

POLITECNICO DI TORINO

Master's degree course in Mechatronic Engineering

Master's Degree Thesis

**Simulation and advanced control of a
professional manipulator**

The educational Robot e.DO



Supervisor
Prof. Marina Indri

Candidate
Stefano PESCE

Supervisor
at Comau S.p.A.
Ing. Ivan Lazzero

APRIL 2018

This work is subject to the Creative Commons Licence

Alle Nonne

*"Sometimes it is the people who no one
imagines anything of, who do the things no
one...can imagine"*

[A. TURING]

Abstract

The project of thesis is developed on the educational robot e.DO in COMAU S.p.A.. COnsorzio MAcchine Utensili was founded in 1973 to bring together companies in the Turin area. The company is part of the FCA Group, it develops and implements automation processes, solutions and production services and is specialized in welding robots.

The manipulator e.DO is composed by six axes. The main aim of the project is to create a model that reflects the real behavior of the robot and to control it improving the performances of the actual situation. The Mathworks software allowed to create the mathematical model in Simulink and simulate the system.

The manipulator includes two types of motor units, one for the arm joints and the other for the wrist joints. Both are DC brushed motors. The approach used was to analyze the DC motor in detail creating a model including the values of the datasheets. Considering an approximate evaluation of the friction coefficient and the inertia due to the motor and the planetary gearbox, the simulation on Simulink of both joints approximated the reality. Then, considering all reduction jumps, the whole structure model was created. Thanks to the Comau library, the dynamic model provided information on the variation of the inertia due to the position of the kinematic chain in space. By adding these values, it was possible to replicate the behavior of the whole robot chain. The software including the original control system was analyzed in order to create an exact copy to control the model of the motors. The original controllers were added on Simulink with the respective saturators, one for each loop: position, velocity and torque. To demonstrate the efficiency of the model, a simulation has been carried out in which the real behavior of the robot is compared with the model's response. The simulated and real signals were very close ensuring that the model satisfactorily reflected reality.

Subsequently, the control is improved to achieve better performance. The structure of the control system has been modified, only two loops have been retained: the position and the velocity ones. Two nested loops are designed in order to made the response of the system close to a prototype of the second order. The canonical form is compared with the transfer function of the system and the parameters were chosen to make the system more prompt keeping the synchronicity and the stability of the entire manipulator. Setting the natural frequency and the damping factor of the transfer function it was possible to improve the transient requirements. The rise time was decreased to make prompt the system, the overshoot was deleted and the settling time was minimized.

Considering the trade-off between the possibility to follow exactly a desired position in Cartesian space and the machine constraints due to the mechanical characteristics of the manipulator, the best controller found in this thesis was modified in order to implement it on the robot in the final step. Finally, the designed control was tested on the robot with satisfactory answers, the transitory response was improved with good results in term of rise time and settling time, on the other hand the steady state response was upgraded without overshoot and tracking error.

The thesis is divided into two large sections, the Background that recalls all the basic knowledges that are used in the project, the part of Project Development that explains in detail how the system has improved and how the improvement has been achieved.

Chapter 1 introduces industrial and educational robotics with a section dedicated to the protagonist of the project e.DO.

From the second chapter starts the Background part, in which the main concepts of the DC motors, the robotics and the manipulator control are explained exhaustively in order to make understandable the following section. Chapters 5-7 in the Project Development part include the design of the control system and the results achieved in this thesis.

Finally Chapter 8 and 9 present the future developments that could improve further e.DO and the conclusions of the thesis.

Contents

List of Tables	5
List of Figures	6
I Introduction	9
1 From industrial to educational robotics	11
1.1 e.DO	12
II Background	13
2 DC Motor	15
2.1 General Principles	15
2.2 Constitutive Relations	17
2.3 Torque Characteristic	17
2.4 Rated Quantities	19
2.5 Machine Dimensions	19
2.6 Commutation	20
2.7 DC Machine Dynamic Model	21
2.7.1 Friction	23
2.7.2 Complete Model in Laplace domain	23
3 Robotics	27
3.1 Kinematics	27
3.1.1 Joints	27
3.1.2 Working space	28
3.1.3 Anthropomorphic robot	29
3.1.4 Kinematic Functions	29
3.1.5 Denavit-Hartenberg Conventions	30
3.1.6 Jacobian Matrices	32
3.1.7 Singularity	33
3.2 Dynamics	34
3.2.1 Lagrangian equations	34
3.2.2 Kinetic Co-energy	34

3.2.3	Potential Energy	35
3.2.4	Virtual Work	35
3.2.5	Generalized Forces	35
3.2.6	Physical interpretation	36
4	Manipulator Control	39
4.1	Decentralized control	40
4.1.1	Cascade Control	42
4.2	Calculated Torque Feedforward Control	44
4.3	Centralized control	45
III	Project Development	49
5	Dynamic Model of the e.DO robot	53
5.1	e.DO features	53
5.2	From Paper to Model	55
5.2.1	Planetary gear	56
5.3	Simulation of model with fixed Inertia	60
5.4	Kinematics and Dynamics of e.DO	61
5.4.1	Arm kinematics	62
5.4.2	Wrist kinematics	63
5.5	Interface to analyze ORL data	66
5.6	Control structure	68
5.6.1	PID controllers	69
5.7	Simulation of the final model	70
6	Control of the e.DO robot	75
6.1	Practical considerations	76
6.2	Anti wind-up	76
6.3	Preliminary analysis	77
6.4	Design	81
7	Results	87
7.1	Performance achieved	87
7.2	Final comparison	92
7.2.1	e.DO writer	93
7.3	Noises analysis	95
8	Future developments	99
8.1	Observer	99
8.2	Friction compensation	99
8.2.1	Stick-Slip	100
9	Conclusions	103
	References	105

List of Tables

5.1	Dc Motor Datasheets of DCX16S GB KL 12V and DCX22L GB KL 12V	54
5.2	Planetary gear Datasheets of GPX16HP 186:1 and GPX26 A 231:1.	55
5.3	Dc Motor Datasheets VS Simulated velocity at 12V	61
5.4	Denavit-Hartenberg parameters of the Arm	63
5.5	Denavit-Hartenberg parameters of the Wrist	64
5.6	Denavit-Hartenberg parameters of the whole manipulator	65
5.7	Maximum values of Inertia matrix	67
5.8	PID values	71
6.1	Step response analysis of the original control system	78
6.2	Nichols margins of the velocity and position loops of both motors.	85
7.1	Step response analysis of the robot with original and the new control	90
7.2	Step response analysis	93
7.3	Time machine on the joints.	93

List of Figures

1.1	e.DO	11
1.2	Overall system and Working Space of e.DO	12
2.1	Internal didactical view of a DC Motor. [8]	15
2.2	Typical Brushed Motor in Cross. [17]	16
2.3	Commutator's effect on the signal. [14]	17
2.4	The output of a one coil armature as opposed to a three coil armature. [18]	17
2.5	The electric equivalent circuit of the armature and the free-body diagram of the DC motor [4]	18
2.6	Linear $T - \omega$ relationship [7]	18
2.7	DC motor diagram with neutral axis[19]	20
2.8	(a) Current time graph of one coil undergoing commutation. [6] , (b) Coils under commutation. [10]	21
2.9	Electrical equivalent circuit of DC motor	21
2.10	Dc Motor Block Diagram	24
3.1	Prismatic and Revolute joint. [16]	28
3.2	Direct and inverse kinematic function. [5]	28
3.3	Anthropomorphic Arm	29
3.4	Eulerian and Roll Pitch Yaw wrists [5]	29
3.5	Denavit Hartenberg convention [11]	30
4.1	General scheme of Joint Space Control [23]	40
4.2	DC motor diagram in open loop	41
4.3	Simplified DC motor diagram in open loop with $L_a = 0$	41
4.4	Diagram with velocity-loop and position-loop	42
4.5	Block scheme of computed torque feedforward control [23]	44
4.6	Global linearization with Inverse Dynamics Control [23]	46
4.7	Inverse Dynamics with P-D actions [23]	46
4.8	Validation and Verification diagram	51
5.1	e.DO six action motion units	53
5.2	The standard flux of communication.	54
5.3	Bode Diagram of the Arm Motor.	56
5.4	Gearmotor diagram of the joint.	57
5.5	Mechanical diagram of the overall joint.	59
5.6	DC motor diagram in open loop.	60
5.7	Velocity in $\left[\frac{rad}{s} \right]$ of Arm and Wrist Motor fed with nominal 12 V voltage. .	61
5.8	Symbolic representation of the e.DO manipulator.	62

5.9	Symbolic representation of the arm of e.DO.	63
5.10	Symbolic representation of the wrist of e.DO.	64
5.11	Communication flow diagram.	67
5.12	Diagonal term of the tensor matrix	68
5.13	Control scheme.	69
5.14	PID control action based on past, present and future [3].	70
5.15	Model Block diagram	72
5.16	In Blue the target, in Green the real, in Red the simulated position of the six motors.	73
5.17	Current varying the friction values	74
6.1	Control diagram with saturation	77
6.2	Time requirements: Overshoot, Rise Time and Settling Time of a prototype of second order function	79
6.3	Step response of prototype 2 nd order system	80
6.4	Control diagram without feedforward	81
6.5	Stability margins in the Nichols Chart	83
6.6	Nichols diagrams of the velocity and position loops of both motors.	84
6.7	Step response and Nichols diagrams of the motors with $\omega_n = 50$ rad/s and $\zeta = 1$	86
7.1	In blue the desired position, in green the position of the previous controller, in red the position of the new controller with $\omega_n = 50$ rad/s and $\zeta = 1$	87
7.2	Manipulator desired trajectories in blue, the real ones in red and the average of the noise of the velocity in yellow for the controller with $\omega_n = 50$ rad/s and $\zeta = 1$	89
7.3	Manipulator data with the controller with $\omega_n = 50$ rad/s, $\omega_n = 20$ rad/s, $\omega_n = 10$ rad/s, $\omega_n = 5$ rad/s, $\omega_n = 2.5$ rad/s and $\zeta = 1$ and the simulation of real system.	91
7.4	Light blue and Green are the Ideal and Real position with the previous control, Red and Blue are the Ideal and Real position with the adopted control, Magenta and Black are the Ideal and Real position with the best control.	92
7.5	e.DO writer.	94
7.6	Test created with the writing support package.	94
7.7	In black the rectangle with the original control and in green the rectangle with the last control system ($\omega_n = 7.5$ rad/S, $\zeta = 1$).	95
7.8	Constant Real Velocity under examination.	96
7.9	Constant Real Velocity and its Matlab derivative under examination.	96
7.10	Real velocity acquisition at $T_s = 1ms$ through the Debug development en- vironment and its Spectrum computed through the Fourier Transform.	98
8.1	Stribeck 's curve.	100
8.2	Stick-Slip effect.	100
9.1	Future of e.DO.	103

Part I

Introduction

Chapter 1

From industrial to educational robotics

The robotics is a branch of the mechatronic engineering. The appearance of the first industrial robots takes place around the seventies. The degrees of freedom and load capacities of these first robots are limited, the first industrial applications are in the automotive field, in fact the robots are equipped with welding pliers to weld and assemble the car bodies. Currently the anthropomorphic robots in the automotive field are used for welding laser and laser systems with one or more sources, this technology reduces time and space and increases the quality of the bodywork. Industrial robots are also heavily used in the industry, especially in difficult working areas, thus safeguarding workers' health. In Italy there is currently one of the largest companies in the world of industrial robotics, Comau. The importance of the Robotics is increasing in several areas, it becomes fundamental to teach children how to approach with the robotics to get an idea of what a robot is and how it can be used. In this context, with more than forty years of expertise, Comau can contribute with a revolutionary idea: to create an anthropomorphic robot in order to bring the students closer to robotics, this robot is called e.DO.



Figure 1.1: e.DO

1.1 e.DO

e.DO is a manipulator created by COMAU S.p.A. (Figure 1.1). The robot is composed by six joints, the first three for the arm and the last three for the wrist. Externally, e.DO can be enhanced with user-developed accessories and configurations. Internally, e.DO is ROS-native, open-source control logic gives users complete freedom to create and execute their own programs and applications.

e.DO can be intended as a conceptual gym for the robotics. Robotics widens its boundaries and opens up to new applications and new user communities. The user, in turn, will be able to intervene on the structure and applications of e.DO, as real makers. The perspective is that of continuous experimentation, training and fun. The possibility to add some applications on the sixth axis as a gripper gives to e.DO the prospect to do everything from executing simple pick and place movements to handling automated activities. The maximum payload is 1kg and it can apply a torque of $4N/m$. In Figure 1.2 can be seen the working space of e.DO with its dimensions:

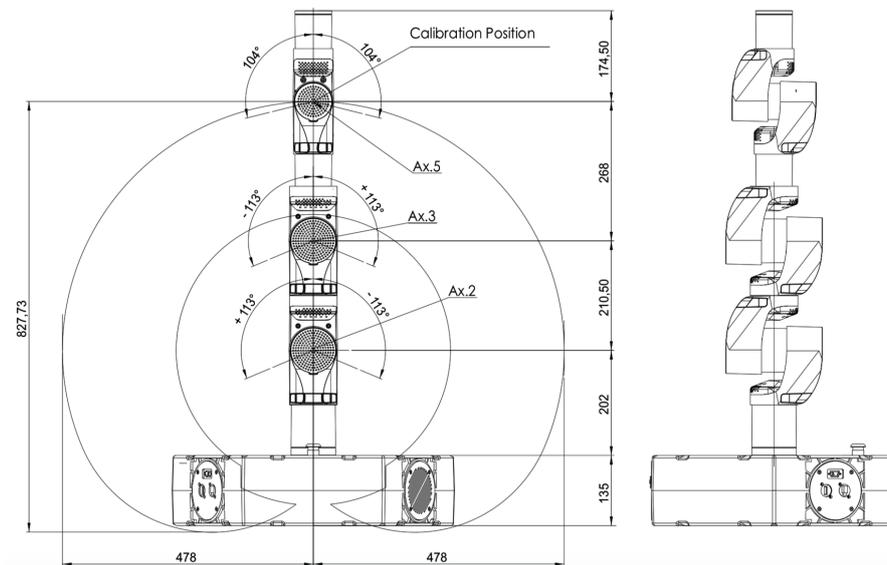


Figure 1.2: Overall system and Working Space of e.DO

Part II

Background

Chapter 2

DC Motor

2.1 General Principles

The DC Motor is one of the protagonists of this project. To carry out the features of this component it is necessary to introduce some recalls about the electrical machines. In an electrical machine where the energy is transformed and where at least one side between Input and Output energy is an Electrical Energy.

In the DC Machine the stator windings are fed with a DC Excitation current to generate a constant Excitation Flux.

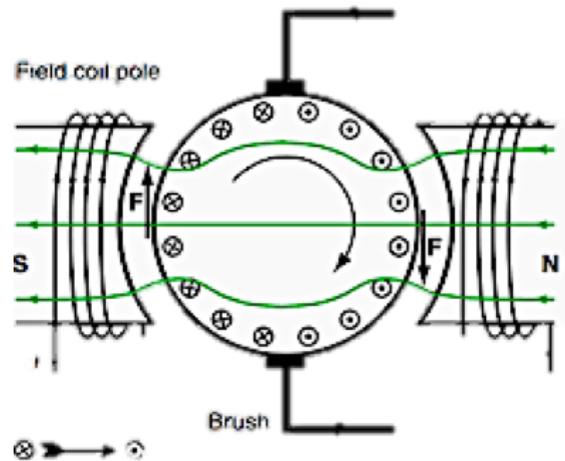


Figure 2.1: Internal didactical view of a DC Motor. [8]

As it is possible to see in Figure 2.1 the shape of the stator allows to obtain the smallest possible reluctance for the excitation flux. When the angle with respect to the neutral axis is null, the flux linked to the rotor windings will be the maximum one and equal to $\lambda = N\Phi_{exc}$, where N is the number of coils and Φ_{exc} is the excitation flux. On the contrary when the angle is 90° the coil surface is parallel to the Φ_{exc} and no flux is linking the coil,

i.e., $\lambda = 0$.

Co-sinusoidal Behaviour:

$$\lambda = N\Phi_{exc} \cos \alpha = N\Phi_{exc} \cos \omega t \quad (2.1)$$

where ω is the Rotor mechanical angular speed.[20]

From the Lenz Law we know that the electromotive force is the derivative of the linked flux, therefore:

$$e = -\frac{\partial \lambda}{\partial t} = N\Phi_{exc}\omega \sin \omega t \quad (2.2)$$

The DC voltage can be obtained directly only from a chemical transformation: the battery. The main problem is that, in this way, it is produced an electromechanical force in a rotating system, this needs a mechanical interface to use a static load. The sliding contacts have this task: two rings are connected to the two terminals of the rotor windings and rotate with the rotor; two brushes are fixed and slide against the rings collecting the current. The load is connected to the brushes terminals. Also with this interface we still get a sinusoidal voltage between the brushes, we need a commutator: we use just one ring but split in two halves (Figure 2.2).

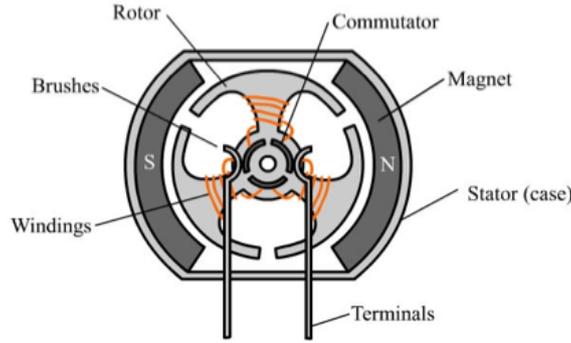


Figure 2.2: Typical Brushed Motor in Cross. [17]

$F = qv \times B$ is the force acting on each electron due to the rotation of the coil inside a flux. This is the force that moves the electrons producing current: thanks to the commutator, the left-hand brush will always be connected to the side of the ring carrying the entering electromotive force, even though the overall shape will still be sinusoidal. In this way, we obtain a mechanical diode with a non-null dc component.[12] (Figure 2.3

In order to fill the rotor with more coils, another interface is required: the segmented commutator. The ring is now made by a series of insulated sections, the electromotive force is entering on the left side and going out on the right side. Each coil carries a slightly more shifted sinusoidal waveform. in Figure 2.4, it is possible to see the difference between one coil armature and three coils armature electromotive force.

Thanks to the commutator, we obtain a very high dc component almost equal to the amplitude of a single winding and a small super imposed ripple. The number of brushes is equal to the number of magnetic poles. [24]

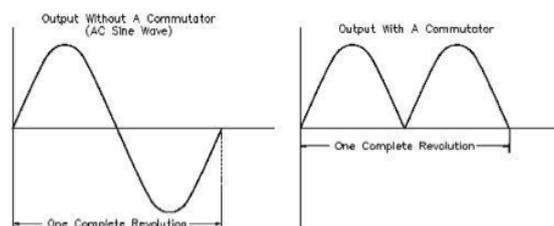


Figure 2.3: Commutator’s effect on the signal. [14]

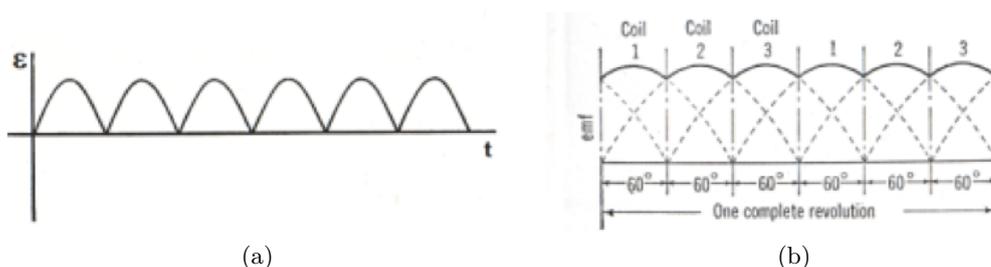


Figure 2.4: The output of a one coil armature as opposed to a three coil armature. [18]

2.2 Constitutive Relations

To evaluate the constitutive relations of the DC Motor it is necessary to highlight the electrical equivalent circuit:(Figure 2.5)

At steady state the behavior of every DC Machine is governed by the followings relations:

1. $E = k\Phi_{exc}\omega$
2. $V = E + I_a R_a$
3. $T = k\Phi_{exc}I_a$

Where E is the electromotive force, V is the output voltage and T is the generated torque. A fourth equation can be taken into account, given by the is the power balance between input and output.

4. $P_{in} = I_a V = I_a E + R_a I_a^2$

where $I_a E$ is the Electrical Power to be transformed into Mechanical Power and $R_a I_a^2$ is the Rotor Joule Losses.

From these equations, it can be noted that at constant Φ_{exc} the torque is proportional to the armature current and ω depends on the electromotive force E and not on the voltage V , even though the only way to control E is by modifying V [13].

2.3 Torque Characteristic

Considering the separate excitation machine, to find the relationship between the torque and the rotational speed (Figure 2.6)starting from the constitutive relations, it can be write

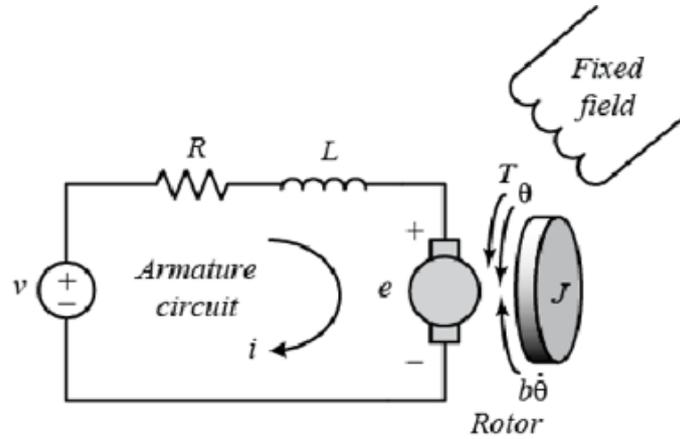


Figure 2.5: The electric equivalent circuit of the armature and the free-body diagram of the DC motor [4]

the following equation:

$$I_a = \frac{V - E}{R_a} \rightarrow T = k\Phi_{exc} \frac{V - E}{R_a} = k \frac{\Phi_{exc}}{R_a} V - k^2 \frac{\Phi_{exc}^2}{R_a} \omega \quad (2.3)$$

Just two points are needed to completely describe the linear characteristic:

- $\omega = 0$, $T = k \frac{\Phi_{exc}}{R_a} V = T_s$
- $T = 0$, $\omega^* = \frac{V}{k\Phi_{exc}}$

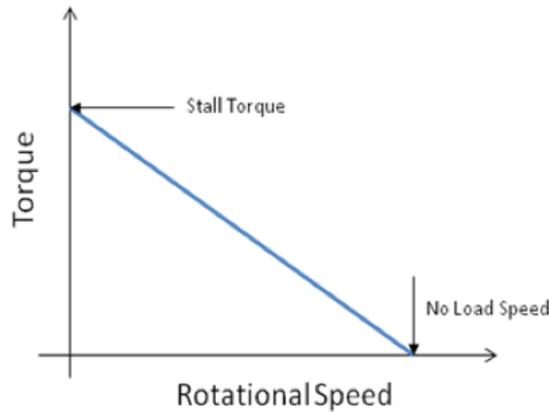


Figure 2.6: Linear $T - \omega$ relationship [7]

It is possible to highlight that the Null Starting Torque T_s is for $T = 0$; in this condition, the motor can't start by itself, it needs to be put in rotation normally and w^* is the mechanical speed associated to a null torque.

2.4 Rated Quantities

- Rated Voltage V_N : the highest voltage applicable to the machine without causing damages.
- Rated Current I_N : the highest current that the machine can absorb.
- Rated Speed n_N : rotational speed expressed in round per minute.
From the above rotational speed, we can compute two other quantities:

- Rated Torque T_N : from the mechanical power $P_{mech} = T\omega \rightarrow T_N = \frac{P_N}{\frac{2\pi n_N}{60}}$

- Efficiency η : $\eta = \frac{P_{out}}{P_{in}} = \frac{P_N}{V_N I_N} = \frac{T_N (\frac{2\pi n_N}{60})}{V_N I_N}$

In the last formulation for the separate excitation machines the Excitation Efficiency should be considered: $\rightarrow \eta = \frac{P_{out}}{P_{in}} = \frac{P_N}{V_N I_N + P_{exc}}$ "Electromagnetic Efficiency"

- Rated Power P_N : the highest power that the machine can provide, it is always an output power.

The Rated Power linked to the thermal problems can be highlighted considering the losses: $\sum Losses = P_{in} - P_{out}$

They increase the temperature; the main risk concerns the insulation system among the windings and the laminations that is very sensitive to the temperature. In this sense, the Rated Power is the maximum power that the machine can produce while remaining at a temperature lower than the maximum temperature that the insulation system can withstand @ $T < T_{insMAX}$

The international standards define the Classes of insulation materials based on the temperatures they can bear from "A" to "H".

2.5 Machine Dimensions

The machine dimensions do not depend only on the rated power. Supposing that the efficiency value is equal to one $\eta = 1$ then the rated power $P_N = E_N I_N = k\Phi_{exc}\omega I_N = T_N \omega_N$, but the excitation flux is equal to the product between the excitation flux density and the iron section $\Phi_{exc} = B_{excN} S_{Fe}$ and the rated current is equal to the product between the current density and the copper section $I_N = J_N S_{Cu}$. Supposing B_{excN} and J_N constants, $P_N = K B_{excN} S_{Fe} \omega_N J_N S_{Cu} = T_N \omega_N$ then :

$$T_N = k_{tot} S_{Fe} S_{Cu} \tag{2.4}$$

The rated torque is proportional to the machine extension, this means that the machine dimension is strongly related to its working torque. Moreover assuming ω_N , B_N , J_N constants, $P_N = k_{tot} S_{Fe} S_{Cu}$ is proportional to the width of the motor powered of four $\rightarrow \alpha l^4$.

