ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DOTTORATO DI RICERCA IN

MECCANICA E SCIENZE AVANZATE DELL'INGEGNERIA

Ciclo 36

Settore Concorsuale: 09/A2 - MECCANICA APPLICATA ALLE MACCHINE

Settore Scientifico Disciplinare: ING-IND/13 - MECCANICA APPLICATA ALLE MACCHINE

HUMAN INTERACTION CONTROL WITH REDUNDANT ROBOTS

Presentata da: Eugenio Monari

Coordinatore Dottorato

Lorenzo Donati

Supervisore

Rocco Vertechy

Esame finale anno 2024

Abstract

Collaborative robotics is an expanding research topic from both an academic and industrial point of view. In fact, there exists a high number of applications that it is not possible or convenient to completely automatize and therefore require the activity of a human operator. At the same time, it is extremely important that such activities do not cause physical or mental fatigue to the worker. In this context, the presence of robots that can safely interact and cooperate with humans is crucial.

This thesis addresses the theme of collaborative robots, focusing in particular on a hand guiding application with a redundant Franka Emika Panda robot, studying the best control strategy for this framework. The analysis has two main parts, namely end effector control and null space control.

As for end effector control, the objective is that the human operator is able to drive the end effector along a desired path in a precise, smooth and effortless way. Impedance control schemes provide poor results, and this is shown to be due to the shortcomings of the torque interface of the Franka robot. Another strategy is tested, called "helping" controller, which sends to the joints control torques aimed at minimizing the force that the user needs to provide to guide the end effector. This scheme has better results, but it is not suitable if also null space commands are sent. Instead, admittance control, gives the best results, also being compatible with null space commands.

As for null space control, two innovative strategies are introduced, finding the null space commands as solutions of constrained optimization. These techniques are compared and tested in a simplified case study on the real robotic platform to show that their computing times are compatible with the real time requirements of the Franka robot.

Contents

Chapter 1

Introduction

1.1 The need for collaborative robotics

Modern robotics have come to automatize numerous tasks that had always been executed by men. Yet, there are tasks that still require the cognitive skills of a human being, especially in non-nominal situations and/or in unstructured environments. In these cases, pre-programming the control laws of the automatic system might be too time-consuming, expensive or sometimes even impossible. As stated in [7], "even highly automated systems, such as electric power networks, need human beings for supervision, adjustment, maintenance, expansion and improvement".

In her influential article, [5], Bainbridge observed that there are two "ironies" in the traditional approach to automation, deriving from the assumption that human operators are less reliable than an automatic system, and it is therefore preferable to replace them. The first irony is that, even in a completely automatized system, software developer's errors can be a source of major problems. The second one is that the designer, though trying to eliminate human operators from the industrial plant, still leaves them to perform the tasks that it is not possible to automatize. These tasks can be either of simple monitoring or of manual take-over in case some unexpected failure happens. In the first case, the worker need to focus his/her attention over a process that will have a regular behavior for most of the time, which was shown to be very critical by vigilance studies [43]. Also manual take-over can be problematic, because it requires skills that can be acquired and remembered only through frequent exercise, but again the cases in which the worker will have to manually perform the task are quite rare in an automatized environment. This leads to the paradox of having boring but high-responsibility jobs, leading to errors, job dissatisfaction and absenteeism. The conclusion is that "by taking away the easy part of his task, automation can make the difficult parts of the human operator's task even more difficult".

Even though [5] was written in 1983, the points that this article made are still valid nowadays, as claimed in [74]. As a matter of fact, even in highly developed countries, it is still very common that workers perform heavy manual works in industrial settings. A European report on work-related musculoskeletal disorders (MSDs, i.e. backache and/or muscular pains in shoulders, neck, upper limbs and/or lower limbs) [13] found that more than 50% of European workers suffer from this kind of problems (Figure 1.1), and that the industrial sector is one of the most affected (Figure 1.2). In fact, many of the risk factors to which human operators are exposed (such as repetitive hand movements, carrying and moving heavy loads and vibration from hand tools and machinery) are frequently encountered in plants (Figure 1.3). This translates to a very significant amount of social burdens for society, namely the overall economical cost of work-related MSDs is estimated to be around 240 billion euros per year in the EU [6].

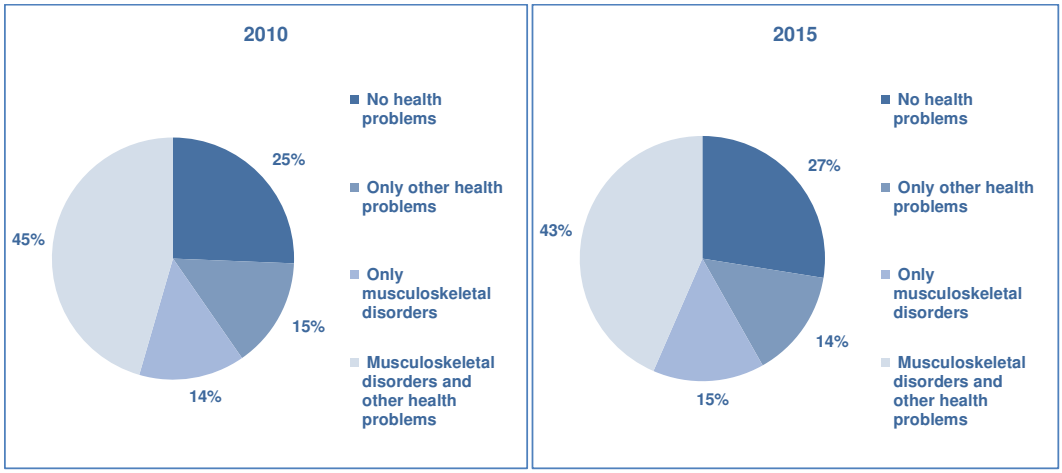


FIGURE 1.1: Percentage of European workers reporting different types of health problems [13].

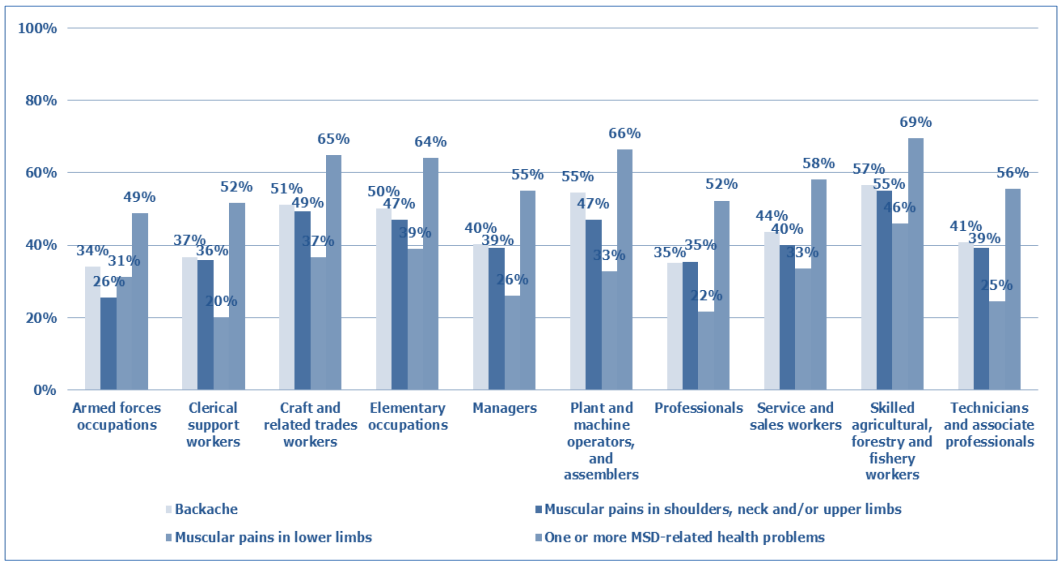


FIGURE 1.2: Percentage of European workers with MSDs by working sector [13].

What emerges from these considerations is that, on the one hand, it is needed to further reduce the factors that lead to MSDs, letting robot take care of the fatiguing part of the human work, on the other hand, this transition cannot be achieved by completely automatizing industrial plants. Instead, starting from the last decade of XX century, research efforts started to be made towards a human-robot collaboration framework, in which the bests of both the human operators (i.e., flexibility) and robots (i.e., efficiency and precision) could emerge [54]. As stated in [44], it is therefore "essential that the goal of automation be changed from the level of automation to the level of collaboration".

This gave rise to a new field of studies and of industrial applications, which is referred to as collaborative robotics, aiming at moving from industrial plants where

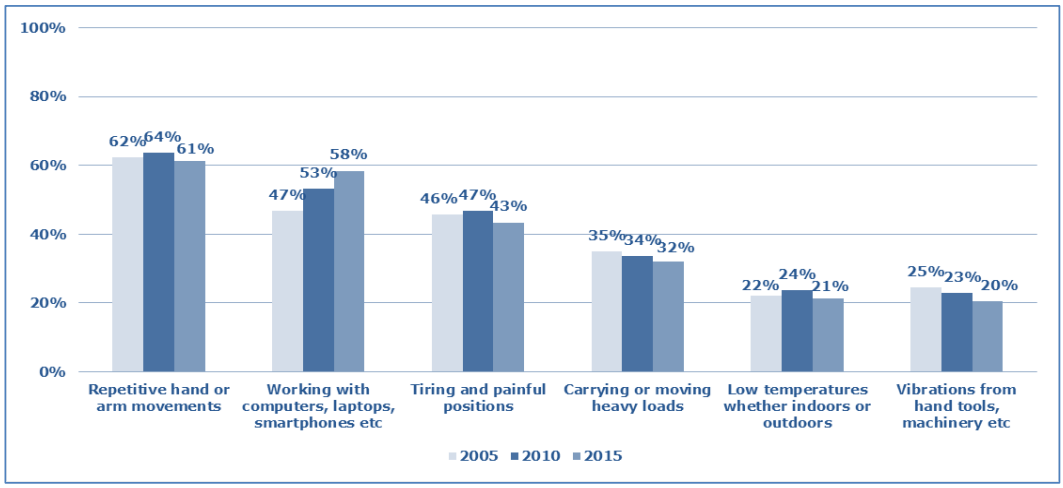


FIGURE 1.3: Percentage of workers reporting that they are exposed to different physical risk factors at their work at least a quarter of the time [13].

robots are confined into safety cages to settings in which robots work alongside humans [45]. At the same time, there are more and more robotic companies that have started to produce collaborative robots, or cobots. These are ordinary robots (generally serial) characterized by being light-weight and equipped with technologies that can detect collisions with humans (usually torque sensors on all the joints), so that the human operators' safety is ensured. Some examples of cobots can be seen in Figure 1.4.

(a) (b) (c)

FIGURE 1.4: Examples of cobots: (a) Franka Emika Panda; (b) Fanuc CRX-5iA; (c) UR5.

A collaborative application is defined as "any application involving a purposeful sharing of the same spaces and resources by humans, while robot systems are therein operating in automatic mode" [78]. By automatic mode it is meant an "operating mode in which the robot control system operates in accordance with the task programme", as opposed to manual mode, which is a "control state that allows for the direct control by an operator", usually by means of a control pendant [68]. Even though the term "cobot" has come into common use, it would be more correct to

only talk about collaborative robotic applications [78]. In fact, the fact that a robot possesses some safety devices and/or control strategies, does not in itself guarantee that an application in which that robot is used is safe: a risk assessment procedure has to be carried out before using the robot. Conversely, also robots which are not classified as cobots and have medium/high payloads can be employed in collaborative scenarios if a proper risk assessment procedure certified that the application is safe.

Safety can be ensured using different methodologies (see also Section 1.2 for a more detailed discussion). Contacts can be recognized in a sensorless or sensorized way. In the first case, estimation is usually based on the difference between nominal torques and observed (through motor current reading) torques. In the second case, estimation is based either on force/torque sensors mounted on the joints and/or on the end effector or on capacitive pads that are mounted on the robot links and also have the positive side effect of absorbing part of the kinetic energy of the collision [78]. If a collision with a part of the human body actually takes place, the control of the robot provides the certainty that it will not lead to an injury to the operator. A representation of the consequence of an impact is shown in Figure 1.5. A high peak of pressure is exerted as soon as the collision happens (typically, the first 150–200 ms). Then, if the robot control is able to react stopping the link that has had the impact, the pressure value will drop (yellow dotted line) below the pain threshold (black dashed line). If instead the robot were not able to do so, elongated peaks or high steady states would be observed, resulting in high-severity contacts (an injury occurs if the pressure on the operator's limb has a high value and at the same time remains above this value for a long enough time). This kind of risk assessment is usually employed to certify the safety of an application because international standards ([73]) provide the values of the pain thresholds for all body limbs, so tests can be designed in a structured and reliable way. Other ways of avoiding dangerous collisions imply predicting the collisions before they happen through capacitive pads or camera vision systems and stopping or slowing down the movement of the robot in advance.

Collaborative robotics can be applied in a very wide range of scenarios, having a beneficial effect in terms of cost optimization, improved ergonomics and operator's well-being [19]. The most typical applications are production lines, where robots help workers assembling products (Figure 1.6(a)), carry heavy weights while the operator just guides the robot in the desired direction (Figure 1.6(b)), or learn a trajectory through which the operator guides the end effector (robot teaching), and which will afterwards be reproduced without human assistance (Figure 1.6(c)).

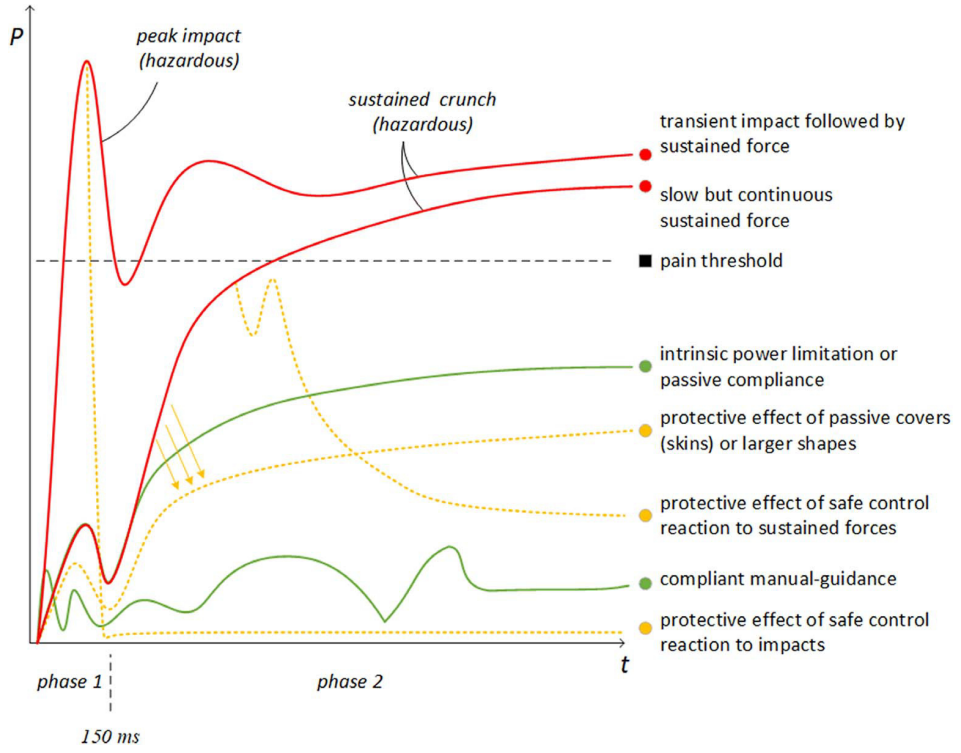
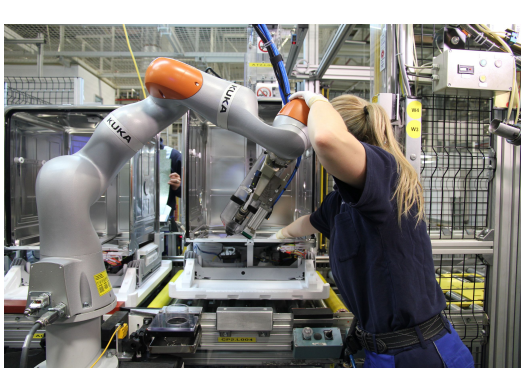
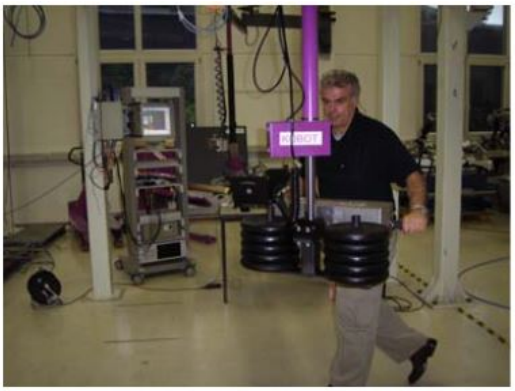


FIGURE 1.5: Representation of the behavior of pressure on a human limb when a collision with a robotic link takes place [78].



(a)



(b)

(c)

FIGURE 1.6: Examples of collaborative applications.

1.2 A taxonomy of collaborative robotics

After providing a general introduction to collaborative robotics and an explanation of its usefulness in the industrial context of nowadays, a possible classification of collaborative solutions will be presented. A complete and clear taxonomy is presented in [44] (see Figure 1.7). The distinction in the Human-Robot Collaboration (HRC) axis is self-evident, regarding the number of robots and operators involved (most often there is only one robot and one operator), but the other two axes deserve more detailed consideration.

On the HRC safety implication axis are the safety features described in the international standard [69], namely (see also Figure 1.8):

- Safety monitored stop (SMS): this is a state in which the robot is halted as soon as an operator enters the hazardous zone. Strictly speaking, this should not be considered as a real collaborative feature because it implies that there is not any kind of interaction between the humans and the robot [78].

- Hand guiding (HG): this method is intended for situations in which the worker is holding the end effector, guiding it to the desired positions and operating it to perform the task. In this case, [68] specifies that the robot enters this mode only when an enabling device, placed on the end effector, is pressed. When the button is released, instead, the robot must be in SMS state if an operator is within its work area. Additionally, the robot must operate with a safety-rated monitored speed function active. This function states velocity limits for the end effector and/or for the axes of the robot to ensure that no hazardous collisions may happen. The values of the speed limits are not provided by the standards but have to be determined by an ad hoc risk assessment on the specific application.

- Speed and separation monitoring (SSM): in this method the position of the human body is continuously tracked and the protection algorithm estimates the amount of time needed by the robot to stop before a collision happens. If the human is far enough from the robot, no speed limitation is set (green area of the SSM sketch in Figure 1.8); if the human is closer but it is still possible to stop the robot on time, the speed is limited but the robot remains operational (yellow area in Figure 1.8); if the distance between the robot and the worker is too short a SMS is performed (red area in Figure 1.8).

- Power and force limiting (PFL): this mode is intended for applications in which no contact is desired between the operator and the robot, but at the same time the human has to stay close to the robot, so that SSM would result to be very unpractical, causing unnecessary stops and slowdowns. Instead, with PFL occasional contacts are acceptable, but they must be detected (through the sensorless or sensorized way examined above), and appropriate reaction must be carried out by the robotic system to avoid that they may be painful or lead to injuries.

On the HRC interaction levels there are five layers (see also Figure 1.9):

- Cell: in this case the workspace of the human and the robot do not intersect and the operator is separated from the robot by a safety cage. In case the worker needs to access the area inside the cage, e.g., to load a piece, a SMS is performed.

- Coexistence: even in this case the human and the robot do not share their workspaces and work on different tasks, but in this case they are not separated by a cage, and SSM strategy is employed.
- Synchronized: in this case the human and the robot share a part of their workspace, but this common area is entered by only one of the two at a time. Typically, the operator has to manually perform some operations on a piece, then will leave the shared region to do something else, and the robot will perform some other task on the initial piece.
- Cooperation: in this case the human and the robot share part of their workspaces and operate in the common area at the same time, but on different components.
- Collaboration: in this case the human and the robot share part of their workspaces and operate in the common are at the same time and on the same component.

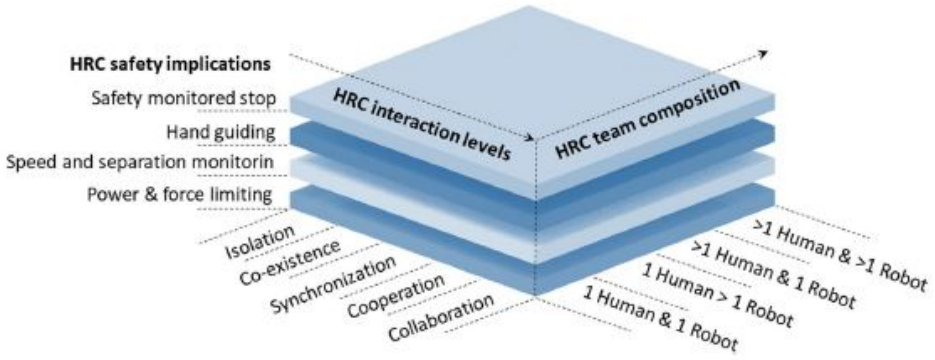


FIGURE 1.7: 3D classification of collaborative applications [44].

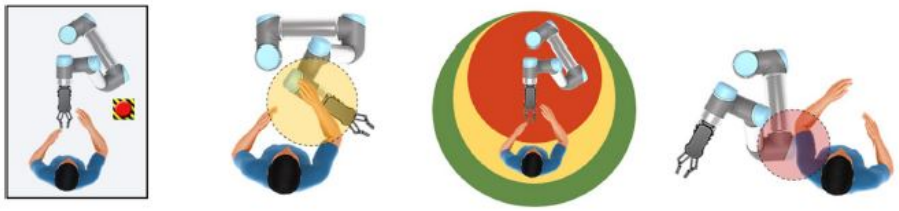


FIGURE 1.8: Collaborative safety features as of [69].

Another possible classification of collaborative application is presented in [78] (see Figure 1.10). The variables of this map are the frequency of access of the human operators to the workspace of the robot and the kinetic energy (i.e., velocity) needed for the application. Such a framework is useful to choose the type of collaboration that fits best the application. If the access of human operators is only needed very rarely, then likely there is no need for a collaborative solution (inefficient collaboration); if very high velocities are required, collaboration may lead to too frequent stops and slowdowns to ensure safety (unpractical collaboration); if the contact is continuous and the velocities are low, HG is the right solution; finally, in case of

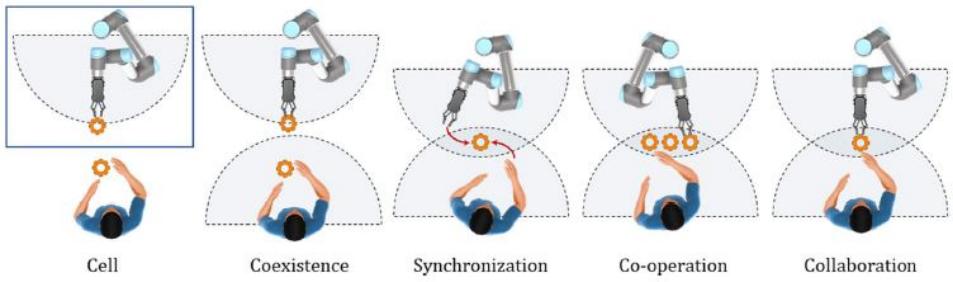


FIGURE 1.9: Interaction levels as of [44].

medium frequency of access and low speeds, PFL is the most suitable options, and in case of medium frequency of access and high velocities, SSM is the best choice.

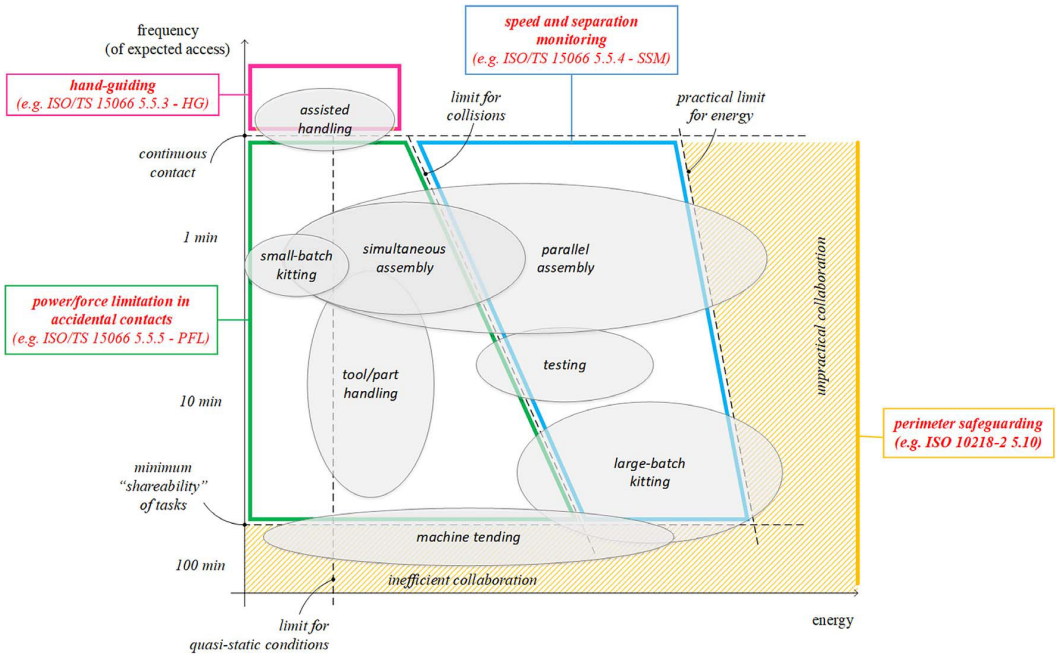


FIGURE 1.10: Energy-frequency map of robotic applications [78].

1.3 Thesis contribution

The present thesis was developed in the context of the research on collaborative robotics at the SAIMA laboratory of the Industrial Engineering department of the University of Bologna. Starting from 2019, the laboratory has been involved in two main projects on this topic, one in collaboration with INAIL, the Italian national agency for accidents at work, and the second one in collaboration with CNR, the Italian National Council of Research, within the European COVR project, aimed at closing the gaps between technological advancements and standardization, and supporting users in the implementation of HRC applications. These projects regarded the development of a collaborative workstation for drilling operations (Figure 1.11), which can be classified as a HG collaborative application. This workstation consists of

- A Franka Emika Panda robot (see Figure 1.4(a)), a seven-degree-of-freedom robot, which is equipped with torque sensors on all the joints and can be commanded through a position, velocity of torque interface with a control cycle time of 1 ms.
- A custom end effector, made up of a pneumatic drill and a 3D-printed handle which fixes the drill and connects it to the rest of the robotic structure. On the handle there is also a trigger that, when pressed, opens a pneumatic valve that allows the flux of compressed air activating the rotation of the drill tip.
- An Axia80 Schunk force/torque sensor, which is mounted between the terminal flange of the Franka robot and the custom handle. This sensor measures the forces and torques that the human operator applies on the end effector along all directions.
- A Shadow Motion Capture IMU suit, made up of 17 IMU units, which are attached in different locations of the human body and track its motion.

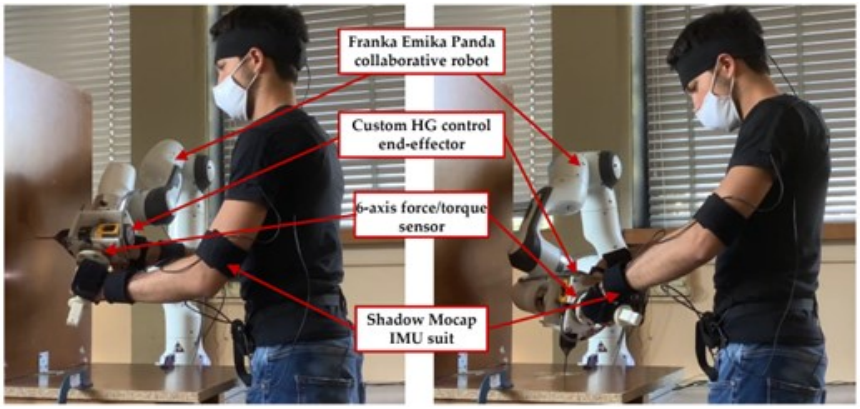


FIGURE 1.11: Collaborative workstation for drilling operations, developed at the SAIMA laboratory.

During the collaboration with INAIL, the safety implications of the workstation were examined, studying how such an application could fit within the current normative and standards framework, and developing a reliable algorithm to track the positions of the human limbs based on IMU data and perform collision avoidance

accordingly (thanks to the fact that the robot is redundant the collision avoidance movement was in the null space, without interfering with the end effector motion).

During the COVR project, instead, the focus was on the ergonomic aspects of the drilling task. In particular, the standards [70, 71, 72], regarding handling of loads by human operators in industrial settings, were adapted to a collaborative scenario, generating an OCRA index (i.e., an index that evaluates the ergonomic risks of industrial activities, defined in [72]) based on the IMU and force/torque sensor data. Also, a MATLAB app (see Figure 1.12) was developed that helps to organize the working shift so that the OCRA index is low, i.e., the activity is ergonomically safe, and performs an online monitoring of the posture of the human operator, providing a notification if he/she is having an ergonomically unsuitable behavior.

This thesis mainly focuses on the following two aspects:

- End effector control: the operator must be able to move the end effector in a smooth and effortless way, with the effort of minimizing the effort needed from the operator to carry the drill. This problem is approached in a rather general way, considering both control schemes that are only translational and keep the orientation of the end effector fixed, and other schemes that also leave free the rotational degrees of freedom of the end effector. In fact, the main aim of the thesis is to provide tools that can be used when designing collaborative solutions rather than studying a specific industrial HG application, as was done in the INAIL and COVR project presented above.

- Null space control: since the robot is redundant, besides end effector control, it is also possible to command additional motions that do not affect the end effector behavior. In the context of this HG application, this is done mainly with three objectives: 1) keeping the kinematic chain of the robot away from the operator, not so much for safety reason (the robot is stopped if a collision is detected through the torque sensors, and, in addition, the joint velocities are quite low in this case, so limiting them after a risk assessment procedure would not degrade the performance) as because it would be very uncomfortable for the operator to have some robot links close to his/her arms impeding his/her movements; 2) avoiding singularities; 3) avoiding joint limits.

The second chapter of the thesis will consider end effector control, examining three possible strategies and discussing their advantages and disadvantages in the specific control framework of the Panda robot.

The third chapter will focus on null space control, presenting some shortcomings of the traditional gradient projection method and introducing two innovative algorithms; these algorithms will also be tested, both in simulation and on the real robotic platform, in a number of case studies different from the actual HG application to better show their characteristics and performances.

The fourth chapter will draw the conclusions of the presented results and indicate some possible future research directions resulting from the present work.

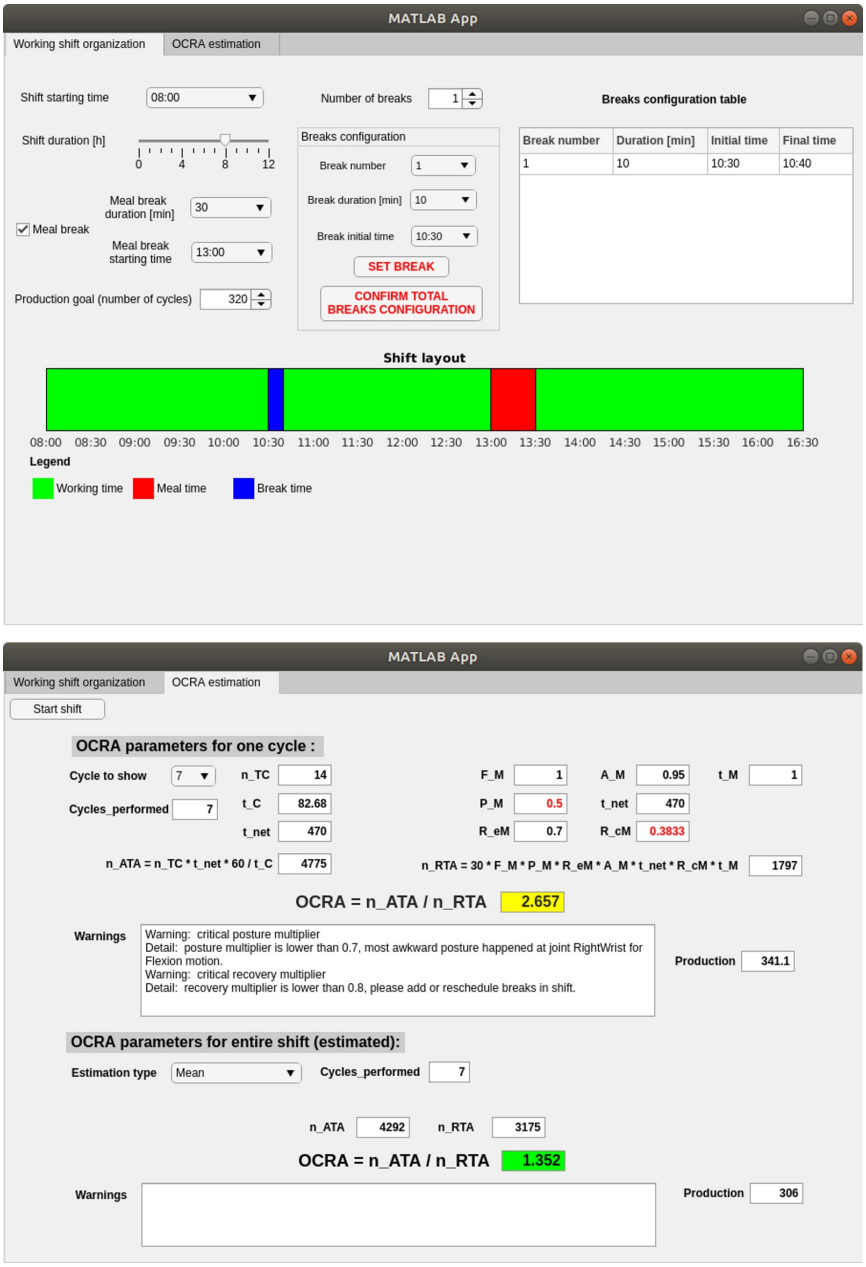


FIGURE 1.12: Two tabs from the MATLAB app developed during the COVR project, showing an organization of the working shift and the resulting OCRA index and the OCRA multipliers (i.e., factors accounting for different aspects of the task, such as effort on the muscles, awkward postures, etc.) [49].

