRC Press: Boca Raton, FL, USA, 2016.
2. Ou, J.; Liu, Y.; Qu, R.; Doppelbauer, M. Experimental and theoretical research on cogging torque of PM synchronous motors considering manufacturing tolerances. *IEEE Trans. Ind. Electron.* **2017**, *65*, 3772–3783. [CrossRef]
3. Boroujeni, S.T.; Jalali, P.; Bianchi, N. Analytical modeling of no-load eccentric slotted surface-mounted PM machines: Cogging torque and radial force. *IEEE Trans. Magn.* **2017**, *53*, 1–8. [CrossRef]
4. Shir Guran, A.; Pfeiffer, F.; Popp, K. *Dynamics with Friction: Modeling, Analysis and Experiment*; World Scientific: Singapore, 2001; Volume 2.
5. Shi, T.; Cao, Y.; Jiang, G.; Li, X.; Xia, C. A torque control strategy for torque ripple reduction of brushless DC motor with nonideal back electromotive force. *IEEE Trans. Ind. Electron.* **2017**, *64*, 4423–4433. [CrossRef]
6. Shin, S.C.; Choi, C.H.; Youm, J.H.; Lee, T.K.; Won, C.Y. Position control of PMSM using jerk-limited trajectory for torque ripple reduction in robot applications. In Proceedings of the IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society, Montreal, QC, Canada, 25–28 October 2012; pp. 2400–2405.
7. Shi, T.; Niu, X.; Chen, W.; Xia, C. Commutation torque ripple reduction of brushless DC motor in braking operation. *IEEE Trans. Power Electron.* **2017**, *33*, 1463–1475. [CrossRef]
8. Boroujeni, M.S.; Markadeh, G.A.; Soltani, J. Torque ripple reduction of brushless DC motor based on adaptive input-output feedback linearization. *ISA Trans.* **2017**, *70*, 502–511. [CrossRef] [PubMed]
9. Wang, C.; Wan, Y.; Liang, X. Model establishment with Stribeck friction and fuzzy PID control for Cartesian picking robot system. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 3070–3074.
10. Dini, P.; Saponara, S. Design of Adaptive Controller Exploiting Learning Concepts Applied to a BLDC-Based Drive System. *Energies* **2020**, *13*, 2512. [CrossRef]
11. Dini, P.; Saponara, S. Cogging torque reduction in brushless motors by a nonlinear control technique. *Energies* **2019**, *12*, 2224. [CrossRef]
12. Dini, P.; Saponara, S. Design of an observer-based architecture and non-linear control algorithm for cogging torque reduction in synchronous motors. *Energies* **2020**, *13*, 2077. [CrossRef]
13. Bernardeschi, C.; Dini, P.; Domenici, A.; Palmieri, M.; Saponara, S. Formal Verification and Co-Simulation in the Design of a Synchronous Motor Control Algorithm. *Energies* **2020**, *13*, 4057. [CrossRef]
14. Adept. Datasheet Adept Cobra i600 SCARA Robot. Available online: <https://manualzz.com/download/10047191> (accessed on 9 October 2021).
15. DIRK, J. *Introduction to Classical Mechanics: Solutions to Problems*; World Scientific: Singapore, 2020.
16. Hughes, A.; Drury, B. *Electric Motors and Drives: Fundamentals, Types and Applications*; Newnes: London, UK, 2019.
17. Gerling, D. *Electrical Machines*; Springer: Berlin/Heidelberg, Germany, 2016.

18. Pereira, A.; Althoff, M. Safety control of robots under computed torque control using reachable sets. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 331–338.
19. Guo, Q.; Zhang, Y.; Celler, B.G.; Su, S.W. Neural adaptive backstepping control of a robotic manipulator with prescribed performance constraint. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *30*, 3572–3583. [[CrossRef](#)] [[PubMed](#)]
20. Boufadene, M. *Nonlinear Control Systems Using MATLAB®*; CRC Press: Boca Raton, FL, USA, 2018.
21. Jang, D.; Choe, G.; Ehsani, M. Vector control of three-phase AC machines: system development in the practice. In *Power Systems*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2015.
22. Astolfi, A.; Marconi, L. *Analysis and Design of Nonlinear Control Systems*; Springer: Berlin/Heidelberg, Germany, 2008.
23. MathWorks. Raspberry Pi Support from Simulink. Available online: <https://it.mathworks.com/hardware-support/raspberry-pi-simulink.html> (accessed on 9 October 2021).
24. Božek, P.; Lozkin, A.; Gorbushin, A. Geometrical method for increasing precision of machine building parts. *Procedia Eng.* **2016**, *149*, 576–580. [[CrossRef](#)]
25. RoboteQ. RoboteQ Official Site Web. Available online: <https://www.roboteq.com/index.php> (accessed on 9 October 2021).
26. Curtis Instruments. Curtis Instrument Official Site Web. Available online: <https://www.curtisinstruments.com/products/motor-controllers/> (accessed on 9 October 2021).
27. IFM Electronic. Processes in Automotive Industry. Available online: <https://www.ifm.com/download/files/ifm-processes-in-the-automotive-industry-gb/\protect\T1\textdollarfile/ifm-processes-in-the-automotive-industry-gb.pdf> (accessed on 9 November 2021).