Whereas from the point of view of losses:

JOULE: $P_J = R_a I_a^2 = \rho_{Cu} \frac{l_{Cu}}{S_{Cu}} J^2 S_{Cu}^2 = \rho_{Cu} J^2 l_{Cu} S_{Cu}$ is proportional to the cube of the width $\rightarrow P_J \propto l^3$

IRON: $P_{Fe} = \text{"Specific Iron Losses" } [\frac{W}{Kg}]$ is proportional to the volume of the Iron $\rightarrow P_{Fe} \propto l^3$

This means that increasing the dimensions the losses have a smaller increase with respect to the rated power, then the efficiency increases with size. The drawback is that with a bigger machine the rated power increase but we have more power dissipated since the thermal exchange is proportional to the cube of the exposed surface, a better cooling system is required.[25]

2.6 Commutation

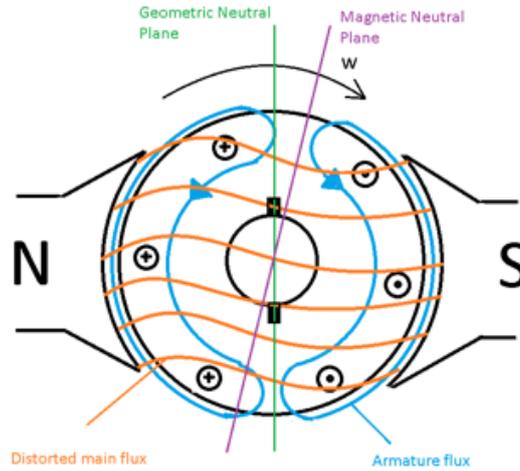


Figure 2.7: DC motor diagram with neutral axis[19]

During the rotation, the current amplitude changes sign when the coil rotates across the neutral axis; Figure 2.8 shows how the current exactly works in the transition. During the commutation, the current flowing in the brush will come from both segments, the amount of current for each segment will be proportional to the contact surface extension between segment and brush. That is because:

$$R_{brush_a} = \rho \frac{l}{S_a} \quad R_{brush_b} = \rho \frac{l}{S_a} \quad (2.5)$$

“The larger the contact surface, the lower the resistance, the higher the current”

From the point of view of the segment current, the ideal linear behaviour will go to zero value, it means that after the commutation no current will flow through the segment without discontinuities. The main problems of this configuration are two:

1. Often the Neutral axis is not exactly vertical but slightly shifted

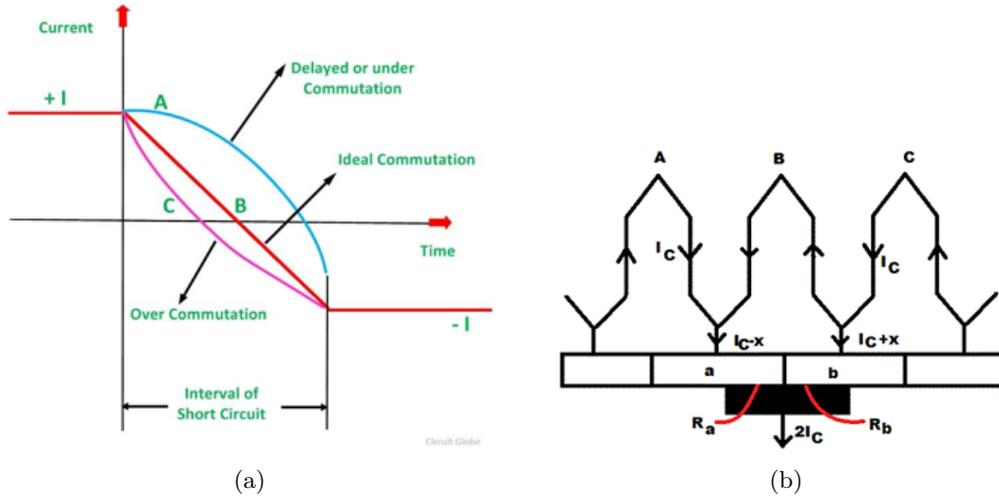


Figure 2.8: (a) Current time graph of one coil undergoing commutation. [6] , (b) Coils under commutation. [10]

2. There is an additional electromotive force contribution due to the inductance of the coils: $E = -L \frac{di}{dt}$

These two problems make the coil under commutation have $E \neq 0$.

The inductive electromotive force causes a current in opposition to the current variation, thus causing a commutation delay for the current and a nonlinear commutation. When the brush closes the contact with the segment there will still be a current $I^* \neq 0$. The sharp variation imposed by the sudden open circuit will cause a big electromotive force. There will be a spark between segment and brush that damage the two components, particularly the segment that is very expensive to replace. The solution to this problem is to add more poles to compensate the electromotive force excess during the commutation.[24]

2.7 DC Machine Dynamic Model

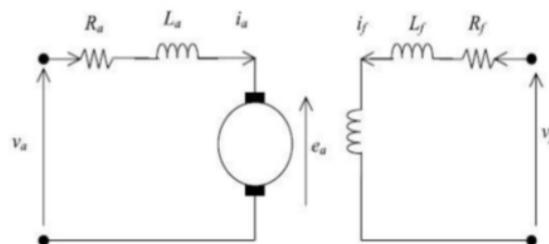


Figure 2.9: Electrical equivalent circuit of DC motor

The mathematical model that describe the dynamic system take under examination the object with a finite number of degrees of freedom that evolves with respect to a deterministic law. With a mathematical and electrical analysis on the equivalent circuit, through the Kirchhoff's law the Armature Equation can be found as:

$$v_a(t) = R_a i_a(t) + L_a \frac{di_a}{dt} + e(t) \quad (2.6)$$

where $e(t)$ is the back electromotive force with a positive sign convention. Another important equation is due to an electromechanical observation that involves these terms:

- $k \rightarrow$ related to the number of conductors and magnetic poles
- $\Phi \rightarrow$ related to the flux per pole
- $K_\omega = k\Phi \rightarrow$ related to the "Back electromotive force constant"

At constant excitation flux Φ_{exc} , the Electromechanical Equation of the motor is:

$$e(t) = k\Phi(t)\omega(t) = K_\omega\omega(t) \quad (2.7)$$

and the Torque-Current Relationship is:

$$T_m(t) = k\Phi(t)i_a(t) = k_\tau i_a(t) \quad (2.8)$$

It can be noted that there are two different constants K_ω and k_τ because the two equations are computed in two different operating condition:

- K_ω at "No Load Condition"
- k_τ at "Short Circuit/Locked Rotor Condition"

The difference is mainly due to the presence or absence of the armature current that produces an additional flux contribution. Even if the average flux produced by the armature is null, it has the negative effect of increasing the density flux B_{exc} in certain regions, this means that in some cases the density flux can enter in the saturation zone. For big machines the effect of the armature reaction are compensated for example by introducing additional stator coils, instead for small machine the effect can be neglected and the difference between K_ω and k_τ is about (5 ÷ 10%).

The dynamic model is completed by the mechanical equation that involves the main terms of the motion:

- $T_m \rightarrow$ Motion Torque
- $J \rightarrow$ Inertia
- $J\dot{\omega} \rightarrow$ Inertia Torque
- $B\omega \rightarrow$ Viscous Friction
- $T_r \rightarrow$ Resistive Torque, independent from ω

The Mechanical Equation is:

$$T_m(t) = J \frac{d\omega}{dt} + B\omega(t) + T_r(t) \quad (2.9)$$

2.7.1 Friction

In order to obtain a linear model, it is convenient to take into account only the time-invariant coefficient parameters. The main types of the frictions are:

- Viscous Friction $\rightarrow B\omega(t)$
- Ventilation Friction $\rightarrow C\omega^2(t)$
- Start up Friction $\rightarrow T_{FS} \neq 0$ only for $\omega = 0$
- Coulomb Friction \rightarrow is dependent from ω

The first and the last frictions are included in the model, the last is considered thanks to the resistive torque, the other two are non-linear frictions and are neglected.

2.7.2 Complete Model in Laplace domain

Under the hypothesis of null initial conditions, translating the previous equations in the Laplace domain:

$$\begin{cases} V_a(s) = R_A I_A(s) + sL_A I_A(s) + E(s) \\ E(s) = K_\omega \omega(s) \\ T_m(s) = k_\tau I_A(s) \\ T_m(s) = sJ\omega(s) + B\omega(s) + T_r(s) \end{cases} \quad (2.10)$$

The inputs and the outputs are respectively V_A , T_r and ω , T_m . Since the current is:

$$I_A(s) = \frac{V_A(s) - E(s)}{R_A + sL_A} \quad (2.11)$$

Then:

$$T_m(s) = k_\tau \frac{V_A(s) - E(s)}{R_A + sL_A} \quad (2.12)$$

It is possible to evaluate some intermediate equations that allow to find the final relationship between the armature voltage and the angular speed.

Torque/Voltage Transfer Function:

$$\frac{T_m(s)}{V_A(s) - E(s)} = \frac{k_\tau}{R_A + sL_A} \quad (2.13)$$

Speed/Torque Transfer Function:

$$\frac{\omega(s)}{T_m(s) - T_r(s)} = \frac{1}{\beta + sJ} \quad (2.14)$$

Electromotive Force/Speed Transfer Function:

$$\frac{E(s)}{\omega(s)} = K_\omega \quad (2.15)$$

From the above equations, it is possible to exploit the block diagram (Figure 2.10) that involves as input the armature voltage and as output the rotational speed.[1, 15, 21, 22]

To find the canonical form it can be used the superposition theorem, since the non linear characteristics are neglected and the system is perfectly linear:

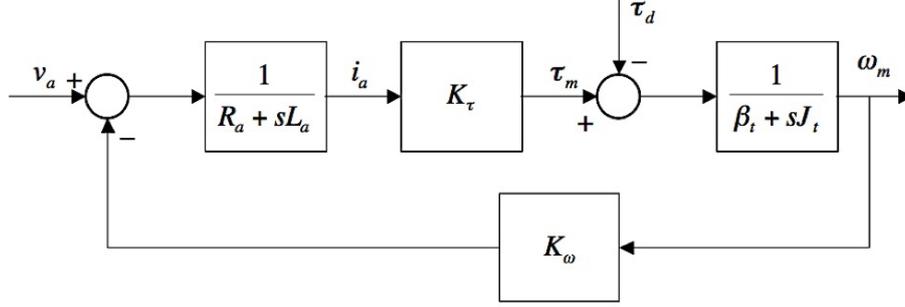


Figure 2.10: Dc Motor Block Diagram

- If the resistance torque is null $T_d = T_r = 0$:

$$\begin{aligned} \frac{\omega(s)}{V_A(s)} &= \frac{G(s)}{1 + G(s)H(s)} = \frac{\frac{k_\tau}{(R_A + sL_A)(\beta + sJ)}}{1 + \frac{k_\tau K_\omega}{(R_A + sL_A)(\beta + sJ)}} \\ &= \frac{k_\tau}{s^2 J L_A + s(J R_A + L_A \beta) + (k_\tau K_\omega + R_A \beta)} \end{aligned} \quad (2.16)$$

The canonical form is:

$$\frac{\omega(s)}{V_A(s)} = \frac{\frac{k_\tau}{J L_A}}{s^2 + s \frac{J R_A + L_A \beta}{J L_A} + \frac{k_\tau K_\omega + R_A \beta}{J L_A}} \quad (2.17)$$

Neglecting friction, two time constants can be highlighted: the Armature time constant $\tau_A = \frac{L_A}{R_A}$ and the Mechanical time constant $\tau_M = \frac{J R_A}{k_\tau K_\omega}$.

The canonical form changes in:

$$\frac{\omega(s)}{V_A(s)} = \frac{\frac{1}{k_\tau \tau_A \tau_M}}{s^2 + \frac{s}{\tau_A} + \frac{1}{\tau_A \tau_M}} = \frac{\frac{1}{K_\omega}}{s^2 \tau_A \tau_M + s \tau_M + 1} \quad (2.18)$$

The roots of the equation are: $s_{1/2} = (-\zeta \pm \sqrt{\zeta^2 - 1})\omega_0$ with $\begin{cases} \omega_0 = \sqrt{\frac{1}{\tau_A \tau_M}} \\ \zeta = \frac{1}{2} \sqrt{\frac{\tau_A}{\tau_M}} \end{cases}$ The system strictly depends on the damping factor in three different ways:

- $\tau_M > 4\tau_A, \zeta > 1 \rightarrow$ The system is Over Damped, there are two real poles
- $\tau_M = 4\tau_A, \zeta = 1 \rightarrow$ The system is Critically Damped, there are two real coincident poles
- $\tau_M < 4\tau_A, \zeta < 1 \rightarrow$ The system is Under Damped, there are two complex conjugate poles

The condition with the Under damped system for $\tau_M = 0$ and $\zeta = 0$ is called perpetual undamped oscillation, but $\tau_M = 0$ means that $R_A = 0$ and there are No Power Losses,

it is an oscillation with a constant amplitude in time domain. This configuration is available only with the superconductors where the armature resistance is very low and can be neglected. In general, the Big Machines have a low damping factor, as opposed to the cheap machines that have high damping factor and accordingly high dissipation.

- If the armature voltage is null $V_A = 0$, then:

$$\frac{\omega(s)}{T_r(s)} = \frac{\frac{1}{\beta+sJ}}{1 + \frac{k_\tau K_\omega}{(\beta+sJ)(JR_A+sL_A)}} = \frac{R_A + sL_A}{s^2 J L_A + s(JR_A + L_A\beta) + (k_\tau K_\omega)} \quad (2.19)$$

It can be seen the same denominator of the previous equation, considering the time constants the transfer function will become:

$$\frac{\omega(s)}{T_r(s)} = -\frac{1}{J} \frac{s + \frac{1}{\tau_A}}{s^2 + \frac{s}{\tau_A} + \frac{1}{\tau_A \tau_M}} \quad (2.20)$$

The negative gain in this formulation means that the higher is the Resistive Torque the lower is the Angular Speed.

Chapter 3

Robotics

“A robot may not injure a human being or, through inaction, allow a human being to come to harm.”

In 1941 Isaac Asimov wrote a short story *“Liar!”* in which the word “Robotics” came out for the first time. The previous citation refers to the main concept of the futuristic novel.[2] The idea was very innovative for the 1941 but it became a fundamental landmark for this century and the future.

3.1 Kinematics

Kinematics allows to feature positions, velocities and accelerations of the multi-body structure points without knowing what generated the motion.

A series of ideal Link/Arms connected by ideal Joints form a kinematic chain, a geometric entity without mass, friction and elasticity. The kinematic chain has a Degree Of Motion (DOM) and may afford a Degree Of Freedom (DOF). On each arm of the structure it is convenient to define a Reference Frame (RF).

3.1.1 Joints

The joints allow one degree of motion between two consecutive connected links, they may have two structures (Figure 3.1):

- Revolute Joint: it permits a rotation between the connected links.
- Prismatic Joint: it permits a translation between the connected links.

The end of the last link in the kinematic chain that can perform a required task is called End Effector. The centre in the end effector is the Tool Centre Point and it is the point that the robot software moves through the space. Depending on the structure of the chain there will be two possible kinds of chain: open if there is only one path between the bases and the tool centre point, closed if there are more possible paths.

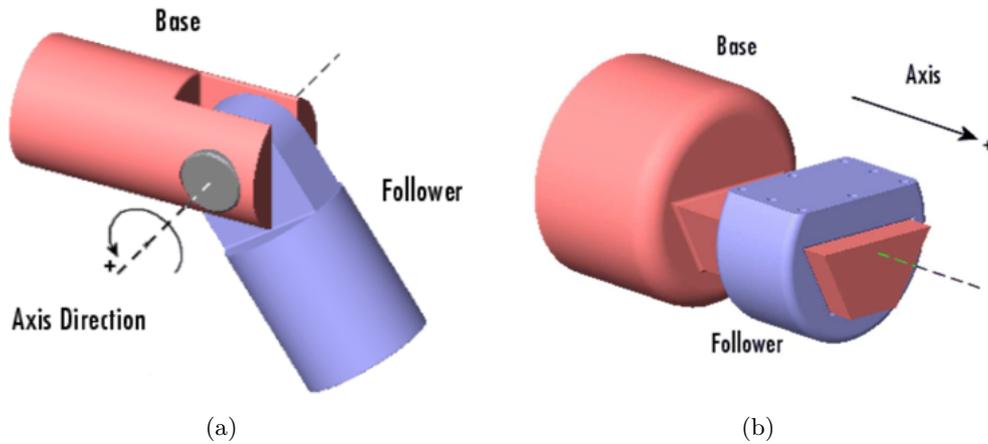


Figure 3.1: Prismatic and Revolute joint. [16]

3.1.2 Working space

The working space can be the Task space or the Joint space: the first is the set of points that the Tool Centre Point can reach in the space, the other is represented by the value of each joint variable q_i . (Figure 3.2)

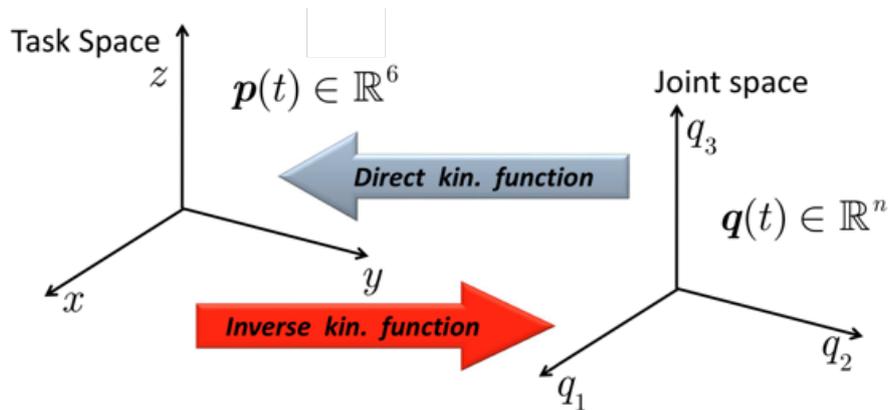


Figure 3.2: Direct and inverse kinematic function. [5]

Defining the robot DOM, the Tool Centre Point DOF and the task DOF m it is possible to characterize four different case studies:

1. $n = n' = m \rightarrow$ The robot has many TCP DOFs as required by the task: *NonRedundant Chain*.
2. $n = n' > m \rightarrow$ The robot has more TCP DOFs than those required by the task: *Redundant Chain*.
3. $n = n' < m \rightarrow$ The robot has less TCP DOFs than required by the task: *Useless Chain*.

4. $n > n' = n \rightarrow$ The robot has more joints than required.

In case 2 the redundancy improves the manipulability or dexterity, i.e., the ability to reach a desired pose avoiding obstacles.

3.1.3 Anthropomorphic robot

A manipulator is composed by the **Arm** and the **Wrist**, the coupling of these two elements create an anthropomorphic robot with six degrees of freedom.

Arm: The structure of arm is composed by three revolute joints disposed in a certain sequence, the first axis is vertical and orthogonal to the others that are parallels between them.(Figure 3.3) It is very similar to the human one: arm and forearm, on which the wrist is grafted. This sequence of joints is often used in the industrial environment because it allows dexterity, even though the accuracy depending on the work space.

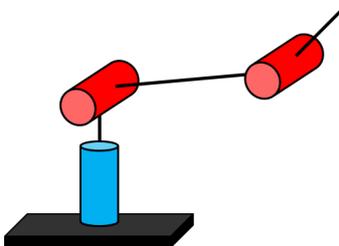


Figure 3.3: Anthropomorphic Arm

Wrist: The wrist in the kinematic chain has the purpose to orient the Tool Centre Point in the space: while the shoulder sets the coordinates, the Wrist orients it. A wrist is spherical when the three axes intersecting in a single point; when computing the inverse kinematics, the presence of a spherical wrist is a sufficient condition for the existence of a closed form solution. It is composed by three consecutive rotational joints and it can be Eulerian or Roll Pitch Yaw.(Figure 3.4



Figure 3.4: Eulerian and Roll Pitch Yaw wrists [5]

3.1.4 Kinematic Functions

There are four functions called kinematic functions that transform the Joint variables into Cartesian variables, these functions describe mathematically the relation between the joint space and the task space:

1. Direct Position: From joint space variables to task space pose.
2. Inverse Position: From task space pose to joint space variables.
3. Direct Velocity: From joint space velocities to task space velocities.
4. Inverse Velocity: From task space velocities to joint space velocities.

In general, the first step is to fix a Reference Frame on each Robot link. It is possible to apply the kinematic functions on any point of the kinematic chain. To move from a Reference Frame to the following one six parameters are needed: three transition values and three angles of rotation.

3.1.5 Denavit-Hartenberg Conventions

Jacques Denavit and Richard S. Hartenberg introduced in 1955 a convention that defines four Degrees of Freedom between two successive Reference Frames, instead of the usual six. Among these four parameters two are related to the translation and the others to the rotation, three of them depend only on the robot geometry and therefore are constant in time, the other depends on the relative motion between two successive links and it is time depending.

Assuming $q_i(t)$ the i -th joint variable, b_i the i -th link, g_i the joint and using the term motion axis for each prismatic or revolute joint:

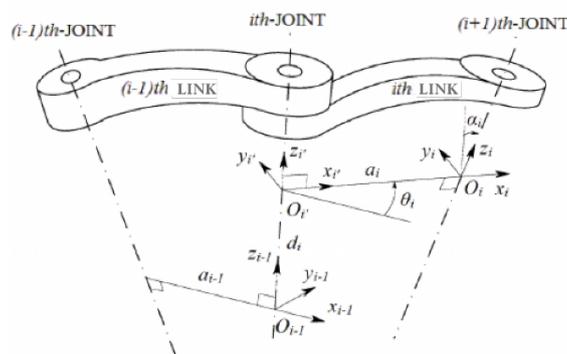


Figure 3.5: Denavit Hartenberg convention [11]

Rules:

1. The Reference frame of the i -th link is positioned on the axis of joint g_{i+1} , at the intersection between this axis and the common normal between this motion axis g_{i+1} and the previous axis g_i . If the two axes intersect, the origin must be in the intersection point. If the axes are parallel the origin can be located in any point depending on the personal choice, usually on the link.
2. Versor z_i is aligned with the motion axis g_{i+1} with the verse indicating positive motion.

3. Versor x_i is set orthogonal to the versor z_{i-1} ; since x_i must be also orthogonal to z_i it is normal to the plane defined by the two versor z_{i-1} and z_i . If those vectors are parallel, x_i lies on the plane normal to them with the direction and verse chosen by the user.
4. At the end, the versor y_i completes the Reference Frame with the right-hand rule.

Two undefined situations:

1. The first Reference Frame \mathbb{R}_0 of the base: the origin cannot univocally be defined since no previous point is present. To fix this ambiguity it is convenient to set only the direction of the versor z_i , and the others two are chosen by the user.
2. The final Reference Frame, in the project the sixth \mathbb{R}_6 : On the tool centre point a following motion axis does not exist, the origin is chosen by the user on the tool centre point or somewhere else and the versor i_6 is usually orthogonal to the z_5 .

Parameters:

- d_i : *is the translation along the motion axis z_{i-1} between the origin of \mathbb{R}_{i-1} and the intersection of the axis defined by z_{i-1} and the axis defined by i_i*
- θ_i : *it is the rotation angle around axis z_{i-1} such that i_{i-1} overlaps i_i*
- a_i : *it is the minimum distance between axis z_{i-1} and z_i along the common normal, measured along i_i*
- α_i : *it is the rotation angle around the motion axis g_i associated to i_i such that k_{i-1} overlaps k_i*

The Denavit-Hartenberg parameters define the transformation between \mathbb{R}_{i-1} and \mathbb{R}_i As said before, depending on the type of the joint we have some parameters fixed:

$$\text{Prismatic} \rightarrow \begin{cases} q_i(t) \equiv d_i(t) \\ \theta_i, a_i, \alpha_i \text{ are fixed} \end{cases} \quad \text{Revolute} \rightarrow \begin{cases} q_i(t) \equiv \theta_i(t) \\ d_i, a_i, \alpha_i \text{ are fixed} \end{cases} \quad (3.1)$$

It is possible to demonstrate that the Roto-Translation can be represented by a homogeneous matrix called Transformation matrix that contains both the rotational and translational terms.

DH homogeneous Roto-Translation matrix:

$$T_i^{i-1} = \begin{pmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} R & t \\ 0^T & 1 \end{pmatrix} \quad (3.2)$$

From the rotation matrix, it is possible to compute the angles: $R(q(t)) \rightarrow \alpha(q(t))$ where $\alpha(q(t))$ can be Euler, RPY, quaternions or the axis angle representation.

The inverse kinematic function is important, since the control actions are applied to the joint motors while the user usually wants to work with Cartesian position and orientation.

$$p(q(t)) = \begin{pmatrix} x(q(t)) \\ \dots \\ \alpha(q(t)) \end{pmatrix} \rightarrow \begin{pmatrix} q_1(t) \\ q_2(t) \\ q_3(t) \\ q_4(t) \\ q_5(t) \\ q_6(t) \end{pmatrix} \quad (3.3)$$

The spherical wrist allows to guarantee a solution, but the problem is complex and there are several possibilities to solve it. Using the linear and angular direct velocity kinematic function, assuming a non-redundant robot with six degree of freedom:

$$\dot{q}(t) = \begin{pmatrix} \dot{q}_1(t) \\ \dot{q}_2(t) \\ \dot{q}_3(t) \\ \dot{q}_4(t) \\ \dot{q}_5(t) \\ \dot{q}_6(t) \end{pmatrix} \rightarrow \dot{p}(t) = \begin{pmatrix} \dot{p}_1(t) \\ \dot{p}_2(t) \\ \dot{p}_3(t) \\ \dot{p}_4(t) \\ \dot{p}_5(t) \\ \dot{p}_6(t) \end{pmatrix} = \begin{pmatrix} \dot{x}(q(t), \dot{q}(t)) \\ \dots \\ \dot{\alpha}(q(t), \dot{q}(t)) \end{pmatrix} = \begin{pmatrix} v(q(t), \dot{q}(t)) \\ \dots \\ \omega(q(t), \dot{q}(t)) \end{pmatrix} \quad (3.4)$$

Where $\dot{q}(t)$ are the velocities in each joint's motor, $\dot{p}(t)$ are the velocities in Cartesian space, $v(q(t), \dot{q}(t))$ are the linear velocities and $\omega(q(t), \dot{q}(t))$ are the angular velocities.

3.1.6 Jacobian Matrices

It is necessary to distinguish two kinds of velocities in the operational space:

Geometrical velocity: it is considered the vector $v = [\dot{x}^T \omega^T]^T \in \mathbb{R}^6$ where $\dot{x} \in \mathbb{R}^3$ is the linear velocity and $\omega \in \mathbb{R}^3$ is the angular velocity of the tool center point.

Analytical velocity: it is the operational space vector $\dot{p} \in \mathbb{R}^6$ obtained deriving the vector $p = [x^T \alpha^T]^T$ of the operational coordinates with respect to time.

Only the linear part \dot{x} of both arrays does not change. The main difference between them is the second member of the vector, in the case of the geometrical ω is a true vector and it is possible to define the linear composition of its term, instead $\dot{\alpha}$ is not a true vector and is not valid the linear composition: $\dot{\alpha} \neq \dot{\alpha}_1 + \dots + \dot{\alpha}_n$. In general, to find the position from $\omega(t)$ the exact relation is:

$$\omega(t) = \dot{\theta}(t)u(t) + \sin \theta(t)\dot{u}(t) + (1 - \cos \theta(t))S(u(t))\dot{u}(t) \quad (3.5)$$

where $S(u(t))$ is the symmetrical matrix.

According to the two types of velocities there are two types of Jacobian: Geometrical J_g and Analytical J_a , which is also called task Jacobian.