Chapter 2

End effector control

2.1 Theoretical background

2.1.1 Notation conventions

The notation conventions that will be used in the present thesis for kinematics, differential kinematics and dynamics are the same as in [67], but for clarity some of the most important and recurring points of this notation will be briefly outlined in the following. These points are:

- $\mathbf{p}^i_A$ is the vector of the coordinates of point A expressed in reference frame i.
- If i and j indicate two reference frames with common origin, the rotation matrix $\mathbf{R}^i_j$ has three equivalent meanings:
  - It is the linear operator mapping $\mathbf{p}^j_A$ to $\mathbf{p}^i_A$, namely $\mathbf{p}^i_A = \mathbf{R}^i_j \mathbf{p}^j_A$.
  - Its columns vectors are the direction cosines of the axes of reference frame j expressed in reference frame i.
  - It is the operator that rotates the vector of the direction cosines (expressed in reference frame i) of each axis of reference frame i to the vector of the direction cosines (always expressed in reference frame i) of each axis of reference frame j, namely, it expresses the rotation from reference frame i to reference frame j.
- If $\mathbf{p}^i_j$ is the vector of the coordinates of the origin of reference frame j expressed in reference frame i, the homogeneous matrix defined as $\mathbf{A}^i_j = \begin{bmatrix} \mathbf{R}^i_j & \mathbf{p}^i_j \\ \mathbf{0}^T & 1 \end{bmatrix}$ is the linear operator mapping $\mathbf{p}^j_j$ to $\mathbf{p}^i_j$, namely $\tilde{\mathbf{p}}^i_j = \mathbf{A}^i_j \tilde{\mathbf{p}}^j_j$, with $\tilde{\mathbf{p}}^i_j = \begin{bmatrix} \mathbf{p}^i_j \\ 1 \end{bmatrix}$.
- $\mathbf{q}$ is a set of generalized coordinates of the system.
- The direct kinematics of a serial manipulator of $n+1$ links connected by n joints is obtained defining $n+1$ reference frames, each one attached to a link, and chain-multiplying the homogeneous matrices $\mathbf{A}^{i-1}_i(q_i)$, each of which is function of a single joint variable, as

$$\mathbf{T}^0_n(\mathbf{q}) = \mathbf{A}^0_1(q_1)\,\mathbf{A}^1_2(q_2)\ldots\mathbf{A}^{n-1}_n(q_n)\ . \tag{2.1}$$

- According to the Denavit-Hartenberg convention (DH), the transformation described by $\mathbf{A}^{i-1}_i$ happens in the following two passages:

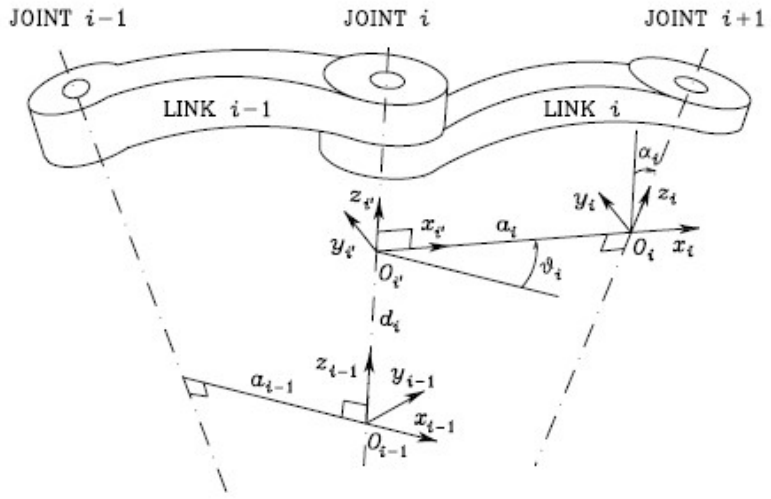


FIGURE 2.1: Denavit-Hartenberg parameters [67].

– Frame $i-1$ is aligned to frame i' translating by d_i along axis z_{i-1} and rotating it by θ_i about axis z_{i-1}, namely

$$\mathbf{A}_{i'}^{i-1} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.2}$$

– Frame i' is aligned to frame i translating by a_i along axis $x_{i'}$ and rotating it by α_i about axis $x_{i'}$, namely

$$\mathbf{A}_{i'}^{i-1} = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.3}$$

Then the overall transformation $\mathbf{A}_i^{i-1}$ is

$$\mathbf{A}_i^{i-1} = \mathbf{A}_{i'}^{i-1}\mathbf{A}_i^{i'}. \tag{2.4}$$

- The geometric Jacobian $\mathbf{J}(\mathbf{q})$ is the operator mapping the joint velocities to linear and angular end effector velocities, namely

$$\mathbf{v}_e = \begin{bmatrix} \dot{\mathbf{p}}_e \\ \boldsymbol{\omega}_e \end{bmatrix} = \mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}} = \begin{bmatrix} \mathbf{J}_P(\mathbf{q}) \\ \mathbf{J}_O(\mathbf{q}) \end{bmatrix} \dot{\mathbf{q}} = \begin{bmatrix} \mathbf{j}_{P1} & \cdots & \mathbf{j}_{Pn} \\ \mathbf{j}_{O1} & \cdots & \mathbf{j}_{On} \end{bmatrix} \dot{\mathbf{q}}, \tag{2.5}$$

where $\mathbf{p}_e$ is the position of the origin of the end effector reference frame, $\boldsymbol{\omega}_e$ is the angular velocity of the end effector reference frame and $\mathbf{j}_{Pi}$ and $\mathbf{j}_{Oi}$ are (3×1) column vectors.

- For the vectors $\mathbf{j}_{Pi}$ of the translational part of the Jacobian, the following equivalence holds:
$$\mathbf{j}_{Pi} = \frac{\partial \mathbf{p}_e}{\partial q_i} . \tag{2.6}$$
- For revolute joints and with reference frames defined according to the DH convention, the following equivalence holds:
$$\begin{bmatrix} \mathbf{j}_{Pi} \\ \mathbf{j}_{Oi} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_{i-1} \times (\mathbf{p}_e - \mathbf{p}_{i-1}) \\ \mathbf{z}_{i-1} \end{bmatrix} . \tag{2.7}$$
- The Jacobian is said to be expressed in the j-th reference frame, and is designated as $\mathbf{J}^j$ if all the vectors $\mathbf{j}_{Pi}$ and $\mathbf{j}_{Oi}$ are expressed in the j-th reference frame.

2.1.2 Mapping between end effector forces and torques and generalized Lagrangian forces

The dynamic equations of motion of a general robotic structure (without considering dissipations in the joints) can be written as

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{J}^T \mathbf{F}_{e,ext} , \tag{2.8}$$

where $\mathbf{q}$ is the vector of generalized coordinates (joint angles for revolute joints and joint positions for prismatic joints), $\mathbf{B}(\mathbf{q})$ is the inertia matrix, $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis vector, $\mathbf{g}(\mathbf{q})$ is the gravity vector, $\boldsymbol{\tau}$ is the vector of the actuation generalized forces (actuation torques for revolute joints and actuation forces for prismatic joints) and $\mathbf{F}_{e,ext}$ is the vector of the external forces $\mathbf{f}_{e,ext}$ applied to the end effector and of the external torques $\boldsymbol{\mu}_{e,ext}$ applied to the end effector with respect to a pole placed at the center of the end effector frame, namely

$$\mathbf{F}_{e,ext} = \begin{bmatrix} \mathbf{f}_{e,ext} \\ \boldsymbol{\mu}_{e,ext} \end{bmatrix} . \tag{2.9}$$

The components of $\mathbf{F}_{e,ext}$ expressed in the end effector reference frame are

$$\mathbf{F}^e_{e,ext} = \begin{bmatrix} f_{e,ext,x} \\ f_{e,ext,y} \\ f_{e,ext,z} \\ \mu_{e,ext,x} \\ \mu_{e,ext,y} \\ \mu_{e,ext,z} \end{bmatrix} . \tag{2.10}$$

Of particular importance for the present thesis is the relation existing between the external end effector forces and torques and their generalized-force counterpart, namely

$$\boldsymbol{\tau}_{ext} = \mathbf{J}^T \mathbf{F}_{e,ext} . \tag{2.11}$$

Most robotics textbooks (e.g., [67, 66]) present this equivalence without providing a detailed demonstration of it. [67] introduces (2.11) in the context of statics, using the virtual work principles, then extends it by analogy to (2.8), but this cannot be considered as a formal proof. The relation was originally introduced by Khatib in [38], which is a paper that had a fundamental importance in the robotics community, totalling more than 2000 citations in the Scopus database. In that work, the relation is

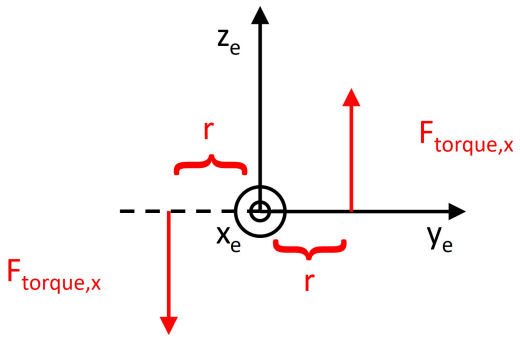


FIGURE 2.2: A pair of forces equivalent to the x-axis torque component of $\mathbf{F}_{e,ext}$, represented in the end effector reference frame.

obtained by comparing the expressions of the dynamics in the operational space and in the joints space. However, some mathematical passages are not completely clarified, especially regarding the partial derivatives of the kinetic energy with respect to the operational space variables. In the following, instead, another formal proof of the equivalence will be shown. This proof, at the best of the author's knowledge, is not present in the literature, and, in addition, will be useful to better illustrate some concepts in Section 2.5.

According to the Lagrangian formulation of dynamics (see for example [8]), in a system described by m generalized coordinates upon which n forces act on n points, the i-th component of the generalized force is defined as

$$Q_i = \sum_{j=1}^{n} \mathbf{F}_j \cdot \frac{\partial \mathbf{P}_j}{\partial q_i} . \tag{2.12}$$

Referring to (2.9), $\mathbf{F}_{e,ext}$ can be equivalently represented by a system of forces consisting of:

- The force $\mathbf{f}_{e,ext}$ applied in the center of the end effector reference frame.
- A pair of forces (see Figure 2.2) $\mathbf{F}_{torque,x}$ and $-\mathbf{F}_{torque,x}$ parallel to the z axis of the end effector frame, having the same norm $F_{torque,x}$ and opposite direction, and applied in two points $\mathbf{P}_{torque,x,1}$ and $\mathbf{P}_{torque,x,2}$ lying on the y axis of the end effector frame, at a positive and negative distance of r from the center of the end effector frame. $F_{torque,x}$ and r are such that $2F_{torque,x}r = \mu_{e,ext,x}$.
- A pair of forces $\mathbf{F}_{torque,y}$ and $-\mathbf{F}_{torque,y}$ parallel to the z axis of the end effector frame, having the same norm $F_{torque,y}$ and opposite direction, and applied in two points $\mathbf{P}_{torque,y,1}$ and $\mathbf{P}_{torque,y,2}$ lying on the x axis of the end effector frame, at a positive and negative distance of r from the center of the end effector frame. $F_{torque,y}$ and r are such that $2F_{torque,y}r = \mu_{e,ext,y}$.
- A pair of forces $\mathbf{F}_{torque,z}$ and $-\mathbf{F}_{torque,z}$ parallel to the y axis of the end effector frame, having the same norm $F_{torque,z}$ and opposite direction, and applied in two points $\mathbf{P}_{torque,z,1}$ and $\mathbf{P}_{torque,z,2}$ lying on the x axis of the end effector frame, at a positive and negative distance of r from the center of the end effector frame. $F_{torque,z}$ and r are such that $2F_{torque,z}r = \mu_{e,ext,z}$.

The vector of generalized forces resulting from such a system of forces is

$$\mathbf{Q}_{ext} = \mathbf{Q}_{force} + \mathbf{Q}_{torque,x} + \mathbf{Q}_{torque,y} + \mathbf{Q}_{torque,z} \,. \tag{2.13}$$

Considering the translational part of (2.13), and remembering (2.6), the i-th component of the force term of the vector of the generalized forces is

$$Q_{force,i} = \frac{\partial \mathbf{p}_e}{\partial q_i} \cdot \mathbf{f}_{e,ext} = \left(\frac{\partial \mathbf{p}_e}{\partial q_i}\right)^T \mathbf{f}_{e,ext} = \mathbf{j}_{P,i}^T \mathbf{f}_{e,ext} \,, \tag{2.14}$$

and thus

$$\mathbf{Q}_{force} = \mathbf{J}_P^T \mathbf{f}_{e,ext} \,. \tag{2.15}$$

Considering now the rotational part of (2.13), the i-th component of the x-axis torque of the vector of the generalized forces is

$$Q_{torque,x,i} = \frac{\partial \mathbf{P}_{torque,x,1}}{\partial q_i} \cdot \mathbf{F}_{torque,x} - \frac{\partial \mathbf{P}_{torque,x,2}}{\partial q_i} \cdot \mathbf{F}_{torque,x} \,. \tag{2.16}$$

The vectors $\mathbf{P}_{torque,x,1}$, $\mathbf{P}_{torque,x,2}$ and $\mathbf{F}_{torque,x}$ can be expressed in any reference frame. For convenience, it is chosen to express them in a reference frame oriented like the base (i.e., 0) reference frame, but with the origin coincident with the origin of the end effector reference frame. With an abuse of notation, also the reference frame in which the vectors will be expressed will be designated as 0. In this way

$$\mathbf{P}^0_{torque,x,1} = -\mathbf{P}^0_{torque,x,2} \,, \tag{2.17}$$

and then

$$Q_{torque,x,i} = 2\frac{\partial \mathbf{P}^0_{torque,x,1}}{\partial q_i} \cdot \mathbf{F}^0_{torque,x} \,. \tag{2.18}$$

At this point, based on (2.1)–(2.4), it is possible to write

$$\begin{aligned} Q_{torque,x,i} &= 2\left(\frac{\partial \mathbf{R}^0_e \mathbf{P}^e_{torque,x,1}}{\partial q_i}\right)^T \mathbf{R}^0_e \mathbf{F}^e_{torque,x} = \\ &= 2F_{torque,x} r \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \frac{\partial \mathbf{R}_e^{i'^T} \mathbf{R}_{i'}^{i-1^T} \mathbf{R}_{i-1}^{0^T}}{\partial q_i} \mathbf{R}^0_{i-1} \mathbf{R}^{i-1}_{i'} \mathbf{R}^{i'}_e \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \,. \end{aligned} \tag{2.19}$$

Since, according to (2.4), among $\mathbf{R}^0_{i-1}$, $\mathbf{R}^{i-1}_{i'}$ and $\mathbf{R}^{i'}_e$, only $\mathbf{R}^{i-1}_{i'}$ depends on q_i, and $\mathbf{R}^{i^T}_j = \mathbf{R}^{i-1}_j$, the following holds

$$\begin{aligned} Q_{torque,x,i} &= 2F_{torque,x} r \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \mathbf{R}_e^{i'^T} \frac{\partial \mathbf{R}_{i'}^{i-1^T}}{\partial q_i} \mathbf{R}_{i-1}^{0^T} \mathbf{R}^0_{i-1} \mathbf{R}^{i-1}_{i'} \mathbf{R}^{i'}_e \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \\ &= 2F_{torque,x} r \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \mathbf{R}_e^{i'^T} \frac{\partial \mathbf{R}_{i'}^{i-1^T}}{\partial q_i} \mathbf{R}^{i-1}_{i'} \mathbf{R}^{i'}_e \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \,. \end{aligned} \tag{2.20}$$

Now, according to (2.2), it is possible to write

$$\frac{\partial \mathbf{R}_{i'}^{i-1^T}}{\partial q_i}\mathbf{R}_{i'}^{i-1} = \begin{bmatrix} -\sin(q_i) & \cos(q_i) & 0 \\ -\cos(q_i) & -\sin(q_i) & 0 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} \cos(q_i) & -\sin(q_i) & 0 \\ \sin(q_i) & \cos(q_i) & 0 \\ 0 & 0 & 0 \end{bmatrix} = \\ = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{2.21}$$

Then, defining the matrix $\mathbf{R}_e^{i'}$ as

$$\mathbf{R}_e^{i'} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \tag{2.22}$$

and substituting (2.21) into (2.20), gives

$$Q_{torque,x,i} = 2F_{torque,x}r\left(R_{12}R_{23} - R_{22}R_{13}\right). \tag{2.23}$$

Since the columns of $\mathbf{R}_e^{i'}$ are the direction cosines respectively of the x, y and z axes of reference frame e expressed in reference frame i' (see Subsection 2.1.1), the following relation holds

$$\begin{bmatrix} R_{12} \\ R_{22} \\ R_{32} \end{bmatrix} \times \begin{bmatrix} R_{13} \\ R_{23} \\ R_{33} \end{bmatrix} = \begin{bmatrix} R_{11} \\ R_{21} \\ R_{31} \end{bmatrix}. \tag{2.24}$$

Then

$$R_{22}R_{33} - R_{23}R_{32} = R_{11} \tag{2.25a}$$

$$R_{32}R_{13} - R_{12}R_{33} = R_{21} \tag{2.25b}$$

$$R_{12}R_{23} - R_{13}R_{22} = R_{31} \tag{2.25c}$$

. Finally, substituting (2.25c) into (2.23) yields

$$Q_{torque,x,i} = 2F_{torque,x}rR_{31}. \tag{2.26}$$

At the same time, defining $\boldsymbol{\mu}_{e,ext,x}$ as a torque vector such that its components expressed in the end effector frame are

$$\boldsymbol{\mu}_{e,ext,x}^{e} = \begin{bmatrix} \mu_{e,ext,x} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2F_{torque,x}r \\ 0 \\ 0 \end{bmatrix} \tag{2.27}$$

and considering (2.7), the following equivalence holds

$$\mathbf{j}_{Oi}^T\boldsymbol{\mu}_{e,ext,x} = \mathbf{z}_{i-1}^T\boldsymbol{\mu}_{e,ext,x}. \tag{2.28}$$

Then, remembering that in the DH convention the z_{i-1} axis and the $z_{i'}$ axis share the same direction (see Figure 2.1) and recalling (2.22), it is possible to write

$$\begin{aligned}\mathbf{z}_{i-1}^{0^T}\boldsymbol{\mu}_{e,ext,x}^{0} = \mathbf{z}_{i'}^{0^T}\boldsymbol{\mu}_{e,ext,x}^{0} = \left(\mathbf{R}_{i'}^{0}\mathbf{z}_{i'}^{i'}\right)^T \mathbf{R}_{e}^{0}\boldsymbol{\mu}_{e,ext,x}^{e} = \mathbf{z}_{i'}^{i'^T}\mathbf{R}_{0}^{i'}\mathbf{R}_{e}^{0}\boldsymbol{\mu}_{e,ext,x}^{e} = \\ = 2F_{torque,x}r\begin{bmatrix}0 & 0 & 1\end{bmatrix}\mathbf{R}_{e}^{i'}\begin{bmatrix}1\\0\\0\end{bmatrix} = 2F_{torque,x}rR_{31}\,.\end{aligned} \tag{2.29}$$

Now, comparing (2.26), (2.28) and (2.29) gives

$$Q_{torque,x,i} = \mathbf{j}_{Oi}^{T}\boldsymbol{\mu}_{e,ext,x}\,, \tag{2.30}$$

which implies

$$\mathbf{Q}_{torque,x} = \mathbf{J}_{O}^{T}\boldsymbol{\mu}_{e,ext,x}\,. \tag{2.31}$$

In an analogous manner it is possible to prove that

$$\mathbf{Q}_{torque,y} = \mathbf{J}_{O}^{T}\boldsymbol{\mu}_{e,ext,y} \tag{2.32a}$$

$$\mathbf{Q}_{torque,z} = \mathbf{J}_{O}^{T}\boldsymbol{\mu}_{e,ext,z}\,, \tag{2.32b}$$

where

$$\boldsymbol{\mu}_{e,ext,x}^{e} = \begin{bmatrix}0\\ \mu_{e,ext,y}\\0\end{bmatrix} \tag{2.33a}$$

$$\boldsymbol{\mu}_{e,ext,x}^{e} = \begin{bmatrix}0\\0\\ \mu_{e,ext,z}\end{bmatrix}\,. \tag{2.33b}$$

Then it is possible to conclude that

$$\mathbf{Q}_{ext} = \mathbf{J}_{P}^{T}\mathbf{f}_{e,ext} + \mathbf{J}_{O}^{T}\boldsymbol{\mu}_{e,ext,x} + \mathbf{J}_{O}^{T}\boldsymbol{\mu}_{e,ext,y} + \mathbf{J}_{O}^{T}\boldsymbol{\mu}_{e,ext,z} = \mathbf{J}_{P}^{T}\mathbf{f}_{e,ext} + \mathbf{J}_{O}^{T}\boldsymbol{\mu}_{e,ext} = \mathbf{J}^{T}\mathbf{F}_{e,ext}\,, \tag{2.34}$$

which proves the initial relation.

This explanation also provides an insight on the internal actions due to motor torques. In fact, when the motor on joint i applies a torque τ_i on the i-th link, the resulting generalized force is $\mathbf{J}_{i}^{i}\begin{bmatrix}0\\\vdots\\\tau_i\end{bmatrix}$, where $\mathbf{J}_i$ is the Jacobian with only the columns from 1 to i, namely

$$\mathbf{J}_{i} = \begin{bmatrix}\mathbf{j}_{P1} & \dots & \mathbf{j}_{Pi} & \mathbf{0} & \dots & \mathbf{0}\\ \mathbf{j}_{O1} & \dots & \mathbf{j}_{Oi} & \mathbf{0} & \dots & \mathbf{0}\end{bmatrix}\,. \tag{2.35}$$

However, due to the third law of dynamics, if a torque τ_i is applied on the i-th link, also a torque $-\tau_i$ is applied on the $i-1$-th link, having a corresponding generalized

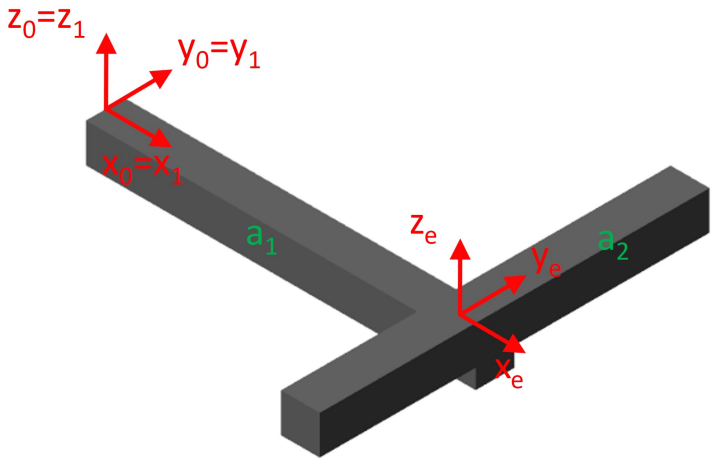


FIGURE 2.3: A simple two-link planar structure (the revolute joints are in correspondence with z_1 and z_e axes).

force of $\mathbf{J}_{i-1}^{i}\begin{bmatrix}0\\ \vdots\\ -\tau_i\end{bmatrix}$. The overall generalized force is therefore, as expected,

$$\mathbf{J}_{i}^{i}\begin{bmatrix}0\\ \vdots\\ \tau_i\end{bmatrix}-\mathbf{J}_{i-1}^{i}\begin{bmatrix}0\\ \vdots\\ \tau_i\end{bmatrix}=\begin{bmatrix}\mathbf{0}^T & \mathbf{0}^T\\ \vdots & \vdots\\ \mathbf{j}_{Pi}^{i^T} & \mathbf{j}_{Oi}^{i^T}\\ \vdots & \vdots\\ \mathbf{0}^T & \mathbf{0}^T\end{bmatrix}\begin{bmatrix}0\\ \vdots\\ \tau_i\end{bmatrix}=\begin{bmatrix}0\\ \vdots\\ \tau_i\\ \vdots\\ 0\end{bmatrix} \tag{2.36}$$

This proof allows to distinguish the action of external forces and torques from the action of the motors actuating the robot joints. This explanation will be made by means of a simple planar example. The planar structure in Figure 2.3 consists of two links connected by two revolute joints, one having length a_1 and connecting the fixed base to one end of the first link and the other one having length a_2 and connecting the other end of the first link to the middle point of the second link. The Jacobian of this structure, expressed in the base reference frame, is

$$\mathbf{J}^0=\begin{bmatrix}-a_1\sin(q_1)-a_2\sin(q_1+q_2) & -a_2\sin(q_1+q_2)\\ a1\cos(q_1)+a_2\cos(q_1+q_2) & a_2\cos(q_1+q_2)\\ 0 & 0\\ 0 & 0\\ 0 & 0\\ 1 & 1\end{bmatrix}. \tag{2.37}$$

If a system of external forces and torques $\mathbf{F}_{e,ext}$ is applied on the end effector such that

$$\mathbf{F}^0_{e,ext} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \tau_z \end{bmatrix}, \tag{2.38}$$

the resulting vector of generalized forces will be

$$\mathbf{J}^{0^T}\mathbf{F}^0_{ext} = \begin{bmatrix} \tau_z \\ \tau_z \end{bmatrix}. \tag{2.39}$$

The result of having a torque component on both joints might lead to the intuitive, but wrong, conclusion, that both links will move. Instead, only the second one moves according to the equations of dynamics. In fact, the τ_z generalized force on the first joint is compensated by the coupled inertia term $\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}}$ from the second link (see (2.8)). Both links, instead, would move if an external torque τ_z were applied on the second link and also another external torque of the same magnitude were applied on the first link.

At the same time, if no external forces are applied on the end effector, but only a motor torque τ_2 by the second motor, the second link will start moving counterclockwise, but also the first link will start moving clockwise, because of the reaction torque $-\tau_2$ acting on the side of the motor fixed to the first link. Instead, if a torque τ_2 is provided by the second motor and a torque τ_1, having the same magnitude of τ_2, is provided by the first motor, only the second link will move, because the reaction torque $-\tau_2$ is compensated by the action of the first motor.

From the previous considerations, it is possible to draw the following important conclusions:

- The vector $\mathbf{J}^T\mathbf{F}_{e,ext}$ is a vector of generalized forces, which must not be confused with a vector of equivalent external torques applied on the links.
- On the contrary, applying $\mathbf{F}_{e,ext}$ on the end effector is equivalent to applying internal motor torques having a value of $\mathbf{J}^T\mathbf{F}_{e,ext}$.

2.2 State-of-the-art control techniques

Physical human-robot interaction is a widely researched and expanding topic, due to the advancements in the collaborative robotics field, described in Chapter 1.

The most common way to implement hand guiding in a collaborative context is imposing a desired end effector behavior, in the form

$$\mathbf{\Lambda}_d \dot{\mathbf{v}} + \mathbf{D}_d \mathbf{v} = \mathbf{F}_{ext}\,, \tag{2.40}$$

where $\mathbf{v}$ is a 6-dimensional vector containing the linear and angular end effector velocities, $\mathbf{F}_{ext}$ is a 6-dimensional vector containing the end effector forces and torques, $\mathbf{\Lambda}_d$ is the desired inertia matrix and $\mathbf{D}_d$ is the desired damping matrix. The inertia and damping matrices have dimension 6x6 and are usually diagonal, to obtain decoupled dynamics.

There are basically two main approaches to treat this kind of problem, which are impedance control and admittance control. Both impose a desired end effector dynamic behavior, with impedance control imposing it at the joint torques level, and admittance control imposing it at the joint velocities level.

Impedance control strategies are founded upon the works of Hogan ([33]) and Khatib ([38]). In this framework the dynamics of the operational space and those of the joints space are put in relation to each other. Remembering the joints space Lagrangian dynamic equations

$$\mathbf{B}\ddot{\mathbf{q}} + \mathbf{c} + \mathbf{g} = \boldsymbol{\tau}_c + \mathbf{J}^T \mathbf{F}_{ext}\,, \tag{2.41}$$

the operational space equations of motion can be represented as

$$\mathbf{\Lambda}\dot{\mathbf{v}} + \boldsymbol{\mu} + \mathbf{F}_g = \mathbf{F}_c + \mathbf{F}_{ext}\,, \tag{2.42}$$

where $\mathbf{\Lambda}$ is the end effector inertia matrix, $\boldsymbol{\mu}$ is the vector of the end effector centrifugal and Coriolis forces, $\mathbf{F}_g$ is the end effector gravity vector and $\mathbf{F}_c$ are the end effector forces and torques corresponding to the joint command torques. For the previous quantities the following relation with the joints space dynamic model hold:

$$\mathbf{\Lambda} = \left(\mathbf{J}\mathbf{B}^{-1}\mathbf{J}^T\right)^{-1}\,, \tag{2.43}$$

$$\boldsymbol{\mu} = \mathbf{\Lambda}\mathbf{J}\mathbf{B}^{-1}\mathbf{c} - \mathbf{\Lambda}\dot{\mathbf{J}}\dot{\mathbf{q}}\,, \tag{2.44}$$

$$\mathbf{F}_g = \mathbf{J}^{\dagger T}\mathbf{g}\,, \tag{2.45}$$

$$\mathbf{F}_c = \mathbf{J}^{\dagger T}\boldsymbol{\tau}_c\,. \tag{2.46}$$

As a result, the desired end effector behavior in (2.40) can be obtained by sending to the motors the following torque commands

$$\boldsymbol{\tau}_c = \mathbf{J}^T\left(\boldsymbol{\mu} + \mathbf{F}_g - \mathbf{\Lambda}\mathbf{\Lambda}_d^{-1}\mathbf{D}_d\mathbf{v} + \left(\mathbf{\Lambda}\mathbf{\Lambda}_d^{-1} - \mathbf{I}\right)\mathbf{F}_{ext}\right)\,. \tag{2.47}$$

It is also possible to apply this approach imposing only a translational behavior, as in [22]. In this case, equations (2.41)-(2.47) are the same, but the end effector force and velocity vectors include only the three translational elements, and the Jacobian includes only the first three rows related to the translational end effector velocity.

This approach is used very frequently in the context of physical human-robot interaction [52, 58, 51, 40, 80]. Often the impedance is made variable during the motion

to improve the performance and guarantee stability ([22, 2, 31, 53, 34]), sometimes also introducing an approximated model of the human arm impedance ([77, 47, 76, 10]) and learning techniques ([75, 30, 61]).

Admittance control ([62, 46, 57, 32]) seeks as well to obtain a desired end effector behavior according to (2.40), yet its output is not a vector of joint torques but of joint velocities, computed based on the forces and torques vector (constituting the input in this context) and through integration of the end effector velocity. Another lower level controller must then take care of tracking the desired admittance joint velocities. Even in the case of admittance control, varying the desired inertial and/or damping parameters is quite frequent in the literature ([50, 4, 21, 16, 16, 65]).

2.3 Hand guiding application setup

As already introduced in Section 1.3, the application considered in the present thesis is a HG collaborative scenario for drilling operations. With respect to Figure 1.11, the IMU tracking system is not included in the scope of this work. In this Section, some details regarding the components of the robotic system will be provided that will allow understanding better the results of the thesis.

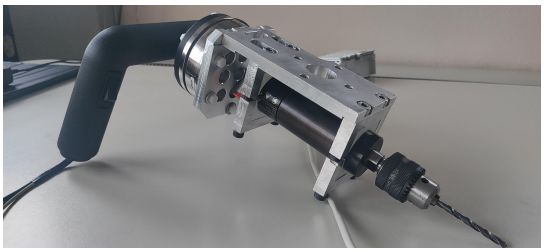
2.3.1 Robotic arm

The Franka Emika Panda robotic system consists of Franka Arm, the actual robotic arm, and Franka Control, a control station connected to the robot through a proprietary-technology cable. This control station contains power-conditioning devices and a computational unit to calculate the commands to be provided to the drivers and to elaborate the data coming from the sensors.

Franka Control can be connected to a workstation PC via an Ethernet cable. The PC can communicate with Franka Control through the Franka Control Interface (FCI). Data from the sensors and commands to the drivers are sent in real time with a cycle time of 1 ms, so the controllers running on the workstation PC must have a computing time of less than 1 ms. If this is not the case, the robot will enter an error state.

The command signals can be sent to the FCI through five possible command interfaces, joint torque interface, joint position interface, joint velocity interface, end effector position interface and end effector velocity interface. In the thesis only the joint torque interface and the joint velocity interface will be used. Regarding the joint torque interface, it needs to be specified that the command signal is the link-side joint torque without the gravity contribution, which is added by default to the command by Franka Control.

The data received from FCI at each cycle time, instead, are the encoders and torque sensors signals, other values obtained from the data from those sensors and the kinematic and dynamic parameters of the robot. In particular, FCI provides the inertia matrix, the Coriolis vector, the gravity vector and the Jacobian. It has to be specified, that, while the kinematic parameters in the Denavit-Hartenberg formulation are made available by the manufacturer, the dynamic parameters (i.e., the masses, centers of mass and inertia matrices of the links) are not provided. They are known internally by the computing unit within Franka Control, but this information is not shared with the user, who only has access to the values of the inertia matrix, Coriolis vector and gravity vector at the current iteration, not to their general expression.



(a)

(b)

FIGURE 2.4: Drill end effector.

2.3.2 End effector

The tests for the present thesis were performed with a drilling end effector mounted on the final flange of the robot. The force/torque sensor already cited in 1.3 was placed before the tool grabbed by the human operator.

The end effector is shown in Figure 2.4. It consists of a metal structure holding a pneumatic drill. The force/torque sensor is connected on one side to the metal structure and on the other side to a 3D-printed handle that is grabbed with both hands by the user. On the part of the handle that is held by the right hand a trigger is present to allow starting the rotation of the drill when the trigger is pressed and stopping the rotation when it is released. This trigger, in fact, is connected to a 4/2 pneumatic valve that allows or avoids the flux of compressed air to the pneumatic drill.

To properly control the robot, it is needed to know the mass and center of mass of the end effector structures. The developers of Franka, in fact, require that these pieces of data be manually inserted within the robotic system when a custom end effector is mounted on the terminal flange.

The mass of the end effector could easily be known by measuring it with a balance, but there was no way to know the center of mass, because a CAD model of the pneumatic drill and of the force torque/sensor is not available.

Then the following method is used to estimate these parameters. According to [28, 27, 37], the dynamic parameters can be retrieved by exploiting the property that the dynamic model (2.8) is linear in the parameters, namely, it can be rewritten as

$$\mathbf{Y}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right) \begin{bmatrix} \boldsymbol{\pi}_1 \\ \vdots \\ \boldsymbol{\pi}_n \end{bmatrix} = \mathbf{Y}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right) \boldsymbol{\pi} = \boldsymbol{\tau}\,, \tag{2.48}$$

where n is the number of joints and $\boldsymbol{\pi}_i$ is the 11-dimensional vector of dynamic parameters of link i, being defined as

$$\boldsymbol{\pi}_i = \begin{bmatrix} m_i & m_i l_{C_i x} & m_i l_{C_i y} & m_i l_{C_i z} & \hat{I}_{ixx} & \hat{I}_{ixy} & \hat{I}_{ixz} & \hat{I}_{iyy} & \hat{I}_{iyz} & \hat{I}_{izz} & I_{m_i} \end{bmatrix}^T, \tag{2.49}$$

where m_i is the mass of the i-th link, $l_{C_i x}$, $l_{C_i y}$ and $l_{C_i z}$ are the positions in x, y and z of the center of mass of the i-th link, $\hat{I}_{ixx}$, $\hat{I}_{ixy}$, $\hat{I}_{ixz}$, $\hat{I}_{iyy}$, $\hat{I}_{iyz}$, $\hat{I}_{izz}$ are the elements of the

inertia tensor of the i-the link and I_{m_i} is the moment of inertia of the i-th motor.

Since the torques on the right side of (2.48) are available from the torque sensors readings, if a large number of measurements are recorded, the vector $\boldsymbol{\pi}$ can be found by stacking these measurements as

$$\bar{\mathbf{Y}}\left(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}\right) \boldsymbol{\pi} = \bar{\boldsymbol{\tau}} \,. \tag{2.50}$$

The regressor matrix $\bar{\mathbf{Y}}$ is in general not full-rank. So it has to be pruned of the linearly dependent columns, obtaining a full column rank regressor matrix $\bar{\mathbf{Y}}_R$. Then a set of independent feasible regrouped parameters (i.e., a linear combination of the real dynamic parameters) is defined as

$$\boldsymbol{\pi} = \mathbf{L}\boldsymbol{\pi}_R \,, \tag{2.51}$$

where $\mathbf{L}$ is the matrix expressing the linear combination of the dependent parameters, and these parameters can be estimated as

$$\boldsymbol{\pi}_R = \bar{\mathbf{Y}}_R^{\dagger} \bar{\boldsymbol{\tau}} \,. \tag{2.52}$$

In the case under consideration, the parameters of the inertia tensors and of the moments of inertia of the motors are not of interest, so the dimension of the vectors $\boldsymbol{\pi}_i$ is only 4, and the measurements needed to estimate the parameters could be taken in static conditions (if also the inertial parameters had been of interest it would have been necessary to execute and record trajectories that could be able to excite the inertial modes of all the links). In particular, the testing procedure consisted of the robot being moved by a random feasible (i.e. without self collisions or collisions with the environment) and of the recording of the joint positions and joint torques once the robot was stopped. The data relative to 200 random positions were collected without any end effector and with the drill end effector. Then the regrouped dynamic parameters were retrieved, and finally the real dynamic parameters were obtained through (2.51).

To estimate the mass and center of mass of the end effector it has to be considered that the mass of center of mass of the seventh link are known from the parameter estimation in the unloaded case, and that, for the loaded case, the elements of $\boldsymbol{\pi}$ relative to the overall seventh link (seventh link of Franka arm + custom end effector) are $m_7 + m_e$, $m_7 l_{C_7 x} + m_e l_{C_e x}$, $m_7 l_{C_7 y} + m_e l_{C_e y}$ and $m_7 l_{C_7 z} + m_e l_{C_e z}$, m_e being the mass of the custom end effector and $l_{C_e x}$, $l_{C_e y}$ and $l_{C_e z}$ being the positions of the center of its center of mass in the x, y and z coordinates, with respect to the flange reference frame. Then, the mass and center of mass of the end effector can simply be found by solving a linear system of four equations in four unknowns.

The resulting estimated mass was then found to be 1.845 kg for the drill end effector. Its real value, measured with a balance, was instead 1.839 kg, allowing to conclude that the method had provided reliable results. The positions of the center of mass in x, y and z were 0.0298 m, -0.0304 m and 0.0519 m. As already noted, the precision of these last results could not be checked exactly. However, an indication of their plausibility is given by the fact that, when these parameters were applied to the system, the robot arm remained in equilibrium when 0 torque was applied, meaning that the internal gravity compensation was effective.

2.3.3 Experimental procedure

The described setup was employed to test several control techniques, which will be described in detail in the following Sections. The tests will involve a human operator guiding the end effector through predefined simple trajectories (with diagonal, triangular or circular paths), which were drawn on a surface so that they could be followed more easily by the experimenter and so that the tests could be more repeatable.

The main goals of these tests are the following:

1. assess how precisely the operator can follow the desired trajectory: if the controller is appropriate the behavior of the end effector should be transparent, meaning that it should be very easy to guide the end effector where it is desired. Instead, if the controller does not perform suitably, there will be the need of frequent corrective actions along the trajectory, resulting in a path that does not resemble very much the desired (diagonal, circular or triangular) one.

2. assess how much effort is needed to perform the trajectory: the data measured by the force/torque sensor along the trajectory will be reported, the aim of the controller being especially to minimize the force needed to execute the task. Additionally, the end effector velocity is also important, because if the controller guarantees a smooth and fluid behavior the operator will go through the trajectory more quickly.

The tests were performed first guiding the end effector in the translational and rotational degrees of freedom, and then only in the translational degrees of freedom, to evaluate the performance of the collaborative robotic platform in different operative conditions.

2.4 Impedance control

2.4.1 Issues with the Franka joint torque interface

The first possible way of controlling the robot that was examined is impedance control, performed according to the framework described in Section 2.2, providing through the Franka torque interface a control command equal to

$$\boldsymbol{\tau}_c = \mathbf{J}^T \left(\boldsymbol{\mu} - \boldsymbol{\Lambda}\boldsymbol{\Lambda}_d^{-1}\mathbf{D}_d\mathbf{v} + \left(\boldsymbol{\Lambda}\boldsymbol{\Lambda}_d^{-1} - \mathbf{I} \right) \mathbf{F}_{ext} \right) + \mathbf{g}. \tag{2.53}$$

In the first test the desired inertia and damping matrices were chosen as diagonal matrices to decouple the dynamics. The translational desired inertia was chosen as 5 kg, the rotational desired inertia as $0.5\,\mathrm{kg\,m^2}$, the translational desired damping as $0.5\,\mathrm{N\,s/m}$ and the rotational damping as $0.5\,\mathrm{N\,s}$. Starting from the initial joint configuration $[0, -0.7854, 0, 2.3562, 0, 2.0071, 0]$ rad that can be seen in Figure 2.5 (all the tests will start from this same initial configuration), the human operator applied a torque around the z axis of the end effector frame, which coincides with the flange frame indicated in Figure 2.6.

The results of this simple experiment are presented in Figures 2.7–2.10. What would be intuitively expected is a movement of only the seventh joint to follow the rotation impressed by the human hand. Instead, as can be observed in Figure 2.7, none of the joints has significant motions, in spite of the fact that a high torque is applied around the z axis (see Figure 2.8). In fact, as can be seen from Figure 2.9,

FIGURE 2.5: Starting position for all the tests.

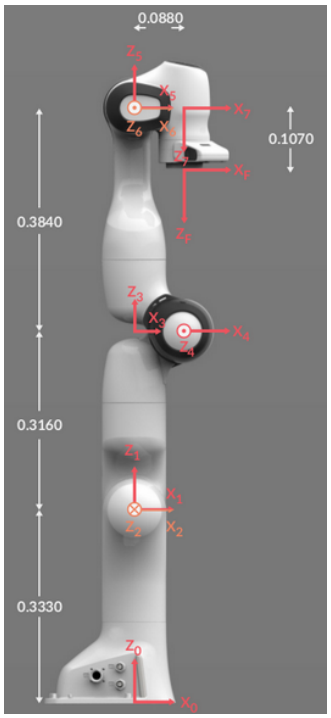


FIGURE 2.6: Reference frames of the robot, defined according to the Denavit-Hartenberg convention.

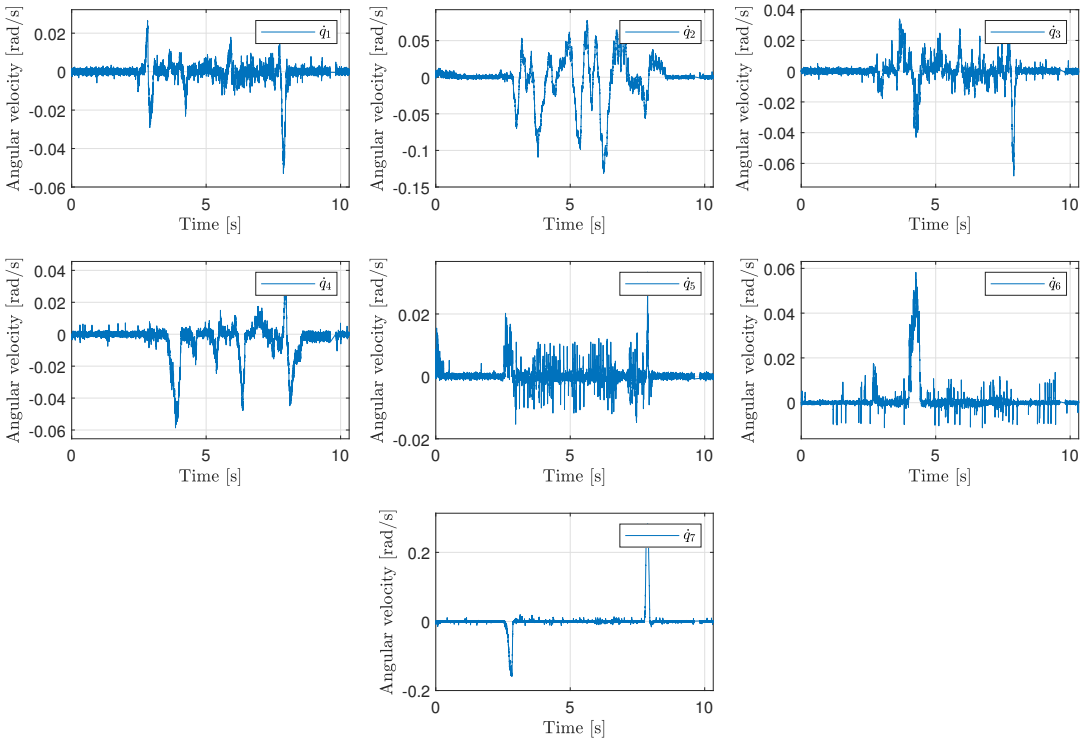


FIGURE 2.7: Joint velocities in the impedance control test on the rotation about the z axis.

the control torque $\tau_{c,7}$ almost completely compensates $\tau_{ext,7}$, which is the seventh element of the vector $\boldsymbol{\tau}_{ext} = \mathbf{J}^T\mathbf{F}_{ext}$. The sum of these two contributions, referred to as $\tau_{genforces,7}$, is almost null, resulting in an absence of motion of the seventh joint.

To provide an explanation for this phenomenon, it is needed to consider, on the one hand, the inertia matrix of the robot, and, on the other hand, the nature of the transmission system within the joints. In the initial configuration, the inertia matrix is equal to

$$\mathbf{B} = \begin{bmatrix} 0.4805 & -0.0061 & 0.4463 & -0.0006 & 0.0447 & 0.0021 & -0.0044 \\ -0.0061 & 1.5087 & -0.0220 & -0.6531 & -0.0088 & -0.0359 & -0.0005 \\ 0.4463 & -0.0220 & 0.9294 & -0.0131 & 0.0506 & 0.0010 & -0.0059 \\ -0.0006 & -0.6531 & -0.0131 & 0.8886 & 0.0208 & 0.1103 & -0.0008 \\ 0.0447 & -0.0088 & 0.0506 & 0.0208 & 0.0204 & 0.0007 & 0.0015 \\ 0.0021 & -0.0359 & 0.0010 & 0.1103 & 0.0007 & 0.0403 & -0.0012 \\ -0.0044 & -0.0005 & -0.0059 & -0.0008 & 0.0015 & -0.0012 & 0.0023 \end{bmatrix} \text{kg m}^2 . \tag{2.54}$$

It can be observed that the inertia of the seventh link is very low, so a very small torque is enough to accelerate this link. As a result, when a high torque is applied with the hand on the z axis starting from the initial position, the impedance control algorithm will apply an opposing torque so that the end effector rotates about the z as if it has a rotational inertia of 0.5 kg m^2 and a rotational damping of 0.5 N s. That is the case of Figure 2.9, where the very small resulting $\tau_{genforces,7}$ should be enough to generate an acceleration about the z axis such that the desired end effector behavior is obtained.

However, this is not what happens in practice, and this is due to the transmission inside the robot joints. In the Franka Emika, in fact, the reducers are harmonic

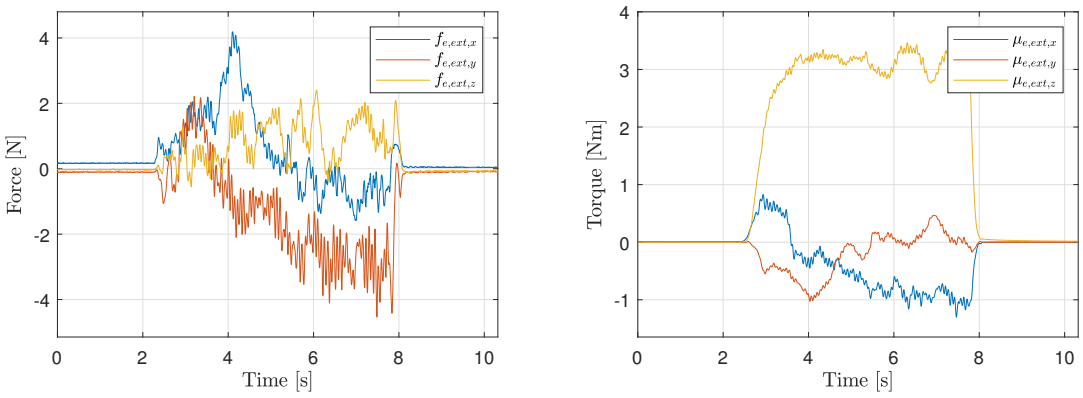


FIGURE 2.8: End effector forces and torques in the impedance control test on the rotation about the z axis.

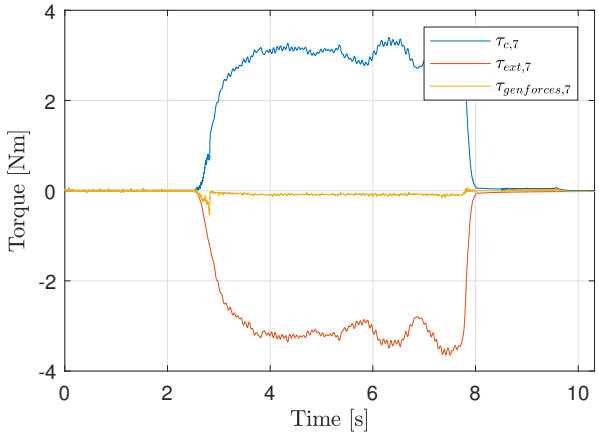


FIGURE 2.9: Control torque and external torque of the seventh joint in the impedance control test on the rotation about the z axis.

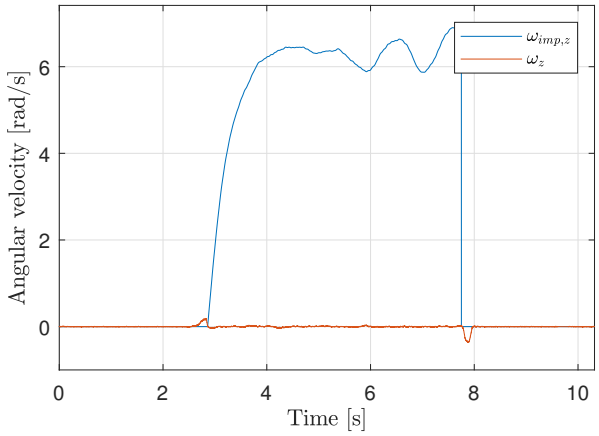


FIGURE 2.10: Measured and theoretical z-axis angular velocity in the impedance control test on the rotation about the z axis.

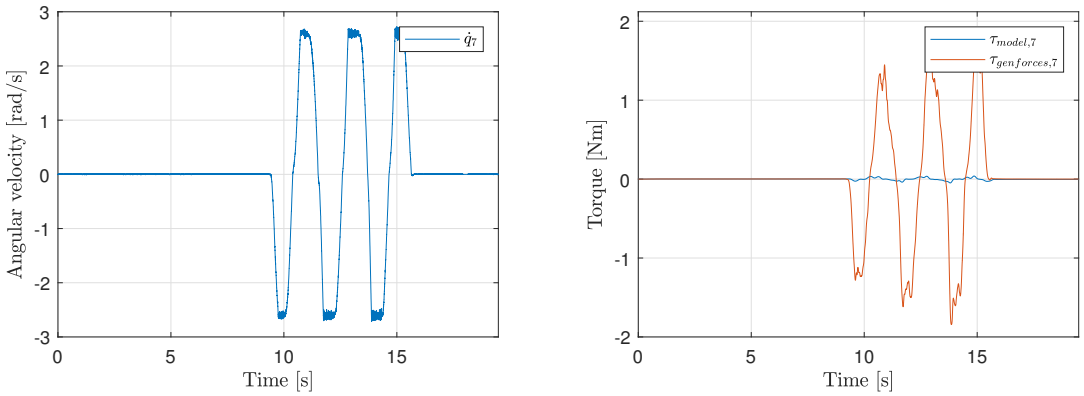


FIGURE 2.11: Seventh joint angular velocity (left) and model torques (right) in the test with only torque compensation.

drives, which are very hard to backdrive. So the problem with the impedance control scheme mainly consists in the fact that the dynamic model in (2.8) is not exact, because it does not consider an unmodelled opposing torque due to the harmonic drive. As a result, Figure 2.10 shows that there is a huge discrepancy between the expected behavior according to the desired impedance behavior (blue line showing the expected end effector angular velocity along the z axis) and the actual behavior (red line showing the measured end effector angular velocity along the z axis).

To further prove this point, another test was performed in which only the torque compensation was provided to the torque command interface. In this case, the seventh joint is actually moved, but the model of (2.8) is shown to be completely inaccurate. Figure 2.11 shows that the user, in this case, is perfectly able to rotate the seventh joint, but the left side of (2.8) ($\mathbf{B}\ddot{\mathbf{q}} + \mathbf{c} + \mathbf{g}$) is much smaller than the right side ($\boldsymbol{\tau}_c + \mathbf{J}^T\mathbf{F}_{ext}$).

An additional observation that can be made regarding the accuracy of the model is that, due to the opposing torque from the harmonic drive (which can be regarded as an energy loss), it is not even longer true that the generalized forces in the model due to the motor torques are simply a vector whose elements are the motor torques themselves, as in (2.36). In fact part of the reaction torque on the link which is previous with respect to the motor is dissipated in the harmonic drive, making the proof of Subsection 2.1.2 inexact. This was shown by applying a constant torque on the seventh joint. In such a case, according to the model, not only the seventh joint, but also the first joint should have moved, due to considerations analogous to those regarding the system in Figure 2.3. Instead, in this test only a motion of the seventh joint was observed.

On the other hand, it could be argued that, even though the model in (2.8) lacks a harmonic drive opposition torque term, it would be possible to take this into account and send a torque command that compensates for this term. However, this is actually not possible because the torque value sent to the Franka interface is not simply the one that will be commanded to the motor, but is elaborated in a "black box" way by the Franka proprietary libraries in the FCI. Considering in more detail the transmission, each of the joints is built as in Figure 2.12. According to the documentation of FCI, the torque command is the desired torque at the link following the joint (i.e., the torque after the harmonic drive). This intrinsically means that there is some control layer that is out of the scope of the user. In fact, the pure torque command would be on the motor (RoboDrive in the figure). In addition, when a certain

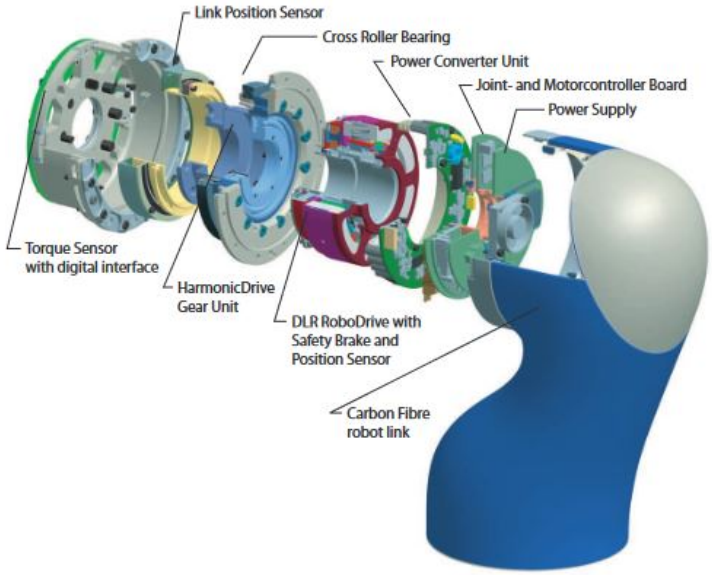


FIGURE 2.12: Transmission system within the Franka links [1].

torque command is sent to the interface, its value does not correspond to the torque that is applied after the harmonic drive.

This is shown by the fact that, when only the gravity compensation torque is applied, as in the test in Figure 2.11, the torque that the user needs to apply is significant, but not extremely high (around 1.5 N m). As a comparison, a test was performed with a structure imitating the joint transmission system (see Figure 2.13), consisting of a harmonic drive, with the same gear ratio of 50 of the Franka robot joints. The circular spline is attached to a fixed frame (simulating the preceding link), the flexible spline is fixed to a force torque sensor connected to a handle (simulating the following link) and the wave generator is connected to an encoder (simulating the motor), which is also attached to the frame. Figure 2.14 shows that the torque needed to slightly move the encoder shaft is much higher. The resulting remark is that even when a zero torque command (with gravity compensation) is sent to the joints, the motors are actively helping the motion, so the "real" torque that is applied after the harmonic drive is not only the gravity compensation.

For all these reasons, using the FCI torque interface to obtain a desired end effector behavior in the context of impedance control is problematic, since the control system of the robot is not open enough to obtain an exhaustive and precise handling of the motor torques, as will be shown in the following tests.

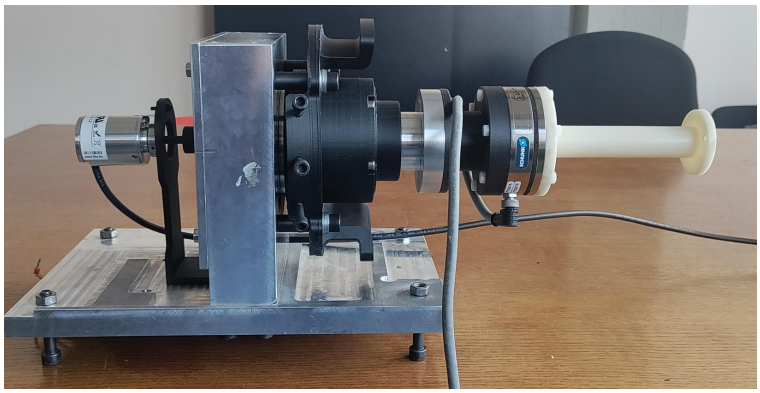


FIGURE 2.13: Test bench to study the backdrivability of the robotic joint.

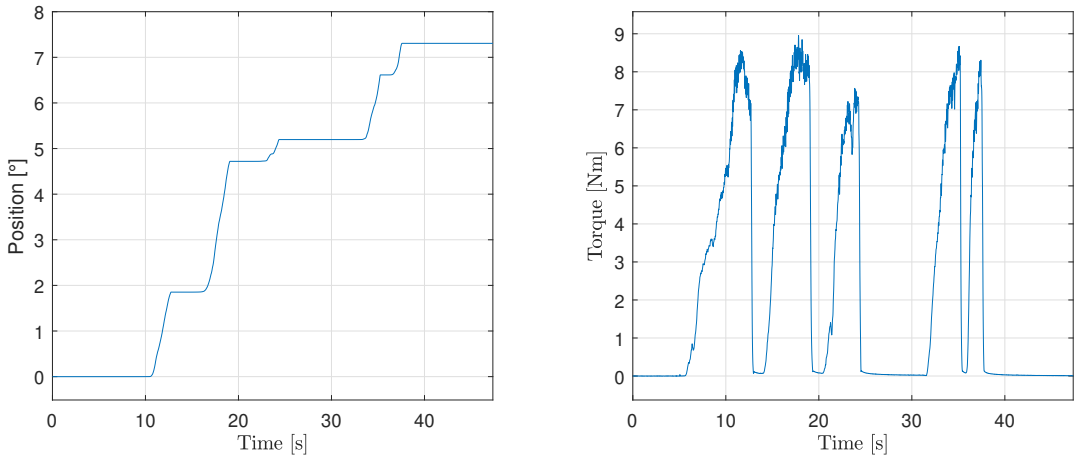


FIGURE 2.14: Encoder position and measured torque in the joint-like structure test.

2.4.2 Impedance control tests

The tests that will be presented in the following consider a case study in which the operator moves the end effector along a diagonal trajectory. In the first test the translational desired inertia was chosen as 5 kg, the rotational desired inertia as 0.5 kg m^2, the translational desired damping as 0.5 N s/m and the rotational damping as 0.5 N s. The results are shown in Figures 2.15–2.23.

As already discussed in the previous case, Figure 2.15 shows there is a significant discrepancy between the torque expected from the dynamic model (left side of (2.8)) and the one from the generalized forces from the motors and from the operator (right side of (2.8)). In Figure 2.16 it can be observed that the forces needed to perform the motion are high, but the resulting end effector velocity is low. Even more significant is the data relative to the orientation of the end effector with respect to the base reference frame. What would be desired is to maintain the same orientation along the whole trajectory, but this is not possible for the same problem that emerged in the preliminary test on the rotation about the z axis: due to the discrepancy in the dynamic model the rotation of the end effector is "locked" and therefore it is not possible to orient it as desired. The result is that, even if the user tries to apply forces to correct the orientation and to maintain it at the initial values (0° for roll and pitch and 180° for yaw), the angles vary significantly.

The outcome of these non-idealities can be seen in Figures 2.17 and 2.18. These figures are complementary: in the first one, the forces/torques that would result from the impedance model based on the measured linear/angular velocity are computed and compared to the real forces/torques; in the second one, the linear/angular velocities that would result from the impedance model based on the measured forces/torques are computed (with forward Euler integration) and compared to the real forces/torques. It can be seen that the desired decoupled dynamics which would guarantee a low effort to the operator are very far from being achieved.

Figure 2.19 shows how the external force varies along the trajectory, Figure 2.20 shows how the orientation angles vary along the trajectory: it can be observed that in the middle point of the diagonal path the orientation is the correct one, but when the end effector is moved to the terminal points of the trajectory the orientation changes even if the user applies a force to try to keep the drill in the initial orientation (Figure 2.21 shows the orientation in one of the terminal points of the diagonal path). Figure 2.22 shows how the velocity varies along the trajectory and, finally Figure 2.23 shows how the height along the z axis varies along the trajectory: it would be desired to keep the end effector approximately at the same height along all the path, but this is not possible to the problems of the impedance control scheme that were mentioned.

The same tests were also performed with lower translational inertia (1 kg) and rotational inertia (0.1 kg m^2). The results are reported in Appendix A (Figures A.1–A.8) for better readability of the thesis. It can be observed that the discrepancy between the model torques and the generalized forces is less significant and that the end effector forces are much lower, but the system shows an unstable behavior, emerging from the very significant oscillations in the graphs, due to the fact that the inertia is too low. Additionally, it is still not possible to maintain the initial orientation.

For both tests, this way of controlling the robot is not satisfactory, especially because it is not acceptable that the orientation of the end effector may change without the possibility for the operator to adjust this, even applying very high forces/torques.

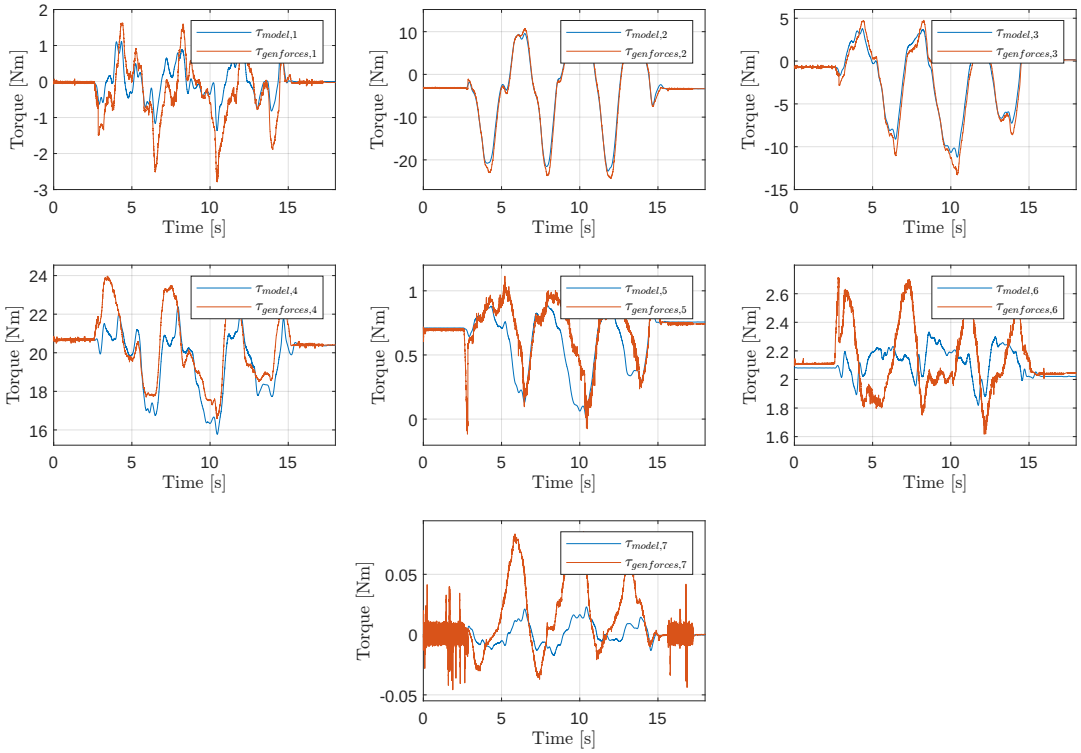


FIGURE 2.15: Diagonal motion test with impedance control, with high inertia and damping parameters: dynamic model accuracy.

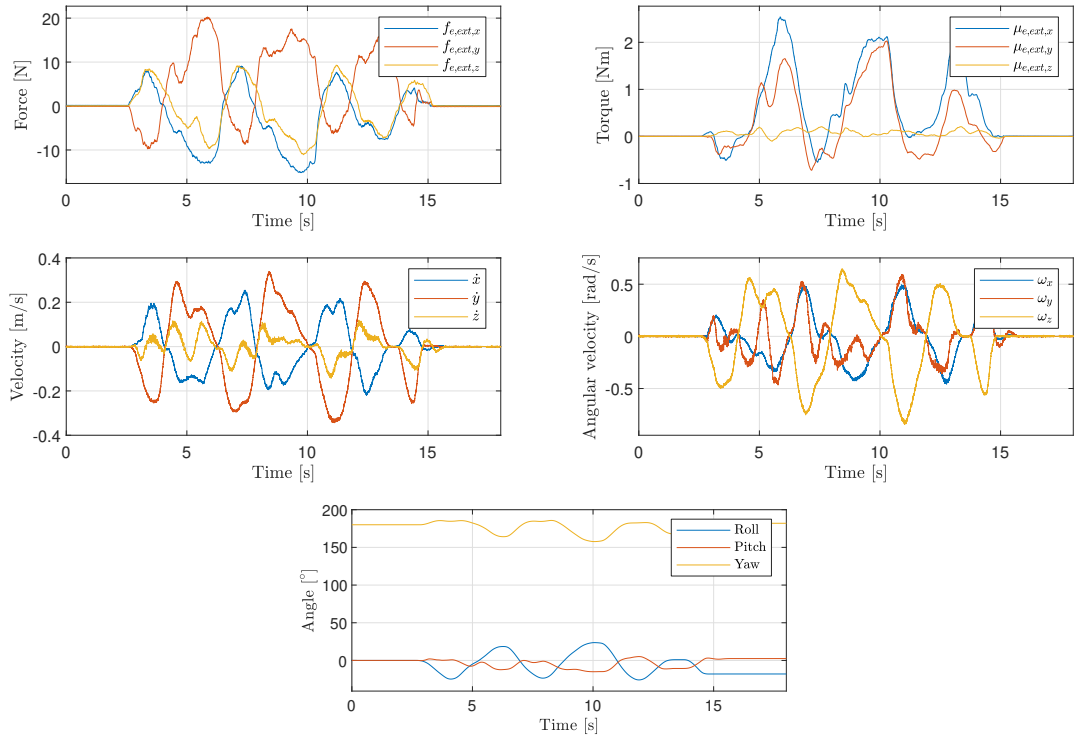


FIGURE 2.16: Diagonal motion test with impedance control, with high inertia and damping parameters: end effector forces and torques, linear and angular velocities and roll pitch and yaw angles.

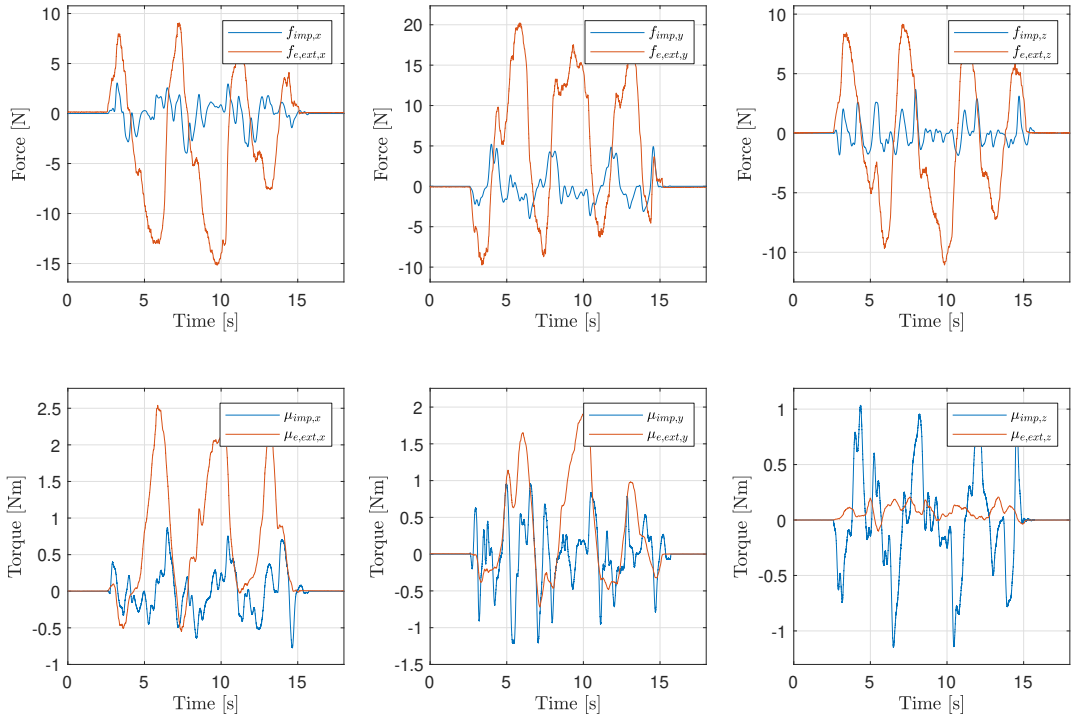


FIGURE 2.17: Diagonal motion test with impedance control, with high inertia and damping parameters: comparison of expected and measured forces/torques.

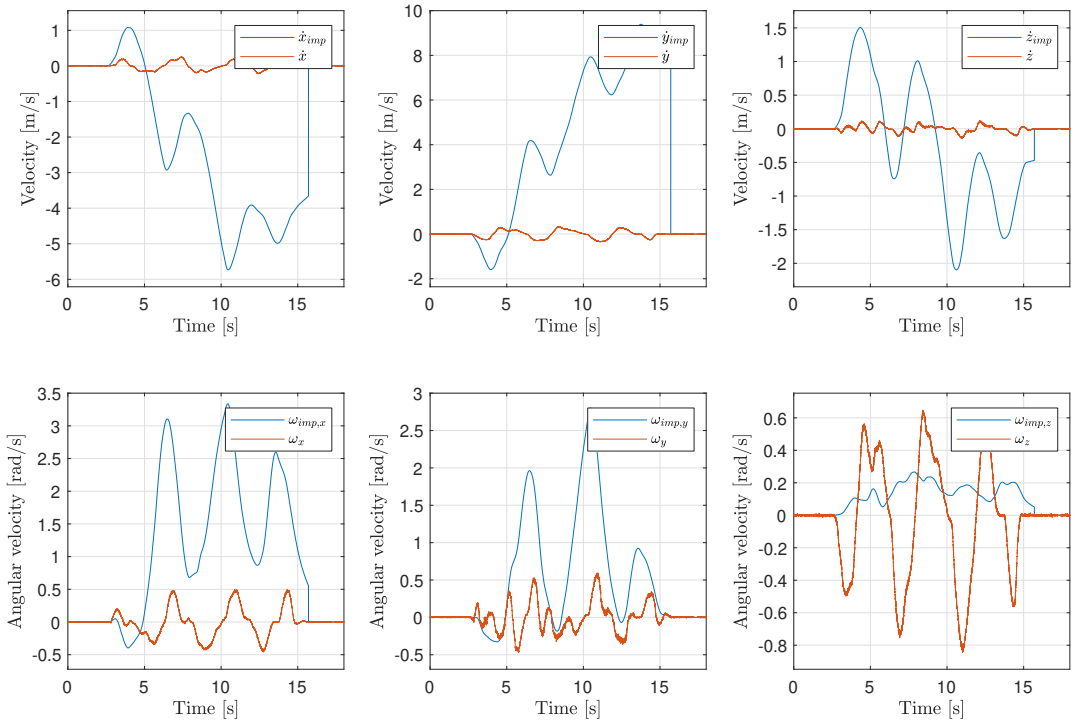


FIGURE 2.18: Diagonal motion test with impedance control, with high inertia and damping parameters: comparison of expected and measured linear/angular velocities.