- *Geometrical velocities:* $v_p = \begin{bmatrix} \dot{x} \\ \omega \end{bmatrix} = J_g \dot{q}$
- *Analytical velocities:* $\dot{p} = \begin{bmatrix} \dot{x} \\ \dot{\alpha} \end{bmatrix} = J_a \dot{q}$

Remembering that the relation $\dot{x} = v$ holds, but it is false that $\dot{\alpha} = \omega$ except on the planar motion it is possible to derive the fundamental formula of the robotics:

$$\dot{p}(t) = J(q(t))\dot{q}(t) \quad (3.6)$$

where J is the analytical Jacobian:

$$J_a(q(t)) = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \cdots & \frac{\partial f_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial q_1} & \cdots & \frac{\partial f_m}{\partial q_n} \end{bmatrix} \quad (3.7)$$

The geometrical Jacobian can be split the matrix in columns as:

$$J_g(q(t)) = \begin{bmatrix} J_{L,1} & J_{L,2} & \cdots & J_{L,n} \\ J_{A,1} & J_{A,2} & \cdots & J_{A,n} \end{bmatrix} \quad (3.8)$$

where $J_{L,i}$ describes how the velocity \dot{q}_i of the i -th joint contributes to the linear velocity of the tool center point, and $J_{A,i}$ how the velocity \dot{q}_i of the i -th joint contributes to the angular velocity. Both $J_{L,i}$ and $J_{A,i} \in \mathbb{R}^3$.

3.1.7 Singularity

“A condition caused by the collinear alignment of two or more robot axes appearing in unpredictable robot motion and velocities.”

The American National Standard for Industrial Robots and Robot Systems-Safety Requirements defined in this way the singularity in Robotics.

For an anthropomorphic robot, the configurations where the geometrical Jacobian has not a full rank are said kinematic singularity. A square matrix is invertible if the determinant is different from zero, in our case if the determinant of the Jacobian matrix is not null: $\det J(q(t)) \neq 0$. When it does not happen, a singularity exists at $q_s(t)$. When the joint coordinates approach a singularity, the joint velocities become very large for small Cartesian velocities. Mathematically speaking:

$$\dot{q} = J^{-1}q(t)\dot{p} = \frac{1}{\det J} \bar{J}\dot{p} \rightarrow \frac{1}{\varepsilon} \bar{J}\dot{p} \rightarrow \infty \quad (3.9)$$

When the joints try to follow a finite Cartesian velocity near the singularity configuration, the velocity becomes excessively large and it is not possible to follow a geometric path. Given a velocity profile, near the singularity conditions it is possible to reduce the Cartesian velocity and follow the geometric path, otherwise to follow the velocity profile but following an approximated path. In the exact singularity situation nothing can be done, the best way to not have problems is to avoid the singularity. There are three types of singularities:

- Completely extended or folded arm: the velocities transit from span a subspace of dimensions Two, the plane, to One dimension space, the tangent line.
- Wrist centre on the vertical: the final velocities cannot be obtained with infinitesimal joint rotation.
- Wrist singularity: there are two axes aligned.

3.2 Dynamics

The relations of the forces and torques between the task space and the joint space when the robot moves are studied by the Dynamics. To determine the dynamic model equation two main approaches can be applied: Lagrange equations based on the energy function and Newton-Euler equations based on the equilibrium of the vector generalized forces. In this project only the Lagrangian approach is treated.

3.2.1 Lagrangian equations

The difference between the total kinetic co-energy and the total potential energy on the manipulator is called Lagrangian function, where the kinetic co-energy and the potential energy are defined as follows:

$$C^*(q(t), \dot{q}(t)) = \sum_{i=1}^n C_i^*(q(t), \dot{q}(t)) \quad P(q(t)) = \sum_{i=1}^n P_i(q(t)) \quad (3.10)$$

So, the Lagrangian function is a state one defined as:

$$\mathcal{L}(q(t), \dot{q}(t)) = C^* - P = \sum_{i=1}^n C_i^*(q(t), \dot{q}(t)) - \sum_{i=1}^n P_i(q(t)) \quad (3.11)$$

The system dynamics can be described by scalar differential equations (the so-called Lagrangian equations):

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \frac{\partial \mathcal{L}}{\partial q_i} + \frac{\partial \mathcal{D}}{\partial \dot{q}_i} = \mathcal{F}_i \quad (3.12)$$

where \mathcal{F}_i is the i -th generalized force, and \mathcal{D} is the dissipative energy defined as follows:

$$\mathcal{D}(\dot{q}) = \sum_{i=1}^n \mathcal{D}_i(\dot{q}) = \frac{1}{2} \sum_{i=1}^n \beta_i \dot{q}_i^2 \quad (3.13)$$

3.2.2 Kinetic Co-energy

The kinetic co-energy is the sum between two contributions, providing the value of the i -th arm kinetic co-energy:

$$C^*(q(t), \dot{q}(t)) = \frac{1}{2} v_{c_i}^T((q, \dot{q})) m_i v_{c_i}((q, \dot{q})) + \frac{1}{2} \omega_i^T((q, \dot{q})) \Gamma_i \omega_{c_i}((q, \dot{q})) \quad (3.14)$$

The first contribution is due to translation and the second to rotation. Using the Jacobian matrices to express the kinetic co-energy as a function of the generalized coordinates q and velocities \dot{q} , it is possible to determine $C^*(q(t), \dot{q}(t))$ starting from the expression of the following jacobians:

$$J_L^i = [J_{L_1} \quad J_{L_2} \quad \dots \quad J_{L_i} \quad 0 \quad \dots \quad 0] \quad (3.15)$$

$$J_A^i = [J_{A_1} \quad J_{A_2} \quad \dots \quad J_{A_i} \quad 0 \quad \dots \quad 0] \quad (3.16)$$

that allow to write velocity as:

$$\dot{p}(t) = J(q(t))\dot{q}(t) \rightarrow \begin{cases} v_c^i = J_{L_1}\dot{q}_1 + \dots + J_{L_i}\dot{q}_i = J_L\dot{q} \\ \omega_i = J_{A_1}\dot{q}_1 + \dots + J_{A_i}\dot{q}_i = J_A\dot{q} \end{cases} \quad (3.17)$$

The kinetic co-energy then becomes:

$$\begin{aligned} C^*(q(t), \dot{q}(t)) &= \frac{1}{2} \sum_{i=1}^n \dot{q}_i^T [J_{L_i}^T m_i J_{L_i}] \dot{q}_i + \dot{q}_i^T [J_A^T \Gamma_i J_A] \dot{q}_i \\ &= \frac{1}{2} \dot{q}_i^T \left[\sum_{i=1}^n (J_{L_i}^T m_i J_{L_i}) + (J_A^T \Gamma_i J_A) \right] \dot{q}_i \\ &= \frac{1}{2} \dot{q}_i^T [H_i(q)] \frac{1}{2} \dot{q}_i = \frac{1}{2} \dot{q}^T [H(q)] \frac{1}{2} \dot{q} \end{aligned} \quad (3.18)$$

where $H(q)$ is the Inertia matrix; it is symmetric, positive definite and depends on the configuration of the robot.

3.2.3 Potential Energy

The potential energy inside the Lagrangian equation can be included with two possible approaches: the forces and the toques due to the gravitational effects are inserted in the equation as external generalized forces \mathcal{F}_i acting on the arms, otherwise the position energy due to the gravitational field is considered as a potential energy term where the position is the center of mass.

$$\mathcal{P}(q) = -m_i G^T r_{0c_i}(q) \quad (3.19)$$

where G is the vector that include the gravitational terms. Moreover, in case of elastic elements in the kinematic chain there will be a potential contribution due to the elastic energy added to the gravitational one.

3.2.4 Virtual Work

The method of Virtual Work is suitable for the multilink structures that change their configuration in time [9]. Depending on the kind of joint, the virtual work can be different. In the revolute joint the virtual work is the sum of the virtual works done by the angular torques, instead in the prismatic joint is the sum of the virtual works done by the linear forces.

$$\delta W = \sum_{K=1}^{N_f} f_K \delta r_K + \sum_{K=1}^{N_\tau} \tau_K \delta \alpha_K \quad (3.20)$$

3.2.5 Generalized Forces

The generalized forces are the forces due to the action of the actuators moving the joints and all the other forces exchanged with the environment neglecting the weight forces, included inside the gravitational potential energy terms, and the friction forces included

in the dissipation energy terms. From the definition of the virtual work it is possible to derive the i -th component of the generalized force acting on the i -th joint.

$$\begin{aligned}\delta W &= \sum_{K=1}^{N_f} f_K \delta_{rK} + \sum_{K=1}^{N_\tau} \tau_K \delta_{\alpha K} = \sum_{K=1}^{N_f} f_K \left[\sum_{i=1}^n \frac{\delta_{rK}}{\partial q_i} \delta q_i \right] + \sum_{K=1}^{N_\tau} \tau_K \left[\sum_{i=1}^n \frac{\delta_{\alpha K}}{\partial q_i} \delta q_i \right] \\ &= \sum_{i=1}^n \left[\sum_{K=1}^{N_f} f_K \frac{\partial_{rK}}{\partial q_i} \right] \delta q_i + \sum_{i=1}^n \left[\sum_{K=1}^{N_\tau} \tau_K \frac{\partial_{\alpha K}}{\partial q_i} \right] \delta q_i = \sum_{i=1}^n \mathcal{F}_{l_i} \delta q_i + \sum_{i=1}^n \mathcal{F}_{\alpha_i} \delta q_i\end{aligned}\quad (3.21)$$

where \mathcal{F}_{l_i} and \mathcal{F}_{α_i} are respectively the linear and the angular generalized forces. If the joints apply an active command torque $\tau_C = [\tau_{C_1} \dots \tau_{C_n}]^T$ the i -th component of the generalized total force is:

$$\mathcal{F}_i = \tau_{C_i} + J_i^T \mathcal{F}_{e_i} = \tau_{C_i} + \tau_{e_i} = \tau_i \quad (3.22)$$

where the first term is related to the command torques provided by actuators and the second term is due to the environment interactions. If friction forces are neglected it can be considered the Lagrangian equation without the dissipative energy:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \frac{\partial \mathcal{L}}{\partial q_i} = \tau_i \quad (3.23)$$

With the differentiations, it can be obtained the i -th dynamic equation:

$$\sum_{j=1}^n H_{ij}(q) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n h_{ijk}(q) \dot{q}_k \dot{q}_j + g_i(q) = \tau_i, \quad i = 1, \dots, n \quad (3.24)$$

where the terms $h_{ijk}(q)$ can be expressed through the so-called Christoffel symbols of the first type, given by:

$$h_{ijk}(q) = \frac{1}{2} \left(\frac{\partial H_{ij}(q)}{\partial q_k} + \frac{\partial H_{ik}(q)}{\partial q_j} + \frac{\partial H_{jk}(q)}{\partial q_i} \right) = h_{ikj}(q) \quad (3.25)$$

Setting $C_{ij}(q, \dot{q})$ the equation became:

$$\sum_{j=1}^n H_{ij}(q) \ddot{q}_j + \sum_{j=1}^n C_{ij}((q), \dot{q}) \dot{q}_j + g_i(q) = \tau_i \quad (3.26)$$

the so obtained dynamic model can be represented in matrix form as:

$$H(q) \ddot{q} + C((q), \dot{q}) \dot{q} + B \dot{q} + g(q) = \tau \quad (3.27)$$

where the dissipative term has been included, defining B as the diagonal matrix of the viscous friction coefficients.

3.2.6 Physical interpretation

It is possible to rewrite the entire equation considering the meaning of the Christoffel terms:

$$H_{ii}(q) \ddot{q}_i + \sum_{j \neq i} H_{ij}(q) \ddot{q}_j + \sum_j h_{ijj}(q) \dot{q}_j^2 + \sum_j \sum_{k \neq j} h_{ijk}(q) \dot{q}_k \dot{q}_j + g_i(q) = \tau_i \quad (3.28)$$

where:

1. $H_{ii}(q)\ddot{q}_i$ is the term due to the acceleration of the link.
2. $\sum_{j \neq i} H_{ij}(q)\ddot{q}_j$ is the term due to the other joint accelerations transmitted to the link.
3. $\sum_j h_{ijj}(q)\dot{q}_j^2$ is due to the centrifugal accelerations acting on the link.
4. $\sum_j \sum_{k \neq j} h_{ijk}(q)\dot{q}_k\dot{q}_j$ is the term due to the Coriolis acceleration acting on the link.
5. $g_i(q)$ is the term due to the weight forces acting on the link.

Chapter 4

Manipulator Control

The objective of the algorithms that are involved in the control of the manipulator is to feed the joint actuators with proper signals so that the tool centre point follows the assigned Cartesian path. Depending on the final target of the robot, the type of controller will be as complex as the path to follow will be difficult. It is possible to distinguish two main cases:

- Trajectory tracking without environment interaction: there are not forces or torques to be applied on other object or surface in the space.
- Trajectory tracking with environment interaction: it is necessary to follow the path and apply a force or torque on another object or surface in the space.

The other variables that affect the task are: the type of robot, its mechanical structure that can be equilibrated or not, the presence of constraints in the reference path that will change trajectories in time and, not last, the desired precision. The type of motors and the sensors used in the manipulator affect the design, too: the DC motors with brush or brushless linked with a gear box to the joint allow to strongly reduce the nonlinear effects on the dynamic model of the manipulator, but they introduce some phenomena that are difficult to model, like the Coulomb friction and the elasticity of the joint, on the other hand the direct-drive motors delete these negative effects but have some nonlinear dynamics. For the sensors, the control structure depends on the types of data that are available on the output: the precision of the velocity, that changes if it is computed via software or not, and the precision of the encoder position are two basic parameters for the closed chain accuracy. In general, it is important to contextualize the controller with respect to the target to achieve. The best controller for an application may be unsuitable for another one, the choice of the controller structure and its parameters depend on the environment and the final desired features of the system. To fulfil the specifications on transient and steady-state response requirements is equivalent in our case to follow the desired Cartesian path, this means that the joints must follow the desired time functions of the actuating torques, force.

A general control scheme can be seen in Figure 4.1, where the main blocks of the chain can be distinguished:

Controller: compares the desired value of the variable to be controlled (position/velocity) with the real one and changes the input processing the resulting error is such a way to minimize it.

Actuators: require the control signal and a source of energy, move and control the mechanism of the system.

Drives: are electronic devices providing power to a motor or servo.

Manipulator: is the system to be controlled.

Transducers: are the sensors that receive a signal or stimulus from a physical system and produce a signal, which represents information about the system, which is used by the control system.

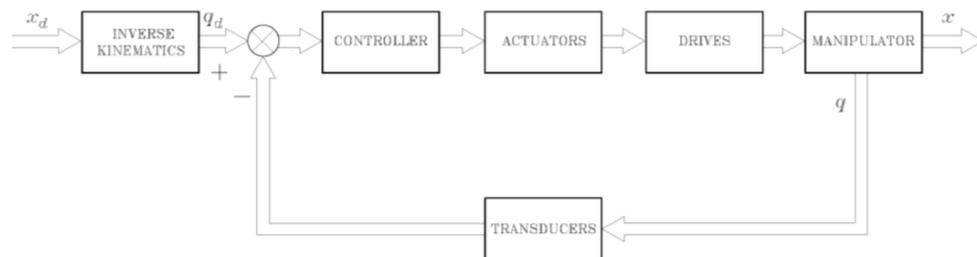


Figure 4.1: General scheme of Joint Space Control [23]

The control system of a manipulator can be made with different approaches: decentralized and centralized control.

4.1 Decentralized control

In the industrial environment, the most common approach is to treat each joint separately, designing a servomechanism that achieves the dynamic requests. The Decentralized Control is also called Independent Joint Control, because each joint motor has a local controller. Each system is Single Input Single Output (SISO) and the variables involved are the position and the velocity of the joint. The controller is designed considering an approximated model of the joint and it is often based on a PID architecture due to its simplicity.

Considering the previous background of the DC motor, starting from Figure 4.2 where the inputs are the armature voltage v_a and the resistive torque $\tau_r = \tau_d$ are seen as disturbance torque and the output is the position of the motor θ_m , it is possible to see how each joint can be modeled.

The choice to use the armature voltage v_a as command variable it is a common feature for the European and Japanese manufacturers, while North American tend to use the armature current i_a . Assuming some approximations, it is possible to simplify the model:

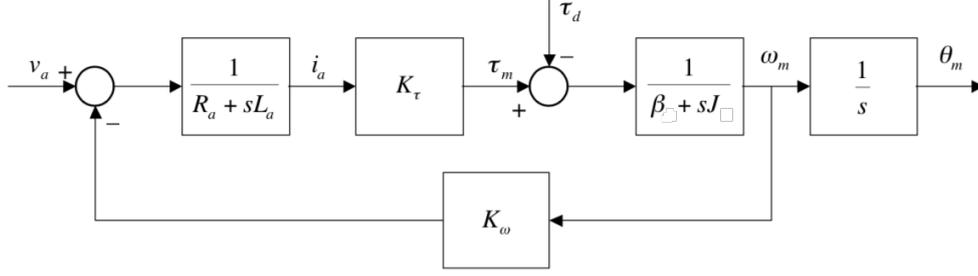


Figure 4.2: DC motor diagram in open loop

It is reasonable to neglect L_a , knowing that the losses associated are usually very small. The equation of the motor will become:

$$V_a - R_a I_a = K_\omega \omega_m \rightarrow V_a - \frac{R_a}{k_\tau} \tau_m = K_\omega \omega_m \quad (4.1)$$

and from the torques equilibrium equations (2.9) it can be obtain:

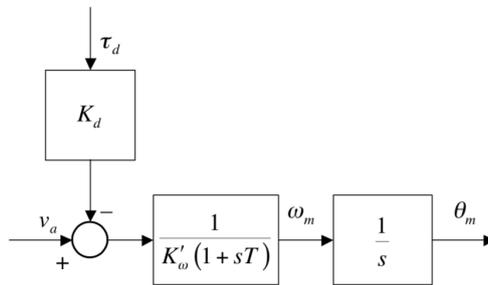
$$V_a - R_a \frac{\tau_d}{k_\tau} = \left[\frac{R_a}{k_\tau} J_t s + \left(K_\omega + \frac{R_a}{k_\tau} \beta_t \right) \right] \omega_m \quad (4.2)$$

The second term inside the square brackets, is very close to K_ω because $\frac{R_a \beta_t}{k_\tau} \omega_m \ll K_\omega \omega_m$. Focusing on the equation:

$$\underbrace{R_a I_a}_{\text{armature losses}} + \underbrace{\frac{\beta_t \omega_m}{k_\tau I_a}}_{\substack{\text{friction torque} \\ \text{torque } \tau_m}} \ll \underbrace{\frac{K_\omega \omega_m}{k_\tau}}_{\text{e.f.m link}} \quad (4.3)$$

The armature losses are multiplied by a small constant, the ratio between the torques lost through the frictions and the driving torque. They will be much less than the electromotive force of the armature because.

From the above equation it is possible to obtain the block diagram reported in Figure 4.3:


 Figure 4.3: Simplified DC motor diagram in open loop with $L_a = 0$

Setting : $T = \frac{R_a J_t}{K_\tau K'_\omega}$ where $K'_\omega = (K_\omega + R_a \beta_t / K_\tau) \approx K_\omega$, the transfer functions between input and output are:

$$\frac{\omega_m(s)}{v_a(s)} = G_\omega(s) = \frac{1}{K'_\omega(1 + sT)} \quad (4.4)$$

$$\frac{\omega_m(s)}{\tau_d(s)} = G_d(s) = -K_d G_\omega(s) = -\frac{T}{J_t(1 + sT)} \quad (4.5)$$

with $K_d = \frac{R_a}{K_\tau}$.

The controller concerned in the project is based on two feedback loop: velocity and position (Figure 4.4).

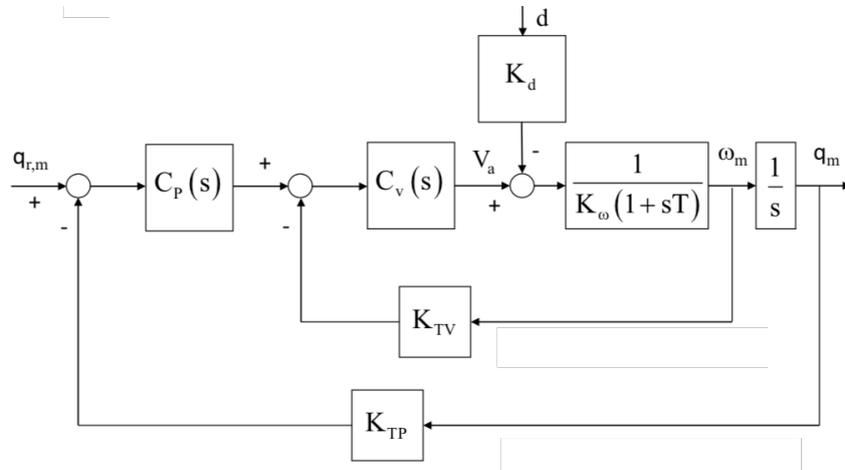


Figure 4.4: Diagram with velocity-loop and position-loop

The inputs are the desired position and the possible disturbance on the direct chain and the output is the position of the motor.

4.1.1 Cascade Control

To regulate the output variable can be convenient to realize a control loop characterized by more loops. The Cascade Control is a control technique that is exploitable when:

1. The dynamics of the process to be controlled can be decomposed into two or more distinct dynamics.
2. The first dynamics is faster than the second one and the following.
3. The variable between the two dynamics can be measured.

The inner control loop shall consist of a velocity loop: the velocity signal is measured through a sensor, an encoder with a gain fixed by the constructor K_{TV} . The external one shall consist of the position loop with its corresponding coefficient K_{TP} . The global controller is designed in such a way to obtain: a high rejection ratio with respect to the noise

through a high gain in the upstream block and an integrative action to delete the gravity effect at steady state. These requests lead to two controllers, one purely *Proportional* (**P**) and another *Proportional-Integrative* (**P-I**).

$$C_P(s) = K_P \quad C_V(s) = K_V \frac{1 + sT_V}{s} \quad (4.6)$$

The time constant T_V can be selected to remove the mechanical pole of the joint:

$$T_V = T \quad (4.7)$$

With this choice the transfer function of the closed loop will be of the second order:

$$W(s) = \frac{K_P K_V}{K_\omega s^2 + K_V K_{TV} s + K_{TP} K_P K_V} \quad (4.8)$$

It can be possible to assign arbitrarily the two poles of $W(s)$ tuning the only free parameters K_P and K_V . Focusing on the function in such a way to exploit the system as a second order type, underlining the damping factor and the natural frequency:

$$W(s) = \frac{\frac{1}{K_{TP}}}{1 + \frac{2\zeta}{\omega_n} s + \frac{s^2}{K_\omega}} \quad (4.9)$$

with:

$$\frac{2\zeta}{\omega_n} = \frac{K_{TV}}{K_P K_{TP}} \quad \frac{1}{\omega_n^2} = \frac{K_\omega}{K_P K_{TP} K_V} \quad (4.10)$$

To obtain the desired poles are imposed:

$$K_V = 2\zeta\omega_n \frac{K_\omega}{K_{TV}} \quad K_P = \frac{K_{TV}}{2\zeta K_{TP}} \omega_n \quad (4.11)$$

Similarly with the superposition effect theorem it is possible to find the transfer function resulting between the disturbance and the output of the system:

$$W_d(s) = \frac{q_m(s)}{d(s)} = -K_d \frac{\frac{s}{K_V K_P K_{TP} (1+sT)}}{1 + \frac{K_{TV}}{K_P K_{TP}} s + \frac{K_\omega}{K_V K_P K_{TP}} s^2} \quad (4.12)$$

The rejection ratio of the disturbance it is the product $K_P K_{TP} K_V$ completely defined imposing K_P and K_V for the desired dynamics. The time necessary to have a significant attenuation of the disturbance is approximable to $T_R = \max[T, 1/(\zeta\omega_n)]$.

The Cascade Control has more advantages with respect to the traditional control:

1. The disturbance can be compensated with the inner loop fast enough to remove the contribute on the output.
2. It is possible to achieve a faster inner dynamics.
3. Greater robustness of the internal system thanks to the second feedback loop and more rejection ratio.
4. Possibility to force a desired dynamics of the inner loop with a consequent simplification of the design of the external controller.

4.2 Calculated Torque Feedforward Control

When it is requested to follow paths with high velocities and accelerations it is possible to reduce the tracking error using the target of these variables in a *Feedforward Compensation*.

This method is a tradeoff between the Independent Joint Control and the Centralized Control.

In particular, this approach can be used when it is necessary to estimate a *Calculated Torque* to remove a known contribute of the exogenous disturbances acting on the system.

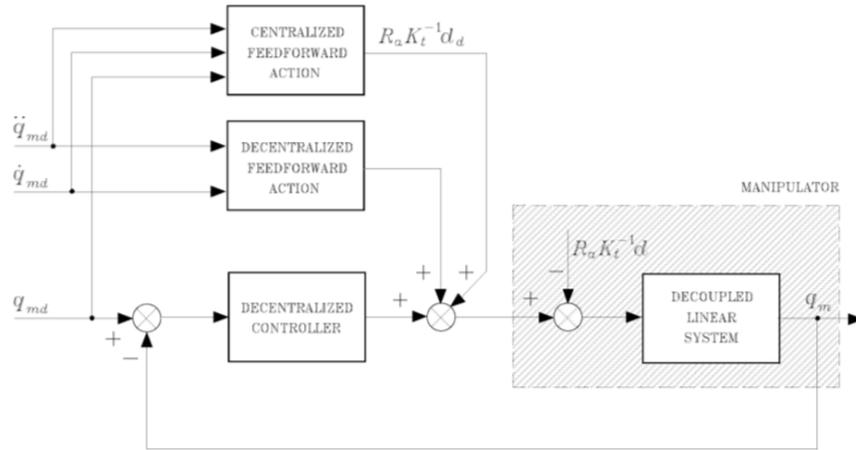


Figure 4.5: Block scheme of computed torque feedforward control [23]

Even though the disturbance term can be canceled $d_r = d_d - d = 0$ only in the ideal case of the perfect tracked trajectory $q = q_d$ and of perfect knowledge of the model. The calculated torque reduce, anyway, the coupling disturbance, this means that it is required less effort to the controllers that can achieve the same rejection ratio with lower proportional gains. The calculation of the feedforward is computationally onerous because it is a centralized control action and the computation time can exceed the sampling time in the real-time systems.

To obtain a right disturbances compensation in the systems where the small sampling time is a limitation, a possible solution is to compute a partial compensation where only the most significant terms will be canceled. In general, the most heavy terms are usually the inertia and the gravitation that change in time. Finally, it should be pointed out that the choice of the properly control structure is always a trade-off between memory capacity and computational requirements.

Independent joint control may be inappropriate when the required operating velocities are too high, the structured disturbance torques dependent on Coriolis' and centrifugal velocities heavily affect the joints behavior and furthermore, it may happen that due to the lack of the gearbox there is no reduction of the non linear coupling effects among the joints. In these cases the tracking error can become too high due to the torque disturbance effects. In order to directly delete these torques, a control strategy containing non linear

compensation terms is used.