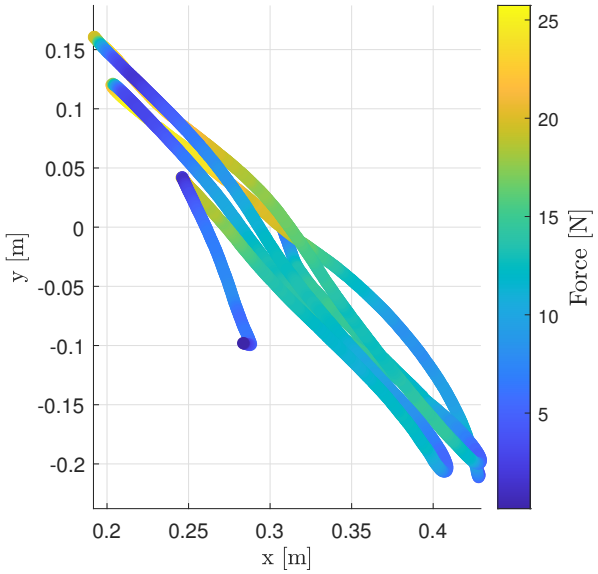


FIGURE 2.19: Diagonal motion test with impedance control, with high inertia and damping parameters: norm of the end effector force vector along the trajectory in the x-y plane.

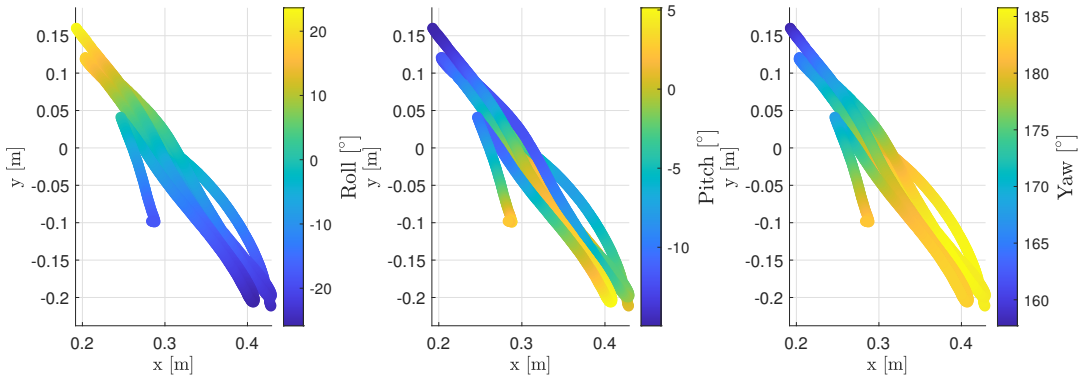
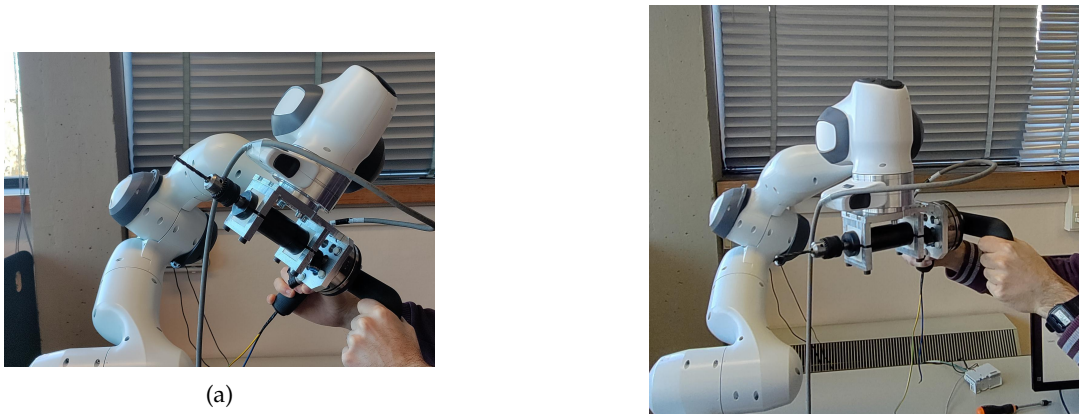


FIGURE 2.20: Diagonal motion test with impedance control, with high inertia and damping parameters: roll, pitch and yaw angles of the end effector along the trajectory in the x-y plane.



(a) (b)

FIGURE 2.21: Diagonal motion test with impedance control, with high inertia and damping parameters: robot configuration at the terminal point of the diagonal (see (a)) compared with the end effector in the same position but in the correct straight orientation (see (b)).

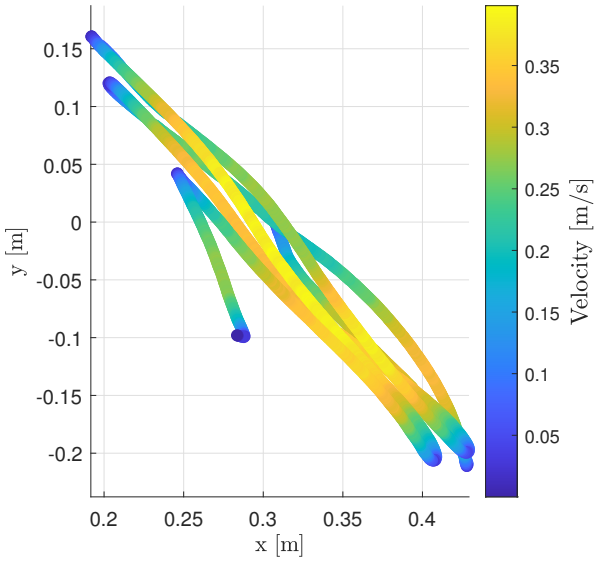


FIGURE 2.22: Diagonal motion test with impedance control, with high inertia and damping parameters: norm of the end effector velocity vector along the trajectory in the *x*-*y* plane.

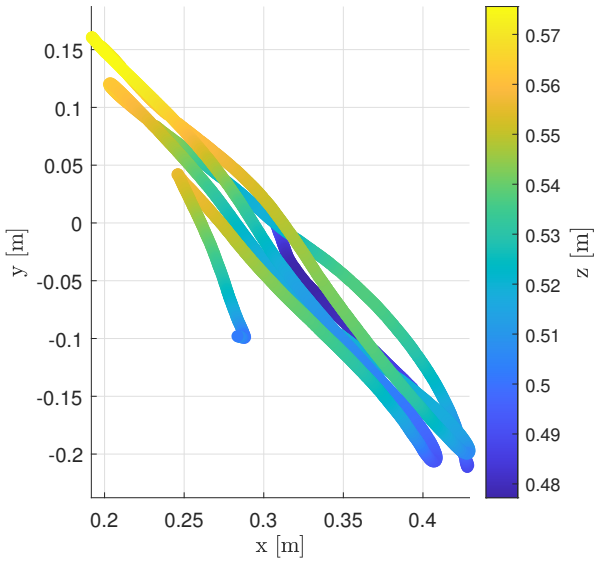


FIGURE 2.23: Diagonal motion test with impedance control, with high inertia and damping parameters: height on the z axis along the trajectory in the x-y plane.

As a result, the impedance control scheme considering only the translational part, introduced in Section 2.2, was also tested. Figures 2.24–2.31 show the results of the translational impedance control with 5 kg as translational inertia and 0.5 N s/m as translational damping. In this case, the orientation problem improves with respect to the previous tests, because the control does not aim at changing the rotational behavior of the robot. However, the forces needed to move the end effector are much larger than in the previous case and the velocity that it is possible to reach is significantly lower.

Instead, if the desired translational inertia is decreased to 1 kg the results improve significantly, as shown in Appendix A (Figures A.9–A.16). In this case, even if the ideal impedance behavior is far from being reached, the forces are lower and the path is performed with a higher velocity.

It is also noticeable that, for the translational control scheme a desired mass of 1 kg generates a more suitable behavior and a mass of 5 kg makes it very hard for the operator to move the robot, while, for the translation and rotational impedance control scheme the 5 kg mass is better and the 1 kg mass generates instability. This discrepancy between the only translational impedance mode and the translational and rotational impedance mode, which at first might seem surprising, is due to the unpredictable way in which the inaccuracies in the model affect the overall result of the two different control schemes.

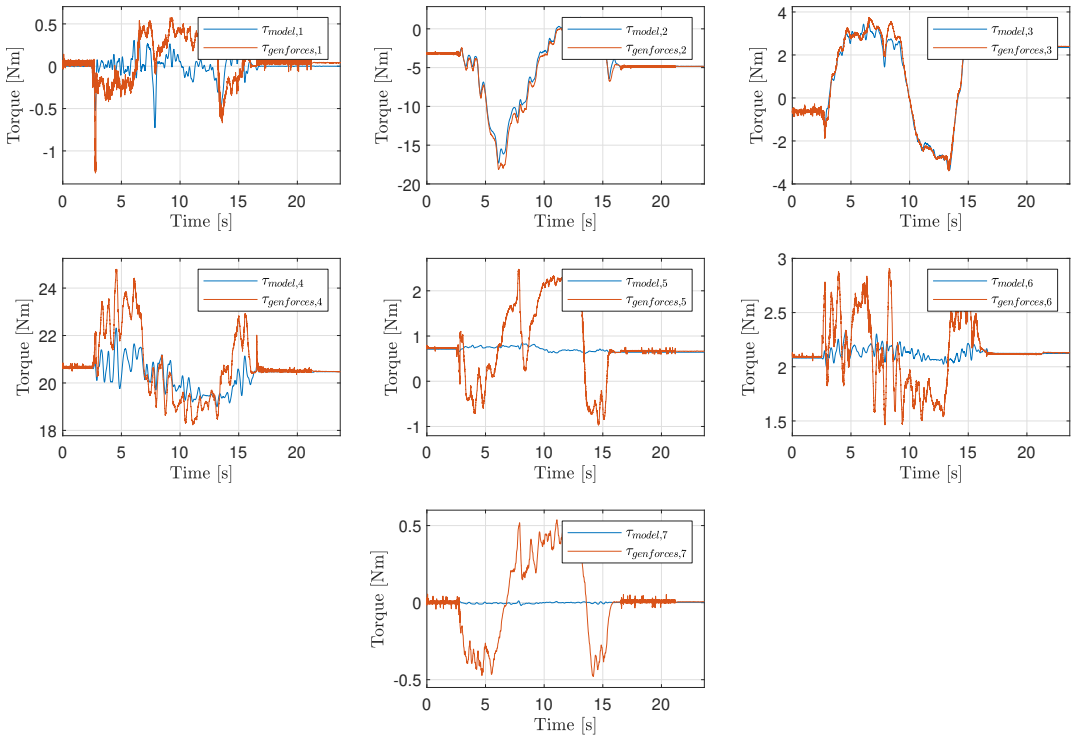


FIGURE 2.24: Diagonal motion test with translational impedance control, with high inertia and damping parameters: dynamic model accuracy.

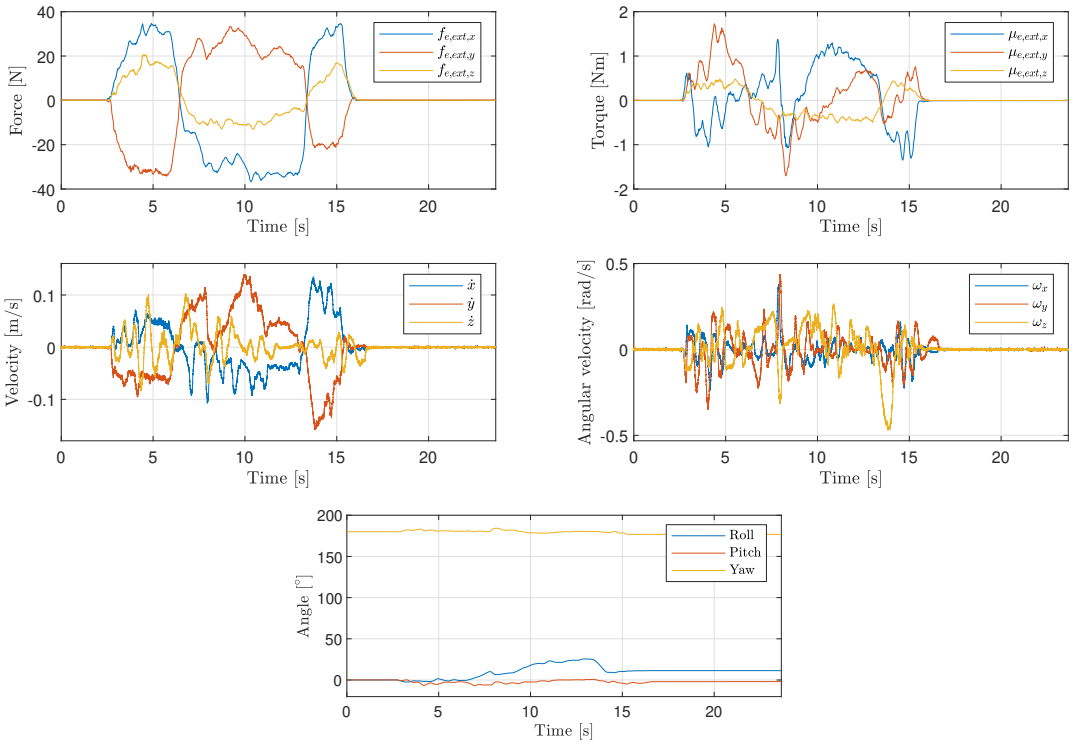


FIGURE 2.25: Diagonal motion test with translational impedance control, with high inertia and damping parameters: end effector forces and torques, linear and angular velocities and roll pitch and yaw angles.

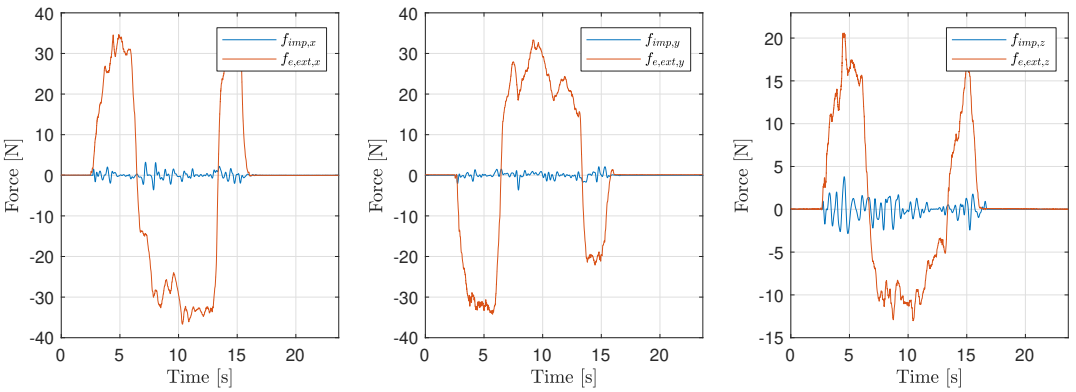


FIGURE 2.26: Diagonal motion test with translational impedance control, with high inertia and damping parameters: comparison of expected and measured forces.

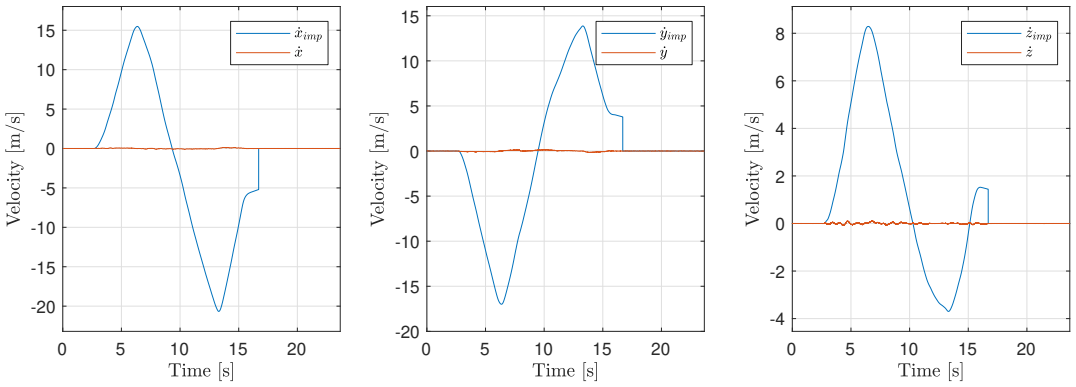


FIGURE 2.27: Diagonal motion test with translational impedance control, with high inertia and damping parameters: comparison of expected and measured linear velocities.

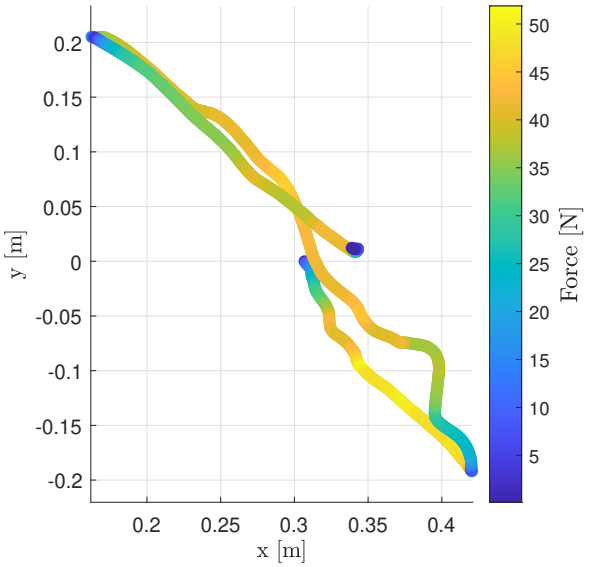


FIGURE 2.28: Diagonal motion test with translational impedance control, with high inertia and damping parameters: norm of the end effector force vector along the trajectory in the x-y plane.

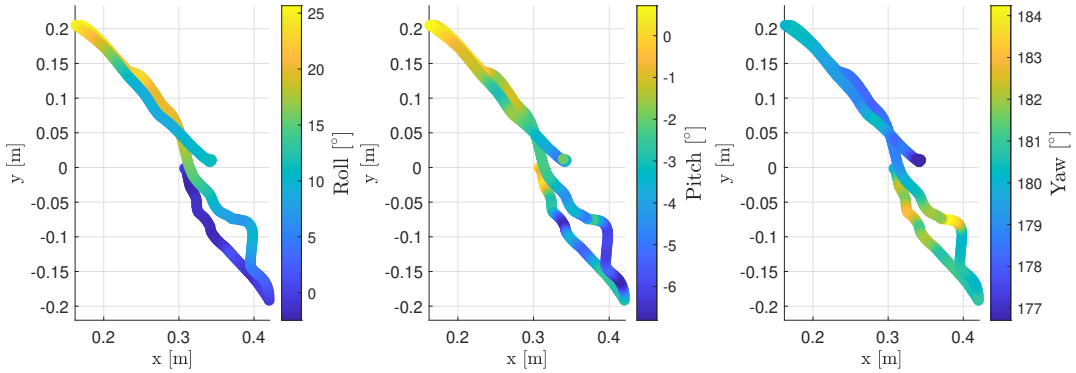


FIGURE 2.29: Diagonal motion test with translational impedance control, with high inertia and damping parameters: roll, pitch and yaw angles of the end effector along the trajectory in the x-y plane.

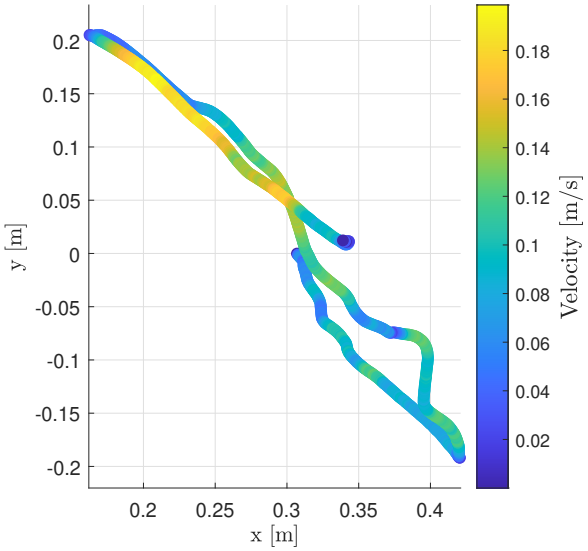


FIGURE 2.30: Diagonal motion test with translational impedance control, with high inertia and damping parameters: norm of the end effector velocity vector along the trajectory in the x-y plane.

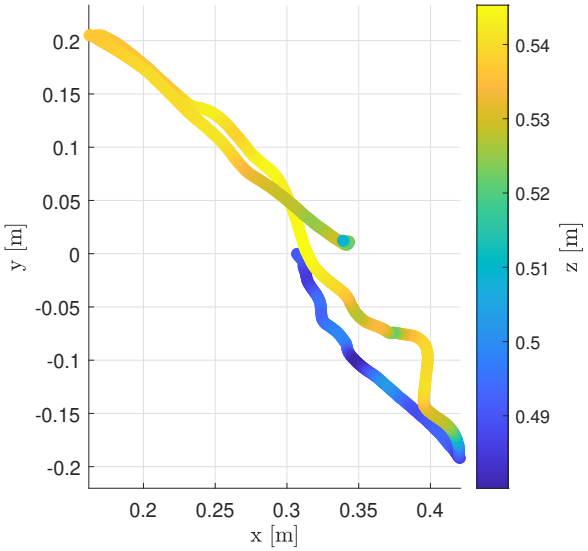


FIGURE 2.31: Diagonal motion test with translational impedance control, with high inertia and damping parameters: height on the z axis along the trajectory in the x-y plane.

The conclusion is that, if it is desired to use an impedance control scheme, it is more appropriate to choose a translational scheme with a low desired mass. To assess the performances of such a controller in a more complete way, three more tests with a 1 kg desired mass were performed. In the first one the path that the user tries to execute is a triangle on the x-y plane, in the second one it is a circle on the x-y plane and in the third test it is a circle on the y-z plane. The data relative to the triangular test are reported in Figures 2.32–2.35, while those relative to the circular tests are reported in Appendix A, because their results are conceptually analogous to those of the triangular path (Figures A.17–A.24).

In these Figures, unlike the previous tests, the data showing the discrepancies in the dynamic model and in the impedance model are omitted for brevity, but obviously even in this case the difference between the expected ideal behavior and the real one was very significant, due to the inherent problems of the impedance control scheme.

What emerges from these tests is that even with the translational impedance control scheme it is quite difficult to follow a desired trajectory. In fact, all the trajectories are executed three times and each time the path is rather different, meaning that the motion imposed by the operator was "deviated" by internal unmodelled opposition forces. In the case of the circular motion in the y-z plane, however, the precision is greater, due to the fact, for that particular joint configuration and that particular trajectory, the unmodelled opposition forces were less significant. This is a further proof of the anisotropy and untrustworthiness of such a control scheme.

The forces needed to move the end effector have peaks around 20-30 N, and are much smaller for the circular motion in the y-z plane. Once again, an anisotropic behavior can be noticed, with the forces being higher in some parts of the trajectories and lower in other parts. Conversely, the velocities are generally higher in the parts of the trajectories where the forces are lower, because there is less "opposition". This is especially evident in the circular motion in the x-y plane (Figures A.17 and A.19).

The orientation of the end effector is not kept constant along the motion (see Figures 2.33, A.18 and A.22). In particular, the roll angle with respect to the base reference frame changes significantly (20-30°) for all three trajectories. However, the situation here is different with respect to the translational and rotational impedance control. In that case, in fact, it was not possible for the user to reorient properly the end effector; in this case, instead, the user could apply forces and torques on the end effector to keep it oriented as desired, but does not do so because this correction would require a significant effort. Therefore, an orientation error is present because the control does not provide any help in this reorientation, as the control is only translational: in particular, the roll angle is the one having the largest error, and in fact it is associated to the rotation around the z axis of the base reference frame. So to keep it at 0° the user should apply a torque on the seventh joint, similarly to the situation described in the preliminary test in Figure 2.11.

Finally, for all three trajectories the height on the x-y plane (y-z plane for the second circular motion) is not kept constant, as would be desired (see Figures 2.35, A.20 and A.24), once again because of the disturbance actions of the unmodelled terms.

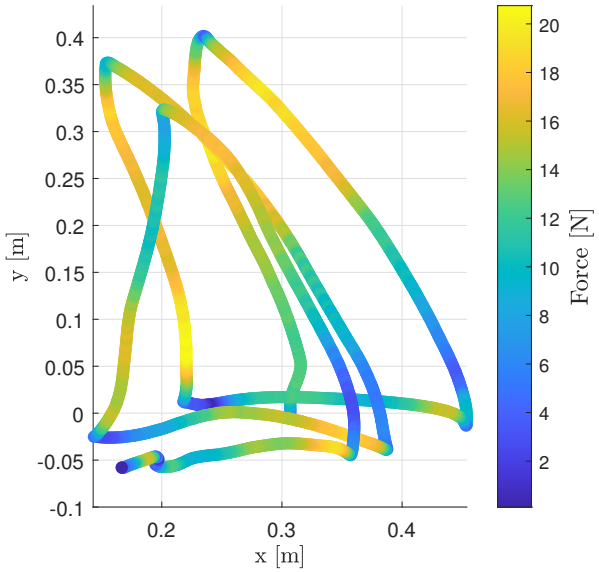


FIGURE 2.32: Triangular motion test with translational impedance control, with low inertia and damping parameters: norm of the end effector force vector along the trajectory in the *x*-*y* plane.

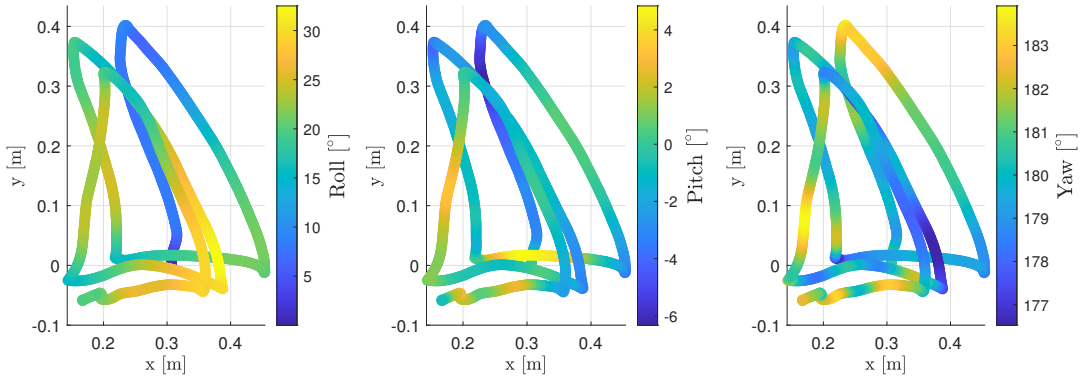


FIGURE 2.33: Triangular motion test with translational impedance control, with low inertia and damping parameters: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

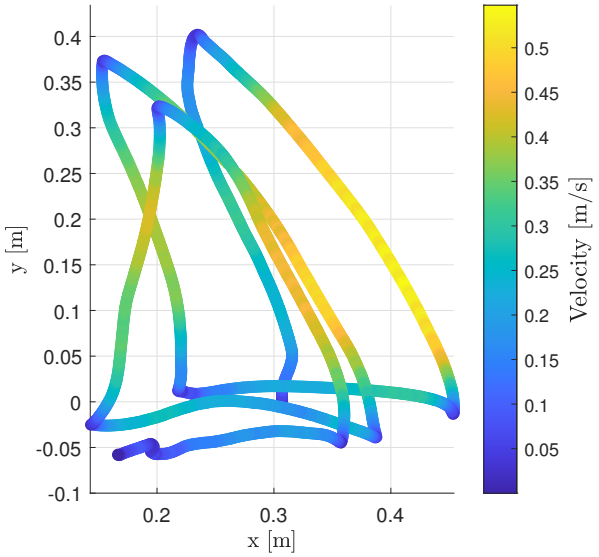


FIGURE 2.34: Triangular motion test with translational impedance control, with low inertia and damping parameters: norm of the end effector velocity vector along the trajectory in the x-y plane.

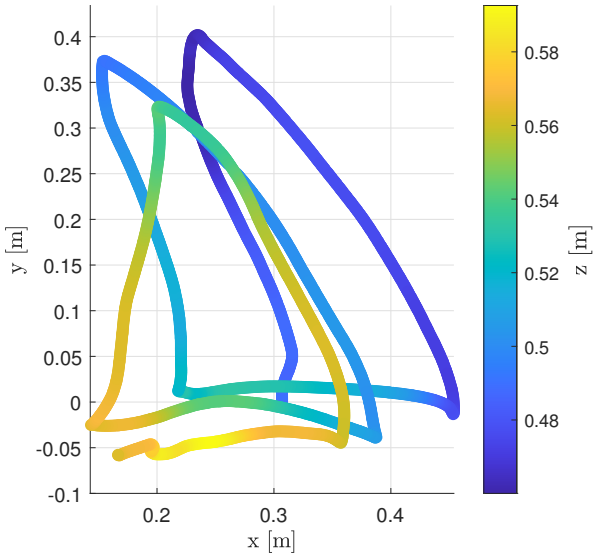


FIGURE 2.35: Triangular motion test with translational impedance control, with low inertia and damping parameters: height on the z axis along the trajectory in the x-y plane.

2.5 Helping control

Due to the intrinsic difficulties that are encountered applying impedance control, another technique using the torque interface was analyzed. Let us consider again the initial case study of the user applying a torque about the z axis, which in the starting configuration coincides with the axis of the seventh joint. It can be argued that, if the user needs to apply a positive/negative torque to move the seventh joint, his effort would be decreased if some more positive/negative torque were provided through the joint torque interface ("helping" control action).

Figure 2.36 shows the velocity of the seventh joint and the external torque (blue) and command torque (red) on the seventh joint. Like in the previous cases, by external torque on the seventh joint it is meant the seventh element of the $\mathbf{J}^T\mathbf{F}_{ext}$ vector, and the command is equal to the external torque multiplied by a gain of 3. Comparing this Figure to Figure 2.11, it can be observed that the behavior of the velocity is similar, and the value of the external torque is significantly smaller, demonstrating that the "helping" action was successful in decreasing the effort of the operator.

Based on the theory presented in Subsection 2.1.2, the idea behind the "helping" controller that is presented in this Section is that, if some torques proportional to the elements of the vector $\mathbf{J}^T\mathbf{F}_{ext}$ are commanded, the the end effector will be driven as desired by the user, reducing his effort. It has to be underlined that this control modality was introduced mainly as a substitute for the impedance control, and is not rigorous from a theoretical point of view. If it had been possible to directly access the "real" torque command, it would also have been possible to compensate the dissipative effects of the harmonic drive, but this strategy could not be implemented because of the "black box" nature of the torque command interface, which, in presence of external forces acting on the system, conditions the control signal sent in a way that is not known by the user.

In fact, if at first it is imagined to have only the external forces and torques acting on the end effector (without joint control torques), and then to add joint control actions that are equal to $\mathbf{J}^T\mathbf{F}_{ext}$, the overall control action will be double with respect to the first case, but it cannot be inferred from the model that the end effector will be guided in the same direction that would have been obtained without the control action. Moreover, the considered formulation once again does not consider the internal dissipations and opposing actions due to the harmonic drives, which affects the very exactness of the equation $\boldsymbol{\tau} = \mathbf{J}^T\mathbf{F}_{ext}$, as already discussed in the previous Section.

Nevertheless, controlling the robot with this very simple law proved to be rather effective, as is shown in the following tests. Like for the impedance control, three trajectories were tested, triangular on the x-y plane (Figures 2.37–2.41), circular on the x-y plane (Figures B.1–B.5) and circular on the y-z plane (Figures B.6–B.10). All the tests are performed with a gain of 3 for all the joints, chosen as the best trade-off after proper trial-and-error tuning as the highest value, decreasing as much as possible the effort of the operator without making the controller unstable. Comparing with the results of the same tests performed with the translational impedance control scheme, it can be observed that the effort needed to move the end effector has decreased drastically, having peaks of around 5 N in some points of the trajectories but staying around 2-3 N for most of the path, meaning that, as desired, the HRI is almost seamless and very smooth. In addition, the precision of the motions has improved: in fact, for all three trajectories, the lines of the successive iterations of the path are almost superimposed and the deviation of roll, pitch and yaw from their initial values is lower, even though the lines in the triangular motion are not straight

as would be desirable, once again due to the deviation effects of the unmodelled opposing torques, and for the same reason the circular trajectories are noticeably deformed in some parts.

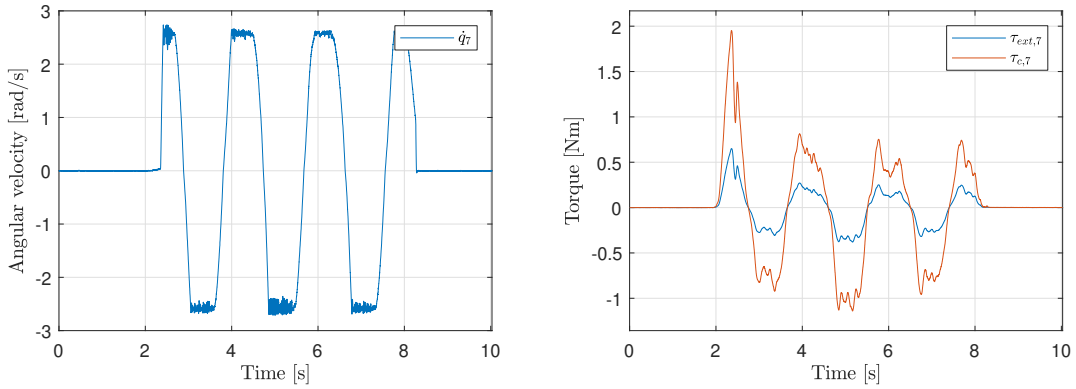


FIGURE 2.36: Test of the "helping" action on the seventh joint.

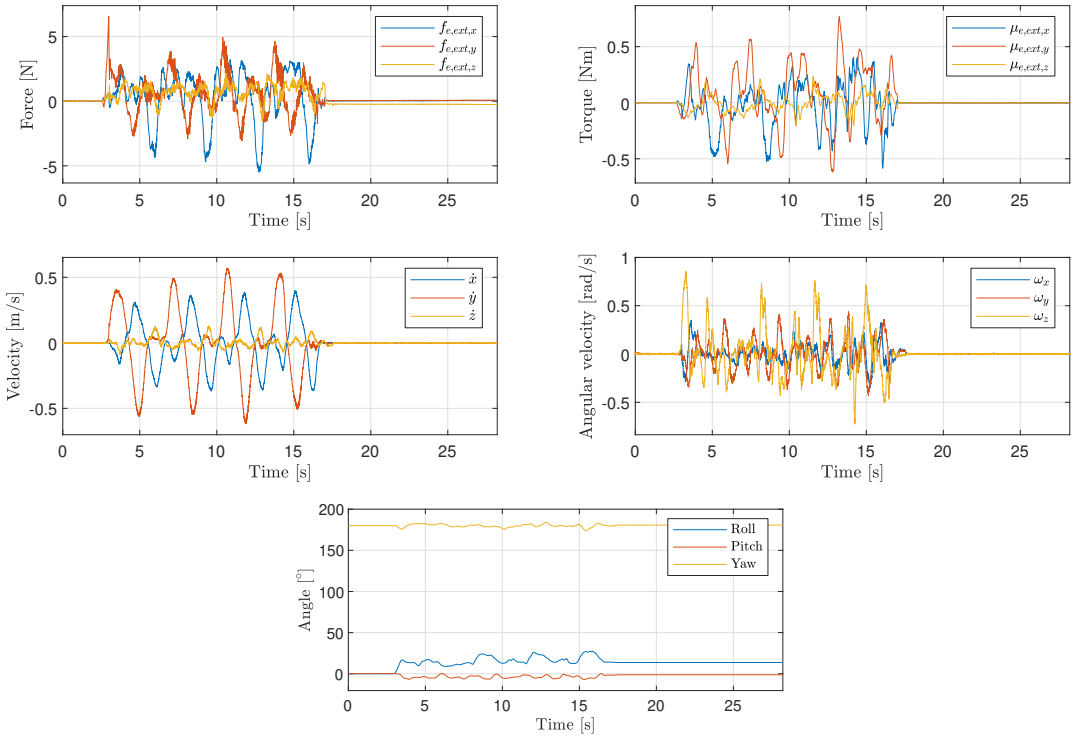


FIGURE 2.37: Triangular motion test with helping control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

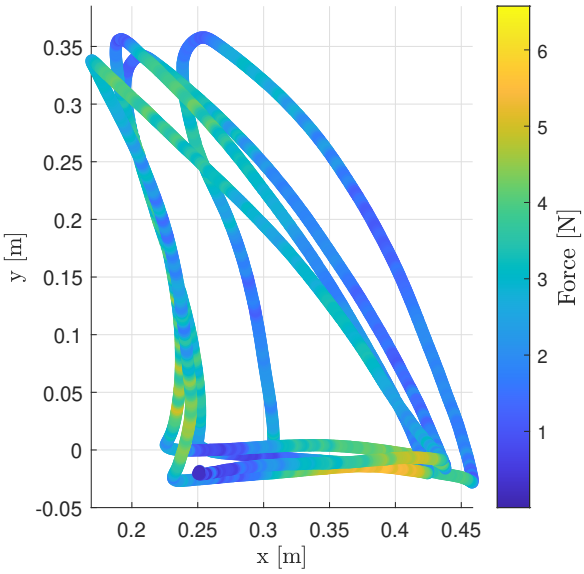


FIGURE 2.38: Triangular motion test with helping control: norm of the end effector force vector along the trajectory in the *x*-*y* plane.

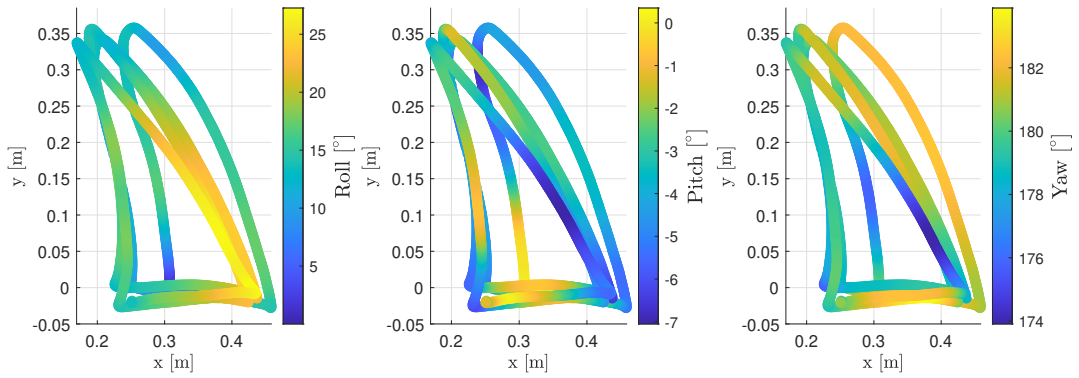


FIGURE 2.39: Triangular motion test with helping control: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

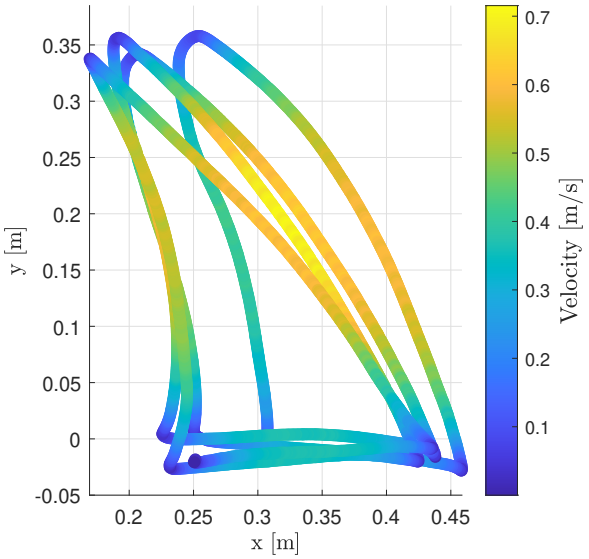


FIGURE 2.40: Triangular motion test with helping control: norm of the end effector velocity vector along the trajectory in the x-y plane.

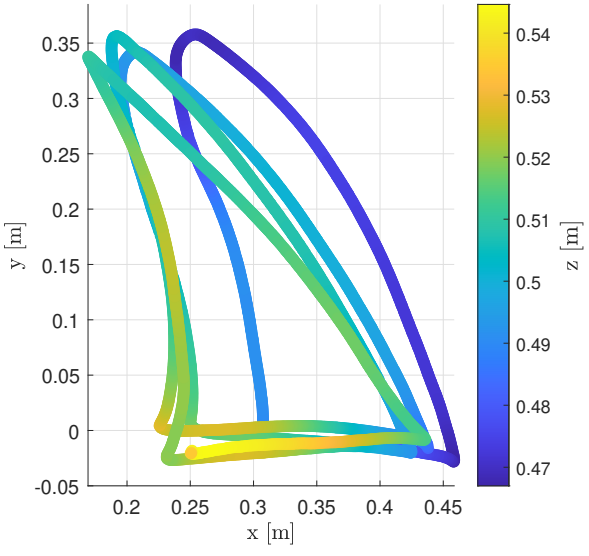


FIGURE 2.41: Triangular motion test with helping control: height on the z axis along the trajectory in the x-y plane.

2.6 Admittance control

As introduced in Section 2.2, a control strategy that is complementary to impedance control is admittance control: in both cases a desired end effector dynamics is imposed, namely a desired inertia matrix and a desired damping matrix are chosen, usually diagonal to have decoupled dynamics. In the case of admittance control, though, the vector of forces and torques is treated as the input, based on which the desired end effector linear and angular velocities are computed to fit the model. The end effector velocity vector is mapped to joint velocities through the pseudo-inverse of the Jacobian, and these velocities are commanded to the robot through the corresponding interface.

Admittance control allows therefore to use the joint velocity interface. This is particularly beneficial because it avoids all the problems related to the joint torque interface. In fact, when a velocity command is sent to the joints an internal controller of Franka (even in this case, in a "black box" way) takes care of tracking the desired signal. In this way, the presence of unmodelled opposition torques inside the transmissions is not of concern. In other words, differently from what happens for the impedance controller scheme, and partially even with the helping controller scheme, the inaccuracies of the dynamic model do not have any negative impact, because the internal controller of Franka is accurate enough to follow successfully the commanded velocity trajectory.

Also for the admittance control case, the usual tests were performed. Figures 2.42–2.48 show the results of the triangular trajectory on the x-y plane, Figures C.1–C.7 show the results of the circular motion on the x-y plane and Figures C.8–C.14 show the results of the circular motion on the y-z plane. All tests were performed with a translational inertia of 2 kg, a rotational inertia of $0.1\,\mathrm{kg\,m^2}$, a translational damping of $0.5\,\mathrm{N\,s/s}$ and a rotational damping of $0.5\,\mathrm{N\,s}$. The most notable results are that the forces and torques required to move the end effector are very low and that the admittance model is matched by measured data (Figures 2.43, 2.44, C.2, C.3, C.9 and C.10). The precision of the trajectory is good: especially in the triangular trajectories the lines are much straighter than in all the other cases analysed up to now. However, there is still some imprecision left both from the point of view of the repeatability of the trajectory and of the orientation angles. This is mainly due to the rotational inertia contribution: in fact, even though an inertia of $0.1\,\mathrm{kg\,m^2}$ is rather small, a noticeable torque is still required to stop a rotation of the end effector, and this tends to generate an oscillatory behavior that is quite disturbing when driving the robot.

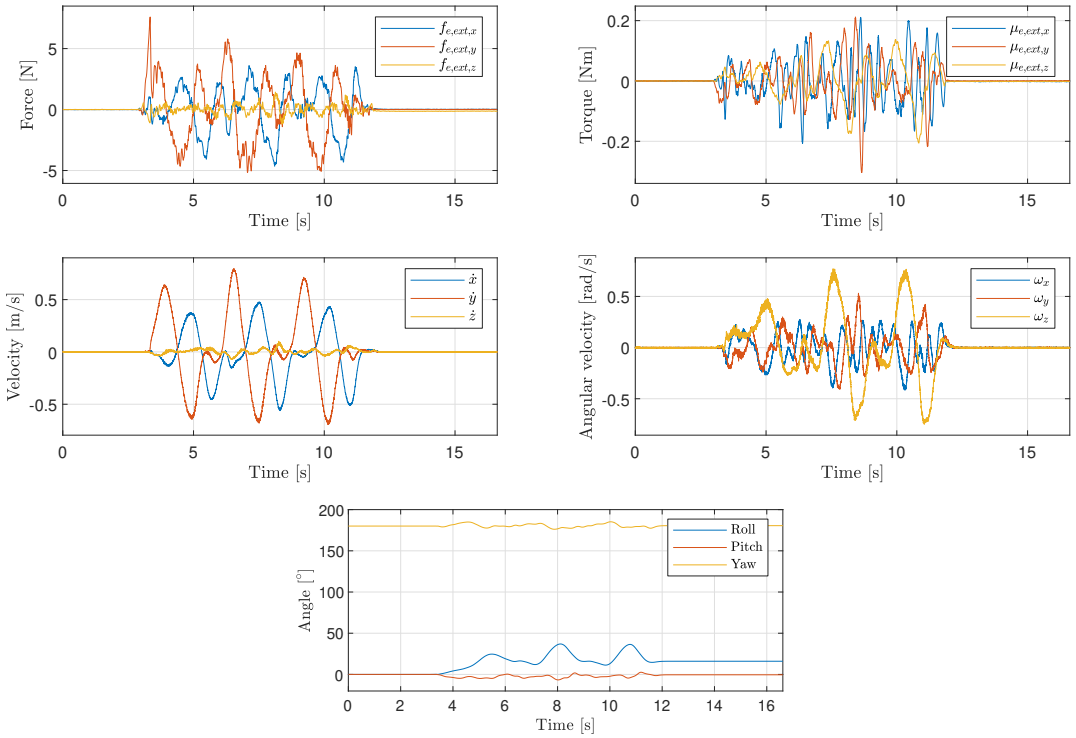


FIGURE 2.42: Triangular motion test with admittance control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

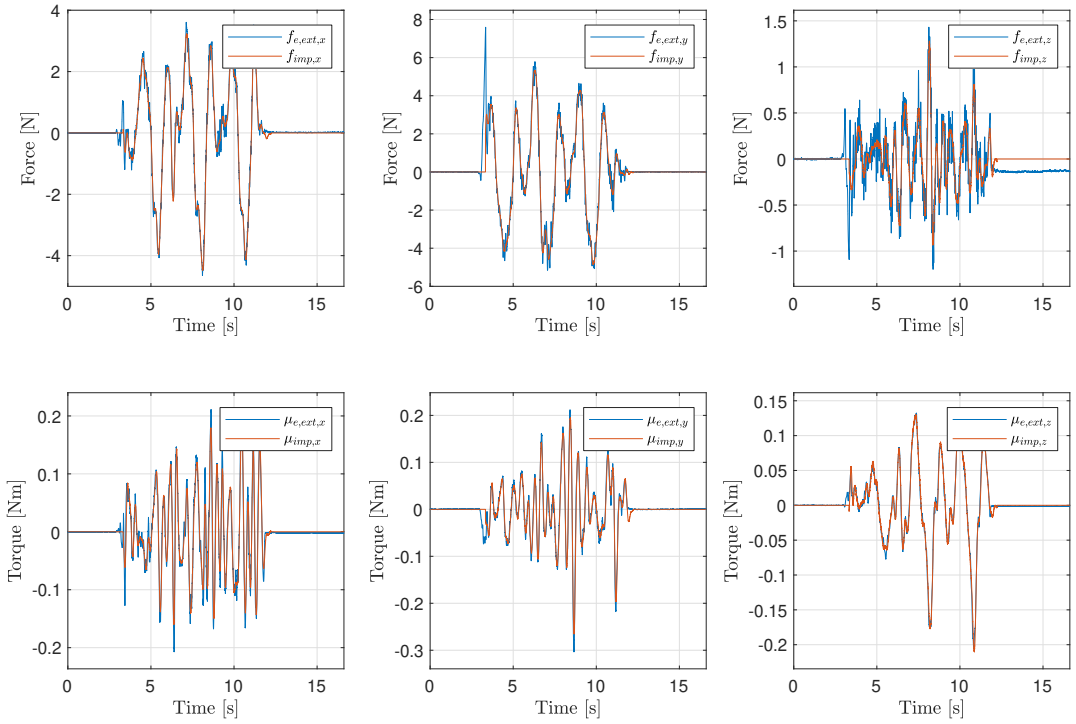


FIGURE 2.43: Triangular motion test with admittance control: comparison of expected and measured forces and torques.

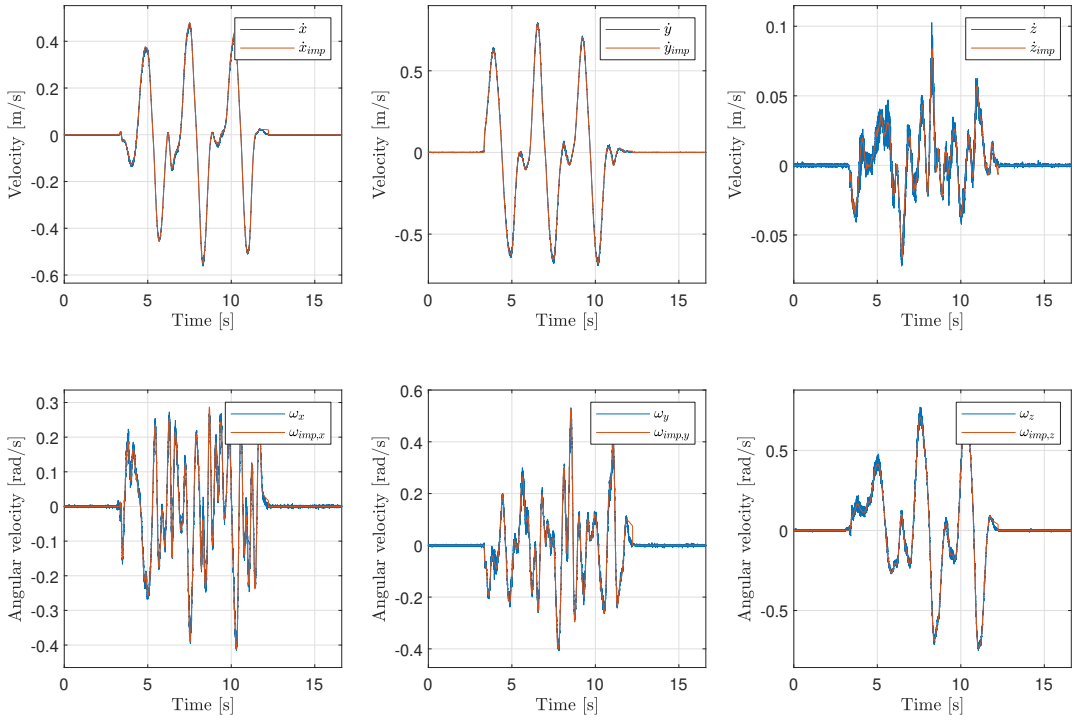


FIGURE 2.44: Triangular motion test with admittance control: comparison of expected and measured linear and angular velocities.

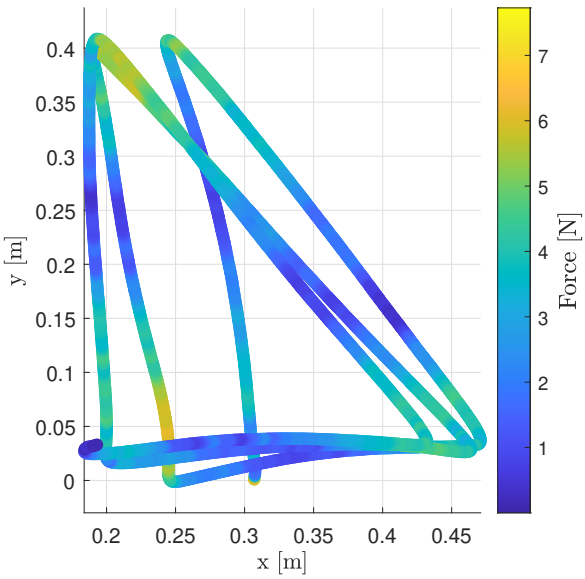


FIGURE 2.45: Triangular motion test with admittance control: norm of the end effector force vector along the trajectory in the *x*-*y* plane.

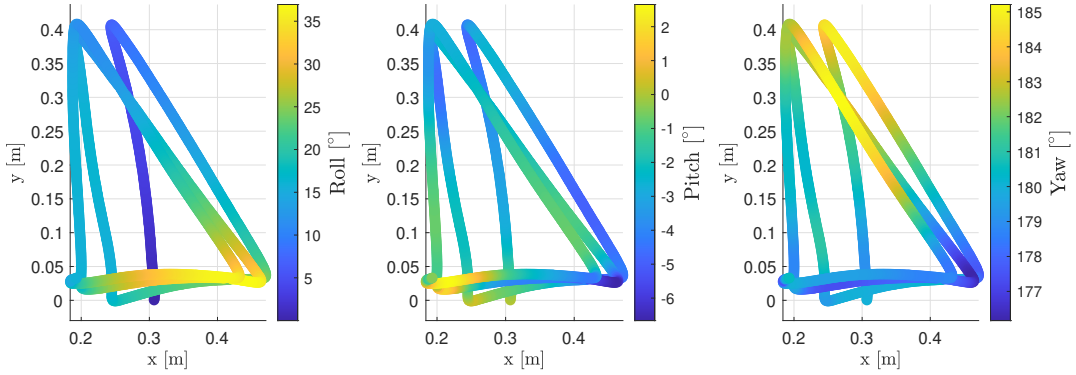


FIGURE 2.46: Triangular motion test with admittance control: roll, pitch and yaw angles of the end effector along the trajectory in the x-y plane.

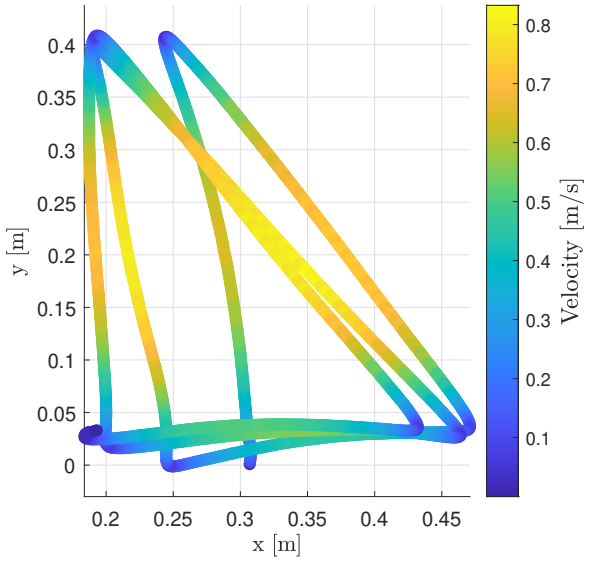


FIGURE 2.47: Triangular motion test with admittance control: norm of the end effector velocity vector along the trajectory in the x-y plane.

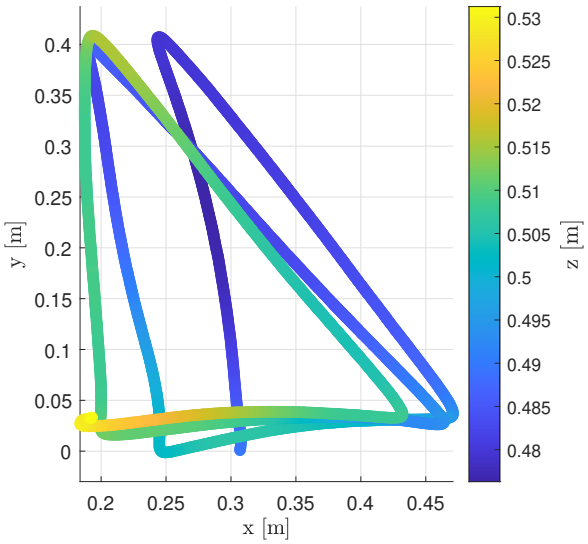


FIGURE 2.48: Triangular motion test with admittance control: height on the z axis along the trajectory in the x-y plane.

To eliminate the imprecisions due to the rotational inertia, and to verify the behavior of the control scheme under a different operative condition, a series of tests implementing an only translational admittance control scheme was performed. Testing once again the triangular motion on the x-y plane (Figures 2.49–2.55), the circular motion on the x-y plane (Figures C.15–C.21) and the circular motion on the y-z plane (Figures C.22–C.28), it can be observed that, differently from the case of translational-only control with the impedance scheme, this time there is actually only translational motion. The effort needed for such a motion is again very low, and the path is followed with great accuracy.

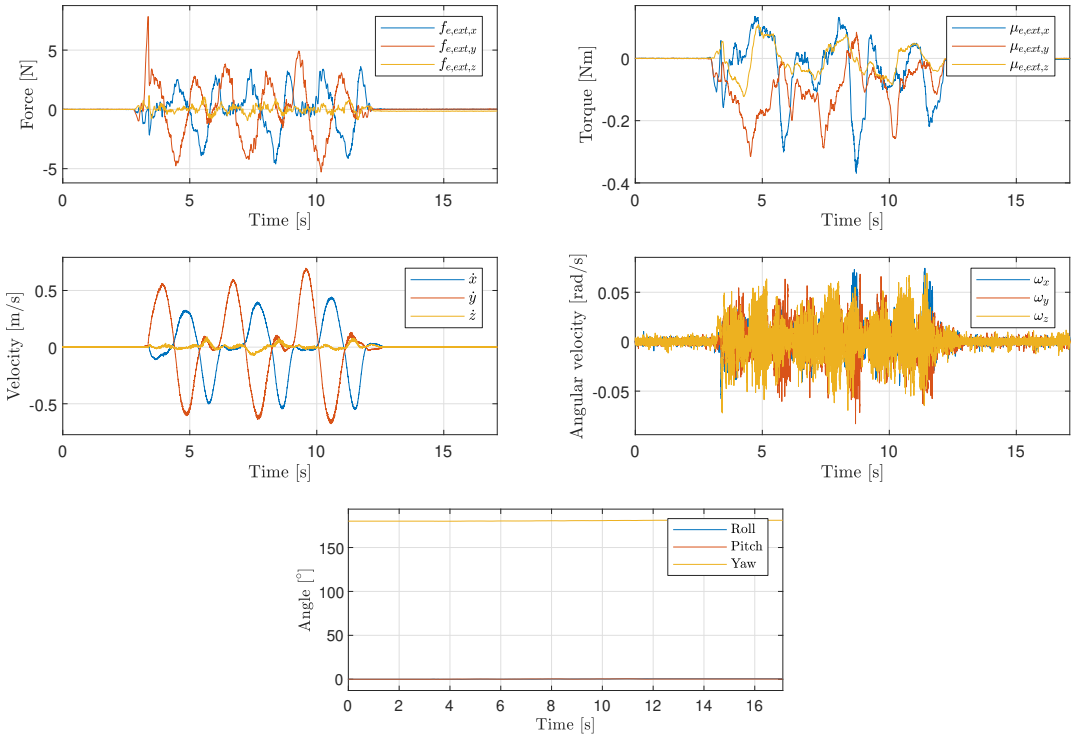


FIGURE 2.49: Triangular motion test with translational admittance control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

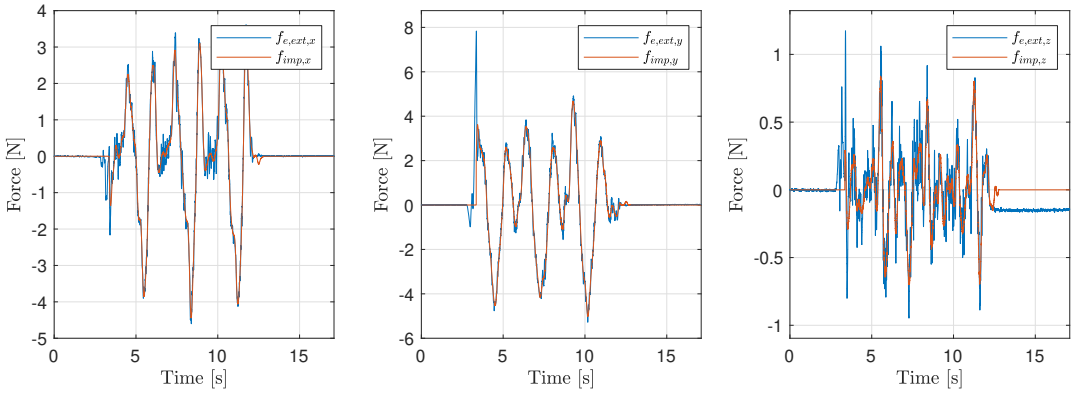


FIGURE 2.50: Triangular motion test with translational admittance control: comparison of expected and measured forces.

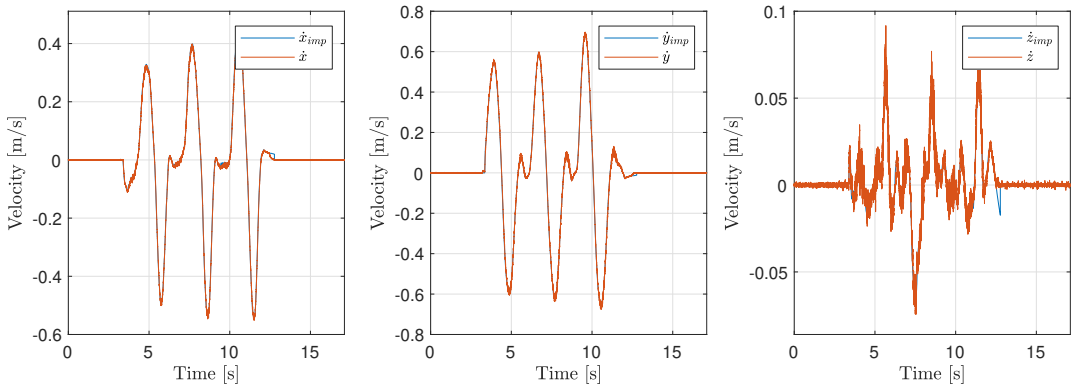


FIGURE 2.51: Triangular motion test with translational admittance control: comparison of expected and measured linear velocities.

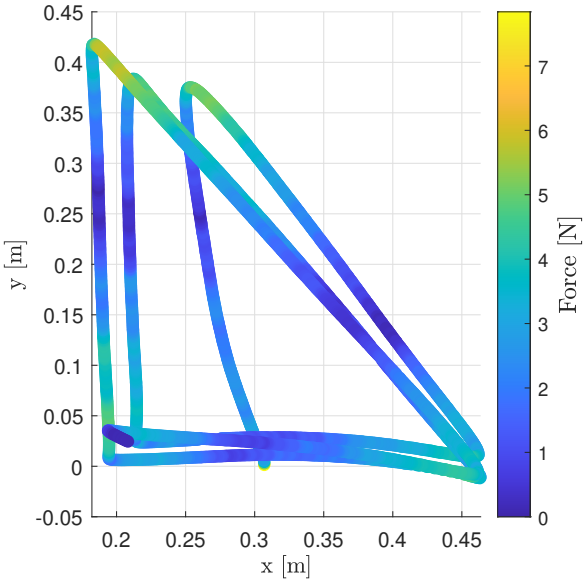


FIGURE 2.52: Triangular motion test with translational admittance control: norm of the end effector force vector along the trajectory in the x-y plane.

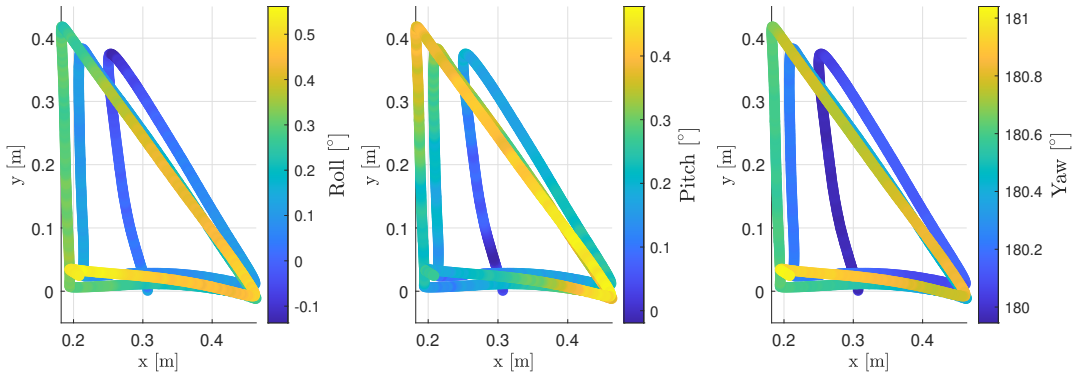


FIGURE 2.53: Triangular motion test with translational admittance control: roll, pitch and yaw angles of the end effector along the trajectory in the x-y plane.

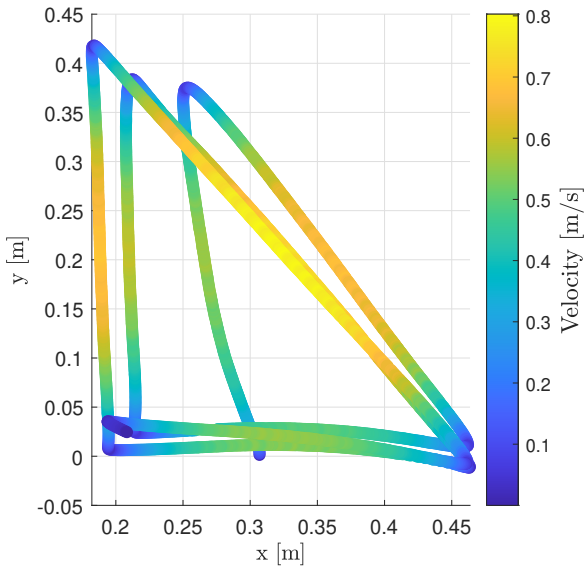


FIGURE 2.54: Triangular motion test with translational admittance control: norm of the end effector velocity vector along the trajectory in the x-y plane.

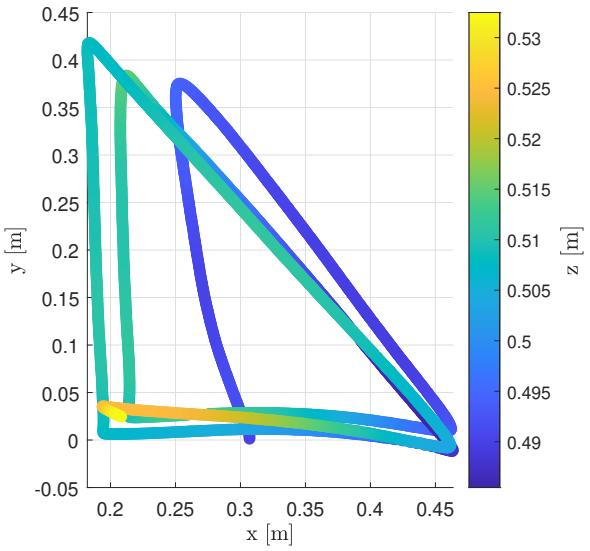


FIGURE 2.55: Triangular motion test with translational admittance control: height on the z axis along the trajectory in the x-y plane.

2.7 Preliminary considerations on null space control

One last consideration that has to be made regarding end effector control is about what happens when also null space commands are sent. In fact, the structure that is being considered is redundant and, in the context of manual guidance, null space motions are necessary for several reasons, such as avoiding obstacles (the human operator himself and/or other obstacles in the workspace), and moving away from joint limits and singularities.

In general, the null space command, when the torque interface is used, takes the form

$$\boldsymbol{\tau}_{null} = k \left(\mathbf{I} - \mathbf{J}^T \mathbf{J}^{\dagger T} \right) \boldsymbol{\tau}_0 \,, \tag{2.55}$$

where k is a free gain and $\boldsymbol{\tau}_0$ is a vector indicating a desired torque direction, that will then be projected onto the null space. The resulting null space torque is such that it will not apply any forces and torques on the end effector.

In the following example the gain was chosen as 5 and the vector $\boldsymbol{\tau}_0$ as $[1, 0, 0, 0, 0, 0, 0]^T$ Nm. The motion resulting from the null space command was recorded. The end effector was left free (the hands of the operator were not grabbing the handle), so what should have happened was a motion of the kinematic structure (with the first joint rotating in the positive direction, because of the choice of $\boldsymbol{\tau}_0$), with the end effector keeping its position and orientation. Instead, as can be observed in Figures 2.56–2.58, there are a very significant displacement and rotation of the end effector. This is once again due to the imprecision of the dynamic model. In fact, as already discussed extensively, the torques sent through the joint torques interface do not correspond exactly to the torque that is expressed at the link level (i.e., after the harmonic drive), and thus there are forces/torques acting on the end effector even when a null space command is intended.

Another interesting way to see the same problem is to perform the same test with the operator grabbing the end effector and trying to keep it in its initial position and orientation. This should be possible without applying any forces/torques, but, in reality, the user struggles to maintain the initial configuration and has to apply significant forces with peaks of around 8 N and torques of up to 2.5 Nm (see Figures 2.59–2.62).

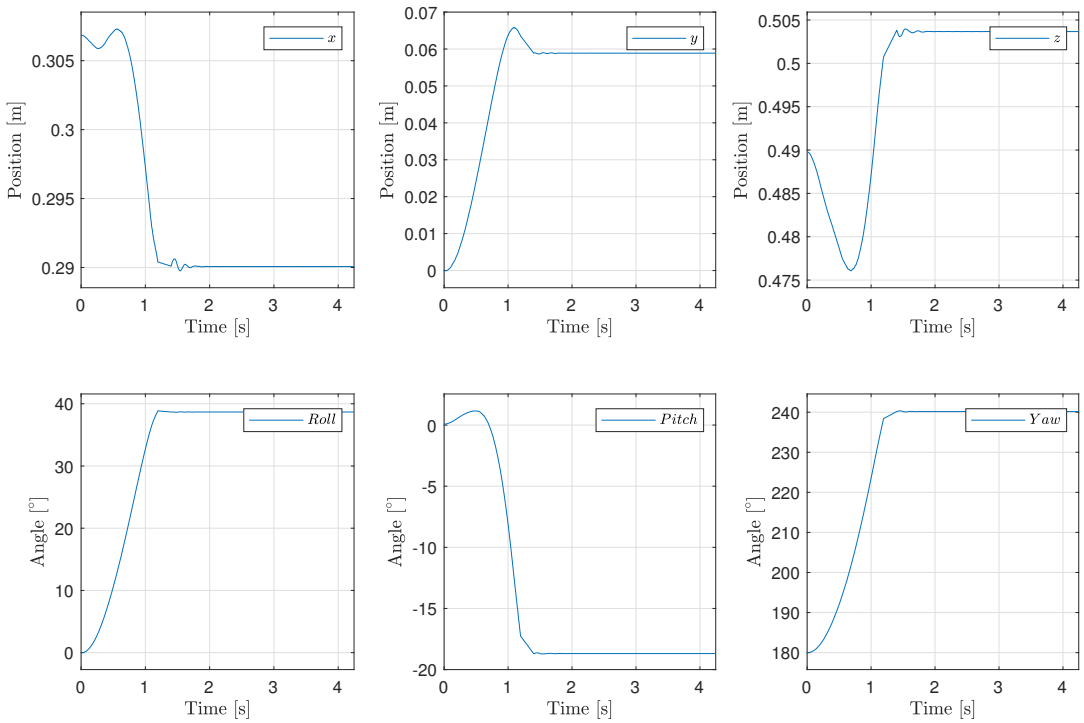


FIGURE 2.56: Null space test with torque control and hands detached from end effector: position in x, y and z of the end effector and roll, pitch and yaw angles of the end effector.