4.3 Centralized control

In the Centralized control the applied torque to each joint is a function of the positions and velocities of the other joints. The manipulator is considered as a MIMO (Multi Input Multi Output) system with n torques applied as inputs to the joints and n positions as outputs that interact among themselves with nonlinear relations.

The centralized control law is necessary non linear, the main approach is the *Inverse Dynamics* one: the applied torque resulting by the dynamic equation of the manipulator is computed starting from the knowledge of the joint variables.

The motors are considered as torque generators because the aim of this control is to apply the torques that will compensate exactly the structured disturbances. Neglecting the value of the armature inductance $L_a \approx 0$ the dynamic equation of the electric motor will be:

$$V_a = G_v V_c = R_a I_a + K_\omega \dot{q}_m \quad (4.13)$$

from this it is possible to find the armature current:

$$I_a = R_a^{-1}(G_v V_c + K_\omega \dot{q}_m) \quad (4.14)$$

Assuming K_r the reduction ratio matrix due to the gearbox, the applied torque will be:

$$\tau_c = K_r \tau_m = K_r K_\tau I_a = K_r K_\tau R_a^{-1}(G_v V_c + K_\omega \dot{q}_m) \quad (4.15)$$

Including another current loop, to obtain a torque generator, the K_w can be neglected, the armature current will be:

$$I_a = G_i V_c \quad (4.16)$$

where G_i is a constant gain matrix and V_c is an array of voltages, this means that:

$$\tau_c = K_r \tau_m = K_r K_\tau I_a = K_r K_\tau G_i V_c = u \quad (4.17)$$

where u is the array of the control input.

As we had seen before the equation of the dynamic model can be written as:

$$H(q)\ddot{q} + C((q), \dot{q})\dot{q} + B\dot{q} + g(q) = \tau_c \quad (4.18)$$

assuming:

$$n(q, \dot{q}) = C((q), \dot{q})\dot{q} + B\dot{q} + g(q) \quad (4.19)$$

then:

$$H(q)\ddot{q} + n(q, \dot{q}) = u \quad (4.20)$$

The inverse dynamic control approach is based on the idea to obtain the linearization of the dynamics of the system through a non linear feedback of the states, so to achieve a linear and decoupled system. Under the hypothesis to have the capacity to compute exactly the matrices $H(q)$ and $n(q, \dot{q})$ of the dynamic model, the vector command u is defined as:

$$u = H(q)a_c + n(q, \dot{q}) \rightarrow \ddot{q} = a_c \quad (4.21)$$

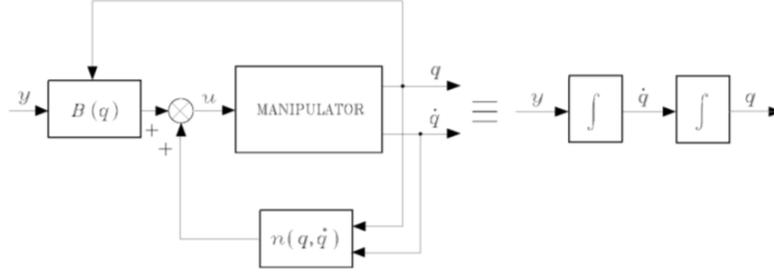


Figure 4.6: Global linearization with Inverse Dynamics Control [23]

where a_c is the new acceleration input to define in order to obtain an asymptotic stability.

The control loop is included to get a exact linearization of the system, so that the resulting system consists of n double integrators as in Figure 4.6.

The decoupled system:

$$\ddot{q} = a_c = y \quad (4.22)$$

can be easily controlled through a loop function. The controller could have a Proportional-Derivative actions (Figure 4.7), to which a feedforward acceleration term can be added:

$$y = K_P \tilde{q} + K_D \dot{\tilde{q}} + \ddot{q}_d \quad (4.23)$$

where q_d is time varying. The closed loop obtained is the following:

$$\ddot{\tilde{q}} + K_D \dot{\tilde{q}} + K_P \tilde{q} = 0 \quad (4.24)$$

The error is now governed by a second order dynamics that can be assign choosing the

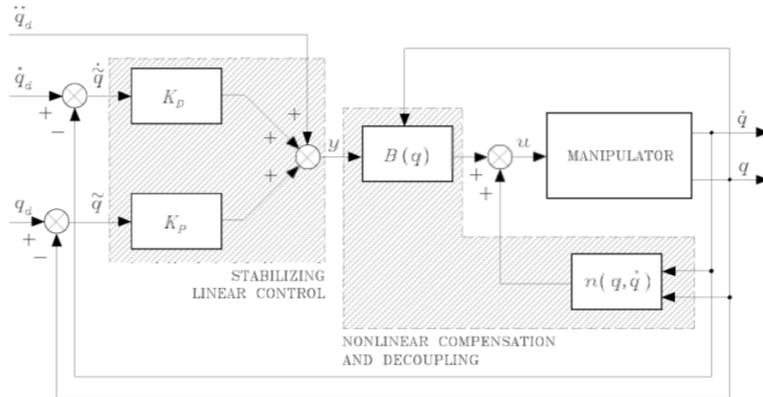


Figure 4.7: Inverse Dynamics with P-D actions [23]

suitable gains in the diagonal matrices K_P and K_D . Assuming the matrices K_P and K_D diagonal, the time evolution of the error is governed by eigenvalues that are roots of the polynomial:

$$s^2 + K_D s + K_P = 0 \quad (4.25)$$

This means that with the inverse dynamics control we have full control on the time history with which the error goes to zero.

Part III

Project Development

The principle aim of this project of thesis is to improve the ability of "e.DO", from the control point of view, i.e., to improve the tracking performance of the robot. It is necessary that each joint follows the desired input trajectories, but with some evident constraints. There are some requested features that are absolutely mandatory to satisfy:

1. The manipulator must have an overall synchronous movement.
2. The stability of the system must be achieved in every condition.
3. Each joint must follow the desired trajectory without overshoot.

To fulfill the requirements it is indispensable to know the model of the system. The model have to comply to the behavior of the real manipulator. Once the model is coherent with the reality, it is possible to modify some features on the pattern, verify if it reacts correctly to the modifications and then make the changes on the real machine. It is a process very similar to the validation and verification that is used in the automotive factory where there are phases to be respected:

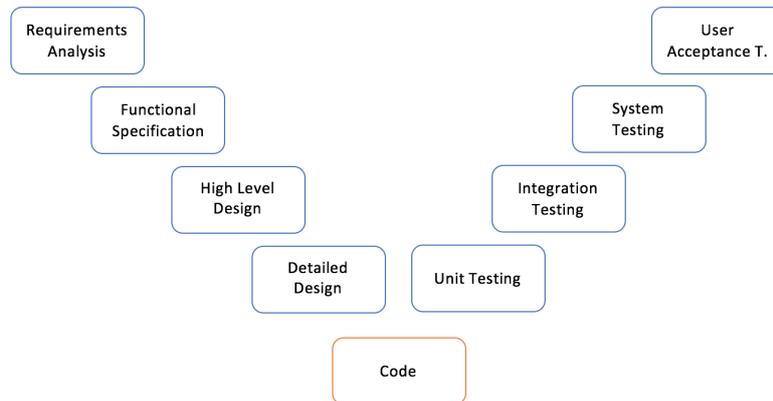


Figure 4.8: Validation and Verification diagram

This continuous iteration in the improvement process suggests to split the following developments in Model and Simulation issues.

Chapter 5

Dynamic Model of the e.DO robot

5.1 e.DO features

The dynamic model is built on the basis of the structure and the components of the manipulator. In this section are listed and depicted the e.DO's components with a brief analysis of the datasheets. e.DO is a six joints manipulator that has two kinds of motors, one for the first three joints and another one the last. This characteristic splits the links too, the brackets that connect the joints together are different depending on the structure (Figure 5.1).

A Top-Down approach is used to examine the problems of the system as a whole.

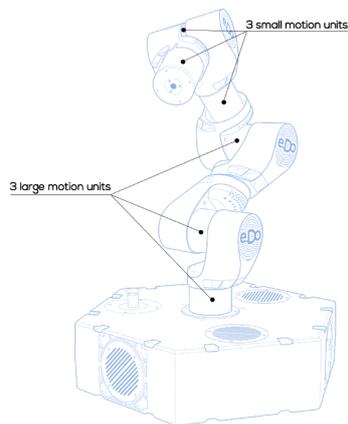


Figure 5.1: e.DO six action motion units

The hardware architecture of the system includes also two main elements:

- Controller Raspberry PI
- Novalabs ServoDrive

The Raspberry is the brain of the manipulator, the control system and the main software algorithms of the project are included in this component. The software is ROS (*Robotic Operating System*) compliant. The second component is the drive that actuates the DC motor. The standard flux of communication is represented in Figure 5.2 where it can be seen that the command received from the tablet will be processed by the RaspberryPI and finally will go to the joints through the Novalabs Servodrive moving e.DO.

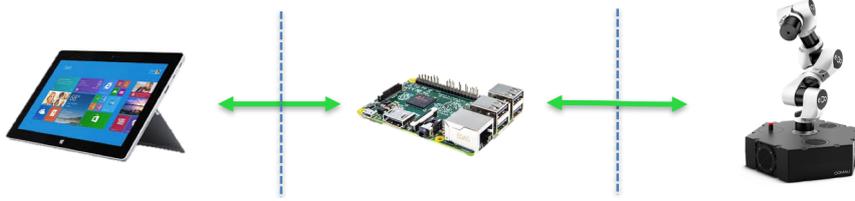


Figure 5.2: The standard flux of communication.

The two kinds of motors were provided by the Swiss company Maxon Motor. In all the project the large motion units are called *Arm motors* and the small motion units *Wrist motors*. The characteristics of these components can be summarized as in Table 5.1, where are listed the most significant values of the datasheets. To achieve high transmissible torque reducing the velocity output at the terminal side of the motor, were added two different planetary gears depending on the motor type; their features are reported in Table 5.2.

	Wrist Motor	Arm Motor
Values at nominal voltage		
Nominal Voltage	12 V	12 V
No load speed	13200 rpm	11700 rpm
No load current	35.4 mA	81.8 mA
Nominal speed	9850 rpm	10700 rpm
Nominal torque	5.36 mNm	30.4 mNm
Nominal current	0.662 A	3.21 A
Max. efficiency	77.3 %	90.6 %
Characteristics		
Max. output power	8.18 W	46.9 W
Terminal resistance	4.53 Ohm	0.335 Ohm
Terminal inductance	0.131 mH	0.0346 mH
Torque constant	8.53 mNm/A	9.73 mNm/A
Mechanical time constant	6.74 ms	3.21 ms
Rotor inertia	1.08 gcm ²	9.06 gcm ²

Table 5.1: Dc Motor Datasheets of DCX16S GB KL 12V and DCX22L GB KL 12V

	Wrist Planetary Gear	Arm Planetary Gear
Gearhead Data		
Reduction	186 : 1	231 : 1
Max. continuous torque	0.8 Nm	4.5 Nm
Max. intermittent torque	1 Nm	6.2 Nm
Max. efficiency	65 %	75 %
Technical Data		
Radial play	0.1 mm	0.1 mm
mm from flange	5 mm	10 mm
Axial play	0 ÷ 0.1 mm	0 ÷ 0.4 mm
Max. radial load	90 N	150 N
Max. axial load (dynamic)	30 N	80 N
Max. continuous input speed	14000 rpm	10000 rpm

Table 5.2: Planetary gear Datasheets of GPX16HP 186:1 and GPX26 A 231:1.

With these data it was possible to create a model of the whole system.

5.2 From Paper to Model

Starting from the analysis of the DC motor from the equations (2.17) and (2.18), the transfer function between the armature voltage V_A and the angular velocity ω in the Laplace domain is given by:

$$\frac{\omega(s)}{V_A(s)} = \frac{k_\tau}{s^2 J L_A + s(J R_A + L_A \beta) + (k_\tau K_\omega + R_A \beta)} = \frac{\frac{1}{K_\omega}}{s^2 \tau_A \tau_M + s \tau_M + 1} \quad (5.1)$$

If the time constants τ_M and τ_A are separated by at least one decade, it is possible to simplify this expression into:

$$\frac{\omega(s)}{V_A(s)} = \frac{\frac{1}{K_\omega}}{(1 + s \tau_M)(1 + s \tau_A)} \quad (5.2)$$

Putting inside the equation the right data from the datasheets only for the Arm Motor the previous transfer function will highlight the poles and the zeros in the formula:

$$\frac{\omega(s)}{V_A(s)} = \frac{8.3511 \times 10^7}{(s + 2241)(s + 375.5)} \quad (5.3)$$

where for simplicity the friction coefficient is considered null.

The engine is already equipped with a negative feedback, so the poles that the system will show on the outside will be different from the mechanical and electrical poles on the direct chain of the block diagram.

This means that the time constants are $\tau_A \approx 3.8388 \times 10^{-4}s$ and $\tau_M \approx 0.0032s$, it is clear that there is at least one decade and the two poles can be highlighted on the Bode diagram that will appear to be the following one:

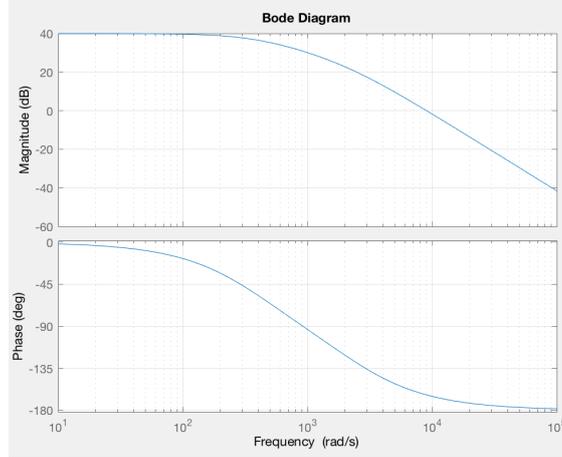


Figure 5.3: Bode Diagram of the Arm Motor.

5.2.1 Planetary gear

To complete the model it is necessary to consider the overall system composed by motor and the planetary gear. The planetary gear has some sensible advantages:

- High reduction ratio
- High transmissible torque
- High radial loads on output shafts

From the mechanical point of view it can be modeled as a parallel shaft gearbox, the essentially aim of this component is to increase the torque to the detriment of the velocity. The planetary gear can be modeled with two ideal gearwheels with radius ρ' and ρ , on the motor side and on the arm side, respectively, with an even number of teeth N' and N . The number of teeth is proportional to the radius: $\frac{\rho'}{\rho} = \frac{N'}{N}$.

The values from the motor side are written with apex, instead the values from the link side are indicated without apex (Figure 5.4).

It is defined *Transformation ratio* the value: $r = \frac{N}{N'} = \frac{\rho}{\rho'}$. This ratio can be higher or lower than one, but in the industrial robot it is always $r > 1$ and the absolute value is written in the form $r : 1$.

During the motion it is necessary that the product between the positions θ' and θ and the respectively rays ρ' and ρ must be equal: $\rho'\theta' = \rho\theta$. The approach is the same for the derivative it means that: $\rho'\omega' = \rho\omega$. From the listed relations follows that $\omega' = r/\omega$ i.e. $\omega = \omega'/r$. It appears that the velocity on the joint side is lower than the velocity on the motor side of the ratio $1/r$.

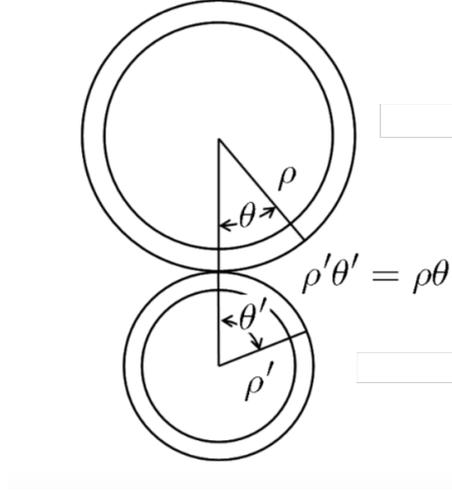


Figure 5.4: Gearmotor diagram of the joint.

If we assume an ideal efficiency equal to one, the relationship between the torques between the planetary gear side and the motor side becomes:

$$\omega_m \tau_r = \omega'_m \tau'_r \quad (5.4)$$

where τ_r is the torque provided to the downstream arm.

Remembering that $\omega'_m = r\omega_m$:

$$\tau_r = r\tau'_r \quad (5.5)$$

$$= r(\tau'_m - \tau'_p) \quad (5.6)$$

$$= r[\tau'_m - J_m \dot{\omega}'_m - \beta_m \omega'_m] \quad (5.7)$$

$$= r[\tau'_m - J_m (r\dot{\omega}_m) - \beta_m (r\omega_m)] \quad (5.8)$$

$$= r\tau'_m - r^2(J_m \dot{\omega}_m + \beta_m \omega_m) \quad (5.9)$$

expressed in joint velocities.

From the point of view of the equilibrium of the torque equation to the downstream gearbox, the output torque is the component of the differential equation of the manipulator dynamics:

$$\tau = H(q)\ddot{q} + C(q, \dot{q})\dot{q} + B(q)\dot{q} + g(q) + J^T F_e \quad (5.10)$$

where:

- $H(q)\ddot{q}$ are the inertial torque of the manipulator
- $C(q, \dot{q})\dot{q}$ are the Coriolis and centrifuges torques
- $B(q)\dot{q}$ are the viscous friction
- $J^T F_e$ are the equivalent joint torques that balance the generalized external force that are created by the interaction between the tool center point and the external environment.

The i-th element of the vectorial previous equation becomes:

$$\tau_{ri} = \sum_{j=1}^n H_{ij}(q)\ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n h_{ijk}(q)\dot{q}_k\dot{q}_j + g_i(q) + \beta_{bi}\dot{q}_i + \tau_{fi}, \quad i = 1, \dots, n \quad (5.11)$$

where τ_{fi} are the i-th torque component of the product $J^T F_e$. Developing the equation and remembering that the angular values to the downstream of the gearmotor are

$$q_i = \frac{\theta'_{mi}}{r_i}; \quad \dot{q}_i = \frac{\dot{\theta}'_{mi}}{r_i}; \quad \ddot{q}_i = \frac{\ddot{\theta}'_{mi}}{r_i} \quad (5.12)$$

and that $H_{ii} = J_{bi}$ is the proper inertia of the i-th arm:

$$\begin{aligned} \tau_{ri} &= \left(\sum_{j \neq i}^n H_{ij}\ddot{q}_j + H_{ii}\ddot{q}_i \right) + \sum_{j=1}^n \sum_{k=1}^n h_{ijk}\dot{q}_j\dot{q}_k + g_i + \beta_{bi}\dot{q}_i + \tau_{fi} \\ &= H_{ii}\ddot{q}_i + \beta_{bi}\dot{q}_i + \sum_{j \neq i}^n H_{ij}\ddot{q}_j + \tau_{ci} + \tau_{gi} + \tau_{fi} \\ &= J_{bi} \frac{\dot{\omega}'_{mi}}{r_i} + \beta_{bi} \frac{\omega'_{mi}}{r_i} + \tau_{Mi} + \tau_{ci} + \tau_{gi} + \tau_{fi} \end{aligned} \quad (5.13)$$

where:

$$\tau_{Mi} = \sum_{j \neq i}^n H_{ij}\ddot{q}_j \quad (5.14)$$

is the torque due to the massive and inertial coupling effects,

$$\tau_{ci} = \sum_{j=1}^n \sum_{k=1}^n h_{ijk}\dot{q}_j\dot{q}_k \quad (5.15)$$

is the torque due to the centrifugal and Coriolis effects, $\tau_{gi} = g_i$ is the torque due to the gravitational effects and τ_{fi} is the torque due to the environment interaction effects. In addition can be inserted another term to consider any errors on the model or on measurements called *structured disturbance torque*:

$$\tau_{di} = \tau_{Mi} + \tau_{ci} + \tau_{gi} + \tau_{fi} \quad (5.16)$$

Finally, it can be obtained these equations:

$$\tau_{ri} = J_{bi}\ddot{q}_i + \beta_{bi}\dot{q}_i + \tau_{di} \quad (5.17)$$

depending on the joint velocities and

$$\tau'_{ri} = \frac{J_{bi}}{r_i}\dot{\omega}'_{mi} + \frac{\beta_{bi}}{r_i}\omega'_{mi} + \tau_{di} \quad (5.18)$$

depending on the motor velocities.

If we try to bring the torque to the upstream of the gearbox:

$$\tau'_{ri} = \frac{1}{r_i}\tau_{ri} = \frac{1}{r_i^2}(J_{bi}\dot{\omega}'_{mi} + \beta_{bi}\omega'_{mi}) + \frac{1}{r_i}\tau_{di} \quad (5.19)$$

remembering that:

$$\tau'_r = \tau'_m - \tau'_p = \tau'_m - (J_m \dot{\omega}'_m + \beta_m \omega'_m) \quad (5.20)$$

and we obtain:

$$\tau'_m = \tau'_r + \tau'_p = \left(\frac{1}{r^2} J_b + J_m\right) \dot{\omega}'_m + \left(\frac{1}{r^2} \beta_b + \beta_m\right) \omega'_m + \frac{1}{r} \tau'_d \quad (5.21)$$

Due to the setting:

$$\begin{aligned} \tau'_d &= \frac{1}{r} \tau_d \\ J'_t &= \left(\frac{1}{r^2} J_b + J_m\right) \\ \beta'_t &= \left(\frac{1}{r^2} \beta_b + \beta_m\right) \end{aligned}$$

we can finally write:

$$\tau'_m = \tau'_r + \tau'_d = J'_t \dot{\omega}'_m + \beta'_t \omega'_m + \tau'_d \quad (5.22)$$

This last equation has a significant meaning: the contribution of the torque τ'_p is related to the total inertia J'_t and the viscous friction β'_t which are mainly composed by the motor inertia J_m and β_m . The inertial contribution and friction due to the link are reduced by the term $\frac{1}{r^2}$. It is reasonable that the effect of the inertia change due to the configuration of the whole robotic structure appears scarcely influent on the torque motor model. This is exclusively due to the introduction of the planetary gearbox.

It will finally appear that:

$$\tau_m = \tau_p + \tau_d = J_t \ddot{q} + \beta_t \dot{q} + \tau_d \quad (5.23)$$

with:

$$J_t = (r^2 J_m + J_b); \quad \beta_t = (r^2 \beta_m + \beta_b) \quad (5.24)$$

This equation shows that the total inertia of joints are mainly the inertia of motors multiplied by r^2 . The relationship among the variables are highlighted in Figure 5.5.

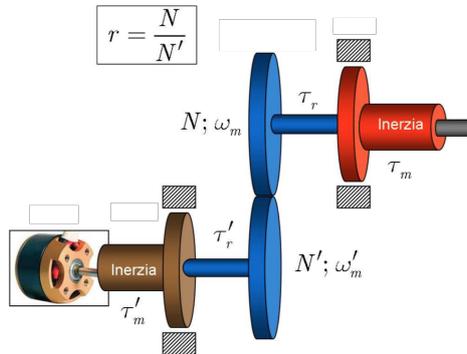


Figure 5.5: Mechanical diagram of the overall joint.

From the datasheets, for the arm motors we have:

$$\begin{cases} J_m = 9.06 \times 10^{-7} \text{ Kgm}^2 \\ J_b = 1.282 \times 10^{-7} \text{ Kgm}^2 \end{cases} \quad (5.25)$$

i.e. the total inertia become:

$$\begin{aligned} J'_t &= \left(\frac{1}{r^2} J_b + J_m \right) \\ &= \left(\frac{1}{231^2} 1.282 \times 10^{-7} + 9.06 \times 10^{-7} \right) \text{ Kgm}^2 \rightarrow J'_t \approx J_m \end{aligned} \quad (5.26)$$

The total inertia is approximated by the motor inertia. Another reduction jump to bring in the model is due to the link: the second ratio is 5.333. This means that the total reduction ratio is the product between the first and second one:

$$\begin{aligned} \text{Arm motors} &\rightarrow r_{tot} = 5.333 \times 231 = 1232 \\ \text{Wrist motors} &\rightarrow r_{tot} = 5.333 \times 186 = 992 \end{aligned} \quad (5.27)$$

Now the further step is to model the total joint with the link. Obviously this consideration can be made only with the joints without the gravity effect, for the first part of this project of thesis I considered only them.

5.3 Simulation of model with fixed Inertia

Using the Simulink development environment with MATLAB it was possible to create a model and simulate the system. After the definition of the values in a program on Matlab, it was possible to proceed by blocks with the diagram presented in Figure 5.6

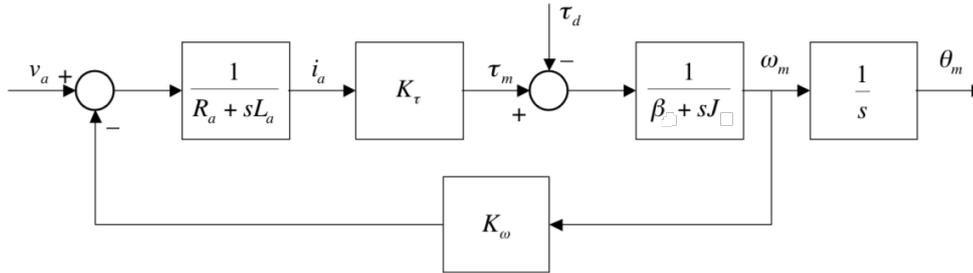


Figure 5.6: DC motor diagram in open loop.

Once created the model it was possible to simulate the behavior; the first step was to verify the nominal velocity with a supply voltage of 12V.

Figure 5.7 shows the step response of the two kinds of motors.

It is evident in the Data Cursor the values of the velocity in $[\frac{rad}{s}]$. In the following table we can highlight the differences from the datasheets value (Table 5.3).

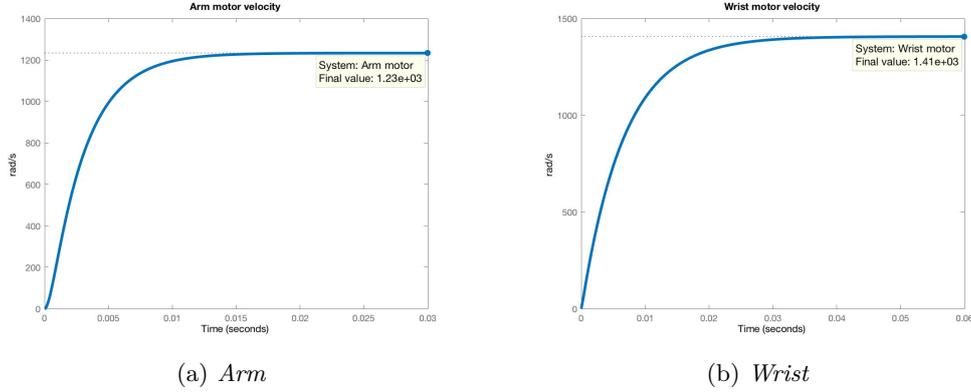


Figure 5.7: Velocity in $\left[\frac{rad}{s} \right]$ of Arm and Wrist Motor fed with nominal 12 V voltage.