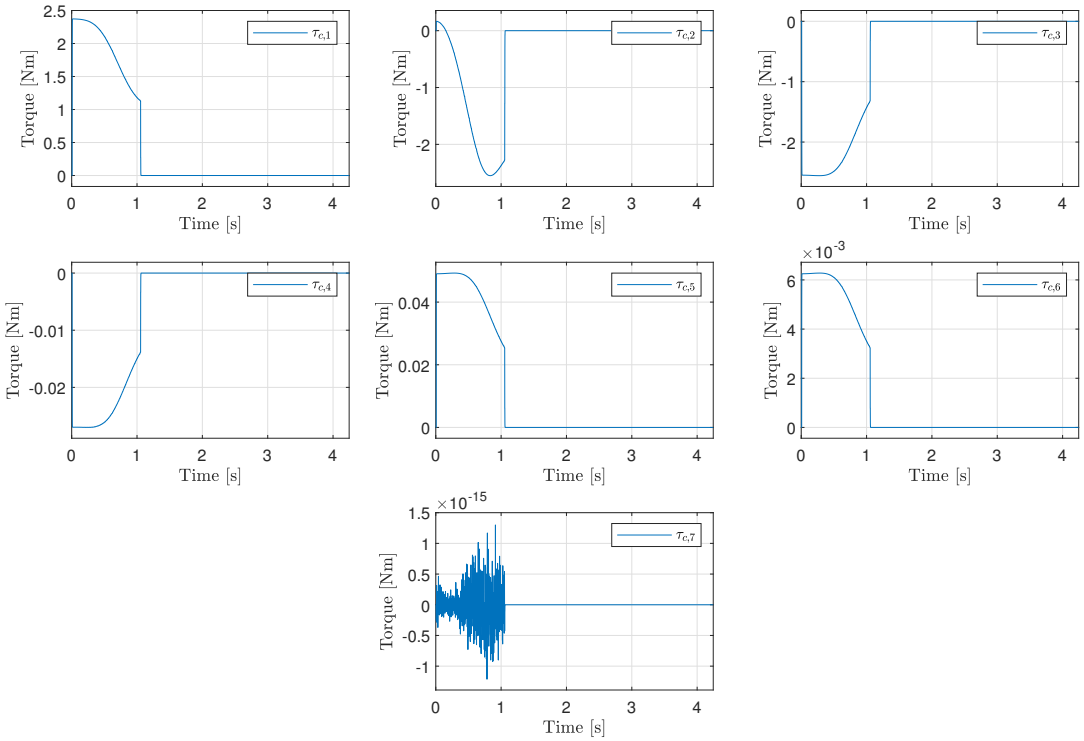


FIGURE 2.57: Null space test with torque control and hands detached from end effector: command torques.

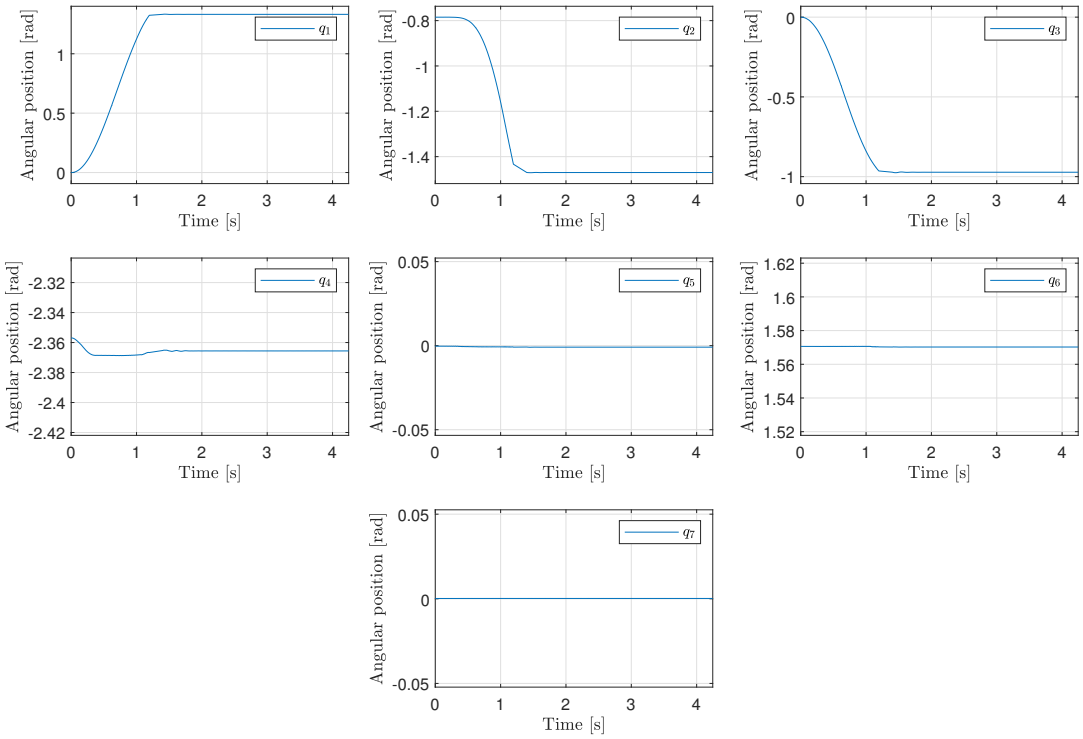


FIGURE 2.58: Null space test with torque control and hands detached from end effector: joint positions.

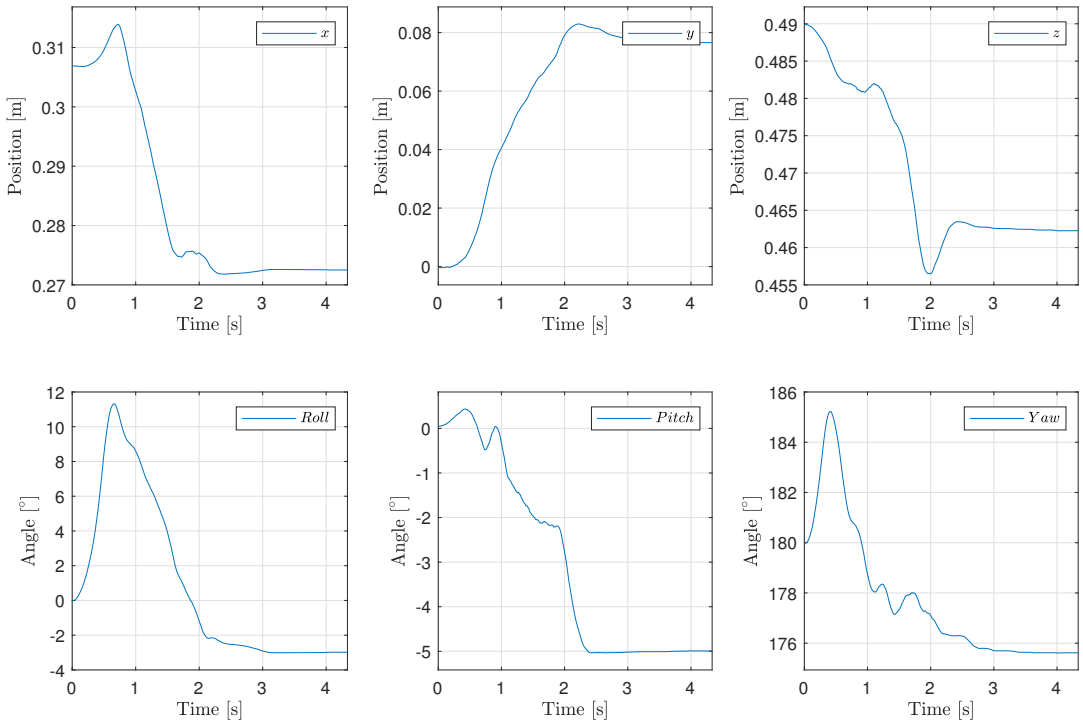


FIGURE 2.59: Null space test with torque control and hands attached to the end effector: position in x, y and z of the end effector and roll, pitch and yaw angles of the end effector.

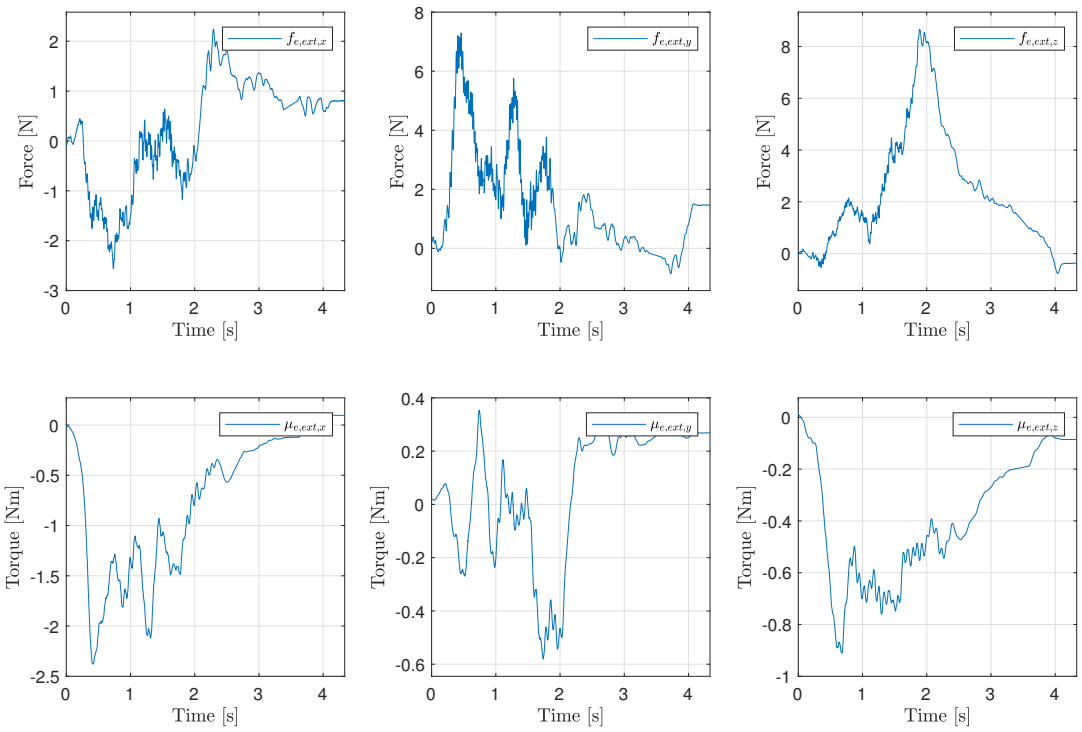


FIGURE 2.60: Null space test with torque control and hands attached to the end effector: forces and torques acting on the end effector.

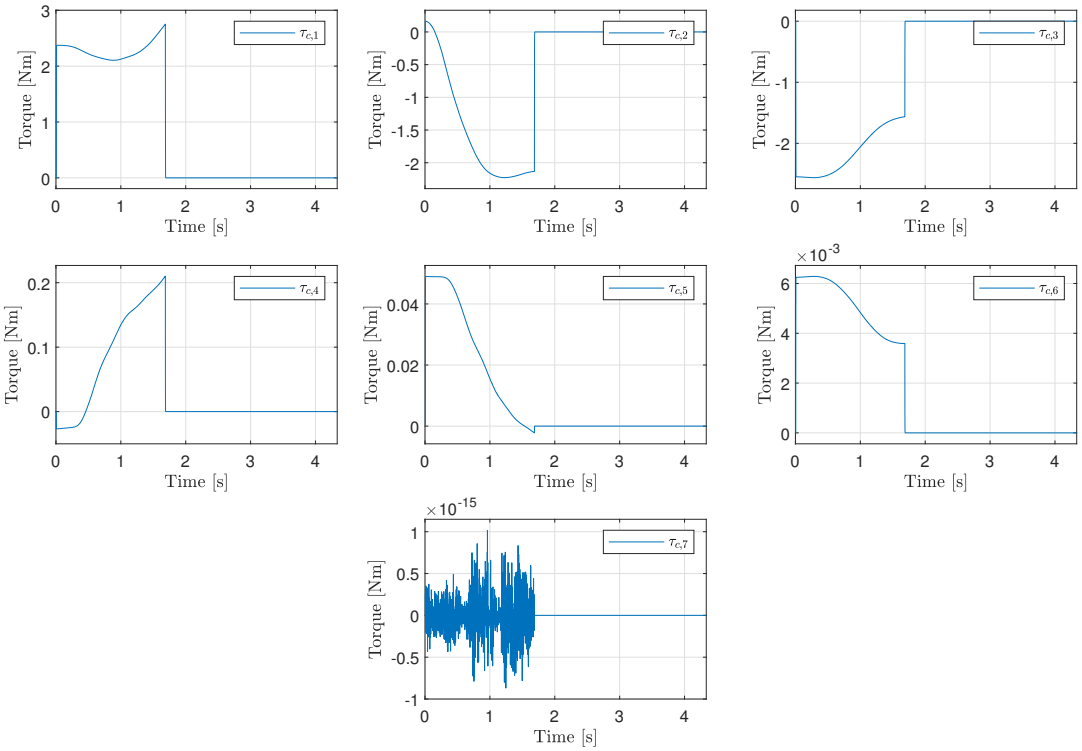


FIGURE 2.61: Null space test with torque control and hands attached to the end effector: command torques.

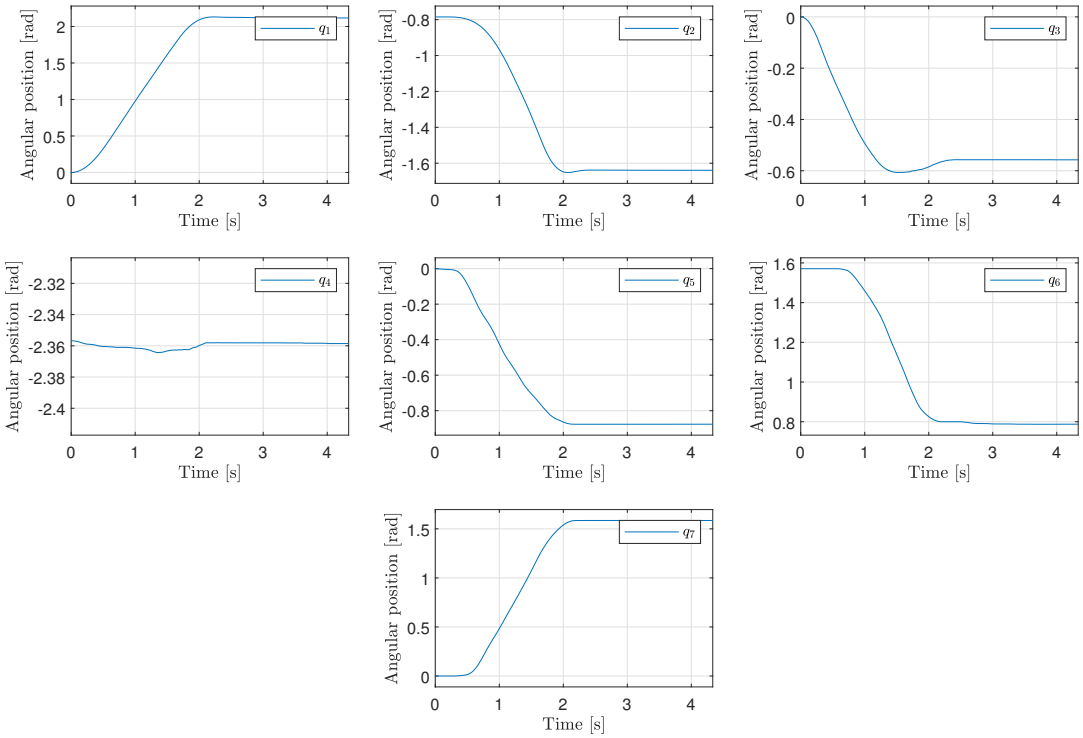


FIGURE 2.62: Null space test with torque control and hands attached to the end effector: joint positions.

Finally, the null space movement was tested with the velocity interface. The principle is the same of the previous cases, but this time the velocity command is expressed as

$$\dot{\mathbf{q}}_{null} = k \left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J} \right) \dot{\mathbf{q}}_0 \,, \tag{2.56}$$

so that the resulting end effector velocity vector is null.

In this case the robot is successful in avoiding the motion of the end effector, due to the effective action of the internal controller, which guarantees that the velocity reference commands provided to the joints are followed with precision. In Figures 2.63 and 2.64, in fact, it can be observed that the position and orientation of the end effector undergo only some minor modifications that can be considered as acceptable residual imprecisions of the control law, but will not impact negatively on the manual guidance application.

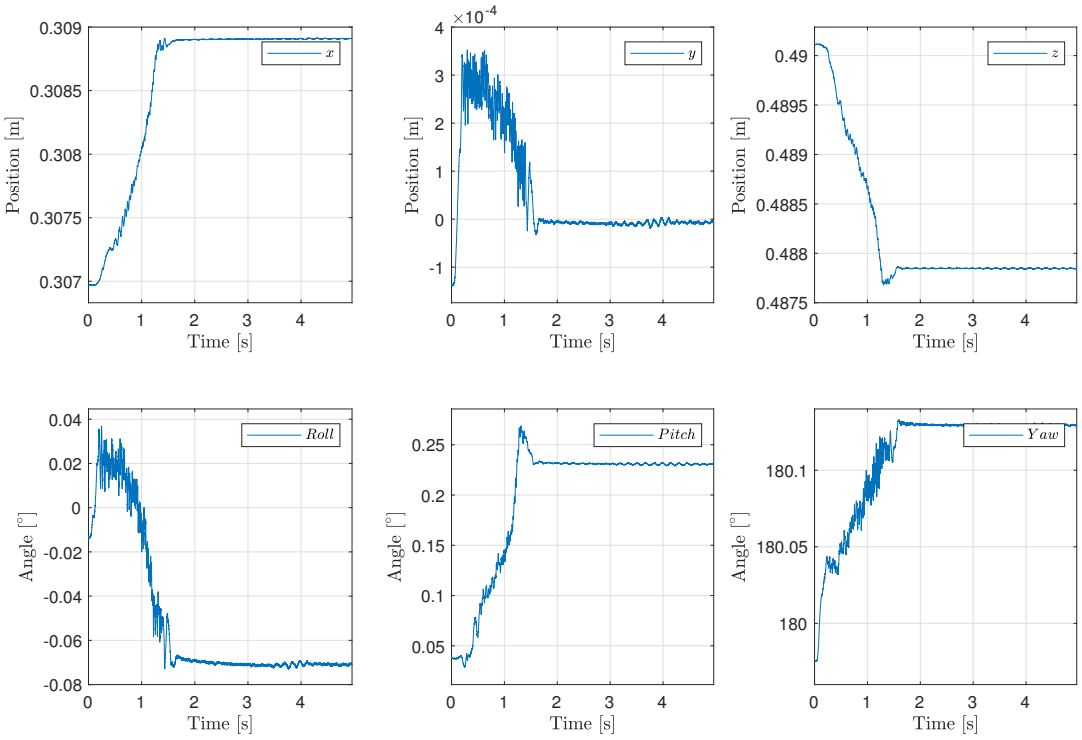


FIGURE 2.63: Null space test with velocity control: position in x, y and z of the end effector and roll, pitch and yaw angles of the end effector.

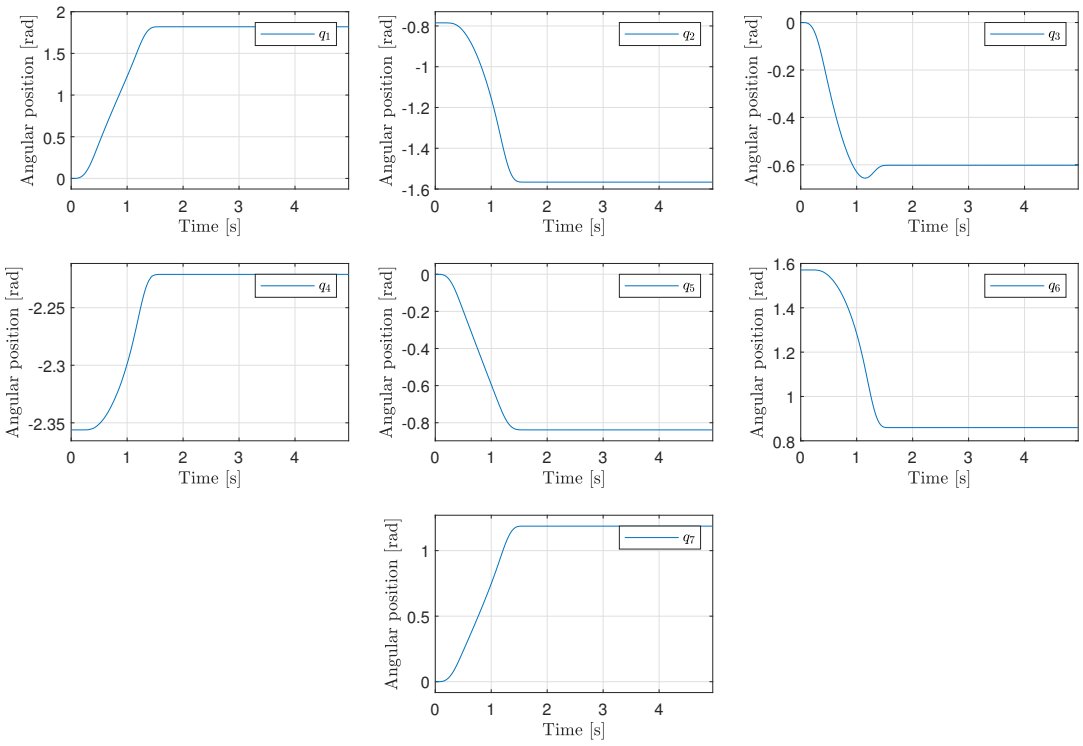


FIGURE 2.64: Null space test with velocity control: joint positions.

2.8 Discussion

The results of the previous tests can be summarized in the following points:

- Impedance control and admittance control should theoretically be equivalent, both imposing a desired end effector decoupled motion law.
- In practice, however, due to the modelling errors and to the limitations of the Franka torque interface, impedance control is not effective in achieving the desired behavior.
- If the impedance control scheme imposes both a translational and a rotational behavior the rotations of the end effector turn out to be "locked", making this control technique completely unviable.
- If the impedance control scheme imposes only a translational behavior, the resulting end effector motion requires a reasonable level of force from the user but is still completely different from the theoretical desired dynamics, resulting in a non-decoupled and anisotropic behavior.
- As an alternative way to use the joint torque interface, the helping controller was introduced. This controller provides good results in terms of forces needed to move the end effector, but it does not result in very accurate trajectories, has a sort of "heuristic" nature and again does not behave isotropically in all directions.
- The admittance control scheme also provides very good results in terms of forces applied by the user, and is also fully adherent to the theoretical desired behavior imposed by model motion law, obtaining a decoupled and isotropic behavior.
- If the admittance scheme is both translational and rotational some involuntary oscillatory motion is sometimes experienced due to the rotational inertia, making the control less precise, while if only the translational part is imposed in the motion law, while the orientation is kept fixed, the movement is cleaner and smoother. Adopting this strategy would not be enough for all operative conditions, because it is needed to change the orientation in the proximity of the area where the end effector needs to operate (e.g., the wall where the drill must perform the hole, or the object that the cooperative robot must lift guided by the human). Some additional reorientation control laws should be developed in those cases, but they would be specific for the needs of the actual application. As a general alternative, it would also be possible to study if it would be possible to just set a translational and a rotational damping (without inertia). This was tried but the results were not acceptable because the end effector exhibited a very vibratory and unstable behavior, due to the oscillations in the force/torque sensor data. As a future development of this thesis, some more sophisticated filtering technique will be applied to assess if a damping-only controller is feasible.
- The null space motions are problematic when using the torque interface, because they generate additional forces on the end effector, i.e., the force null space is not really a null space, and, for the intrinsic limitations of the torque interface, there is nothing that can be done to avoid or mitigate this problem; instead, the velocity null space works in a perfectly satisfactory way.

A synthetic comparison of the results is presented in Figures 2.65–2.67, where the considered controllers are the translational impedance controller with 5 kg translational inertia and 0.5 N s/m translational damping, the helping controller, the full (translational and rotational) admittance controller with 1 kg translational inertia, 0.1 kg m^2 rotational inertia, 0.5 N s/m translational damping and 0.5 N s rotational damping and the translational admittance controller with 1 kg translational inertia and 0.5 N s/m translational damping (the full impedance controller was not considered because of its rotational "locking" problems). The Figures show the mean values of some quantities of interest to evaluate the performance of the controllers, namely the mean value of the force, velocity, distance with respect to the desired plane of motion, and difference with respect to the desired values of roll, pitch and yaw (namely, 0°, 0° and 180°, because it is desired to keep fixed the initial straight orientation of the end effector). The controller is best if the mean velocity is maximized (because the operator is trying to guide the end effector as quickly as possible, so the velocity will be higher if little resistance is felt) and all the other parameters are minimized.

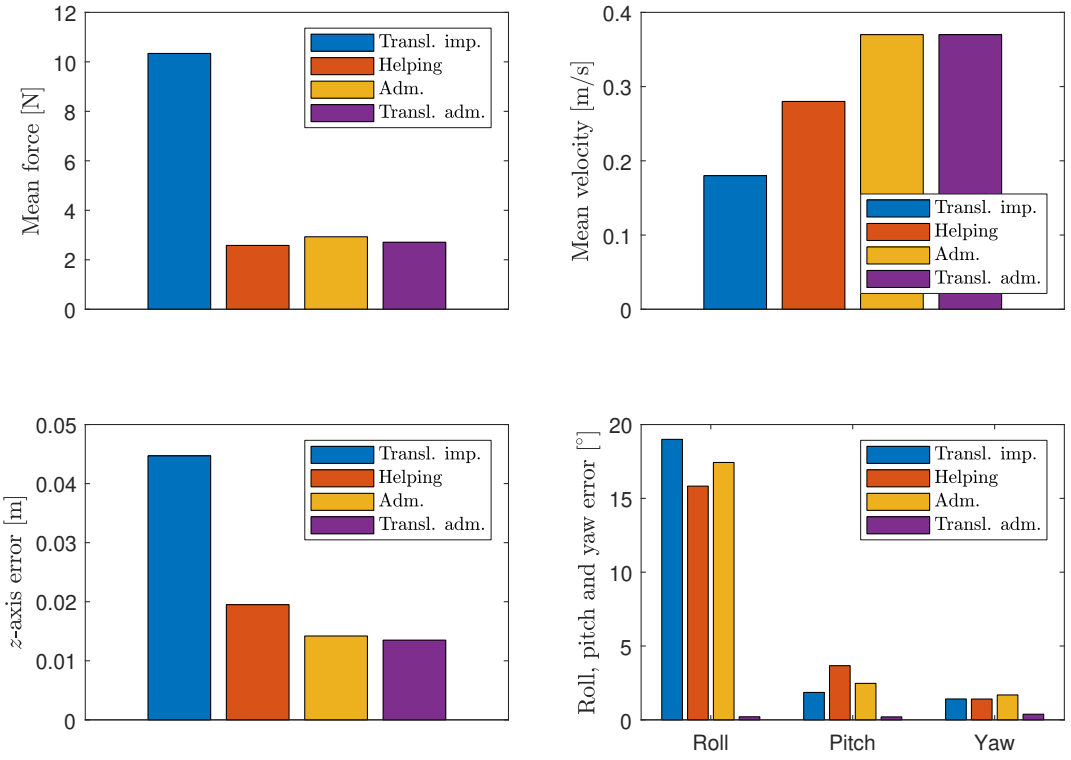


FIGURE 2.65: Evaluation parameters for the triangular motion test.

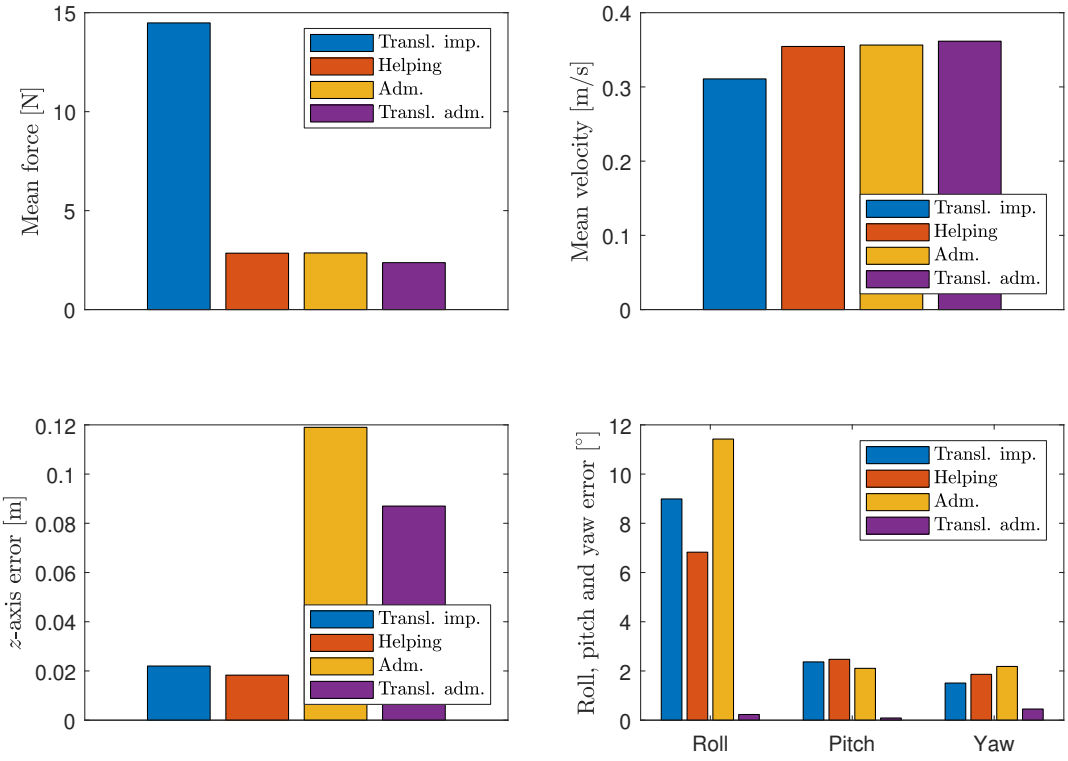


FIGURE 2.66: Evaluation parameters for the circular motion test on the x-y plane.

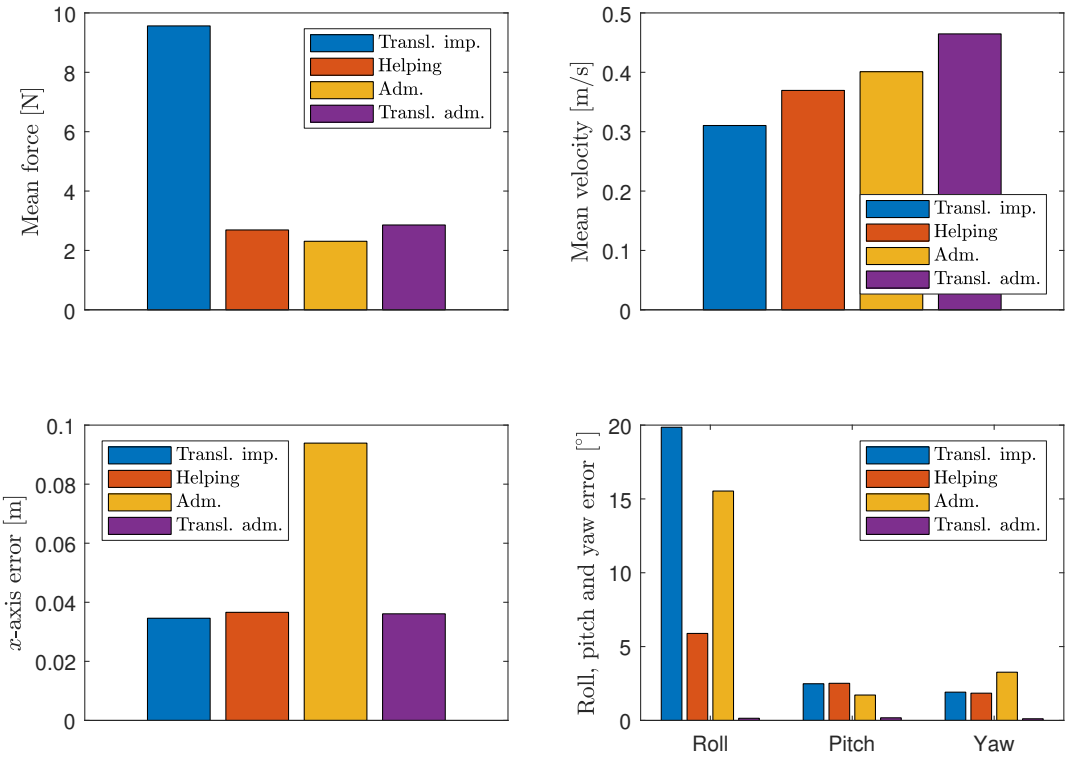


FIGURE 2.67: Evaluation parameters for the circular motion test on the y-z plane.

The conclusions that can be drawn for these results are the following:

- The admittance control scheme is surely the one to be preferred in the present context: impedance control, even in its translational only version, has poorer results; helping control, instead, has comparable results in itself, but would pose several problems when it had to be associated with null space command. In fact, the null space motion would also generate end effector forces that could not be distinguished from the "clean" end effector forces indicating the desired motion directions, which would affect very negatively the quality of control. With admittance control instead, the null space velocity command would not disturb in any way the end effector motion.

- It has to be underlined that such a conclusion is valid in the context of the Franka robot control framework: in fact, the problems related to the impedance and helping control schemes are mainly due to two factors: 1) an imperfect knowledge of the internal mechanics of the transmission system in the joints, resulting in an inaccurate dynamic model; 2) the limitations and "black box" nature of the joint torque interface. If the robotic system had been completely open, it would have been possible to command the real motor torques, i.e., the torques acting on the shafts before the harmonic drives. Moreover, with a complete knowledge of the model, it would have been possible to implement as well the impedance control scheme, and to avoid the errors in the torque null space command.

- Nevertheless, it could be argued (but this is a merely hypothetical statement, due to the impossibility to prove this with data from the robotic system) that in any case the admittance scheme is preferable. In fact, using the torque interface relies on an extremely good knowledge of the model, because even small errors can propagate and be amplified in an unpredictable way, especially in impedance control. Instead, inaccuracies of the model would be less detrimental for the admittance control scheme, even if the internal controller commanding the torques to the joints in order to track the desired velocity trajectory had to be developed instead of being provided by Franka. In fact, a knowledge of the model would be helpful to improve the performance of such a controller, but basic proportional, integral and derivative actions, or other more sophisticated strategies, could compensate for modelling errors, even if they were significant.

Chapter 3

Null space control

3.1 Introduction

As already introduced, once the strategy for the end effector control has been defined, it is necessary to also compute the null space command so as to avoid singularities, joint limits and collisions of the human operator with the kinematic structure of the robot. For the reasons explained in Section 2.7, only velocity null space commands will be considered, because of the inherent problems with torque null space. In this framework, during the PhD two innovative null space control techniques have been introduced, and they have been presented at the 2023 International Conference of Robotics and Automation (ICRA) in London ([48]). In the following, these algorithms will first be contextualized in the literature about null space, then they will be formalized, tested in a relevant case study and proved to be adequate for real-time implementation with the Franka Emika Control system.

The most common approach to solve the inverse kinematics of a redundant robot is the Gradient Projection Method (GPM), originally introduced by Liegeois [41]. GPM computes the null space command as the product of the opposite of the gradient of some cost function by a matrix which projects it on the null space of the Jacobian, with the projection matrix being such that the obtained null space vector is the most similar to the opposite of the gradient.

Several cost functions are presented in the literature, for example in [41] and [81] a function of the distance from the central joint position is used for joint position limit avoidance; in [42], [12] and [82] obstacle avoidance is obtained using different types of functions of the distance between the robot and obstacles in the workspace; [79] and [39] propose to use different types of measures of the manipulability in order to ensure singularity avoidance.

Other methods ([20], [29]) recur to the solution of a quadratic programming problem, enforcing constraints on joint limits and on the distance from obstacles in the workspace; [35] extends [20] to include several objectives with different levels of priority. Workspace augmentation ([64]) uses a method for kinematic inversion based on the transpose of the Jacobian and extends it including constraints in the form of an error which is made to converge to zero. Full Space Parametrization and Parametrization through Null Space ([55] and [36]) seek a linear combination of the elements of a basis of the null space of the Jacobian to satisfy the constraints.