	Arm Motor	Wrist Motor
Values at nominal voltage from datasheets		
Nominal Voltage [V]	12	12
No load speed [rpm]	11700	13200
No load speed [rad/s]	1225.22	1382.3
Simulated with Simulink		
Simulated speed [rad/s]	1233.3	1406.8

Table 5.3: Dc Motor Datasheets VS Simulated velocity at 12V

The differences between the model and the reality are very small, this means that the model meets the needs, however the small gap can be due to different reasons like the fact that the frictions has not been included in the model, until now.

In order to obtain a correct model of the whole system, it is necessary to find the right values or a good approximate estimate of the Joint's Inertia that changes on time, because depending on the configuration of the manipulator the inertia that acts on the engine changes. From this point of view a big help is given by the Comau Softwares that with their library ORL can provide the model of the manipulator in real time.

5.4 Kinematics and Dynamics of e.DO

The structure of the e.DO manipulator can be interpreted distinguishing the first three degrees of freedom, that constitute an anthropomorphic arm, and the last three degrees of freedom corresponding to the wrist. Figure 5.8 shows a symbolic representation of the manipulator, including how frames have been attached to the links.

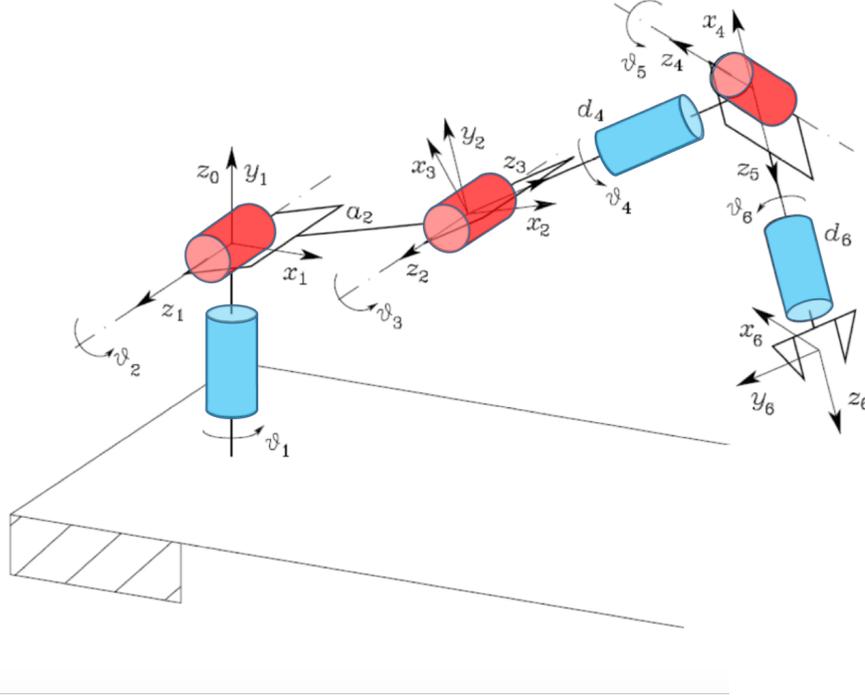


Figure 5.8: Symbolic representation of the e.DO manipulator.

To solve the *Direct kinematics* of the manipulator consist on determine the Transformation matrix that links the reference frame of the tool center point to the base reference frame. In the case of a six-joints manipulator:

$$T_6^0(q) = \begin{bmatrix} R_6^0(q) & t_6^0(q) \\ 0^T & 1 \end{bmatrix} \quad (5.28)$$

where t_6^0 links the translations and R_6^0 the rotations between the two reference frames. Starting from the reference frame on the base it is possible to find all the matrices that allow to find the relation between the tool center point and the base. All rotation matrices can be calculated as product of basic rotation around the z-axis and the x-axis:

$$R_{z,\theta} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad R_{x,\theta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (5.29)$$

where θ is the rotation angle.

It is possible to analyze the arm and the wrist of the manipulator separately.

5.4.1 Arm kinematics

Considering the Denavit-Hartenberg parameters in Table 5.4, the transformation matrices that link the first three reference frame are given by:

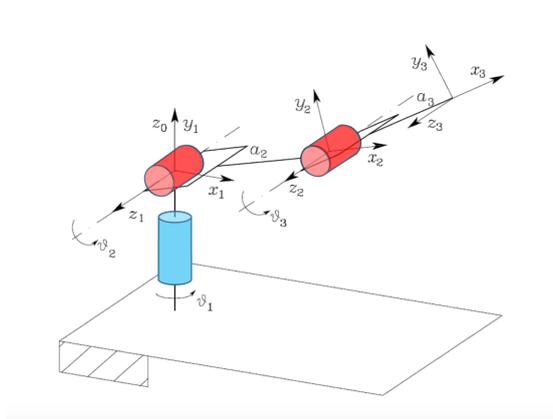


Figure 5.9: Symbolic representation of the arm of e.DO.

Arm				
	\mathbf{a}_i	α_i	\mathbf{d}_i	θ_i
1	0	$\pi/2$	0	θ_1
2	a_2	0	0	θ_2
3	a_3	0	0	θ_3

Table 5.4: Denavit-Hartenberg parameters of the Arm

$$T_1^0(q_1) = \begin{bmatrix} \cos(q_1) & 0 & -\sin(q_1) & 0 \\ \sin(q_1) & 0 & -\cos(q_1) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.30)$$

$$T_2^1(q_1) = \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & a_2 \cos(q_2) \\ \sin(q_2) & \cos(q_2) & 0 & a_2 \sin(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.31)$$

$$T_3^2(q_1) = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 & a_3 \cos(q_3) \\ \sin(q_3) & \cos(q_3) & 0 & a_3 \sin(q_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.32)$$

5.4.2 Wrist kinematics

The wrist represented in Figure 5.10 shows the last three axes with the Denavit-Hartenberg parameters, that are reported in Table 5.5.

In the wrist the transformation matrices that link the last three reference frames are

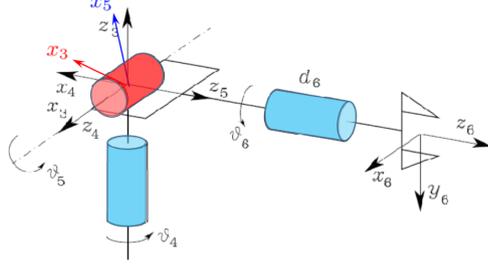


Figure 5.10: Symbolic representation of the wrist of e.DO.

Wrist				
	\mathbf{a}_i	α_i	\mathbf{d}_i	θ_i
4	0	$-\pi/2$	0	θ_4
5	0	$\pi/2$	0	θ_5
6	0	0	d_6	θ_6

Table 5.5: Denavit-Hartenberg parameters of the Wrist

given by:

$$T_4^3(q_4) = \begin{bmatrix} \cos(q_4) & 0 & -\sin(q_4) & 0 \\ \sin(q_4) & 0 & -\cos(q_4) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.33)$$

$$T_5^4(q_5) = \begin{bmatrix} \cos(q_5) & 0 & \sin(q_5) & 0 \\ \sin(q_5) & 0 & -\cos(q_5) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.34)$$

$$T_6^5(q_6) = \begin{bmatrix} \cos(q_6) & -\sin(q_6) & 0 & 0 \\ \sin(q_6) & \cos(q_6) & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.35)$$

Analyzing the whole structure, the transformation matrix that links the tool centre point and the base it can be rewritten as:

$$T_6^0 = \begin{bmatrix} R_p^0(q) & t_p^0(q) \\ 0^T & 1 \end{bmatrix} = \begin{bmatrix} n_p^0(q) & s_p^0(q) & a_p^0(q) & t_p^0(q) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.36)$$

where $a_p^0(q)$ is usually the versor that indicates the direction of the gripper on the tool center point, $s_p^0(q)$ is perpendicular to $a_p^0(q)$ in the sliding surface and the versor $n_p^0(q)$ is normal to the previous ones.

$$t_p^0(q) = \begin{bmatrix} a_2c_1c_2 + d_4c_1s_{23} + d_6(c_1(c_{23}c_4s_5 + s_{23}c_5) + s_1s_4s_5) \\ a_2s_1c_2 + d_4s_1s_{23} + d_6(s_1(c_{23}c_4s_5 + s_{23}c_5) - c_1s_4s_5) \\ a_2s_2 - d_4c_2s_3 + d_6(s_2s_3c_4s_5 - c_2s_3c_5) \end{bmatrix} \quad (5.37)$$

$$n_p^0(q) = \begin{bmatrix} c_1(c_{23}(c_4c_5c_6 - s_4s_6) - s_{23}s_5c_6) + s_1(s_4c_5c_6 + c_4s_6) \\ s_1(c_{23}(c_4c_5c_6 - s_4s_6) - s_{23}s_5c_6) - c_1(s_4c_5c_6 + c_4s_6) \\ s_{23}(c_4c_5c_6 - s_4s_6) + c_{23}s_5c_6 \end{bmatrix} \quad (5.38)$$

$$s_p^0(q) = \begin{bmatrix} c_1(-c_{23}(c_4c_5c_6 + s_4s_6) + s_{23}s_5c_6) + s_1(-s_4c_5c_6 + c_4s_6) \\ s_1(-c_{23}(c_4c_5c_6 + s_4s_6) + s_{23}s_5c_6) - c_1(-s_4c_5c_6 + c_4s_6) \\ -s_{23}(c_4c_5c_6 + s_4s_6) - c_{23}s_5c_6 \end{bmatrix} \quad (5.39)$$

$$a_p^0(q) = \begin{bmatrix} c_1(c_{23}c_4s_5 + s_{23}c_5) + s_1s_4s_5 \\ s_1(c_{23}c_4s_5 + s_{23}c_5) - c_1s_4s_5 \\ s_{23}c_4s_5 - c_{23}c_5 \end{bmatrix} \quad (5.40)$$

$$(5.41)$$

Table 5.6 lists the Denavit-Hartenberg's parameters of e.DO that are computed starting from the real dimensions of the structure (Figure 1.2).

	\mathbf{a}_i [mm]	$\boldsymbol{\alpha}_i$ [rad]	\mathbf{d}_i [mm]	$\boldsymbol{\theta}_i$ [rad]
1	0	$\pi/2$	0	θ_4
2	210.50	0	0	θ_5
3	0	$\pi/2$	0	θ_6
4	0	$-\pi/2$	268.00	θ_4
5	0	$\pi/2$	0	θ_5
6	0	0	174.50	θ_6

Table 5.6: Denavit-Hartenberg parameters of the whole manipulator

The inverse kinematic problem consists in computing the joints coordinates from the knowledge of the transformation matrix T. In a general case, a 9 non-linear equations system in six variables (q_1, \dots, q_6) should be solved but a solution in a closed form is difficult to be computed. It can be satisfactorily resolved with a recursive development, for example the Newton-Raphson.

From the dynamics point of view, it can be made reference to the Dynamic section in the background part. From the Lagrangian or the Newton-Euler method are derived the dynamic equations as:

$$H(q)\ddot{q} + C((q), \dot{q})\dot{q} + B\dot{q} + g(q) = \tau \quad (5.42)$$

It is possible to obtain the state equations of the system placing:

$$x = (x_1, x_2, x_3, x_4, \dots, x_{11}, x_{12}) = (q_1, \dot{q}_1, q_2, \dot{q}_2, \dots, q_6, \dot{q}_6) \quad (5.43)$$

The resulting system will be nonlinear with the input that can be the applied torques τ or the target velocities \dot{q} depending on the control structure. From the input torques τ it

can be possible, with the integration of the differential equations, to compute the joints coordinates.

The *Inverse dynamics*, instead, consists in providing the joints coordinates, together with the first and second derivatives (q, \dot{q}, \ddot{q}) and computing the applied torques to the joints, these torques need to solve the problem of the robot's control. The most significant issue is that, being the terms $H_{ij}\ddot{q}_j, h_{ijk}$ and g_i depending on the configuration q , the computational cost increases proportionally with the growth of the degrees of freedom n . It is possible to demonstrate that the number of the products is proportionally to n^4 , indeed the complexity is $O(n^4)$. For a robot with six degrees of freedom it was computed that the approach through the Lagrange equation requires each step 66271 products. From the 1980 have been introduced some algorithms that reduce the computationally cost (for example the Luh, Walker, Pul with $O(n)$). In our electronic structure, fortunately, thanks to decades of experience of the company is not a problem and the Inverse dynamic can be achieved without problems through Comau's algorithms.

5.5 Interface to analyze ORL data

In this phase of the thesis it is necessary to highlight some contextualizations with the project: this project of thesis was made with a Mac-OS system, even though the Comau's library is compatible only with a Linux system. To emulate the Linux machine it was created an operating system on Virtual Machine called Ubuntu, and after different trials the best performance was found with "Parallels Desktop" with the version of "Ubuntu Linux 16.04". In order to analyze the data given from the library ORL it was created a Server. With the protocol TCP/IP it was possible to create a communication between the Server (ORL based on the virtual machine) and the Client (represented by my Matlab program on the Mac-Os system). Figure 5.11 shows the communication flows through two programs in Matlab on Mac-Os(blue) and in C on Ubuntu(green).

The results from this process can be appreciated with a list of Inertia matrices that are generated changing the whole configuration of the manipulator in time. Now, the following step is to compare the diagonal inertia value given by the matrices and the inertia of the datasheets in order to check if the contribution, that must be included in the model, can be significant or less. Providing to ORL a sequential input signals that change all joints position, velocity and accelerations over time, the inertia diagonal terms given by ORL are plotted in Figure 5.12.

As it is evident, these terms are so small that their values divided by the square of the reduction ratio will be irrelevant with respect to the Inertia of the motor. The reliability of the data seems to be good, demonstrated by the sixth term that is null because there is no other connection to the downstream of the flange. Through the computations with Matlab, it can be highlighted that at the end the final significant result is that the inertia time-variant can be neglected:

$$J(q(t)) = J_{tot} + \frac{1}{k^2} \bar{H}_i(t) \approx J_{tot} \quad (5.44)$$

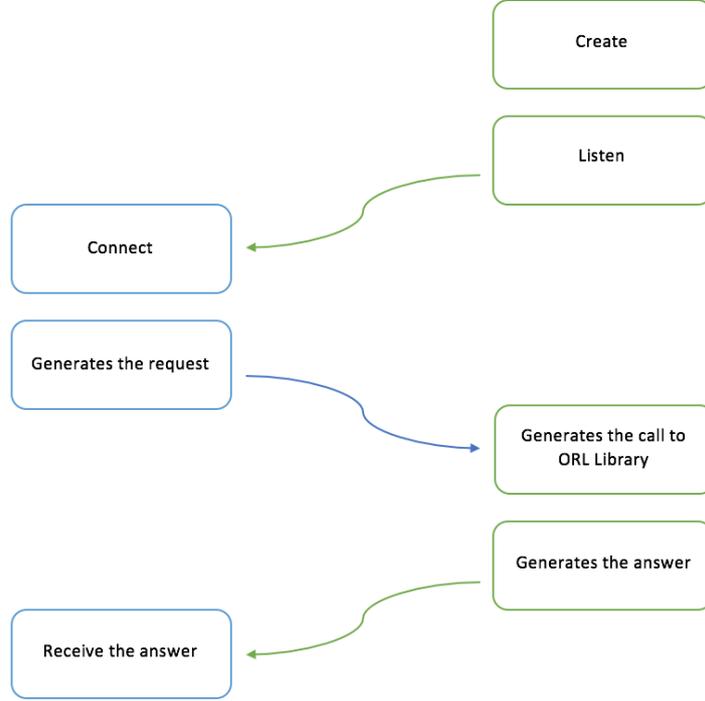


Figure 5.11: Communication flow diagram.

where \bar{H} is the tensor matrix:

$$\bar{H}(q(t)) = \begin{bmatrix} \bar{H}_{11} & & \\ & \ddots & \\ & & \bar{H}_{66} \end{bmatrix} \quad (5.45)$$

Table 5.7 lists the maximum values of the Inertia matrix when the joints are in the position of maximum gravity.

Max Inertia diagonal[Kgm ²]	Inertia on motor side[Kgm ²]
$\max \bar{H}_{11} = 3.924 \times 10^{-7}$	$J_1 = 9.060 \times 10^{-7} + \frac{3.924 \times 10^{-7}}{1232^2} \approx 9.060 \times 10^{-7}$
$\max \bar{H}_{22} = 3.833 \times 10^{-7}$	$J_2 = 9.060 \times 10^{-7} + \frac{3.833 \times 10^{-7}}{1232^2} \approx 9.060 \times 10^{-7}$
$\max \bar{H}_{33} = 1.373 \times 10^{-7}$	$J_3 = 9.060 \times 10^{-7} + \frac{1.373 \times 10^{-7}}{1232^2} \approx 9.060 \times 10^{-7}$
$\max \bar{H}_{44} = 0.220 \times 10^{-7}$	$J_4 = 1.282 \times 10^{-7} + \frac{0.220 \times 10^{-7}}{992^2} \approx 1.282 \times 10^{-7}$
$\max \bar{H}_{55} = 0.244 \times 10^{-7}$	$J_5 = 1.282 \times 10^{-7} + \frac{0.244 \times 10^{-7}}{992^2} \approx 1.282 \times 10^{-7}$
$\max \bar{H}_{66} = 0.000 \times 10^{-7}$	$J_6 = 1.282 \times 10^{-7} + \frac{0.000 \times 10^{-7}}{992^2} \approx 1.282 \times 10^{-7}$

Table 5.7: Maximum values of Inertia matrix

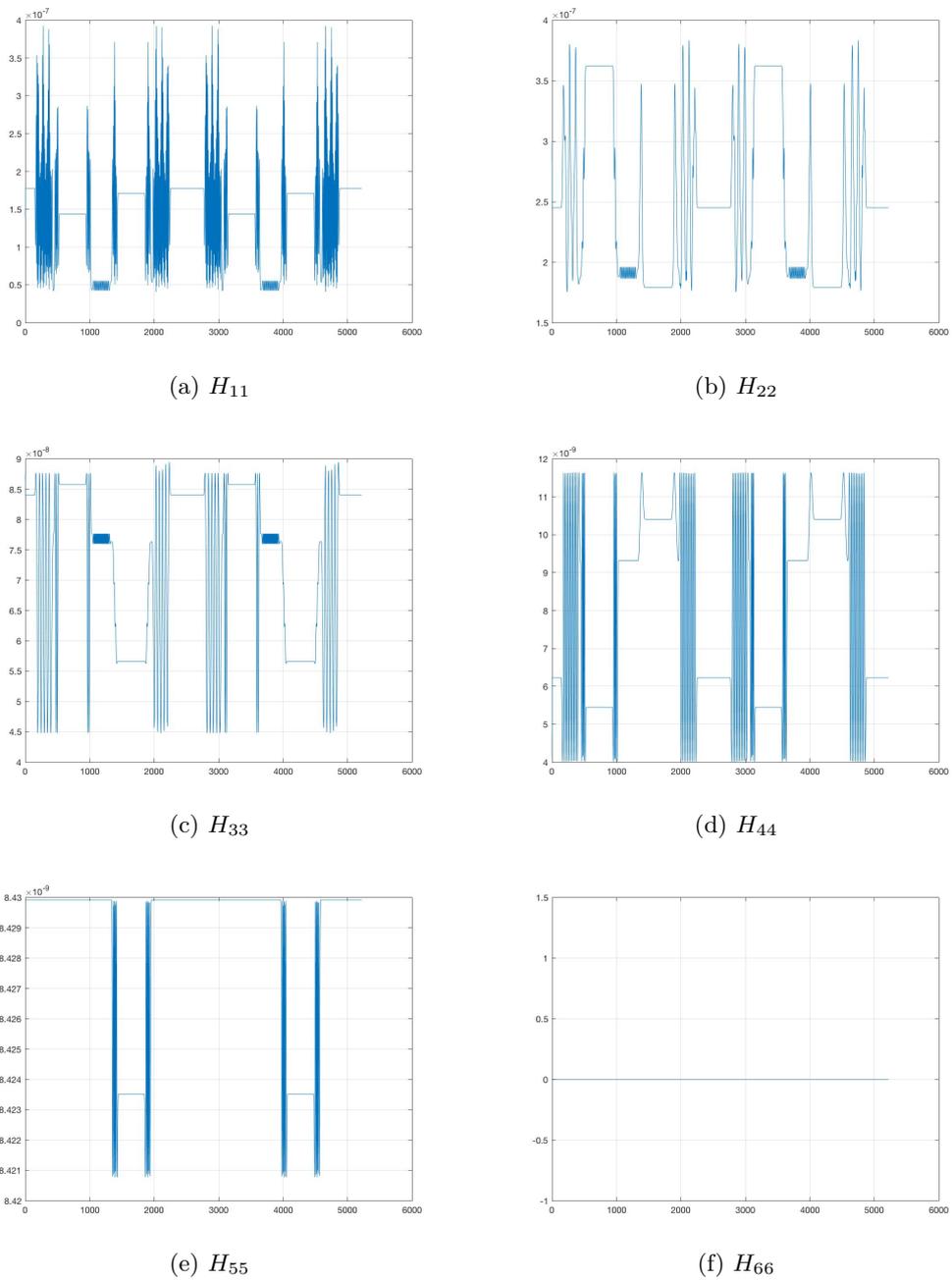


Figure 5.12: Diagonal term of the tensor matrix

5.6 Control structure

To compare the model with the real manipulator it is necessary to complete the system with the the control structure developed on the Raspberry. It was a nested control loop:

one loop for the position, one for the velocity and the last for the torque. Thanks to an Hall effect probe was measured the real current, that multiplied by the constant K_τ returns the torque measured. The control scheme is the most general in order to have the possibility of changing the structure of the project over time. The main elements of the loops are three:

The Controllers: three PID controllers.

The Feedforward: three possible feedforward terms that could improve the tracking.

The Saturators: three saturators that could limit the control variables in case of dangerous peaks.

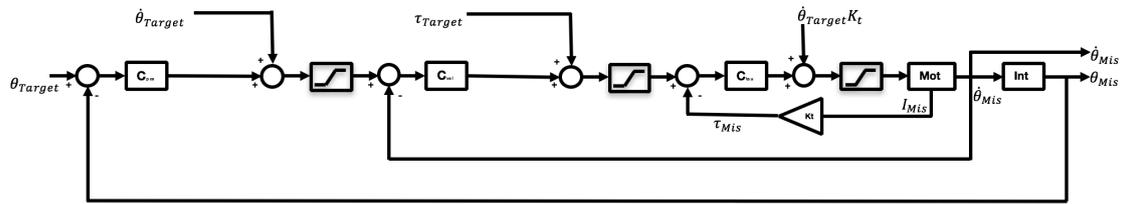


Figure 5.13: Control scheme.

The control system is looped every 1 ms, the trajectories are sent every 10 ms and in the Novalabs drive there is an Interpolator that generates ten consecutive samples interpolating the points with an Hermitian cubic spline. The same block diagram was replicated on Simulink, taking into account that the target and the real signals generated by the sensors are sampled every 10 ms.

5.6.1 PID controllers

The PID controller is the most common solution in Industrial process, this because it is a solution easy and cheap to practical control problems. The controllers with proportional, integrative and derivative actions have been used in the 1930s with pneumatic controllers, the development of them were applied in electronic mechanic and computer technology. The microprocessors increased their use from the 1980s. The PID control, as the name says, is the combination of three actions [3]:

Proportional: The characteristic of the response is proportional to the control error for small errors, the control signal will be proportional to the error:

$$u = K_p(y_{target} - y_{real}) = K_p e \quad (5.46)$$

where K_p is the controller gain.

Integrative: if the process variable with the only proportional gain K_p deviates from the set point, this drawback can be avoided by introducing a term proportional to the integral of the error:

$$u = K_i \int_0^t e(\tau) d\tau \quad (5.47)$$

This takes into account the history of the signal; one of the amazing property of this integral action is that if there is a steady state error e_0 with a constant control signal u_0 :

$$u_0 = K_i e_0 t \quad (5.48)$$

Since the signal u_0 is constant, e_0 must be null. This demonstrate that for a steady state signal if an integral action is present, the steady state error is always zero; this is also true for a PI controller:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau \quad (5.49)$$

Derivative: this term takes into account the future of the signal by a linear prediction of the error for T_d times:

$$u(t) = K_d \frac{de(t)}{dt} \quad (5.50)$$

The proportional terms of the Integrative and Derivative actions can be provided also depending on the time values, the relationship between them are given by:

$$T_i = \frac{K_p}{K_i} \quad T_d = \frac{K_d}{K_p} \quad (5.51)$$

where T_i is the integration time, and T_d the derivation time.

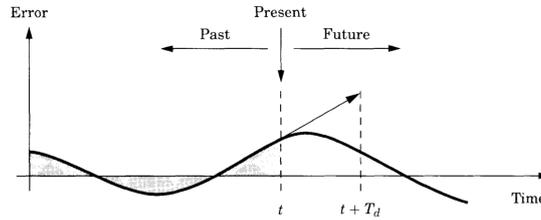


Figure 5.14: PID control action based on past, present and future [3].

5.7 Simulation of the final model

The final model is the resulting one represented in Figure 5.15.

In the DC Motor Block there is the same block seen before with the data of the datasheets; inside the control loop the PID values are set equal to the original ones adopted as reported in Table 5.8.

With these values a simulation test has been carried out choosing trajectories allowing the motion of all the joints. Figure 5.16 is significative, because it shows how the model follows the real system. It can be noted a little discrepancy due to different causes: in particular the backlash on the joints and the friction that have not been taken into consideration in the mechanical model of the system.