Also [56], [59] and [60] recur to the solution of an optimization problem in which the decision variables are the coefficients of the linear combination of the null space of the Jacobian, but [56], when applied to obstacle avoidance, finds the solution that is most similar to a desired “escape velocity", chosen without taking into account the joint limits, while [59] and [60] aim at finding the time-optimal trajectory respecting the joint limits.

It would theoretically be possible to address redundancy resolution using inverse dynamics, such as in [15], but in this Chapter only velocity null space approaches will be treated, because torque null space is not applicable with the Franka joint torque interface, for the reasons discussed in Section 2.7.

3.2 Formalization of GPM and NSBM as constrained optimization problems

The idea for these algorithms came from the observation of a collision avoidance case study with the Franka Emika Panda robot, represented in Figure 3.1. The robot has to distance itself from a planar obstacle, while the end effector must not move. The null space command is computed according to the GPM formulation

$$\dot{\mathbf{q}}_{null} = -k\left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)\frac{\partial w\left(\mathbf{q}\right)}{\partial \mathbf{q}}, \tag{3.1}$$

where $w\left(\mathbf{q}\right)$ is a function that it is desired to minimize (in this case, the opposite of the distance between the robot and the plane) and k is a gain that can freely be chosen by the user.

Starting from the initial joint configuration $[0, \text{-}0.7854, 0, 2.3562, 0, 2.0071, 0]$ rad, it can be observed, that, however large k is chosen, the null space command will always be null, so there will be no distancing from the plane. At the same time, if the position of the first joint is slightly changed, so that the starting configuration is instead $[0.1, \text{-}0.7854, 0, 2.3562, 0, 2.0071, 0]$ rad, and k is chosen as 10, the resulting joint velocities are very low, and therefore the robot manages to distance itself very little from the plane (see Figure 3.2). Instead, when k is set as 50, the joint velocities increase along time, and they reach large values around 5 rad/s towards the end of the motion, so that the distancing from the obstacle is much more significant this time (see Figure 3.3).

These examples show that the choice of the gain is far from obvious, because the effect of a certain gain can change a lot depending on the current joint configuration of the robot. For example the gain of 50 generates a very slow motion in the initial configuration of Figure 3.3, but when the configuration has changed the same gain generates a really fast movement, even violating the maximum joint velocity limits.

As a result, it is useful to find a method to choose the gain in a "scientific" way, without having to tune it manually, even because the joint velocity interface requires sending commands that comply with the joint limits on position, velocity, acceleration and jerk.

The solution is to treat the choice of the null space command as a constrained optimization problem. The traditional GPM, as already introduced, is based on the following solution of the differential inverse kinematics problem

$$\dot{\mathbf{q}} = \mathbf{J}^{\dagger}\mathbf{v}_e - k\left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)\frac{\partial w\left(\mathbf{q}\right)}{\partial \mathbf{q}}. \tag{3.2}$$

m is the dimension of the operational space task, and n is the dimension of the joints space, so $\mathbf{J} \in \mathbb{R}^{mxn}$, I is an nxn identity matrix, $\mathbf{v}_e \in \mathbb{R}^m$ is the vector of desired end effector linear/angular velocity and $\left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right) \in \mathbf{R}^{nxn}$ is the matrix projecting the gradient of the function $w\left(\mathbf{q}\right)$ onto the null space of the Jacobian.

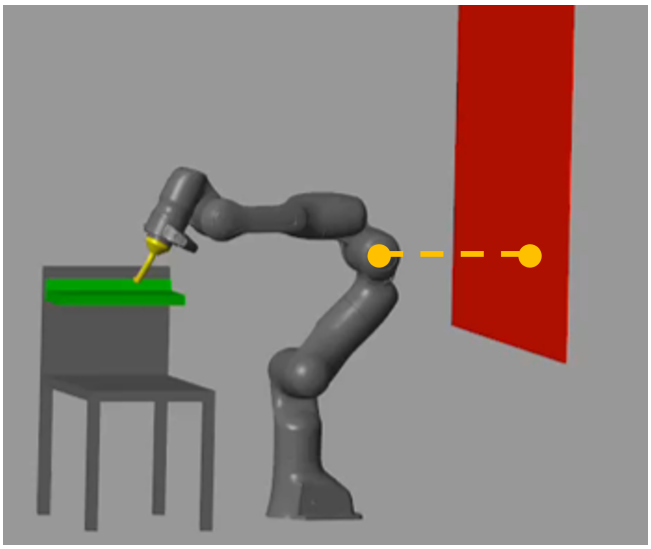


FIGURE 3.1: Collision avoidance case study.

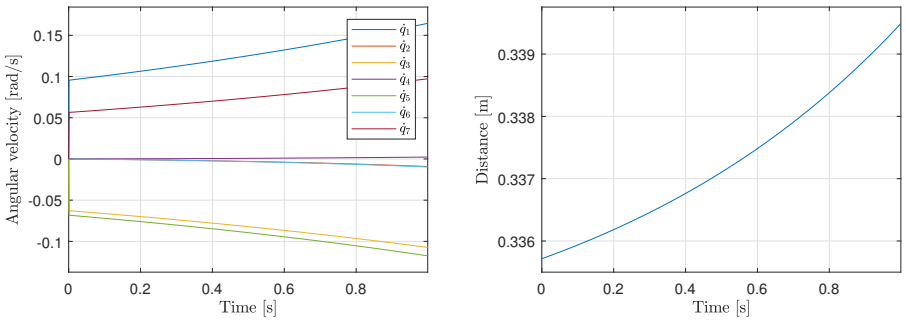


FIGURE 3.2: Joint velocities and distance between the robot and the plane with a gain of 10.

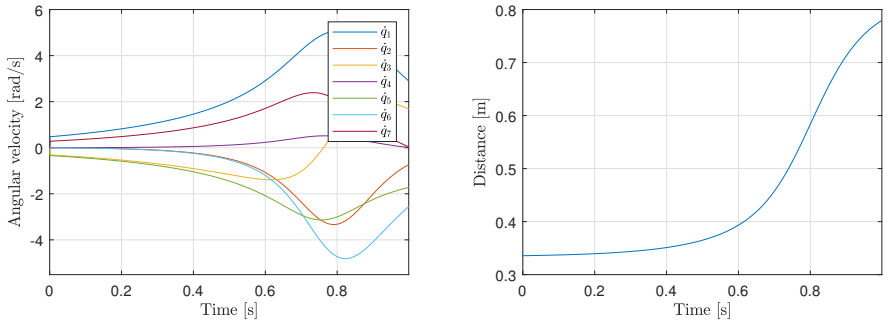


FIGURE 3.3: Joint velocities and distance between the robot and the plane with a gain of 50.

To formalize GPM as a constrained optimization problem it is needed to define a cost function and a set of constraints. As for the cost function, it is a known result in the literature ([17]) that choosing k as large as possible is always the choice guaranteeing the maximum decrease of $w(\mathbf{q})$, unless the configuration for which $w(\mathbf{q})$ is minimal is near. In that case, the best option would be to choose k so that the robot reaches the optimal configuration: if instead the maximum possible k is chosen, the robot starts oscillating around the optimal configuration, as observed in [18]. This case, however, will never happen in practice in the proposed framework, because the null space will be deactivated when the function $w(\mathbf{q})$ exceeds a certain threshold. So the cost function of the optimization problem will be as simple as $-k$.

As for the constraints, we consider a robotic system which imposes asymmetric position bounds $\mathbf{q}_{max}/\mathbf{q}_{min}$ and symmetric velocity, acceleration and jerk bounds $\dot{\mathbf{q}}_{max}/-\dot{\mathbf{q}}_{max}$, $\ddot{\mathbf{q}}_{max}/-\ddot{\mathbf{q}}_{max}$ and $\dddot{\mathbf{q}}_{max}/-\dddot{\mathbf{q}}_{max}$. It has to be observed that it is not sufficient to simply set the limits on the maximum/minimum positions, velocities, accelerations and jerks, as

$$\mathbf{q}_{min} \leq \mathbf{q} \leq \mathbf{q}_{max} \tag{3.3}$$
$$\dot{\mathbf{q}}_{min} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{max} \tag{3.4}$$
$$\ddot{\mathbf{q}}_{min} \leq \ddot{\mathbf{q}} \leq \ddot{\mathbf{q}}_{max} \tag{3.5}$$
$$\dddot{\mathbf{q}}_{min} \leq \dddot{\mathbf{q}} \leq \dddot{\mathbf{q}}_{max}\,. \tag{3.6}$$

In fact, supposing that a joint is travelling towards its velocity limit at maximum acceleration, the acceleration should drop instantly to 0 when the joint velocity at the next iteration would exceed the maximum velocity. But in this case, bringing to 0 the acceleration in just one iteration may overcome the limit on the minimum jerk. Instead, the decrease of the acceleration should be gradual, taking into account also the jerk limit. An analogous consideration holds for the velocity decrease when a joint is approximating its position limit.

So the constraints have to be given with a certain safety margin, as

$$\dot{\mathbf{q}}_{min,safe} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{max,safe} \tag{3.7}$$
$$\ddot{\mathbf{q}}_{min,safe} \leq \ddot{\mathbf{q}} \leq \ddot{\mathbf{q}}_{max,safe} \tag{3.8}$$
$$\dddot{\mathbf{q}}_{min} \leq \dddot{\mathbf{q}} \leq \dddot{\mathbf{q}}_{max}\,, \tag{3.9}$$

where the limit on position is not explicitly present any more because it is "included" in the safety margin of the velocity limit, and also the velocity limit is "included" in the safety margin of the acceleration limit.

In the following it will be explained how to compute an adequate numeric value for $\dot{\mathbf{q}}_{min,safe}$, $\dot{\mathbf{q}}_{max,safe}$, $\ddot{\mathbf{q}}_{min,safe}$ and $\ddot{\mathbf{q}}_{max,safe}$. We will treat the case of $\ddot{\mathbf{q}}_{max,safe}$, which can easily be extended to all others. Similarly to the approach presented in [23], we consider that the joint acceleration limit must be such that, at any iteration, it is possible to reduce the acceleration to 0 applying a constant minimum jerk without exceeding the maximum velocity.

At each iteration h and for each joint $i = 1, \ldots, n$ the velocity at the next iteration is approximated with backward Euler method as

$$\dot{q}_{i,h+1} = \dot{q}_{i,h} + \ddot{q}_{i,h+1} T \tag{3.10}$$

where T is the sampling time. Based on these considerations, and given the initial velocity $\dot{q}_{i,0}$ and acceleration $\ddot{q}_{i,0}$ of a sequence of iterations, it is desired to find the value of acceleration $\ddot{q}_{i,1}$ such that the maximum velocity value in the sequence is

$\dot{q}_{i,max}$ if the acceleration decreases at a rate of $-\dddot{q}_{i,max}$.
In this case, the values of acceleration and velocity at the h-th iteration are

$$\ddot{q}_{i,h} = \ddot{q}_{i,1} - (h-1)\,\dddot{q}_{i,max} T \tag{3.11}$$

$$\dot{q}_{i,h} = \dot{q}_{i,0} + h\ddot{q}_{i,1} T - \frac{(h-1)\,h}{2}\,\dddot{q}_{i,max} T^2. \tag{3.12}$$

So the acceleration reaches zero (and the velocity reaches its maximum value) at

$$h_{max} = \frac{\ddot{q}_{i,1}}{\dddot{q}_{i,max} T} + 1. \tag{3.13}$$

As a result, the maximum allowable value for $\ddot{q}_{i,1}$ is the solution of the equation

$$\dot{q}_{i,h_{max}} = \dot{q}_{i,max} \tag{3.14}$$

namely

$$\ddot{q}_{i,1,max} = -\dddot{q}_{i,max} \left(\frac{T}{2} - \sqrt{\frac{\dddot{q}_{i,max} T - 8\dot{q}_{i,0} + 8\dot{q}_{i,max}}{4\,\dddot{q}_{i,max}}} \right). \tag{3.15}$$

In this way the value for $\ddot{\mathbf{q}}_{max,safe}$ has been found. Extending (3.15) to $\ddot{\mathbf{q}}_{min,safe}$, $\dot{\mathbf{q}}_{max,safe}$ and $\dot{\mathbf{q}}_{min,safe}$, the constraints on velocity and acceleration can be ultimately cast as

$$\dot{q}_{i,max,safe} = \min \left(\dot{q}_{i,max}, \max \left(0, -\ddot{q}_{i,max} \left(\frac{T}{2} - \sqrt{\frac{\ddot{q}_{i,max} T^2 - 8q_i + 8\,(q_{i,max} - q_{i,mar})}{4\ddot{q}_{i,max}}} \right) \right) \right) \tag{3.16}$$

$$\dot{q}_{i,min,safe} = \max \left(-\dot{q}_{i,max}, \min \left(0, \ddot{q}_{i,max} \left(\frac{T}{2} - \sqrt{\frac{\ddot{q}_{max,i} T^2 + 8q_i + 8\,(q_{i,min} - q_{i,mar})}{4\ddot{q}_{i,max}}} \right) \right) \right) \tag{3.17}$$

$$\ddot{q}_{i,max,safe} = \min \left(\ddot{q}_{i,max}, \max \left(0, -\dddot{q}_{i,max} \left(\frac{T}{2} - \sqrt{\frac{\dddot{q}_{i,max} T^2 - 8\dot{q}_i + 8\,(\dot{q}_{i,max} - \dot{q}_{i,mar})}{4\,\dddot{q}_{i,max}}} \right) \right) \right) \tag{3.18}$$

$$\ddot{q}_{i,min,safe} = \max \left(-\ddot{q}_{i,max}, \min \left(0, \dddot{q}_{i,max} \left(\frac{T}{2} - \sqrt{\frac{\dddot{q}_{max,i} T^2 + 8\dot{q}_i + 8\,(\dot{q}_{i,max} - \dot{q}_{i,mar})}{4\,\dddot{q}_{i,max}}} \right) \right) \right) \tag{3.19}$$

where $q_{i,mar}$ and $\dot{q}_{i,mar}$ are safety margins that take into account the fact that the number of iterations to reach 0 velocity/acceleration as expressed in (3.14) may not be an

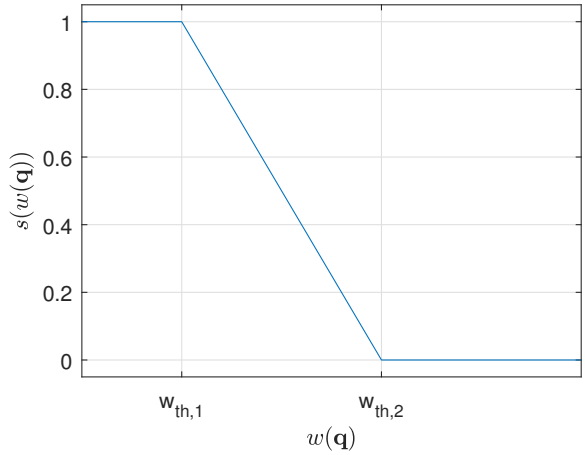


FIGURE 3.4: Behavior of the null space activation/deactivation factor $s\left(w\left(\mathbf{q}\right)\right)$.

integer number and that some additional space is needed to bring the acceleration from its initial value to $-\ddot{q}_{i,max}$ without violating the jerk limit to enforce the position limit.

In addition to the constraints (3.16)–(3.19), more constraints are added to guarantee a gradual activation and deactivation of the null space command (3.1), in the form

$$-s\left(w\left(\mathbf{q}\right)\right)\dot{\mathbf{q}}_{max} \leq \dot{\mathbf{q}}_{null} \leq s\left(w\left(\mathbf{q}\right)\right)\dot{\mathbf{q}}_{max}\,, \tag{3.20}$$

where $s\left(w\left(\mathbf{q}\right)\right)$ is a factor ranging from 0 to 1 defined as

$$s\left(w\left(\mathbf{q}\right)\right) = \begin{cases} 1 & \text{if } w\left(\mathbf{q}\right) \leq w_{th,1} \\ m_{th}w\left(\mathbf{q}\right) + q_{th} & \text{if } w_{th,1} \leq w\left(\mathbf{q}\right) \leq w_{th,2} \\ 0 & \text{if } w\left(\mathbf{q}\right) \geq w_{th,2} \end{cases} \tag{3.21}$$

with $m_{th} = \frac{1}{(w_{th,1} - w_{th,2})}$ and $q_{th} = \frac{w_{th,2}}{w_{th,2} - w_{th,1}}$.

In practice, the minimum/maximum values of the null space command are the minimum/maximum velocity when $w\left(\mathbf{q}\right)$ is lower than a threshold $w_{th,1}$ and then linearly decrease until they become 0 when a second threshold $w_{th,2}$ is crossed ($w_{th,1} < w_{th,2}$) (see Figure 3.4). $w_{th,1}$ is the value below which it is desired to fully exploit the robot potential in terms of decrease of the cost function; $w_{th,2}$, instead, is tuned to avoid abruptly and repeatedly switching between the activation and deactivation of the null space.

So the overall optimized version of GPM, which from now on will be called Optimized Gradient Projection Method (OGPM) can be formalized as

$$\min_{k} \quad -k \tag{3.22a}$$

$$\text{s.t.} \quad \dot{\mathbf{q}}_{min,safe} \leq \dot{\mathbf{q}}\left(k\right) \leq \dot{\mathbf{q}}_{max,safe} \tag{3.22b}$$

$$\ddot{\mathbf{q}}_{min,safe} \leq \ddot{\mathbf{q}}\left(k\right) \leq \ddot{\mathbf{q}}_{max,safe} \tag{3.22c}$$

$$-\dddot{\mathbf{q}}_{max} \leq \dddot{\mathbf{q}}\left(k\right) \leq \dddot{\mathbf{q}}_{max} \tag{3.22d}$$

$$-s\dot{\mathbf{q}}_{max} \leq \dot{\mathbf{q}}_{null}\left(k\right) \leq s\dot{\mathbf{q}}_{max}, \tag{3.22e}$$

where $\dot{\mathbf{q}}$ (and therefore also $\ddot{\mathbf{q}}$ and $\dddot{\mathbf{q}}$) and $\dot{\mathbf{q}}_{null}$ depend on k according to (3.2) (3.1).

Applying OGPM would be very helpful to choose appropriately the gain to decrease efficiently the value of $w(\mathbf{q})$ while at the same time respecting the limits of the joint interface. However, the first example that was considered showed a case in which, whatever the value of the gain was, the robot was not able to move away. This would seem surprising at first, because it is very evident that, starting from the configuration of Figure 3.1, it is possible to move the robot to increase the distance from the plane without moving the end effector. The reason why GPM and OGPM fail to bring about a null space motion is that in the starting position the gradient $\frac{\partial w(\mathbf{q})}{\partial \mathbf{q}}$ is [0, -0.2818, 0, 0, 0, 0, 0] and this vector belongs to the null space of the projection matrix $\left(\mathbf{I} - \mathbf{J}^{\dagger}\mathbf{J}\right)$. Actually, this is a very unfortunate case. In fact, it is enough to slightly change the initial configuration (choosing a value different from 0 for the first joint, even a very small value) to solve this issue. As in the previous example, if the initial configuration is [0.1, -0.7854, 0, 2.3562, 0, 2.0071, 0] rad, the gradient does not belong any more to the projection matrix, but, since the configuration for which the gradient does belong to the null space is quite close, the projection of the gradient in this configuration is still a very small vector. This is the reason why, in the test of Figure 3.3, a gain of 50 initially generates a very slow motion.

These considerations lead to the conclusion that GPM and OGPM don't necessarily lead to the best possible choice of the null space command. In fact, as stated in [11], GPM constrains the null space vector to have the same direction of $\left(\mathbf{I}_n - \mathbf{J}^{\dagger}\mathbf{J}\right)\left(\frac{\partial w(\mathbf{q})}{\partial \mathbf{q}}\right)^T$, namely GPM allows exploring only a one-dimensional subspace of the null space of $\mathbf{J}$, but the best possible solution does not necessarily belong to this subspace. In fact, the vector $-\left(\frac{\partial w(\mathbf{q})}{\partial \mathbf{q}}\right)^T$ is along the direction of maximum decrease of the cost function, but since this vector has to be projected onto the null space of $\mathbf{J}$ and the limits (3.22b)–(3.22e) have to be enforced, it is not guaranteed that the direction $-\left(\mathbf{I}_n - \mathbf{J}^{\dagger}\mathbf{J}\right)\left(\frac{\partial w(\mathbf{q})}{\partial \mathbf{q}}\right)^T$ is the optimal one.

As a result, an alternative formulation is introduced. This formulation is based on the following solution of the differential inverse kinematics problem (see [60]), which is more general than (3.2),

$$\dot{\mathbf{q}} = \mathbf{J}(\mathbf{q})^{\dagger}\mathbf{v}_e + \mathbf{B}(\mathbf{q})\mathbf{a} \tag{3.23}$$

where $\mathbf{B} \in \mathbb{R}^{nxr}$ is a matrix whose columns are the vectors of a basis of the null space of $\mathbf{J}$ ($r = n - m$ is the dimension of the null space) and $\mathbf{a} \in \mathbb{R}^r$ is a vector of arbitrary coefficients, so that the null space command is a linear combination of the vectors of the null space basis.

In such a framework, it is possible to define

$$\mathbf{q}_{new} = \mathbf{q} + \left(\mathbf{J}(\mathbf{q})^{\dagger}\mathbf{v}_e + \mathbf{B}(\mathbf{q})\mathbf{a}\right)T, \tag{3.24}$$

which is the joint configuration that will be obtained at the next iteration for a certain choice of $\mathbf{a}$. Remembering that the optimization problem that is being solved is a local one, namely it considers only a one-iteration horizon, it can be concluded that the best possible null space command is the one for which $w(\mathbf{q}_{new})$ is minimized.

As a result the optimization problem can be redefined, having $\mathbf{a}$ instead of k as optimization variables, and $w(\mathbf{q}_{new})$ as the cost function. The constraints do not change, with the exception of (3.22e), where the expression of $\dot{\mathbf{q}}_{null}$ is not anymore

(3.1), but is substituted within

$$\dot{\mathbf{q}}_{null} = \mathbf{B}\left(\mathbf{q}\right)\mathbf{a}\,. \tag{3.25}$$

So the overall formulation of the NSBM optimization problem is

$$\begin{aligned}
\min_{\mathbf{a}} \quad & w\left(\mathbf{q} + \left(\mathbf{J}\left(\mathbf{q}\right)^{\dagger}\mathbf{v}_e + \mathbf{B}\left(\mathbf{q}\right)\mathbf{a}\right)T\right) && \text{(3.26a)}\\
\text{s.t.} \quad & \dot{\mathbf{q}}_{min,safe} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{max,safe} && \text{(3.26b)}\\
& \ddot{\mathbf{q}}_{min,safe} \leq \ddot{\mathbf{q}} \leq \ddot{\mathbf{q}}_{max,safe} && \text{(3.26c)}\\
& -\dddot{\mathbf{q}}_{max} \leq \dddot{\mathbf{q}} \leq \dddot{\mathbf{q}}_{max} && \text{(3.26d)}\\
& -s\dot{\mathbf{q}}_{max} \leq \dot{\mathbf{q}}_{null} \leq s\dot{\mathbf{q}}_{max}. && \text{(3.26e)}
\end{aligned}$$

This way of computing the velocity command, differently from GPM and OGPM, spans the entire solution space and therefore intrinsically provides the locally optimal solution, and in case OGPM already provides the optimal solution, the solution of NSBM is eventually the same as that of GPM.

3.3 Simulation results of a case study

The proposed algorithms were tested in a relevant collision avoidance case study and the results are presented in this Section. The testing scenario is the same as in Figure 3.1.

In the first test the end effector must remain in a fixed position and orientation and also the plane is parallel to the y_0-z_0 plane, is initially at $x_0 = -0.5$ m and travels towards the robot at a speed of 0.4 m/s until it reaches the position $x_0 = -0.1$ m (the reference frames are the same as in Figure 2.6): the joints of the robot must move so as to increase the distance between the robot and the plane. It is the same situation of the tests in Figures 3.2 and 3.3, with the only difference that the gain is not chosen arbitrarily, but according to OGPM. The initial configuration is [0.1, -0.7854, 0, 2.3562, 0, 2.0071, 0] rad, to avoid the problem of the gradient belonging to the null space of the projection matrix. The function $w(\mathbf{q})$ that is used in the test is chosen as the opposite of the distance between the center of the fourth reference frame and the plane, which is an approximation of the distance between the robot and the plane. It could be argued that considering only the center of the fourth reference frame is not a good estimation of the distance between the robot and the plane in general. However, in the considered case studies, the point of the robot which is nearest to the plane is always close to the center of the fourth reference frame, so this approximation is reasonable for the considered tests. Additionally, the focus of this test is not on finding a cost function that allows to obtain the best possible obstacle avoidance, but on the algorithms to compute null space commands. The joint limits are those expressed in Table 3.1 and the thresholds $w_{th,1}$ and $w_{th,2}$ for the activation of the null space are set to 0.15 m and 0.25 m respectively.

The results of the test are shown in Figures 3.5 and 3.6. Initially the distance is greater than 0.25 m, so the null space command is inactive (the limits (3.22e) avoid the null space motion), then, when the threshold $w_{th,2}$ is crossed, the velocities start to increase, until they reach a full activation of the null space when the $w_{th,1}$ threshold of 0.15 m is crossed, just before 0.5 s. At this point the constraint that is acting is the one of not anymore (3.22e), but (3.22b) on the maximum velocity: in fact the velocity of the first joint stabilizes at its maximum value of 2.175 rad/s. At 1 s the

TABLE 3.1: Joint bounds, as declared in the documentation of Franka Emika ([25]).

	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6	Joint 7	Unit
q_{max}	2.8973	1.7628	2.8973	-0.0698	2.8973	3.7525	2.8973	rad
q_{min}	-2.8973	-1.7628	-2.8973	-3.0718	-2.8973	-0.0175	-2.8973	rad
$\dot{q}_{max}$	2.175	2.175	2.175	2.175	2.61	2.61	2.61	rad/s
$\ddot{q}_{max}$	15	7.5	10	12.5	15	20	20	rad/s^2
$\dddot{q}_{max}$	7500	3750	5000	6250	7500	10000	10000	rad/s^3

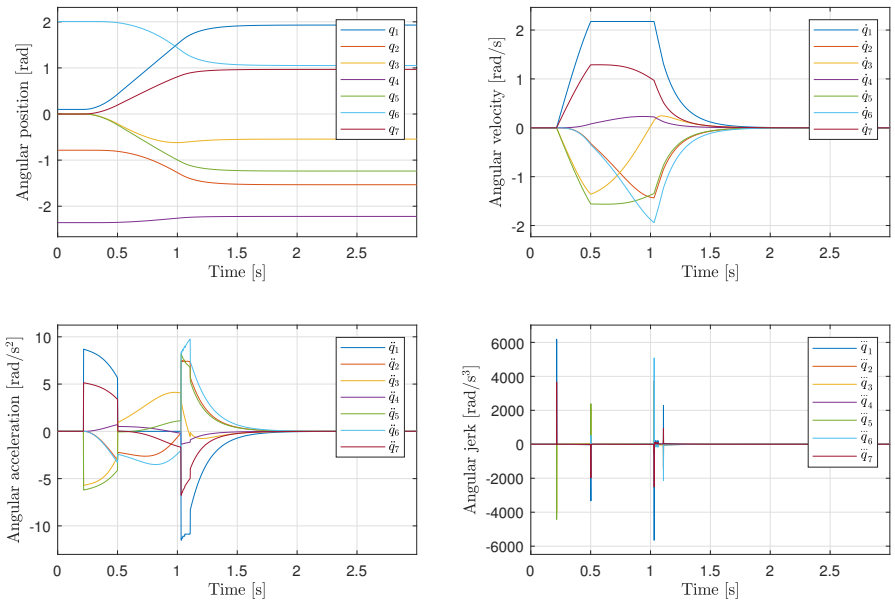


FIGURE 3.5: Positions, velocities, accelerations and jerks in the test with OGPM.

planar obstacle stops, and the distance from the obstacle increases until the null space deactivation value of 0.25 m is reached again. It can be observed that, since the obstacle was approaching at a high speed, the distance from the robot decreased in spite of the null space motion. Yet, that null space motion, though initially not fast enough to increase the distance, managed to avoid the collision with the obstacle. Additionally, even before the end of the motion of the planar obstacle, around 0.75 s, the distance began to increase, showing even better the efficacy of the null space command. Considering the behavior along the test of the value of the gain k in Figure 3.6, it can be observed that it changes very significantly, and it would have been very difficult to properly choose it in this way without the optimization approach that was introduced.

In the previous case only OGPM was tested, because the task space had dimension 6 (the end effector could neither change its position nor its orientation), so the dimension of the null space was 1. In such a situation NSBM will always provide the same result as OGPM, because there is only one possible direction of the null

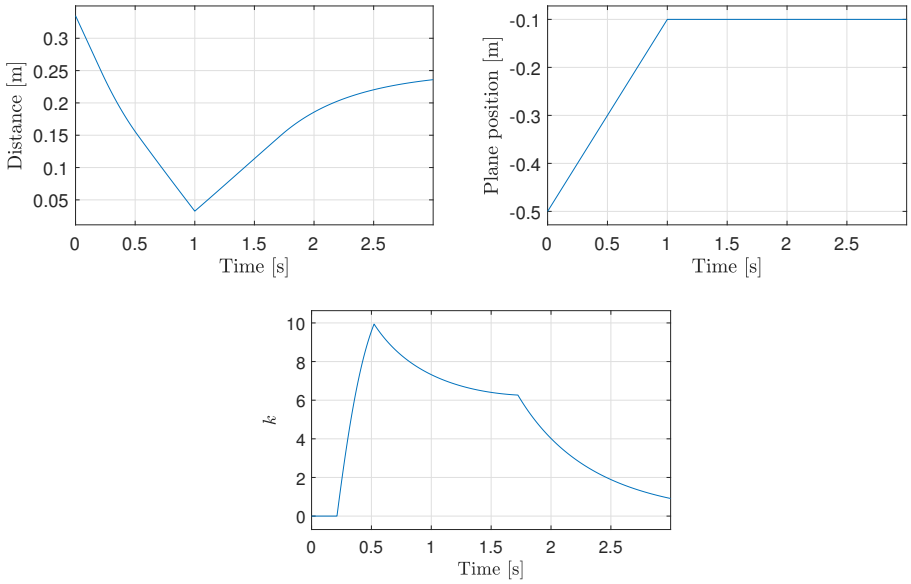


FIGURE 3.6: Distance from the plane, position of the plane and value of the gain in the test with OGPM.

space, so the gradient of any vector will always be projected onto the same direction individuated by the null space basis of NSBM.

To compare OGPM and NSBM another case study is introduced in which the dimension of the task space is 5 instead of 6. In particular, we consider a welding application in which the welding tip attached to the end effector must maintain its position and orientation with respect to the welding surface, but can freely rotate about its own axis. It is stressed that, in the present context, a non-collaborative welding application was chosen just to better exemplify the difference between GPM and NSBM. In fact, the motions of the end effector that are related to hand-guiding tasks involve six degrees of freedom. Even in the considered case of translational-only hand-guiding, the rotational degrees of freedom are locked, so their related angular velocities are fixed to zero and therefore the dimension of the operational space task is 6.

In the welding example application, the velocity command is set as $\mathbf{v}_e = \mathbf{0} \in \mathbb{R}^5$ and does not include the angular velocity around y_e, while the Jacobian does not include the row related to the rotation around y_e as well. As in the previous case, there is a planar obstacle travelling towards the robot, and the null space motion must be such to avoid the collision, as shown in Figure 3.7. To better emphasize the potential of NSBM to improve the performance of OGPM, the limit on the maximum velocity of the second joint is set to 0.5 rad/s instead of 2.175 rad/s.

Figures 3.8–3.12 show the results of this second comparative test: NSBM performs much better than OGPM. In fact, the projection of the gradient has a large component on the second joint, but the speed limit of this joint is soon reached, and therefore GPM performs rather poorly in this case, being unable to increase the distance while the plane is moving in the positive x direction. Actually, there exist null space directions which could guarantee a better performance than the projection of the gradient, but GPM is "unaware" of the fact that the joint bounds may deteriorate the performances along the gradient projection direction and cannot explore those

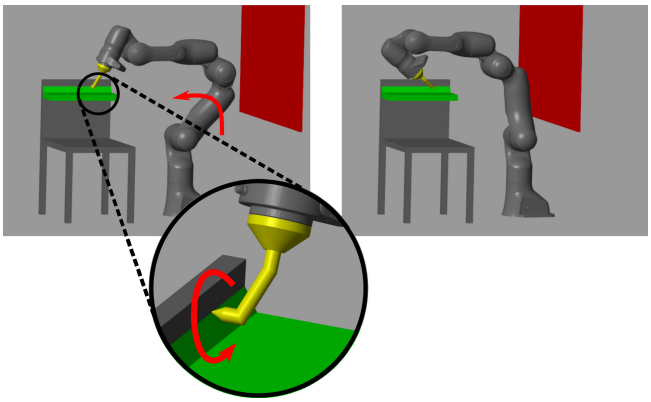


FIGURE 3.7: Two frames from the welding application: the null space command enables the tip of the welder to rotate about its axis, as shown in the right frame.

directions. NSBM, instead, exploits the knowledge of the joint bounds to choose the direction of the null space command, generating a motion of all other joints, which allows not to saturate the velocity limit on joint 2 while increasing successfully the distance between the center of the fourth reference frame and the plane. This is also shown visually in Figure 3.13, where it can be observed that the motion according to OGPM causes a collision between the planar obstacle and the robot, while NSBM allows to reconfigure the kinematic chain so as to avoid such a situation.

To provide a more accurate comparison of the performance of the two algorithm in the considered case study, also the "distance increase" was considered in Figure 3.14, namely, for each iteration of the control, the value of the distance at the next iteration obtained without the null space command was subtracted from the value of the distance obtained with the null space command, thus providing a metric of how better NSBM performs over OGPM. Additionally, in the same Figure it is also shown which are the mean values of the distance for OGPM and NSBM, weighting those values with the null space activation parameter $s\left(w\left(\mathbf{q}\right)\right)$.

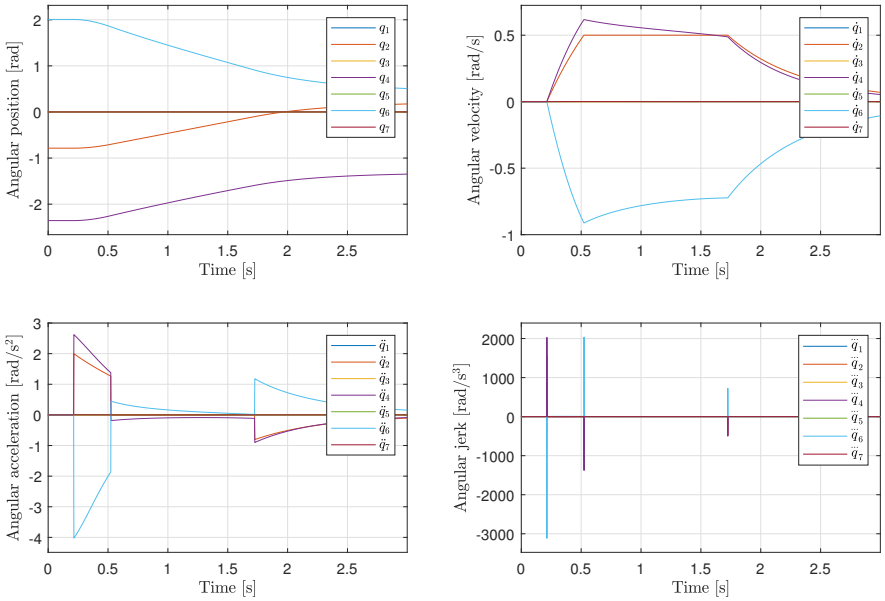


FIGURE 3.8: OGPM positions, velocities, accelerations and jerks in the comparison test between OGPM and NSBM.