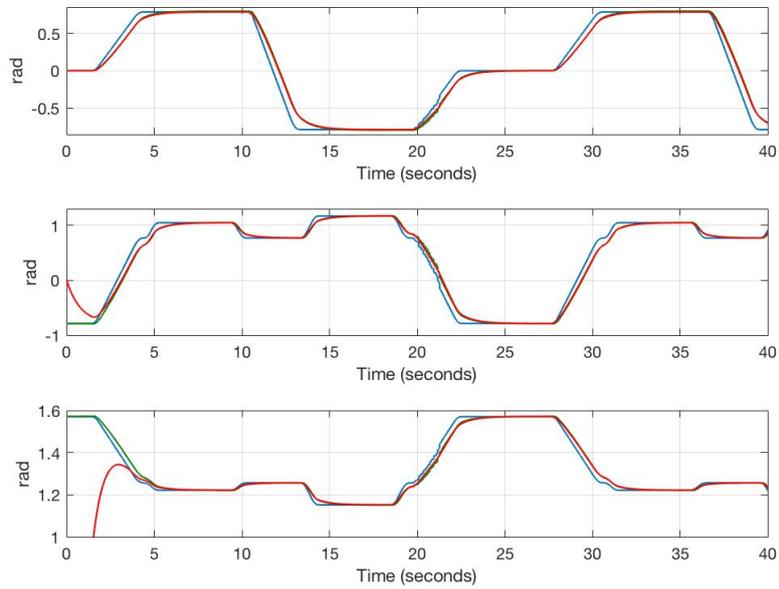
In order to improve the model, an estimate of the friction was made with Matlab. The pursuit of an approximated value of the friction coefficient was made through an iterative

	Arm Motor	Wrist Motor
	1	4
Position	$K_p = 1.3 \ t_i = 30000 \ t_d = 0 \ K_{ff} = 0.5$	$K_p = 2 \ t_i = 50000 \ t_d = 0 \ K_{ff} = 0.3$
Velocity	$K_p = 0.15 \ t_i = 2500 \ t_d = 0 \ K_{ff} = 0$	$K_p = 0.2 \ t_i = 2000 \ t_d = 0 \ K_{ff} = 0$
Torque	$K_p = 0.0015 \ t_i = 30 \ t_d = 0 \ K_{ff} = 0$	$K_p = 0.004 \ t_i = 50 \ t_d = 0 \ K_{ff} = 0$
	2	5
Position	$K_p = 1.3 \ t_i = 60000 \ t_d = 0 \ K_{ff} = 0.5$	$K_p = 2 \ t_i = 50000 \ t_d = 0 \ K_{ff} = 0.3$
Velocity	$K_p = 0.15 \ t_i = 1500 \ t_d = 0 \ K_{ff} = 0$	$K_p = 0.2 \ t_i = 2000 \ t_d = 0 \ K_{ff} = 0$
Torque	$K_p = 0.0015 \ t_i = 30 \ t_d = 0 \ K_{ff} = 0$	$K_p = 0.004 \ t_i = 50 \ t_d = 0 \ K_{ff} = 0$
	3	6
Position	$K_p = 1.3 \ t_i = 50000 \ t_d = 0 \ K_{ff} = 0.5$	$K_p = 2 \ t_i = 50000 \ t_d = 0 \ K_{ff} = 0.3$
Velocity	$K_p = 0.15 \ t_i = 1500 \ t_d = 0 \ K_{ff} = 0$	$K_p = 0.2 \ t_i = 2000 \ t_d = 0 \ K_{ff} = 0$
Torque	$K_p = 0.0015 \ t_i = 30 \ t_d = 0 \ K_{ff} = 0$	$K_p = 0.004 \ t_i = 50 \ t_d = 0 \ K_{ff} = 0$

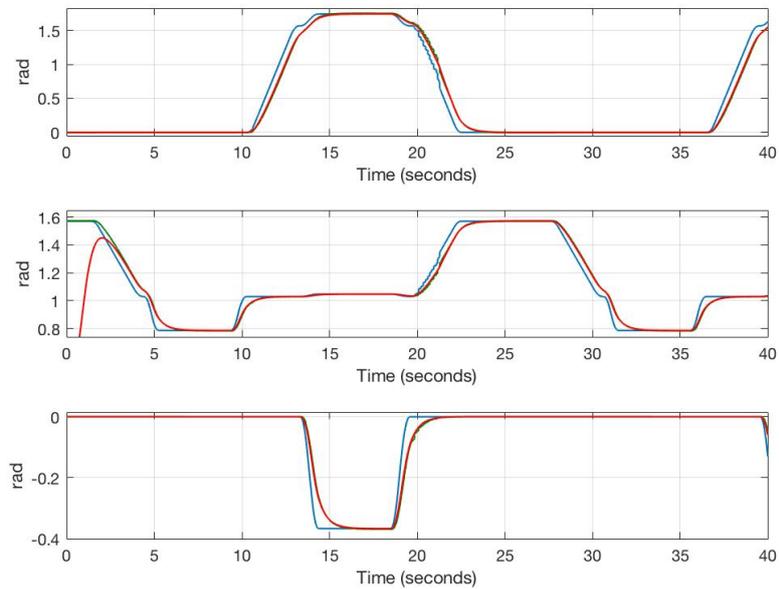
Table 5.8: PID values

procedure: the viscous friction affects the overall torque and consequently the current; it was performed a simulation where, varying β , the current, with respect to the real behavior, decreased until the desired trajectory was reached. (Figure 5.17)

A correct estimate was reached for $\beta \approx 10^{-4}$ by varying β in a range from 0 to 10^{-5} decreasing, each step, the friction value of 0.1.



(a) *Arm motors of the first three axes*



(b) *Wrist motors of the last three axes*

Figure 5.16: In Blue the target, in Green the real, in Red the simulated position of the six motors.

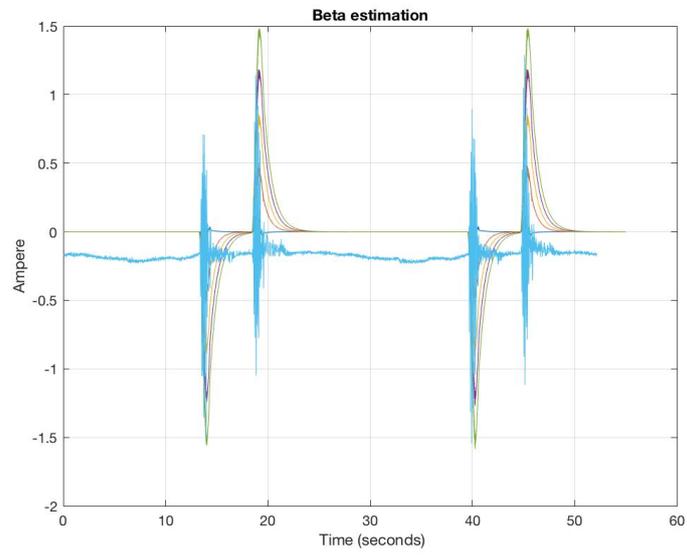


Figure 5.17: Current varying the friction values

Chapter 6

Control of the e.DO robot

The aim of this thesis is to improve the performance of the controller: the accuracy of the tool center point positioning and the capability to track the desired trajectory. The chances of control are multiple: to modify the control structure, to modify the controller gains inside the same control structure. The adopted solution is the result of a compromise between them, this option was very smart because the code written inside the Raspberry is not changed. Only the control system structure was modified. It was possible to choose how to connect the feedback loops and the feedforward terms of the control system by varying the input parameters. All controller PID gains can be modified by changing values. These software features allowed us to quickly test ideas during the project.

The input terms in the software to modify the control loops are the following:

1. Input variables:

- Target Position
- Target Velocity
- Target Torque

2. PID values:

- Proportional
- Integrative
- Derivative

for the all the control loops.

3. Feedforward gains:

- Position
- Velocity
- Torque

4. Saturations values

Only the final and used controller will reported with details in the thesis, the best one in term of satisfactory behavior and requests met.

6.1 Practical considerations

It is useful to make some practical considerations: in a real application we move away from the theoretical model because there exist some "parasitic dynamics". There are some nonlinear relations that are approximated in the model with purely proportional term. For example the armature inductance L_a is very small and it is considered null, in this way the electric pole in $s = R_a/L_a$ will be $-\infty$. There are many cases of mismatch between the model and the real system as:

Saturating actuators The presence of the saturations block on the feedback loop is modeled including in the loop a non linear block with the following transfer function:

$$y(t) = \begin{cases} s_{max} & \text{if } u(t) > u_{max} \\ ku(t) & \text{if } u_{min} \leq u(t) \leq u_{max} \\ s_{min} & \text{if } u(t) < u_{min} \end{cases} \quad (6.1)$$

Elasticity in the mechanical structure The presence of the gearboxes and other transmission systems include a structural elasticity and consequent resonances deleterious for the kinematic chain. The resonance effects can be attenuated designing the bandwidth of the controlled system. Knowing that this must not be too low, which would made the system response rather low, but not too high, in order to not excite the neglected elastic dynamics. A trade off between these two behaviors must be achieved. It is advisable, in order to not excite the resonance structural frequency, that the bandwidth is not so high. If for simplicity it is equal to the natural frequency ω_n , it must not be greater than $0.5\omega_r$ where ω_r is the resonance frequency of the manipulator. This frequency depends on the used material and it can be obtained from a simple dynamic model of the elastic system, not damped and not forced by external torques:

$$J_t \ddot{\theta}_m(t) + k_t \theta_m(t) = 0 \quad (6.2)$$

where k_t is the elastic constant of the joint, the natural frequency of the system will be:

$$\omega_r = \sqrt{\frac{k_t}{J_t}} \quad (6.3)$$

Non linear frictions in the joints The Coulomb frictions and the stiction component made non linear and discontinuous in the origin the relation between joint velocity and resistance force. It follows that the relation between the applied force on the joint and the resulting force has a dead bandwidth. These effects make the position control inaccurate specially for low velocities where it can be observed the establishment of limit cycles. (*Stick and Slip effect*)

6.2 Anti wind-up

The presence of non-linear elements in the control loop, in particular the saturation of the actuators, and the integral action of the control determine a nonlinear phenomenon called

wind-up or *integral capacity* that deteriorates the performance of the control system. It is analyzed the general problem with a **PI** controller in a feedback control loop as shown in Figure 6.1

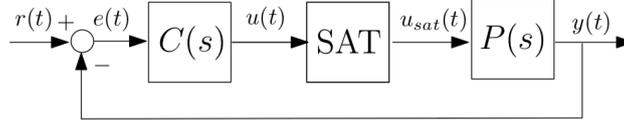


Figure 6.1: Control diagram with saturation

When the error $e(t)$ keeps its sign unchanged, positive for example, for a time period the integral action of the controller integrates the error even though the input variable of the process saturates. When the error becomes negative, changing sign, it is necessary to wait that the output of the controller assumes again values inside the range (discharging the integral action). The wind-up phenomenon is due to the controller dynamics that are not affected by the presence of the limitations on the input variable. There exist different *antiwind-up* techniques dealing with this problem. In this project an antiwind-up technique is adopted, it is implemented via software inside the control loop, which allow not to cross the phenomenon. The strategy actuated in the control software is the following: when the value $u(t)$ is greater or lower than the boundaries, the value in input to the motor $u_{sat}(t)$ will be the saturated one chosen a-priori, but the Integral value for the following step is frozen. In this way the integrator of the PID controller is stopped only if it is violated the saturation bound. The integral term can be expressed as $(\sum_i^{k-1} e_i + e_k)$ in this case what it is made is to delete the last term of the saturation e_k , keeping the summation of the previous terms till the value does not return in the range limited by the saturation boundaries.

6.3 Preliminary analysis

The intent is to find some a-priori constraints in order to design the controllers and achieve a-posteriori requirements. The project is developed in the frequency domain; there are two kinds of diagrams that allow to see if the requirements are satisfied, in particular they are very useful to analyze the stability of the system.

Being the transfer function:

$$W(j\omega) = \frac{G(j\omega)}{1 + G(j\omega)} \quad (6.4)$$

Nyquist: it is a representation on polar coordinates with Imaginary and Real part of the transfer function when varying the angular frequency ω_n , it is used only one reference diagram to take into account the magnitude and frequency that are figured into two diagrams on Bode.

The direct chain can be written as:

$$G(j\omega) = Re(\omega) + jIm(\omega) \quad (6.5)$$

using the polar representation of a complex number it can be written:

$$W(j\omega) = M(\omega)e^{j\phi(\omega)} = \frac{Re(\omega) + jIm(\omega)}{1 + Re(\omega) + jIm(\omega)} \quad (6.6)$$

from this relation it can be demonstrated that:

$$\begin{cases} M(\omega) = \frac{\sqrt{Re(\omega)^2 + Im(\omega)^2}}{\sqrt{[1+Re(\omega)]^2 + Im(\omega)^2}} = f_1[Re(\omega), Im(\omega)] \\ \phi(\omega) = \angle[Re(\omega) + jIm(\omega)] - \angle[1 + Re(\omega) + jIm(\omega)] = f_2[Re(\omega), Im(\omega)] \end{cases} \quad (6.7)$$

In this chart each point with coordinates $Re(\omega) + jIm(\omega)$ coincide with a couple M, ϕ read or interpolated by the nearest values of the module and constant phase places.

Nichols: it is a representation with magnitude and phase of the transfer function when varying the angular frequency ω_n , it is a stricter representation than the Bode diagrams. The direct chain can be written as:

$$G(j\omega) = A(\omega)e^{j\alpha(\omega)} \quad (6.8)$$

the whole transfer function becomes:

$$W(j\omega) = M(\omega)e^{j\phi(\omega)} = \frac{G(j\omega)}{1 + G(j\omega)} = \frac{A(\omega)e^{j\alpha(\omega)}}{1 + A(\omega)e^{j\alpha(\omega)}} \quad (6.9)$$

where:

$$\begin{cases} M(\omega) = \frac{A(\omega)}{|1+A(\omega)e^{j\alpha(\omega)}|} = F_1[A(\omega), \alpha(\omega)] \\ \phi(\omega) = \alpha(\omega) - \angle[1 + A(\omega)e^{j\alpha(\omega)}] = F_2[A(\omega), \alpha(\omega)] \end{cases} \quad (6.10)$$

The previous performance with the original control are highlighted in Table 6.1 where are listed the transitory characteristic of a step response.

Original Control		
	Arm Motor	Wrist motor
Rise Time	2.8914 s	1.2971 s
SettlingTime	5.7128 s	3.2893 s
SettlingMin	0.9039°	0.9007°
SettlingMax	1.0155°	1.0017°
Overshoot	1.1094%	0.0969%
Undershoot	0.0000%	0.0000%
Peak	1.0155°	1.0017°
PeakTime	11.0561 s	7.4053 s

Table 6.1: Step response analysis of the original control system

The requirements to improve the system are relative to the following issues:

- The Rise Time is the time required for the response to rise from 0% to 100% of its final value. In our system it is necessary to decrease the Rise Time to make the system more reactive.
- The maximum Overshoot \hat{s} is defined as

$$\hat{s} = \frac{y_{max} - y_{\infty}}{y_{\infty}} \quad (6.11)$$

In our system it is needed to delete the Overshoots.

- The Settling Time $\pm\alpha * 100\%$, $t_{s,\alpha\%}$ is the amount of time required for the step response to reach and stay within a range about $\pm\alpha * 100\%$ of the steady-state value y_{∞} . In our system it is necessary to decrease the Settling Time.

In addition:

- The Synchronicity of the whole system must be kept.
- The Stability of the whole system must be guaranteed.

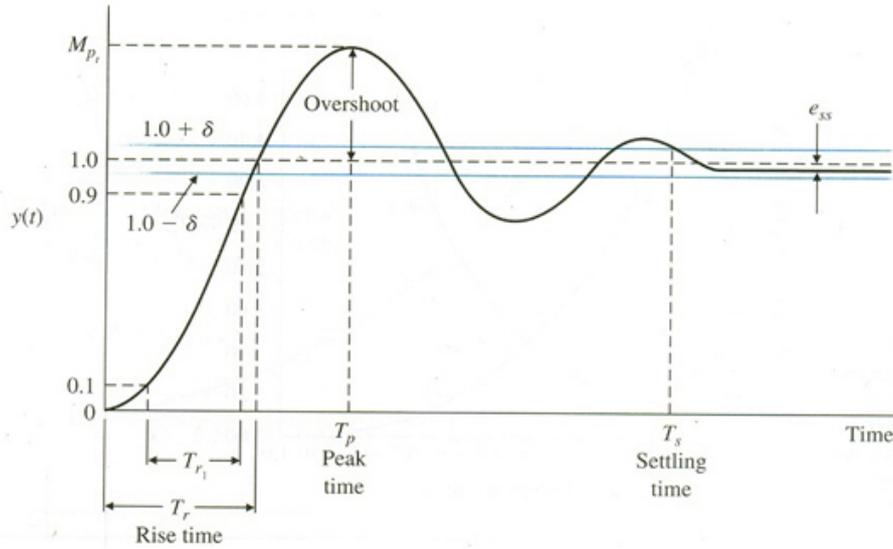


Figure 6.2: Time requirements: Overshoot, Rise Time and Settling Time of a prototype of second order function

The given time requirements will be translated in the frequency domain, in particular on the Nichols chart. The strategy to control the system is to compare the system as a prototype of 2nd order of the form:

$$T(s) = \frac{1}{1 + \frac{2\zeta}{\omega_n}s + \frac{s^2}{\omega_n^2}} \quad (6.12)$$

The unit step response is given by:

$$y(t) = 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1 - \zeta^2} \sin[\omega_n t + \tan^{-1} \frac{\sqrt{1 - \zeta^2}}{\zeta}]}, t \geq 0 \quad (6.13)$$

If $0 < \zeta < 1$ (complex poles): the system is **underdamped**.

If $\zeta = 1$ (equal real poles): the system is **critically damped**.

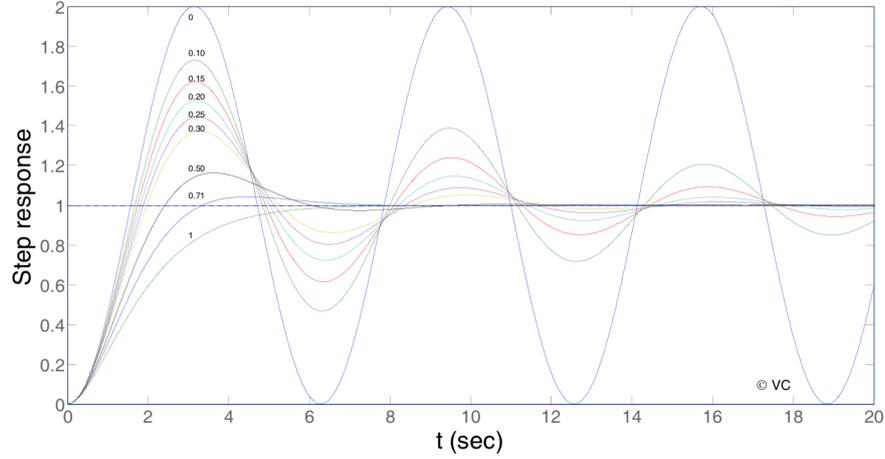


Figure 6.3: Step response of prototype 2nd order system

It can be noted that an underdamped system with $0.5 < \zeta < 0.8$ gets close to the final value more quickly than a critically damped system with a minimum overshoot.

The main transient response values are \hat{s}, t_r and $t_{s,\alpha\%}$. They can be expressed as function of the damping factor ζ and the natural frequency ω_n :

$$\hat{s} = e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}} \quad (6.14)$$

$$t_r = \frac{1}{\omega_n \sqrt{1-\zeta^2}} (\pi - \arccos(\zeta)) \quad (6.15)$$

$$t_{s,\alpha\%} = -\frac{\ln \alpha}{\omega_n \zeta} \quad (6.16)$$

$$(6.17)$$

To translate the time requirements in frequency domain three frequency responses can be defined:

- the frequency response $L(j\omega)$ of the loop function $L(s)$
- the frequency response $S(j\omega)$ of the sensitivity function $S(s) = [1 + L(s)]^{-1}$
- the frequency response $T(j\omega)$ of the complementary sensitivity function $T(s) = 1 - S(s)$

The maximum peak of these functions depends on the damping factor, while the cross-over

frequency depends on the natural frequency according to the following relationships:

$$T_p = \frac{1}{2\zeta\sqrt{1-\zeta^2}} \quad (6.18)$$

$$S_p = \frac{2\zeta\sqrt{2+4\zeta^2+2\sqrt{1+8\zeta^2}}}{\sqrt{1+8\zeta^2+4\zeta^2-1}} \quad (6.19)$$

$$w_c = \frac{(\pi - \arccos(\zeta))\sqrt{\sqrt{1+4\zeta^4}-2\zeta^2}}{\sqrt{1-\zeta^2}/t_r} \quad (6.20)$$

The values of the resonance peaks T_p and S_p of the sensitivity function and its complementary obtained via the \hat{s} requirements can be used to draw the corresponding constant magnitude loci on the Nichols plane. These loci can be considered as constraints which should not be violated by the Nichols plot of the frequency response of the loop function $L(j\omega)$

6.4 Design

Once modeled the system, the following step was to find a controller that improves the previous performance keeping the stability and the synchronicity of the whole manipulator. The developed control is structured with two nested loops as seen in the background section and only two controllers are designed. The inputs of the chain are the desired position and the resistive torque that is considered as a disturbance on the direct chain of the system.

The adopted the structure is sketched in Figure 6.4.

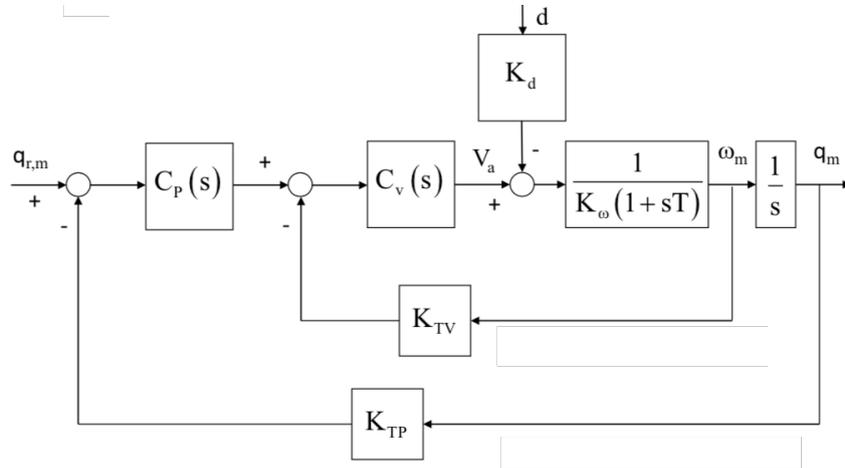


Figure 6.4: Control diagram without feedforward

The Controllers $C_p(s)$ and $C_v(s)$ are designed so to reduce the tracking error. The motor transfer functions are:

$$G_{arm}(s) = \frac{8.3511 \times 10^7}{(s + 2241)(s + 375.5)} \quad (6.21)$$

$$G_{wrist}(s) = \frac{3.5103 \times 10^8}{(s + 1.998 \times 10^4)(s + 149.8)} \quad (6.22)$$

for the Arm and the Wrist motors respectively.

A first analysis of the system is based on the time constants and the paced frequencies:

$$\tau_m = \frac{JR_a}{K_\omega K_\tau} = \begin{cases} \text{Arm motors} \rightarrow 0.0032 \text{ s} \\ \text{Wrist motors} \rightarrow 0.0067 \text{ s} \end{cases} \quad \tau_a = \frac{L_a}{R_a} = \begin{cases} \text{Arm motors} \rightarrow 3.8388 \times 10^{-4} \text{ s} \\ \text{Wrist motors} \rightarrow 4.96691 \times 10^{-5} \text{ s} \end{cases} \quad (6.23)$$

To eliminate the mechanical pole of the joint, as seen in the background in the cascade control section for the controllers written in (4.6), the main idea is to place the controller pole in the same frequency. In the real system it will not be exactly the same frequency but it will work reasonably because the zero and the pole of the whole structure are very close.

This technique is called pole placement, these frequencies are, therefore, included in the Arm and Wrist controllers of the velocity loops as follows:

$$C_{v_{arm}} = \omega_{n_b}(2\zeta_b)K_\omega \frac{1 + s\frac{1}{375.5}}{s} \quad (6.24)$$

$$C_{v_{wrist}} = \omega_{n_s}(2\zeta_s)K_\omega \frac{1 + s\frac{1}{149.8}}{s} \quad (6.25)$$

The positions controllers are designed to exploit the system as a second order type:

$$C_{p_{arm}} = \frac{\omega_{n_b}}{2\zeta_b} \quad (6.26)$$

$$C_{p_{wrist}} = \frac{\omega_{n_s}}{2\zeta_s} \quad (6.27)$$

The step response is the most difficult input to follow, it is important because allows to study the behavior of the system in the transient between two stable points. It highlights the time requirements of the transient, and verifies whether the system truly meets our expectations. The only two parameters that can be tuned with this control structure, imposing the pole of the velocity loop, are the damping factor (ζ) and the natural frequency (ω_n).

Now the idea is to achieve the desired behavior monitoring the stability and the synchronicity of the whole system.

Obviously the strategy is to:

1. Enlarge the bandwidth of the system.
2. Reduce the rise time making the system more prompt.
3. Do not create an overshoot.

Possible negative effects could occur:

1. Decreasing too much the damping factor, the settling time could be too high.

2. Increasing the natural frequency, enlarging the bandwidth, the current of the system increases, and dynamic components at high frequency, not included in the model, could be excited.
3. Closing the bandwidth the system becomes slow, so a proper tuning of the damping factor is required.

The requirements are then set as:

- $\hat{s} = 0.01$;

for the velocity loop, and

- $\hat{s} = 0.001$;

for the position loop.

The stability of the control loop is monitored at each step of the design highlighting the phase and gain margin in the Nichols diagram. As it is possible to see in Figure 6.5 the Phase and Gain margins can be directly found and it is possible with a graphical analysis to discuss the stability of the system.

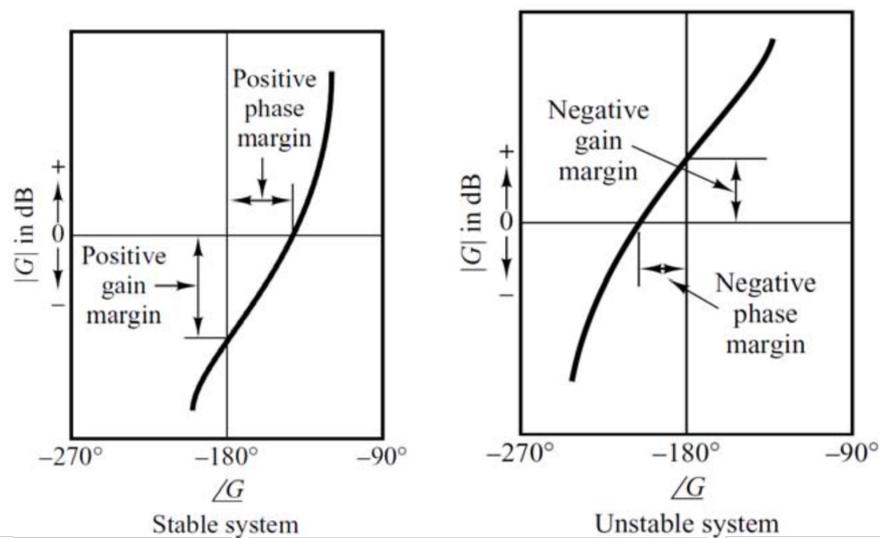


Figure 6.5: Stability margins in the Nichols Chart

The first approach is to reduce the overshoot generated by the previous controller. To ensure the desired behavior, being the system a prototype of a second order, the damping factor is set equal to one. This choice as seen in the previous paragraph guarantees that the system will be underdamped and the objective will be achieved. The range of the possible control choices with this configuration can be seen in Figure 6.6, that shows two simulations in which the natural frequency is changed in a range from 3 to 60 for the velocity loops and the position loops of the two kinds of motors.

The constraint on the overshoot is achieved because the plots do not exceed the constant magnitude locis. The analysis of the above figures is resumed in Table 6.2, which reports the stability margins in the worst cases for the two motors.

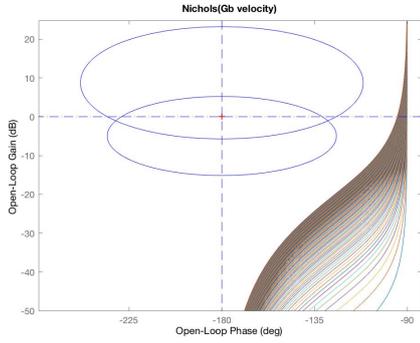
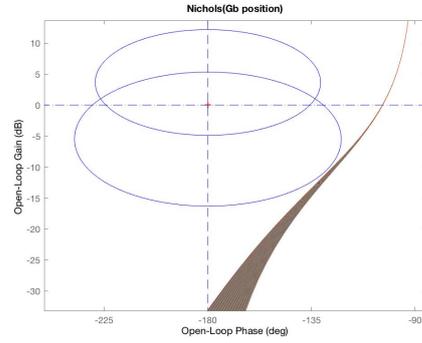
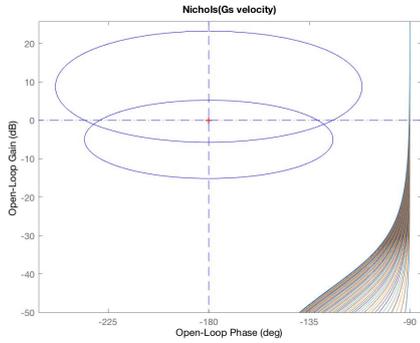
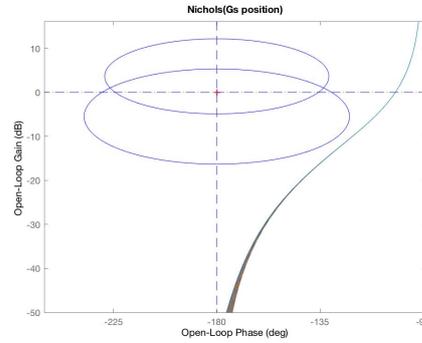
(a) *Velocity loop of the big motor*(b) *Velocity loop of the small motor*(c) *Position loop of the big motor*(d) *Position loop of the small motor*

Figure 6.6: Nichols diagrams of the velocity and position loops of both motors.

In order to take into account the synchronicity of the whole manipulator, it is necessary that all the joints will finish their task at the same time. Being the second joint, the slowest one because it is affected by the biggest gravity component due to the contribute of the other four joints that follow in the chain and an eventual gripper. In the worst case the rise time of the last joint, that probably is the fastest, will must be greater or equal to the slowest joint, the second one. To achieve this requirement, knowing that the rise time depends on the variable ω_n and ζ , since for the damping factor is not a problem because is unitary for both for the overshoot specification, the natural frequency is placed equal again for both motors imposing in this way the poles of the system at the same frequency. The maximum target is fixed, the further step is to simulate the system with the maximum possible frequency. To obtain the PID parameters from the frequency transfer functions, a Matlab tool is used to translate the transfer function in a parallel PID form as:

$$C = K_p + \frac{K_i}{s} + \frac{K_d s}{T_f s + 1} \quad (6.28)$$

	Velocity	Position
Arm motor		
Phase margin	85.1°	75.8°
Delay margin	0.00772 <i>sec</i>	0.0271 <i>sec</i>
At frequency	1924 <i>rad/s</i>	48.7 <i>rad/s</i>
Closed loop stable?	<i>Yes</i>	<i>Yes</i>
Wrist motor		
Phase margin	89.7°	77.9°
Delay margin	0.0137 <i>sec</i>	0.0556 <i>sec</i>
At frequency	114 <i>rad/s</i>	24.5 <i>rad/s</i>
Closed loop stable?	<i>Yes</i>	<i>Yes</i>

Table 6.2: Nichols margins of the velocity and position loops of both motors.

Starting with a natural frequency equal to $\omega_n = 50$ rad/s the velocity controllers become:

$$C_{v_{(arm)}} = 2.5912 \times 10^{-3} \frac{(s + 375.5)}{s} \rightarrow K_p = 3.11 \times 10^{-3} \quad K_i = 0.973 \quad (6.29)$$

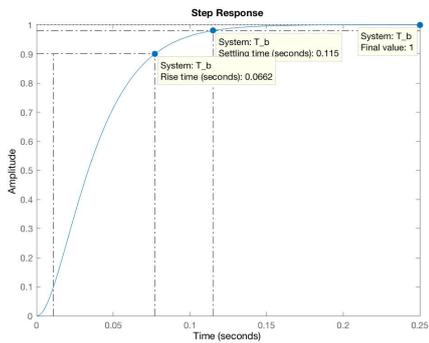
$$C_{v_{(wrist)}} = 6.4953 \times 10^{-3} \frac{(s + 149.8)}{s} \rightarrow K_p = 6.99 \times 10^{-3} \quad K_i = 0.973 \quad (6.30)$$

and the position controllers become:

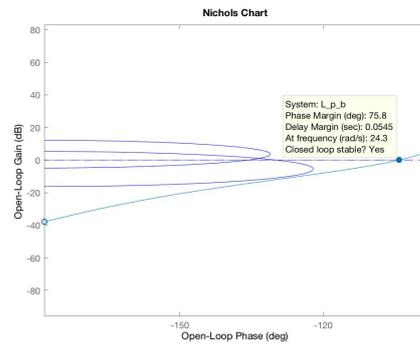
$$C_{p_{(arm)}} = 25 \rightarrow K_p = 25 \quad (6.31)$$

$$C_{p_{(wrist)}} = 25 \rightarrow K_p = 25 \quad (6.32)$$

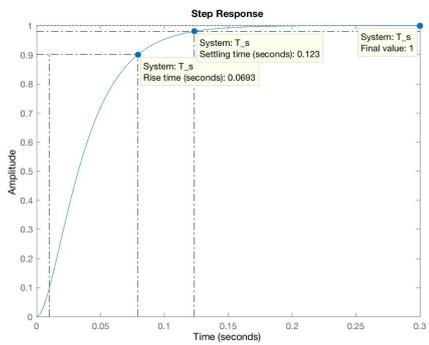
The step response of the system and the Nichols plot of the whole system controlled with a target position are reported in Figure 6.7; it can be seen that the requirements are achieved and the rise time is very lower than the first one with the previous control.



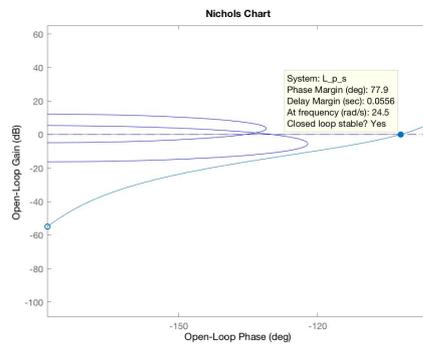
(a) Step response Arm motor



(b) Nichols diagram for Arm motor



(c) Step response Wrist motor



(d) Nichols diagram for Wrist motor

Figure 6.7: Step response and Nichols diagrams of the motors with $\omega_n = 50$ rad/s and $\zeta = 1$.

Chapter 7

Results

7.1 Performance achieved

A simulation with the development environment Simulink was created giving as inputs the real trajectories tested on e.DO, in order to excite all the behaviors of the motors of the joints. The results can be observed in Figure 7.1. These pictures highlight how the situation is improved. The new controller has the best tracking error that it is achieved in this thesis project.

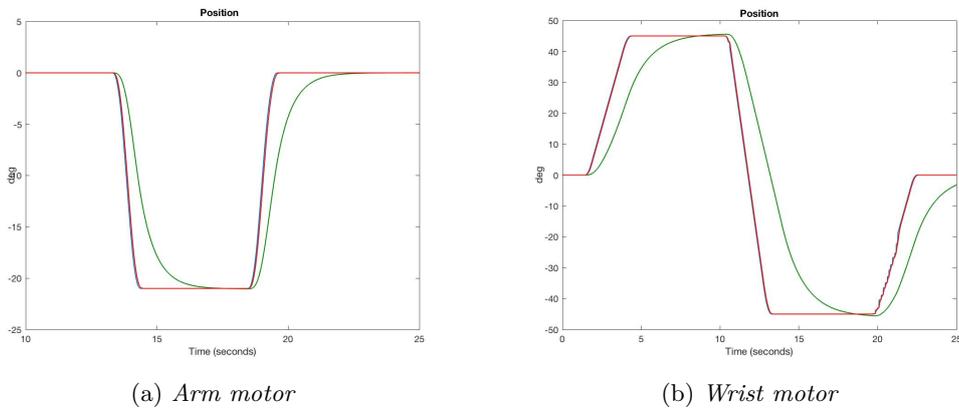


Figure 7.1: In blue the desired position, in green the position of the previous controller, in red the position of the new controller with $\omega_n = 50$ rad/s and $\zeta = 1$.

The performance in the simulation is great, all the previous desired objectives are achieved :

- Stability
- Better rise time
- Better settling time

- No overshoot

The analysis on the synchronicity of the whole manipulator is still missing, which can only be noticed with the machine time of the real joints and will be discussed later. Now the critical phase is to implement the new control scheme on the manipulator and check how it works in the real environment. Through a wi-fi it was established a connection with the e.DO robot, by terminal it was sent the inputs configuration to manage the control structure. The main changes compared to the original scheme were the modified gains of the controllers and the absence of the feedforward terms and the current loop. A program generated through the Tablet was created to let the robot move in some predefined positions, reaching a wide range of working space. Before the experimental test with the robot, the data acquisition program was started; this program through the USB protocol creates a text file where every 10 ms the position, velocity and current values are stored. Real and target data are collected and via Wi-fi it is possible to download these contents linking on the same connection of the Raspberry. Once the data are unpacked and processed by Matlab, a new program was written to print all the data for a subsequent analysis. Through this program it has been possible to align the time line of both the real and desired signals, in order to evaluate the control performances.

Figure 7.2 shows how the variables change over time for the first and the last joints, one per type of motor. In particular it can be noted how the disturbances affect the current and the velocity. This problem will be treated in the following chapter.

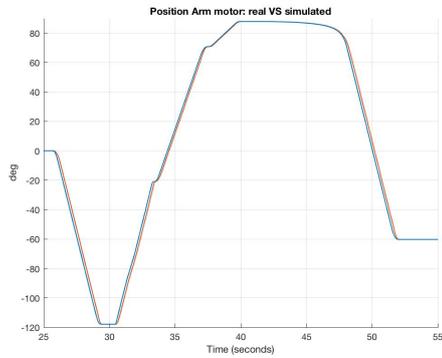
The objective is achieved under the control domain and the improvement from the original control can be appreciated also in Table 7.1, where are listed the transient performances achieved. The issue is that the best motor control of the joints in this application is not really satisfactory. This because when a control is applied on the development environment there are a lot of other requirements that can be involved in the system. In this way, two schools of thought branch off:

- Only the parameters that we can analyze are controllable: since the only block that is controllable is the motor, the best controller is the one that follows slavishly the target variables.
- The parameters that can be controlled have to be consistent with the behavior of the whole system: it does not matter if the motor is controlled in the best possible way, it needs that the behavior of the whole structure will be feasible with the environment requirements.

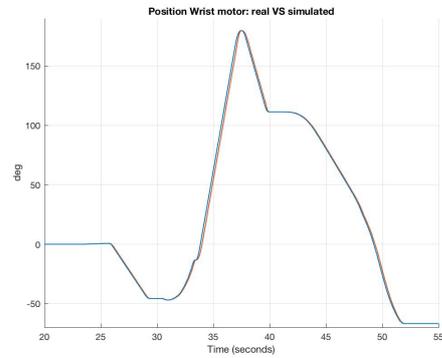
This means that, for the control point of view, the goal is not only to find the best way to follow the desired trajectory but to find the best trade-off among a tolerable tracking error, a good current limitation that means lower auditory noises, a good movement depending on the specific working space, a prompt system in the environment where the mechanical structure is not so rigid will highlight the defects.

In this practical case, the structure composed by plastic staff and the plays in the gear boxes does not allow a very prompt system. This means that the control has to be modified reducing the promptness of the system until a good execution of the movement is reached keeping the synchronicity of the six axes. Another practical issue is the auditory noise, in order to reduce the noise it is needed that the current absorbed by the motors decreases. With a lower current, especially for the arm motors, the motion will result silent. All these

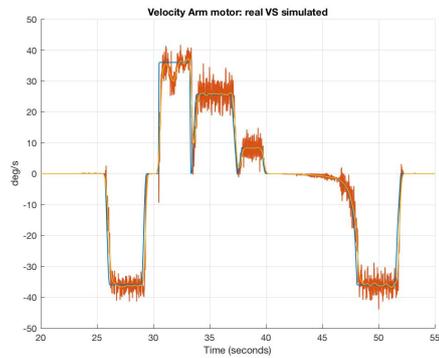
7.1 – Performance achieved



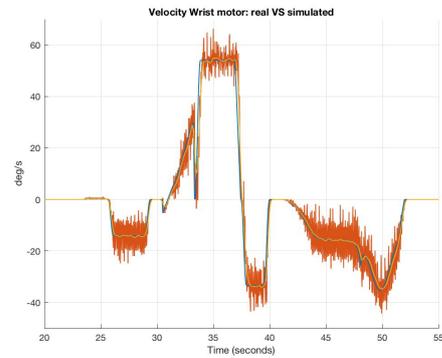
(a) Position of the Arm motor



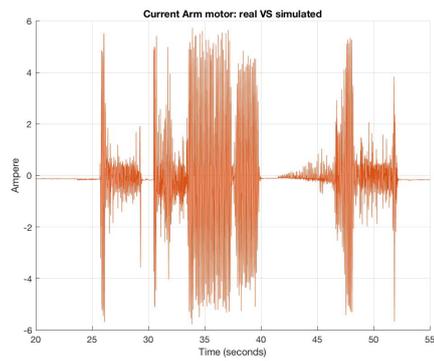
(b) Position of the Wrist motor



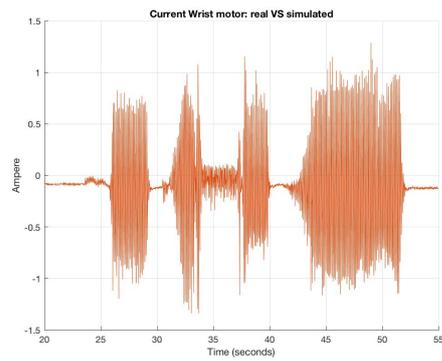
(c) Velocity of the Arm motor



(d) Velocity of the Wrist motor



(e) Current of the Arm motor



(f) Current of the Wrist motor

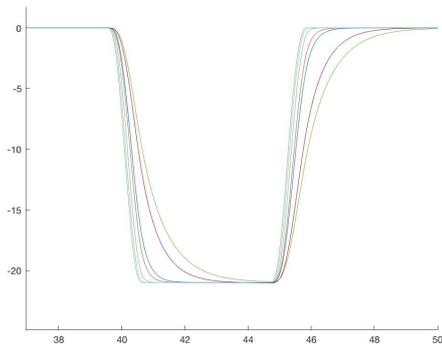
Figure 7.2: Manipulator desired trajectories in blue, the real ones in red and the average of the noise of the velocity in yellow for the controller with $\omega_n = 50$ rad/s and $\zeta = 1$.

Original Control $\omega_n = 50$ rad/s		
Arm Motor		
Rise Time	2.8914 s	0.0662 s
SettlingTime	5.7128 s	0.1151 s
SettlingMin	0.9039°	0.9009°
SettlingMax	1.0155°	1.0000°
Overshoot	1.1094%	0.0000%
Undershoot	0.0000%	0.0000%
Peak	1.0155°	1.0000°
PeakTime	11.0561 s	0.2480 s
Wrist motor		
Rise Time	1.2971 s	0.0693 s
SettlingTime	3.2893 s	0.1233 s
SettlingMin	0.9007°	0.9006°
SettlingMax	1.0017°	1.0000°
Overshoot	0.0969%	0.0000%
Undershoot	0.0000%	0.0000%
Peak	1.0017°	1.0000°
PeakTime	7.4053 s	0.3193 s

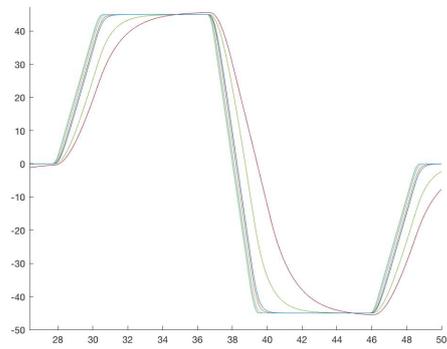
Table 7.1: Step response analysis of the robot with original and the new control

considerations bring to a mathematical solution: since the current of the motor depends on the torque that must be applied by the system to move the motor in a specific position, better is the promptness of the system higher will be the current that the control have to fed to the motors, higher the noises developed, more highlighted the negative effects due to the mechanical structure. The objective becomes to find the best compromise in terms of performance and "esthetic" outcome (noise and motion achievement). In order to get these objectives, the natural frequency of the controller is reduced, and some experimental tests are made till a good enough result is reached. Then the controllers designed have been implemented on the real robot, and the esthetic and auditory features are checked and verified.

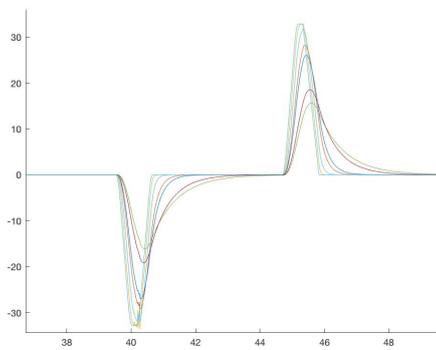
Figure 7.3 shows the real signals of the robot for decreasing values of the natural frequency ω_n , compared with those obtained by the previous controller starting choice of $\omega_n = 50$ rad/s. The result is evident on the current plot where the decreasing of the amplitude varies from 3.55×10^{-2} A to 5.6×10^{-3} A for the Arm motors, and from 2.06 A to 0.95 A for the Wrist motors.



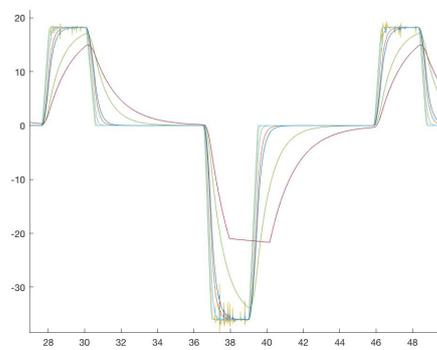
(a) *Position of the Arm motor*



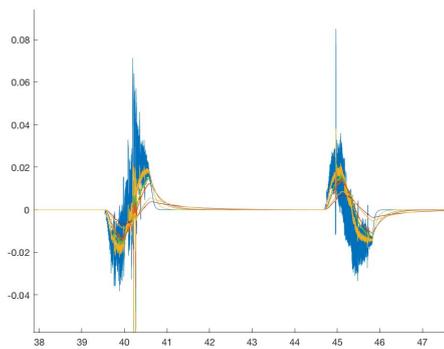
(b) *Position of the Wrist motor*



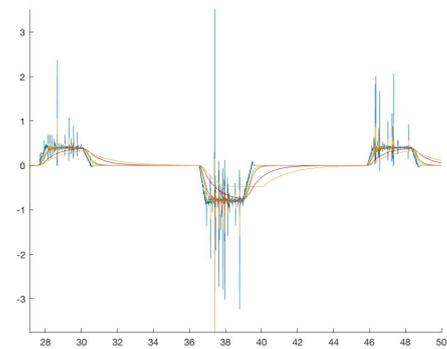
(c) *Velocity of the Arm motor*



(d) *Velocity of the Wrist motor*



(e) *Current of the Arm motor*



(f) *Current of the Wrist motor*

Figure 7.3: Manipulator data with the controller with $\omega_n = 50$ rad/s, $\omega_n = 20$ rad/s, $\omega_n = 10$ rad/s, $\omega_n = 5$ rad/s, $\omega_n = 2.5$ rad/s and $\zeta = 1$ and the simulation of real system.

7.2 Final comparison

The best trade-off is finally found for a natural frequency $\omega_n = 7.5 \text{ rad/s}$. With this configuration all the requirements are achieved, esthetically and in term of noise. Although the rise time is decreased the control keeps the previous characteristic without overshoot improving the tracking error with respect to the first one controller as can be seen in Figure 7.4. Table 7.2 highlights how the system is improved. The transient performances are listed for three different situations of the project: the original control system, the new one with $\omega_n = 7.5 \text{ rad/s}$ $\zeta = 1$ and in the control system with $\omega_n = 50 \text{ rad/s}$ $\zeta = 1$, in order to see the difference with the best that is achieved in this project of thesis.

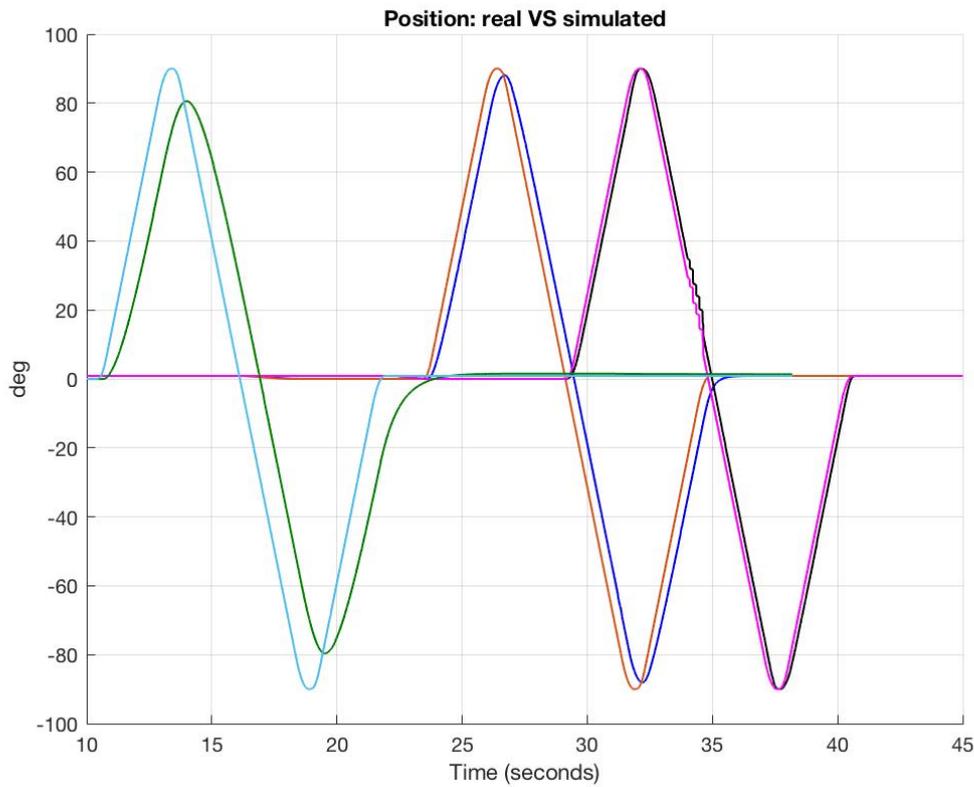


Figure 7.4: Light blue and Green are the Ideal and Real position with the previous control, Red and Blue are the Ideal and Real position with the adopted control, Magenta and Black are the Ideal and Real position with the best control.

By inspecting the delay due to the time machine on each joint, as reported in Table 7.3, the requirements on the synchronicity of the whole structure is achieved: the last joint has a time machine greater or equal then all the others, in particular of the second that is the slower.

	Original Control	$\omega_n = 7.5 \text{ rad/s}$	$\omega_n = 50 \text{ rad/s}$
Arm Motor			
Rise Time	2.8914 s	0.4437 s	0.0662 s
SettlingTime	5.7128 s	0.7652 s	0.1151 s
SettlingMin	0.9039°	0.9018°	0.9009°
SettlingMax	1.0155°	0.9991°	1.0000°
Overshoot	1.1094%	0.0000%	0.0000%
Undershoot	0.0000%	0.0000%	0.0000%
Peak	1.0155°	0.9991°	1.0000°
PeakTime	11.0561 s	1.1958 s	0.2480 s
Wrist motor			
Rise Time	1.2971 s	0.4619 s	0.0693 s
SettlingTime	3.2893 s	0.8221 s	0.1233 s
SettlingMin	0.9007°	0.9023°	0.9006°
SettlingMax	1.0017°	1.0000°	1.0000°
Overshoot	0.0969%	0.0000%	0.0000%
Undershoot	0.0000%	0.0000%	0.0000%
Peak	1.0017°	1.0000°	1.0000°
PeakTime	7.4053 s	3.0165 s	0.3193 s

Table 7.2: Step response analysis

	Original Control[s]	$\omega_n = 7.5 \text{ rad/s}$ [s]	$\omega_n = 50 \text{ rad/s}$ [s]
Time machine			
t_1	0.80	0.36	0.16
t_2	0.81	0.36	0.16
t_3	0.76	0.34	0.16
t_4	0.55	0.34	0.16
t_5	0.54	0.34	0.15
t_6	0.56	0.36	0.16

Table 7.3: Time machine on the joints.

7.2.1 e.DO writer

In order to appreciate the real behavior of e.DO's performances, a test was carried out in which the writing support package was added on the sixth axis (Figure 7.5).

It was programmed a trajectory on the Tablet to move the manipulator in Cartesian space. Two parallel lines were drawn on a sheet of paper, changing each time the control system. Figure 7.6 highlights the improvement with the increasing of the controllers' natural frequency. Considering that the writing support package is not fixed but has some



Figure 7.5: e.DO writer.

backlashes due to the elastic deflection, the lines written with the controllers designed in this thesis are more parallel than the original control system ones. Programming the test the difference among them has been noted also in the repeatability of the plot that in the last cases are very close to the ideal one.

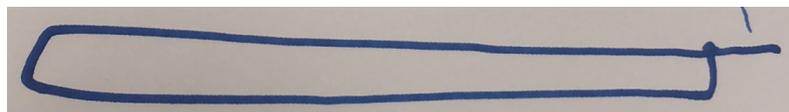
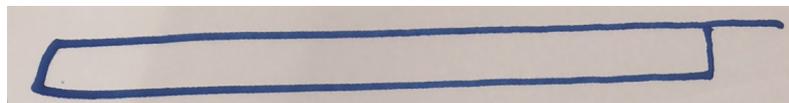
(a) *Original control system*(b) $\omega_n = 7.5 \text{ rad/s}$ (c) $\omega_n = 25 \text{ rad/s}$ (d) $\omega_n = 50 \text{ rad/s}$

Figure 7.6: Test created with the writing support package.

The final test highlights the main improvements of the control system from the practical point of view. It was drawn a rectangle with the original controller and the final used one. The results in Figure 7.7 show the improved performance in this project. In black the original controller cannot track the desired path contrary to the last control system that follow it without problems.

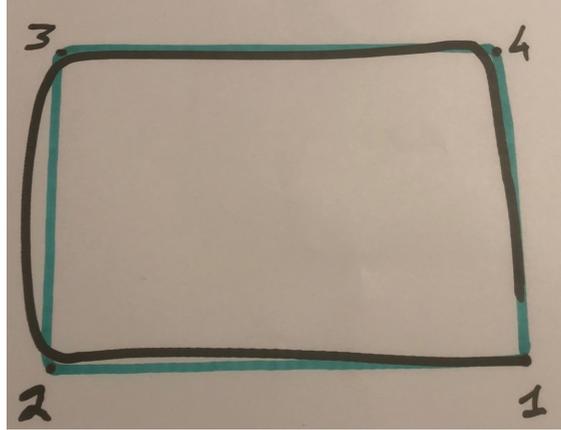


Figure 7.7: In black the rectangle with the original control and in green the rectangle with the last control system ($\omega_n = 7.5 \text{ rad/S}$, $\zeta = 1$).

7.3 Noises analysis

The noises produced on the current and the velocity are due to some issues referred on the encoder and the Hall's effect probe inside the motors or on the collected data from the devices. If for the current the problem is mainly due to the accuracy of the probe, for the velocity the noise disturbance is very weird because the position signal is "clean". To find the solution to this problem the actual signal acquired at constant velocity is analyzed (Figure 7.8).

In order to find the source of the problem, if hardware or software, the first check was to verify if the derivative of the position w.r.t. time computed with Matlab is equal to the signal given by the code. Figure 7.9 shows that the two signals are very close, this means that the derivative made by software is efficient. It can be noted that there are different values that become null. Since the software make the derivative as:

$$\omega_i = \frac{\theta_i - \theta_{i-1}}{T_s} \quad (7.1)$$

where T_s is the sampling time, this behavior is due typically for the sample repetitions. If two consecutive samples are equal, the difference between them becomes null and the velocity goes to zero, but the source of the issue can be purely software in this case. A package sent in the communication was lost and a position value was repeated. In order to understand the entire chain of communication, it is needed to list the components belonging to the chain:

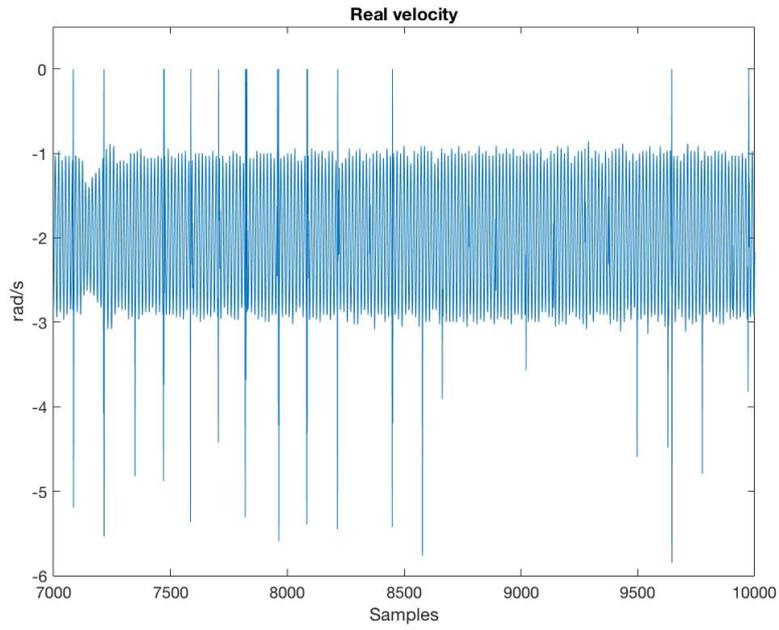


Figure 7.8: Constant Real Velocity under examination.

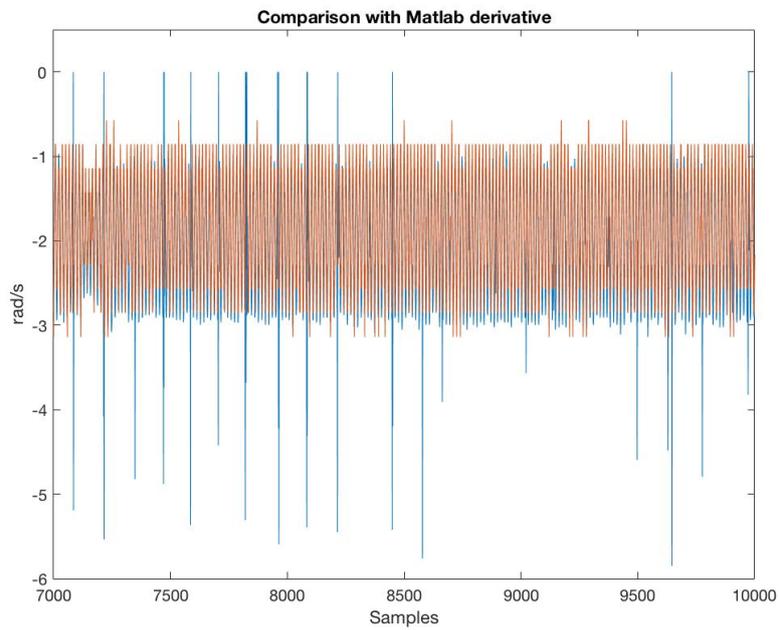


Figure 7.9: Constant Real Velocity and its Matlab derivative under examination.

- The NovaLabs circuit board on the joints provides the message with the values of the encoders and the Hall's effect probe every 10 ms.
- An USB communication sends the message to the Raspberry that gives the informations and writes them in the text file.
- A CAN bus is the protocol that behaves as a bearing between the two above components.

The critical point of the chain is the USB that can lose some packages. The right working of this protocol is not verifiable, but the losses of the samples are certainly cause of noise on the signal, in particular the zeroing of the velocity value. To find if the problem is linked exclusively to the communication chain, it is checked if the buffer filled up by the encoder contains the same noised values. In order to inspect the arrays containing the encoder values, a move action was sent to the motor through terminal. The Debug was started and the program was stopped in the part of the code when the arrays of the position and velocity were full. The result of this procedure is shown in Figure 7.10, which reports the array and the examined spectrum. The Fourier Transform highlights the excited frequencies and suggests which is the highest noise frequency to be cut through a proper filter. But the weird behavior is due to the fact that there is no sinusoidal component that can be considered as dominant, it means that the noise is casual random process and the cause of this has to be found somewhere else.

The main hypothesis is the following: there is a significative and random delay between the sent message and the real sampling time. Supposing the acquisition clean without loss of samples, the differentiations between the values of the samples is exact, the only term that can create noise is the sampling time. This means that the time that elapses between two consecutive samples is not always the same, and a little variation of this variable creates a big noise in the signal, as it is possible to see in the real plots. To solve this problem, that is hardware predominantly, it must find a fixed time clock for all the system. It must be applied on the whole structure because it is not only important the real time of the system, but its overall synchronicity.

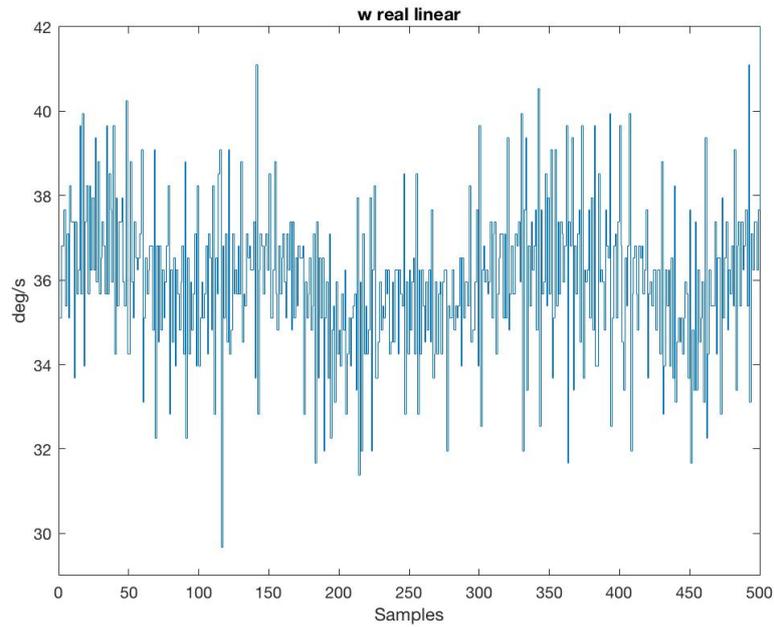
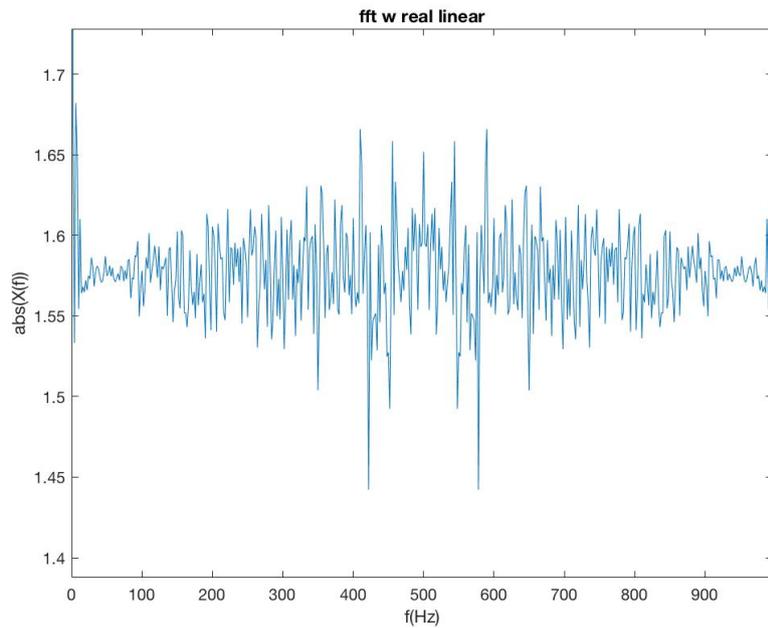
(a) *Real velocity acquisition*(b) *Spectrum of the real velocity acquisition*

Figure 7.10: Real velocity acquisition at $T_s = 1ms$ through the Debug development environment and its Spectrum computed through the Fourier Transform.

Chapter 8

Future developments

An analysis to improve the project was done in parallel, highlighting two main issues for future works:

1. Cleaning of the velocity variable
2. Elimination of the friction effect at low velocity

8.1 Observer

Even if the hypothesis on the sample time will be verified, it is possible to take another strategy. The first is to create a filter that makes an average of the noised value. This can be a good strategy, but a filter implies a delay in the chain, with possible problems if the lateness is too big because it affects the performance of the system. The solution to the problem is to create an observer of the state. The observer computes the value of the velocity in parallel to the system, and each sample time gives an estimate of the velocity in feedback to the system.

8.2 Friction compensation

A weird behavior is verified when the manipulator is moved at low velocity. At low velocity there are four zones depending on four lubrication regimes, that can be distinguished in the Stribeck's curve as in Figure 8.1.

1. Static friction
2. Lubrication border
3. Partial lubrication
4. Full lubrication

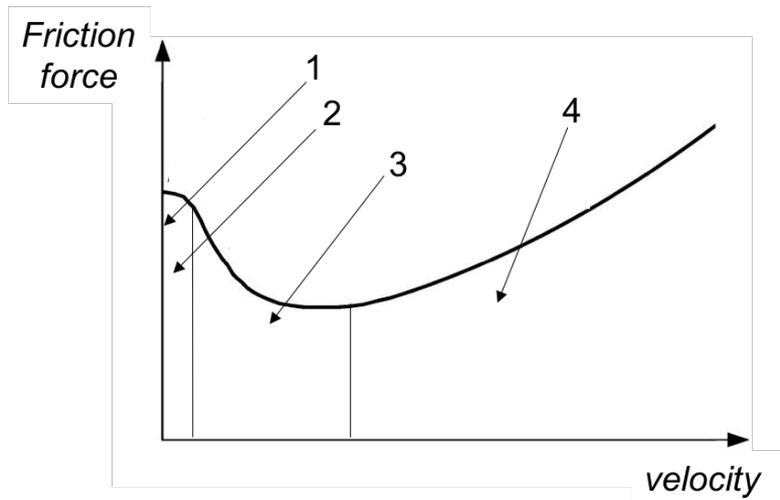


Figure 8.1: Stribeck 's curve.

8.2.1 Stick-Slip

The Stick-Slip effect occurs at low velocities, it cannot be verified with a simple model of Coulomb and static friction. Figure 8.2 shows the force applied by a spring in a servo machine. Initially the force increases till the static frictional force is reached. The sliding phase begins till t_1 , the slip time, when the spring is charging till the stick phase intervenes for the time t_2 when the spring is charged. The spring cannot achieve the value of the

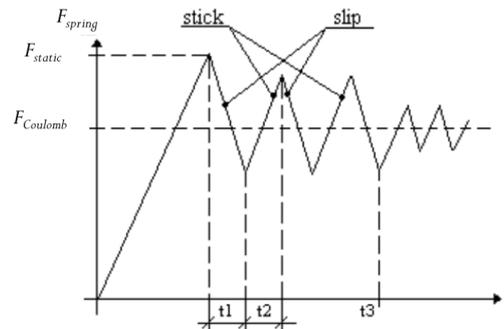


Figure 8.2: Stick-Slip effect.

static friction, because t_2 , the dwell-time is not sufficiently large to allow it. Therefore a stable limit cycle is established, with alternative stopping (stick) and sliding (slip) phases. If the time t_3 the velocity of the platform increases from v_1 to v_2 there is a reduction of the oscillations because the charging period of the spring (dwell-time) decreases.

In this condition the friction contrasts the applied torque and the external effect is an oscillatory movement. To eliminate such an undesired behavior there are two solutions: the first is purely mechanic, i.e. adding lubricant, the second is purely software, compensating

friction with a feedforward term, by identifying a proper friction model, including all its significant components (and not only the viscous one, like in the model currently adopted).

Chapter 9

Conclusions

The concepts described in this thesis and the design of the control system can be applied on any anthropomorphic robot. The structure and the complexity of the problem, even though the robot under examination is a small manipulator, are the same that can be found on an industrial robot. The results achieved seen previously are significant and can be appreciated using the manipulator. In particular the ability to follow a desired path with a good accuracy in the cartesian space is a fundamental requirement for using the robot. In this project of thesis are seen the main concepts of robotics applied on the manipulator, dealing with various topics in different fields but always related to robotics. Starting from the single joint, with the characterization of the DC motor, passing through the kinematics and dynamics of a manipulator until reaching the main arguments of the control engineering. I think that this is the scope of e.DO now. e.DO is thought as a gym for the Robotics, this is an optimal solution to test the knowledges. Obviously for the mechanics adopted, in particular the use of plastic, it cannot provides the accuracy of an industrial robot. It can happen that, with a change of mechanics e.DO will improve its workspace and its performances. e.DO can become a 3D-printer or a spacecraft, one day. Conceptually is a revolution, it can create a domino effect that will achieve big results in a future. Now it is only the begin.

Dream the impossible things is the first step to realize them.



Figure 9.1: Future of e.DO.

References

- [1] M. K. Mohammed A. Mahfouz and F. A. Salem. “Modeling, Simulation and Dynamics Analysis Issues of Electric Motor, for Mechatronics Applications, Using Different Approaches and Verification by MATLAB/Simulink (I)”. In: *Modern Education Computer Science* (2013).
- [2] I. Asimov. *Liar!* Street and Smith Publications, 1941.
- [3] Karl J Åström and Tore Hägglund. *Advanced PID control*. 2006.
- [4] R. Shrestha B. Joshi and R. Chaudhar. “Modeling, Simulation and Implementation of Brushed DC Motor Speed Control Using Optical Incremental Encoder Feedback”. In: *Researchgate* (Oct. 2014).
- [5] B. Bona. *Robotics Kinematics chain*. Feb. 2015. URL: http://www.ladispe.polito.it/corsi/meccatronica/01PEEQW/2014-15/Slides/Robotics_2015_02_Kinematics_chain.pdf.
- [6] circuitglobe. *What is commutation or commutation in dc machine*. Online. URL: <https://circuitglobe.com/what-is-commutation-or-commutation-in-dc-machine.html>.
- [7] D. Collins. *What is linear behavior for DC motors?* Online. July 2017. URL: <https://www.motioncontroltips.com/what-is-linear-behavior-for-dc-motors/>.
- [8] E. Csanyi. *basics-of-dc-motors-for-electrical-engineers*. Online. June 2015. URL: <http://electrical-engineering-portal.com/basics-of-dc-motors-for-electrical-engineers-beginners>.
- [9] K. Dasgupta. *Lecture19-KD*. Online. URL: <http://www.iitg.ac.in/kd/Lecture%20Notes/ME101-Lecture19-KD.pdf>.
- [10] electrical4u. *methods-of-improving-commutation*. Online. URL: <https://www.electrical4u.com/methods-of-improving-commutation/>.
- [11] S. Baglioni F. Cianetti C. Bracessi and L. Landi. “Parametric Multibody Modeling of Anthropomorphic Robot to Predict Joint Compliance Influence on End Effector Positioning”. In: (Nov. 2013).
- [12] H. S. Jacobs. *Commutator for Magneto Electric Machines*. 1881.
- [13] C. W. Lander. “Power Electronics”. In: 1993. Chap. 8 D.C. Machine Control.
- [14] R. Langaliya. *Commutation In Dc Generator*. Online. Jan. 2011. URL: <http://rushilangaliya.blogspot.it/2011/01/commutation-in-dc-generator.html>.

-
- [15] A. R. Valib M. Abdoa A. R. Toloeb and M. R. Arvan. “Modeling, Control and Simulation of Cascade Control Servo System for One Axis Gimbal Mechanism”. In: *International Journal of Engineering* (Jan. 2014).
- [16] mathworks. *modeling degrees of freedom*. Online. URL: <https://it.mathworks.com/help/physmod/sm/mech/ug/modeling-degrees-of-freedom.html>.
- [17] C. McGrady. *Research and Event*. Online. July 2016. URL: <https://www.arrow.com/en/research-and-events/articles/powering-a-battlebot>.
- [18] J. Miller. *Electromagnetic induction. Emf induced in a moving conductor. Faraday’s law. Lenz’s law. Self-induction. Self-induced emf. Self-inductance of a coil of n turns. Energy stored in an inductor. Electrical oscillations. Electric generator, motor*. Online. URL: <http://www.solitaryroad.com/c1048.html>.
- [19] precisionmicrodrives. *The practicalities of reversing Dc Motors*. Online. June 2016. URL: <https://www.precisionmicrodrives.com/tech-blog/2016/06/30/practicalities-reversing-dc-motors..>
- [20] R. K. Rajput. *A Textbook of Electrical Engineering*. Laxmi publications, 2003.
- [21] F. A. Salem. “Dynamic Modeling, Simulation and Control of Electric Machines for Mechatronics Applications”. In: *International Journal Of Control, Automation and Systems* (Apr. 2013).
- [22] F. A. Salem. “Modeling, Simulation and Control Issues for a Robot ARM; Education and Research”. In: *Modern Education and Coputer Science* (2014).
- [23] B. Siciliano et al. *Robotics–Modelling, Planning and Control. Advanced Textbooks in Control and Signal Processing Series*. London, UK: Springer-Verlag, 2009.
- [24] M. U.A.Bakshi. “D.c. Machines and Synchronous Machines”. In: 2008. Chap. 1.7.5 Commutator.
- [25] M. U.A.Bakshi. “D.c. Machines and Synchronous Machines”. In: 2008. Chap. 3 Testing of DC machine.

Acknowledgements

Sono stati anni stupendi, colmi di speranze e aspettative. Ora che il traguardo è vicino tengo a ringraziare coloro che hanno contribuito a rendere questo percorso speciale.

I primi ringraziamenti sono rivolti alla professoressa Marina Indri, la quale con la sua disponibilità e gentilezza ha, dapprima, reso possibile questo progetto e, successivamente, dispensando preziosi consigli, illuminato la meta. Grazie all'Ingegnere Ivan Lazzerò: un maestro nelle giornate trascorse a risolvere i problemi ingegneristici, un amico sincero nelle belle serate post lavoro, ma soprattutto un esempio per la sua dedizione e passione per la robotica. Durante questo progetto ho avuto la fortuna di lavorare accanto a persone straordinarie che sono stati e saranno per me punto di riferimento. In particolare ringrazio l'Ingegnere Enrico Mauletti che con la sua competenza e professionalità è stato un esempio tangibile di leadership. Ringrazio tutto il team e.DO e AGV di Comau, soprattutto il Dott. Pietro Gonizzi che in questi mesi non ha mai esitato a darmi un aiuto sincero; l'Ing. Guido Cargnino, "Funsuuna? Tuca nen", esempio di umiltà e competenza; l'Ingegnere Andrea Ivaldi, senza il quale non sarebbe stato possibile inserirmi all'interno del team. Volgo inoltre i miei ringraziamenti più sentiti all'Ingegnere Valerio Perna che con i consigli, i ragionamenti costruttivi e gli insegnamenti mi ha trasmesso la passione per il lavoro e l'ambizione di non fermarmi alla prima difficoltà o alla prima soluzione, perché entrambe, difficoltà e soluzioni, sono soprattutto "Good job opportunities".

Nonostante il cambiamento continuo che comporta la vita universitaria, ci sono state persone che mi sono sempre rimaste vicino. Durante gli anni vissuti a Torino ero consapevole che a Casa avrei trovato loro ad aspettarmi, sempre disposti a concedermi un consiglio e festeggiare le mie conquiste: grazie a Gre, Matte e Galli, i compagni di viaggio di una vita.

A Torino ho imparato a vivere senza i genitori, ma non sono mai stato solo. Da compagni di Liceo a compagni di Università e di stanza, i miei coinquilini sono stati fondamentali in questa esperienza. Grazie Andre, da un calcio al pallone fino alla laurea sempre vicini nel periodo più bello della vita. Un discorso a parte lo riservo a Gian, grazie perché in questi anni è stato il fratello con cui ho condiviso la mia esperienza universitaria. Ha smorzato le delusioni ed esaltato le vittorie, trovando in ogni occasione le parole giuste e un abbraccio sincero.

Grazie a tutti voi se sono riuscito ad apprezzare ogni singolo momento di questo viaggio.

Vento nel vento, questi anni hanno avuto una colonna sonora, Viola, con la quale ho condiviso i momenti più belli di questo percorso. Grazie per la dolcezza e sincerità, perché con la tua spontaneità tutto si fa chiaro. Fiero di noi per aver raggiunto insieme questo traguardo, spero di averti regalato e regalarti le stesse emozioni.

Grazie ai miei genitori per avermi sempre supportato sotto tutti i punti di vista, con il vostro esempio di umiltà e dedizione al lavoro mi avete spinto a non demordere nel raggiungimento degli obiettivi. Grazie perché le vostre parole e atteggiamenti sono stati le fondamenta più solide per arrivare alla meta. Grazie alle nonne, a cui dedico la tesi, che non hanno mai smesso di sognare e, con una chiamata o un abbraccio, mi hanno sempre fatto sentire a mio agio anche nei momenti più difficili.

La vita si sviluppa su due variabili: il tempo e lo spazio...così è iniziata la mia prima lezione al Politecnico con il docente di Analisi I Paolo Cortese, al quale volgo gli ultimi ringraziamenti come chiusura del cerchio. Per me un esempio di professionalità e di vita. Le parole che lascia ai suoi studenti ogni anno rimangono scolpite nella mente e regala loro la possibilità, se colte, di crederci davvero perseverando nel raggiungimento dei propri obiettivi.

Grazie!

They have been wonderful years, full of hopes and expectations. Now that the goal is near, I would like to thank those who helped make this special journey.

The first thanks are addressed to Professor Marina Indri; who with his availability and kindness first made this project possible and, subsequently, dispensing precious advice, enlightened the goal. Thanks to the engineer Ivan Lazzero, for me a teacher in the days spent solving engineering problems, a sincere friend in the best evenings spent finding solutions and an example for his dedication and passion for robotics. During this project I was fortunate to work alongside extraordinary people who have been and will be for me a point of reference. In particular I thank the Engineer Enrico Mauletti who with his competence and professionalism is a tangible example of leadership. I thank all the team e.DO and AGV of Comau, especially Dr. Pietro Gonizzi who in recent months has never hesitated to give me sincere help; Eng. Guido Cargnino, "Fungsiuna? Tuca nen", an example of humility and competence; the engineer Andrea Ivaldi, without which it would not have been possible to join the team. I also want my sincere thanks to the engineer Valerio Perna who with the advice, the constructive reasoning and the teachings transmitted to me the passion for work and the ambition not to stop at the first difficulty or the first solution, because both difficulties and solutions are "Good job opportunities".

Despite the constant change in university life, there have been people who have always been close to me. During my years in Turin I was aware that at home there were them, always willing to give me advice and celebrate my achievements: thanks to Gre, Matte and Galli, the traveling friends of a lifetime. In Turin I learned to live without parents, but I was never alone. From high school classmates to university, my roommates have been fundamental in this experience. Thanks Andre, from soccer to football to graduation always close in the most beautiful period of life. I reserve a special mention to Gian, thanks because in these years he was the brother with whom I shared my university experience. He has dampened the disappointments and exalted the victories, finding in every occasion the right words and a sincere embrace. I will give you a special mention to you, Gian thanks, because in these years you have been the brother with whom I have shared all my university experience. You have dampened the disappointments and exalted the victories finding the right words and a sincere hug on every occasion. Thank you so much, because it is thanks to all of you if I have managed to appreciate every single moment of this journey.

"Vento nel Vento", these years have had a soundtrack, thanks Viola, with you I shared the best moments of this journey. Thank you for your sweetness and your sincerity, because with your spontaneity everything becomes clear. I am proud of having reached this goal together, I hope I have given you the same emotions.

Thanks to my parents for always supporting me from all points of view, with your example of humility and dedication to work you pushed me not to give up in achieving the goals. Thank you because your words and attitudes have been the most solid foundations to reach the goal. Thanks to the grandmothers to whom I dedicate the thesis, which have never stopped dreaming and with a call or a hug have always made me feel at ease even in the most difficult moments.

Life develops on two variables: time and space ... so began my first lesson at the Polytechnic with the teacher of Analysis I Paolo Cortese, to whom I turn my last thanks as the closing of the circle. For me, an example of professionalism and life. The words that he leaves to his students every year remain carved in the mind and gives them the opportunity, if caught, to really believe and persevere in achieving their goals.