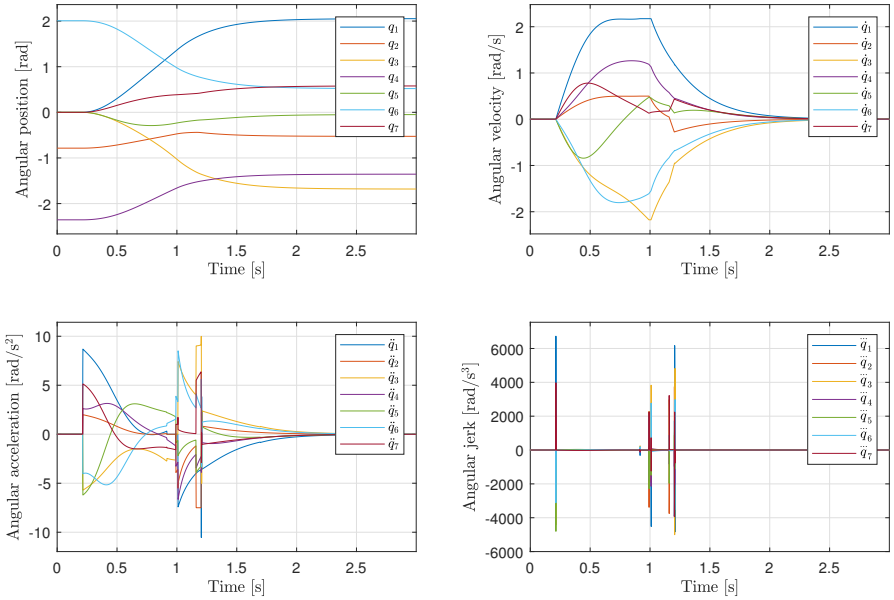


FIGURE 3.9: NSBM positions, velocities, accelerations and jerks in the comparison test between OGPM and NSBM.

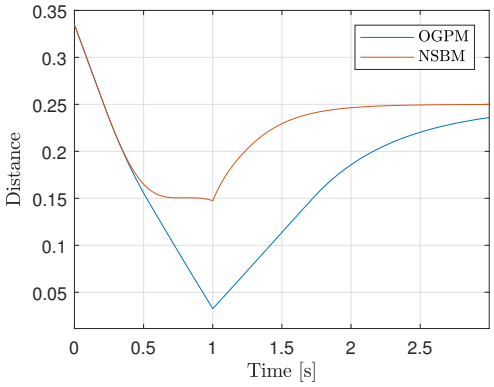


FIGURE 3.10: Comparison of the distance between the robot and the plane with OGPM and with NSBM.

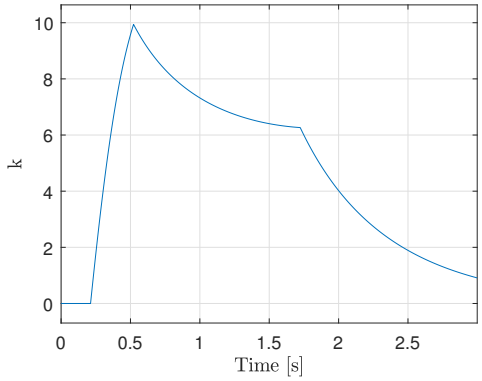


FIGURE 3.11: OGPM value of the gain k in the comparison test between OGPM and NSBM.

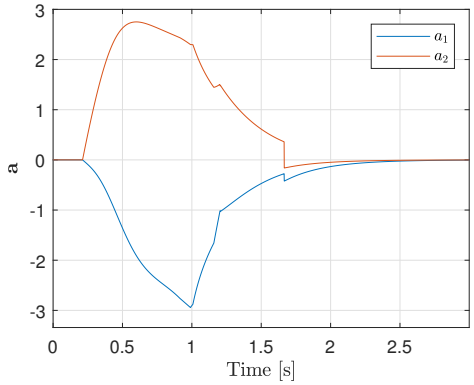
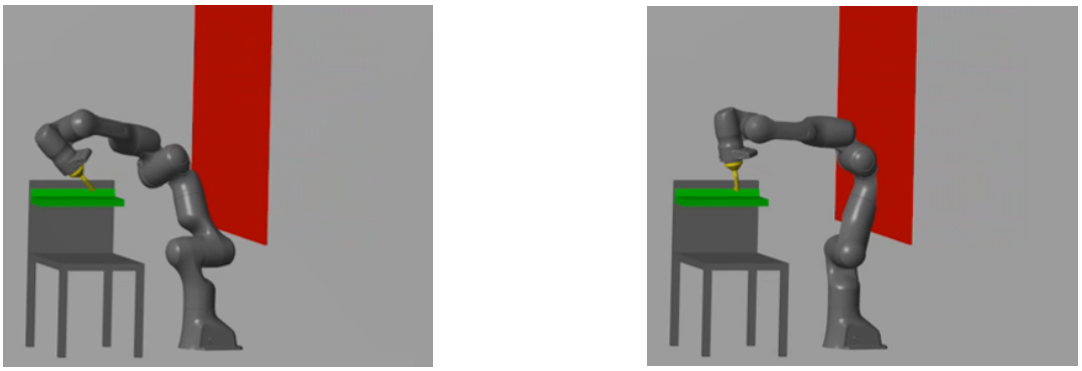
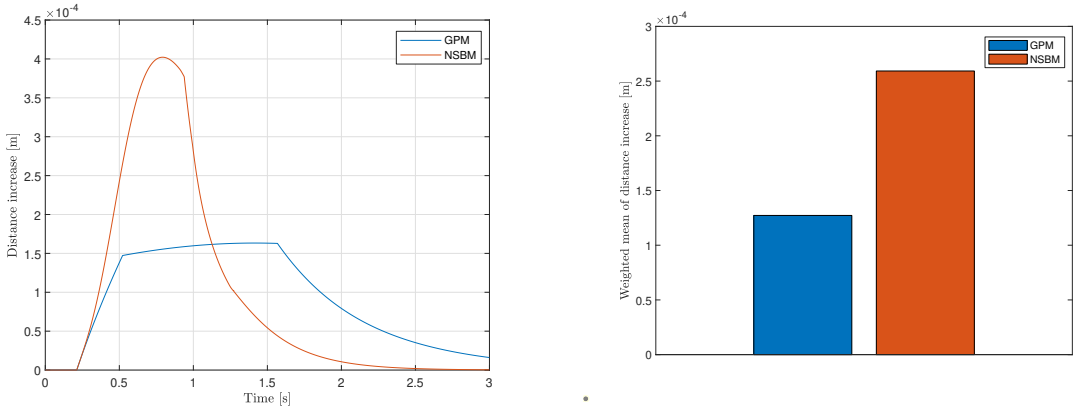


FIGURE 3.12: NSBM value of the gains a_1 and a_2 in the comparison test between OGPM and NSBM.



(a) (b)

FIGURE 3.13: Frames from an animation showing that with OGPM a collision occurs (see (a)), while NSBM manages to avoid the collision (see (b)).



(a) (b)

FIGURE 3.14: Distance increase thanks to OGPM and NSBM, over time ((a)) and mean value weighted on the null space activation level ((b)).

3.4 Real-time implementation

The outlined algorithms were also tested on the real robotic platform, using the joint velocity interface described in Chapter 2. It is therefore needed to meet the real-time constraints of the robot, i.e., the sampling time is 1 ms, so, at each iteration, all the computations needed to provide the joint velocity command must be executed within this period, using a C++ library, instead of the `fmincon` MATLAB function ([24]) that was used for the tests of the previous section. The real-time tests were performed on a PC with a 6-core Intel i7-5820K CPU, running at 3.3 GHz.

The GPM algorithm is intrinsically a linear optimization problem, because both the function (3.22a) to be minimized and the constraints (3.22b)–(3.22e) are linear in the decision variable k. As a result, using the linear programming package of the ALGLIB library [9], the problem was solved obtaining the same solutions as in MATLAB and meeting the requirements.

As for the NSBM algorithm, the function (3.26a) to be minimized is non-linear in $\mathbf{a}$. However, it can be observed that, in the proximity of the origin of $\mathbb{R}^2$, which is the space containing all the possible choices for $\mathbf{a}$, the function can be approximated with a linear function, namely a plane.

So (3.26a) is substituted with its linear approximation

$$\bar{w}\left(a_1, a_2\right) = w\left(0,0\right) + \frac{\partial w\left(0,0\right)}{\partial a_1} a_1 + \frac{\partial w\left(0,0\right)}{\partial a_2} a_2 \tag{3.27}$$

where a_1 and a_2 are the components of $\mathbf{a}$ and the distance function was written only as a function of $\mathbf{a}$ because $\mathbf{q}$ and $\mathbf{v}_e$ are known and $\mathbf{J}$ and $\mathbf{B}$ of (3.24) are functions of $\mathbf{q}$, so all these quantities can be considered as constants at each iteration.

Figure 3.15 shows in the left the behavior of the $w\left(\mathbf{q}\right)$ function at time $t = 0.5$ s with a_1 and a_2 ranging from -10000 to 10000. It can be observed that the function is highly non-linear, and this is due to the sines and cosines involved in the distance function. However, looking at the right part of Figure 3.15, the same $w\left(\mathbf{q}\right)$ function is represented, but for a smaller range of a_1 and a_2. Additionally, the linear (planar) approximation of the function is also represented. It can be observed that for large values of a_1 and a_2 the approximation error is noticeable, but in the proximity of the origin the planar approximation is very accurate. More specifically, considering the whole time history of the case study, the maximum percentage approximation error is 24.53% considering values of a_1 and a_2 ranging from -500 to 500, but decreases to $2.88 \cdot 10^{-4}$% considering values ranging from -5 to 5.

So, since in the application under consideration the values of the components of $\mathbf{a}$ stay in the contour of the origin (their absolute values are always lower than 5 for both components), the linear approximation can safely be used. A further proof of the appropriateness of the approximation is given by the percentage error on the velocity command computed with the linearization with respect to its non-linearized version: this error takes a maximum value of $7.05 \cdot 10^{-3}$%.

As a result, also NSBM was computed in real-time using the ALGLIB linear optimization solver, thus meeting the real-time requirements and reducing the computational cost. It should also be mentioned that the calculation of a null space basis of a matrix is computationally not problematic, and can be performed by several linear algebra libraries without raising real-time issues (in the tests reported here the Armadillo library was used [63]).

The overall computation time was always lower than 0.5 ms for both algorithms, therefore no errors due to the violation of the 1-ms time constraint happened during

the tests.

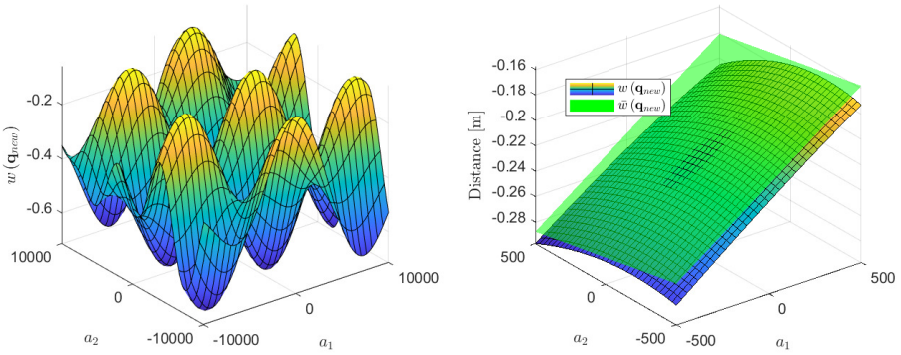


FIGURE 3.15: $w(\mathbf{q}_{new})$ and $\bar{w}(\mathbf{q}_{new})$ as a function of $\mathbf{a}$, at time $t = 0.5$ s.

3.5 Discussion

This Chapter presented a framework for the optimal gain selection of GPM and, most importantly, the NSBM algorithm for the computation of the null space command in redundant robots, whose performance is always better or at least equal with respect to the commonly used GPM, and whose computational cost is not significantly higher than that of GPM and is compatible with the real-time constraints of most robots.

On the one hand, these results have a general importance for the study of null space. In fact, existing techniques available in the literature which use a linear combination of the elements of the null space basis either require long computational times because they were developed for a priori path planning ([59] and [60]) or are not suitable for usage with a general form of the cost function and require to choose a desired operational space “escape velocity" for obstacle avoidance ([56]).

Instead, NSBM uses as objective function the value of a desired criterion at the next iteration for a certain choice of the decision variables, and can therefore easily be applied to any case, such as joint limits or singularity avoidance.

Finally, unlike NSBM, the approaches that treat obstacle avoidance using constraints ([20, 36, 29, 35, 64, 55]), do not necessarily provide the solution that causes the greatest possible decrease of the cost function.

On the other hand, these algorithms are particularly important in the context of the manual guidance application studied in this thesis because they provide an efficient way to compute null space commands so that the kinematic structure of the robot stays away from the operator, avoids singularities and joint limits. The tests presented in this Chapter were about a simplified collision avoidance case study in a welding application, that is not related to a collaborative scenario. Yet, they proved that the proposed algorithms are useful to choose in an optimal way a joint velocity null space command such that the end effector desired velocity is not modified and the joint limits are respected. This is fundamental because the joint velocity interface of Franka Emika strictly demands that the commanded signal fulfills the joint constraints, otherwise it enters an error state.

Future work will involve choosing proper cost functions to be minimized so that a suitable behavior in terms of singularity, joint limits and collision avoidance is obtained. In other words, the value of the results of this Chapter in the context of this thesis is to have provided useful tools to treat null space command computation in an optimized way, which is necessary to develop a fully functional collaborative application.

Chapter 4

Conclusions and future developments

This thesis studied which control strategies are most suitable for a hand guiding application with a Franka Emika Panda robot. The results of this work are summarized in the following points:

- Impedance control schemes, which are commonly used in the literature to deal with hand guiding scenarios, are not appropriate in the considered context. This is not due to intrinsic shortcomings of the impedance methods themselves, but to the functioning of the Franka joint torque interface. In fact, the interface does not provide direct access to the motor torque, instead allowing the user the set the link-side joint torque (the torque after the harmonic drive transmission). However, as was proved in detail in Chapter 2, very significant modelling errors are present due to the losses of the harmonic drive, and, in addition, the user-defined link-side torque command is further conditioned by the Control in a "black box" way, making it hard to implement precise torque control schemes.
- These results regarding inherent problems of the Franka joint torque interface have a strong significance because, at the best of the author's knowledge, they are not documented yet in the literature.
- As a consequence of these shortcomings of the torque interface, a complete impedance control scheme, regulating both the translational and the rotational behavior, is not possible, because the rotations appear "locked", for the reasons detailed in Chapter 2. Instead, if an impedance scheme regulating only the translation is chosen, a better behavior can be achieved, though being far from respecting the set end effector dynamics.
- Another control scheme using the joint torque interface was tried, called "helping" controller, based on providing additional torque to the joints according to the forces and torques exerted on the end effector, in order to compensate for the opposing actions due to the harmonic drives. This controller provides good results in terms of reducing the forces that the human operator needs to apply to drive the end effector.
- However, if null space motions are needed, even the helping controller is not suitable, because the torque null space, again because of the problems with the torque interface, generates forces and torques on the end effector (so it is only nominally a null space, but de facto it is not due to the modelling errors).
- Admittance control schemes provide good results as well in terms of reducing the forces that the human operator needs to apply to drive the end effector. The

control command is in the form of joint velocities, that are sent to the Franka joint velocity interface. In this case, the internal (and proprietary) controller of the robot is able to track the desired trajectories precisely, so the set dynamic behavior of the end effector is respected.

- The velocity null space does not present the same problems of the torque null space.
- Two innovative algorithms for the computation of the null space command have been introduced, Optimized Gradient Projection Method (OGPM) and Null Space Basis Optimal Linear Combination Method (NSBM). These algorithms are suitable for providing joint velocity commands that comply with the constraints on joint position, velocity, acceleration and jerk that are imposed by the Franka joint velocity interface.
- The algorithms have been proved to be efficient in a simplified collision avoidance case study and also to respect the real-time requirements of the Franka Emika Control framework, having an update rate of 1 ms.
- Even if the algorithms for the null space were exemplified in a non-collaborative case study, the results of the second and of the third Chapters come together to define a fully comprehensive control scheme that integrates the admittance end effector control with the null space commands that allow to avoid collision and singularity problems that the end effector control alone would not be able to solve.

In conclusion, it can be stated that in this thesis, the best control framework to treat the considered collaborative scenario has been defined, but there are more steps that need to be developed to obtain a complete and functional collaborative application:

- The rotational part of the admittance controller needs to be further studied, because the results of the controller in Chapter 2 were not completely satisfactory under this respect.
- The OGPM and NSBM algorithms, already tested in a simplified case study, need to be applied to the relevant cases of joint limits avoidance, singularity avoidance and collision avoidance.
- For the collision avoidance, the position of the human operator needs to be tracked. This will be done using IMU sensors. This work has already been started in the context of the SAIMA laboratory ([3]), and will be continued in the following years.
- OGPM and NSBM are local optimization algorithms, meaning that their results are optimal considering only the next iteration: a useful improvement would be to consider an extended time horizon, iteratively solving an optimal control problem. In this way it would be possible to follow more sophisticated predictive strategies to stay away from singularities, joint limits and obstacles.
- In order to obtain a more versatile collaborative application, the final goal is to mount the Franka robot on a mobile platform, similarly to the collaborative system developed by the IIT of Genova (see Figure 4.1). Such a solution meets the increasing demand for human-robot collaboration in industrial settings, and can be applied to a very wide range of contexts. The drilling workstation

FIGURE 4.1: Mobile collaborative application developed at the IIT of Genova [26]

mentioned in the introduction is just a case study, but there are many situations in which such a mobile platform would be very advantageous, especially when there is a need to carry heavy loads and the activity cannot be automatized. The relevance of such a scenario is also testified by the significant amount of research on active and passive exoskeletons to reduce physical load for human operators ([14]). Collaborative (mobile) platform could certainly play a role as a valid substitute of these systems in a number of industrial scenarios.

Appendix A

Additional figures of the impedance control tests

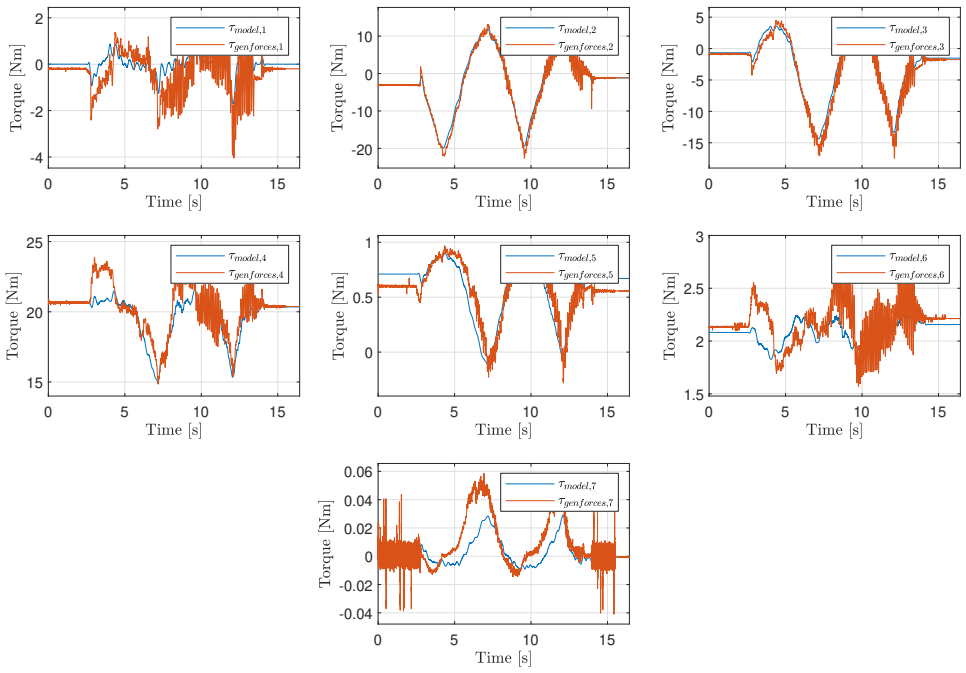


FIGURE A.1: Diagonal motion test with impedance control, with low inertia and damping parameters: dynamic model accuracy.

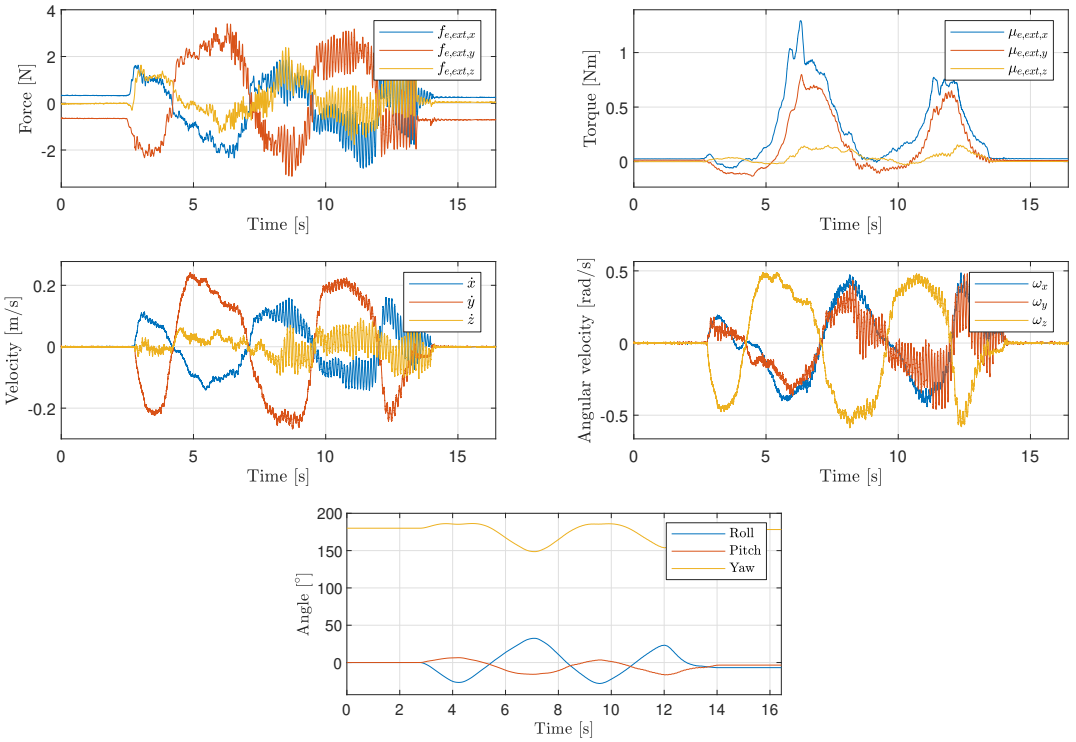


FIGURE A.2: Diagonal motion test with impedance control, with low inertia and damping parameters, linear and angular velocities and roll pitch and yaw angles.

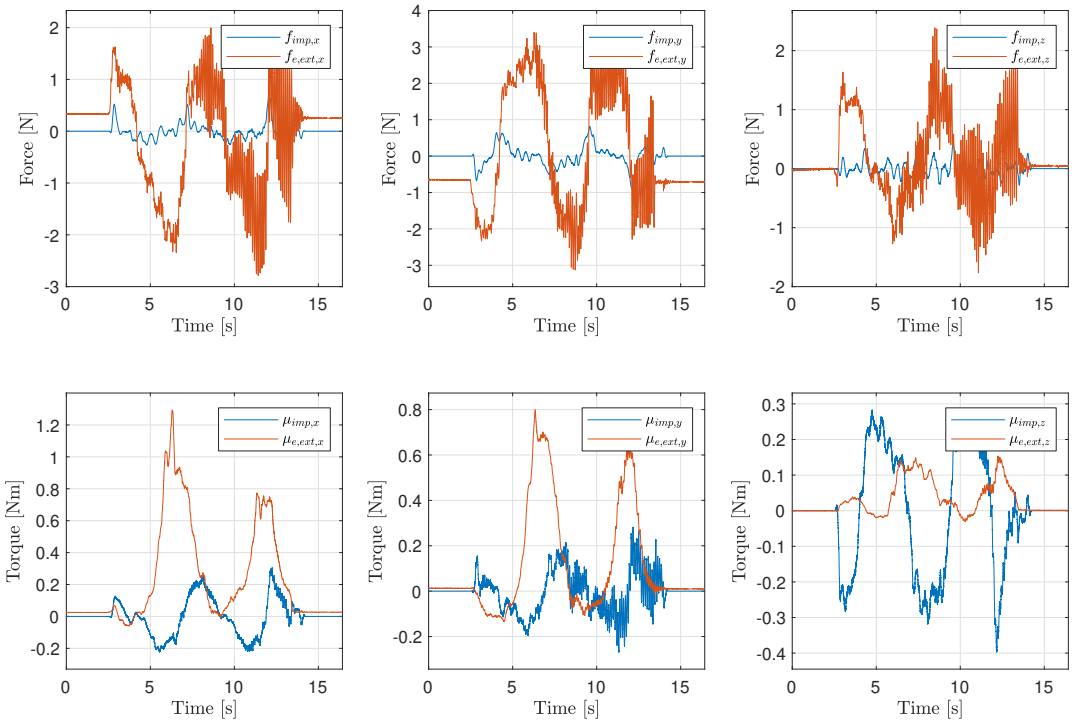


FIGURE A.3: Diagonal motion test with impedance control, with low inertia and damping parameters: comparison of expected and measured forces/torques.

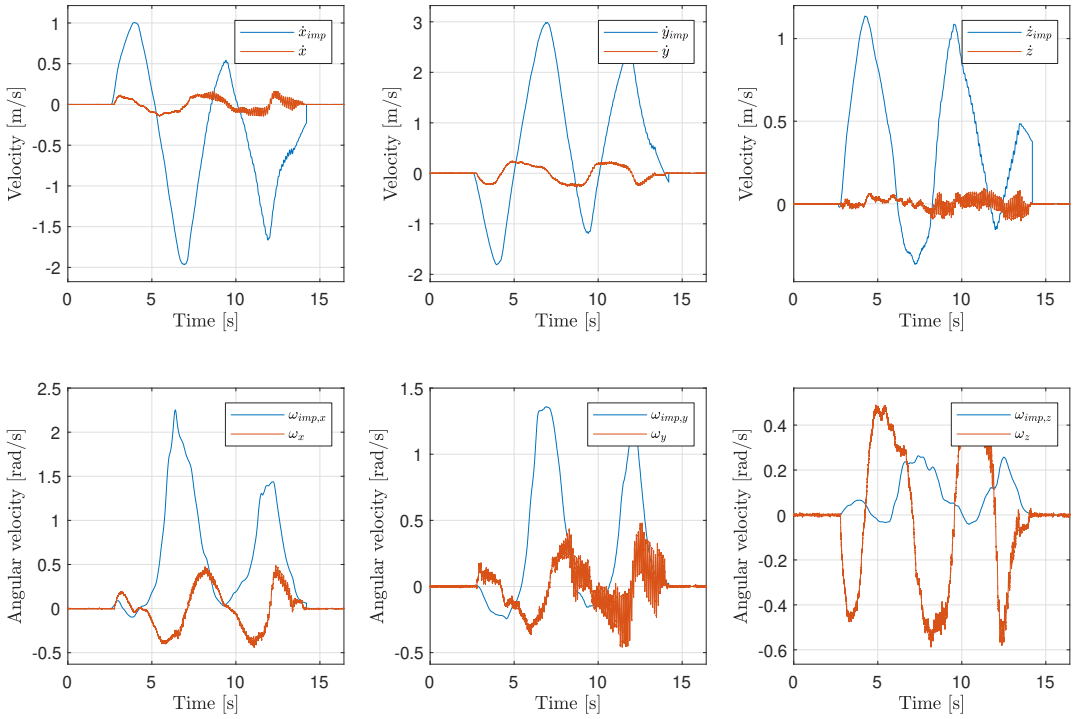


FIGURE A.4: Diagonal motion test with impedance control, with low inertia and damping parameters: comparison of expected and measured linear/angular velocities.

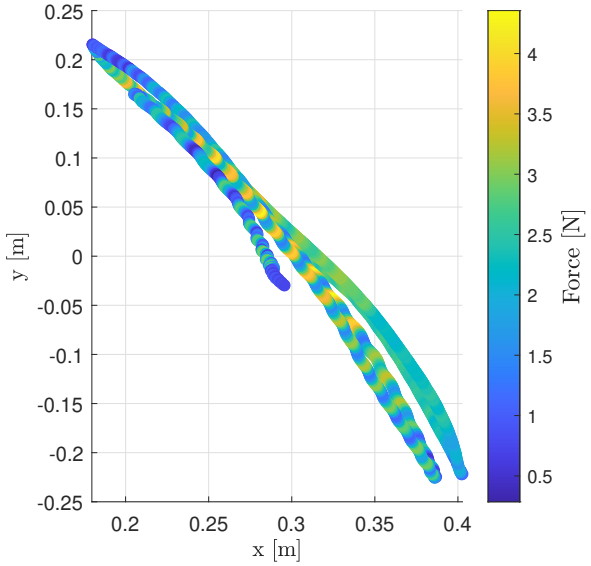


FIGURE A.5: Diagonal motion test with impedance control, with low inertia and damping parameters: norm of the end effector force vector along the trajectory in the *x*-*y* plane.

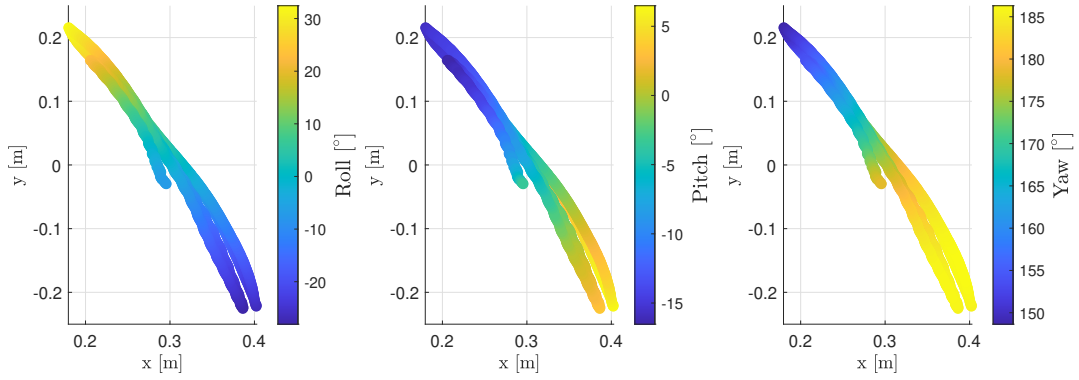


FIGURE A.6: Diagonal motion test with impedance control, with low inertia and damping parameters: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

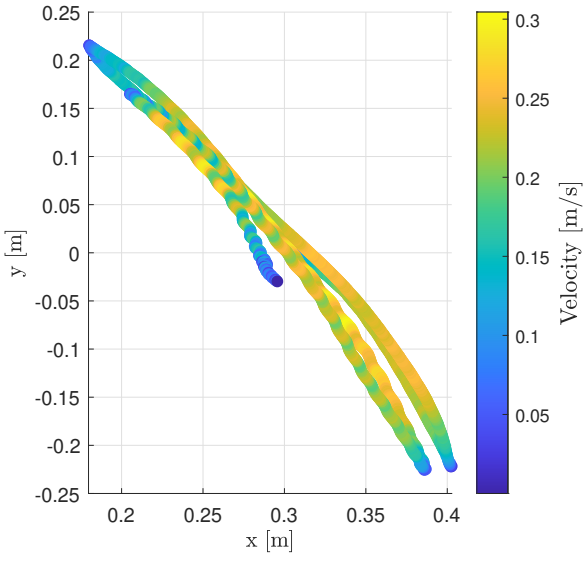


FIGURE A.7: Diagonal motion test with impedance control, with low inertia and damping parameters: norm of the end effector velocity vector along the trajectory in the *x*-*y* plane.

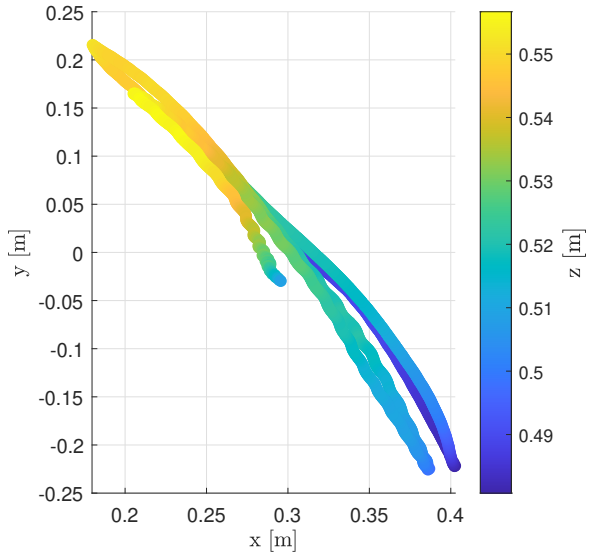


FIGURE A.8: Diagonal motion test with impedance control, with low inertia and damping parameters: height on the z axis along the trajectory in the x-y plane.

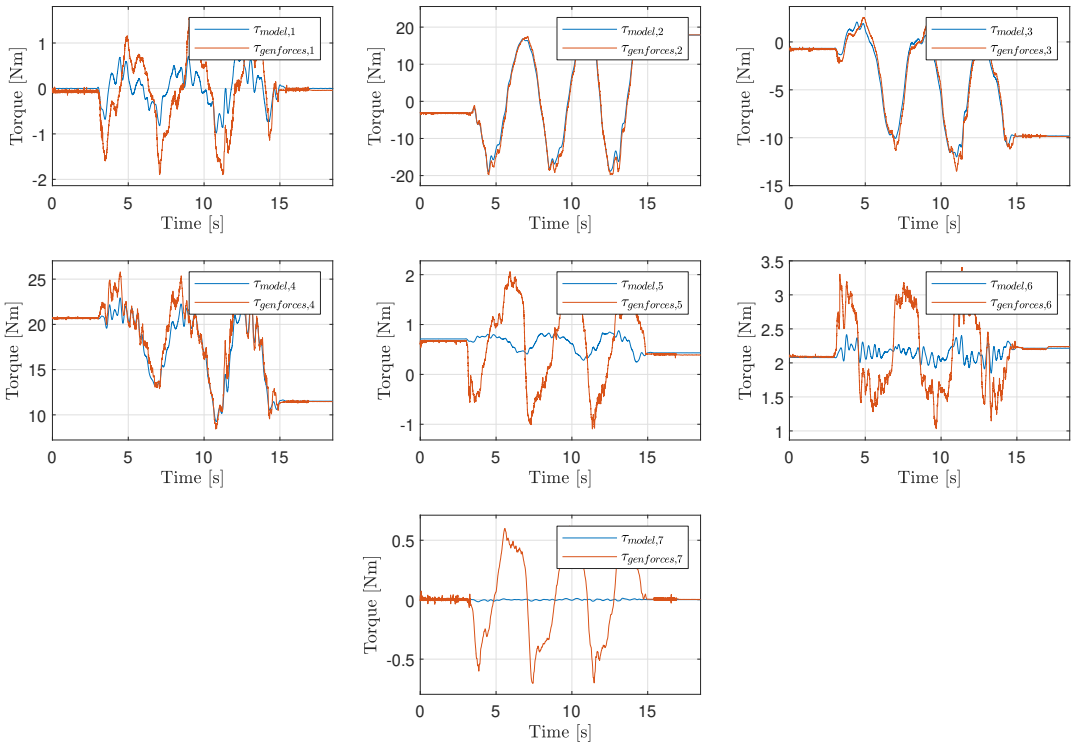


FIGURE A.9: Diagonal motion test with translational impedance control, with low inertia and damping parameters: dynamic model accuracy.

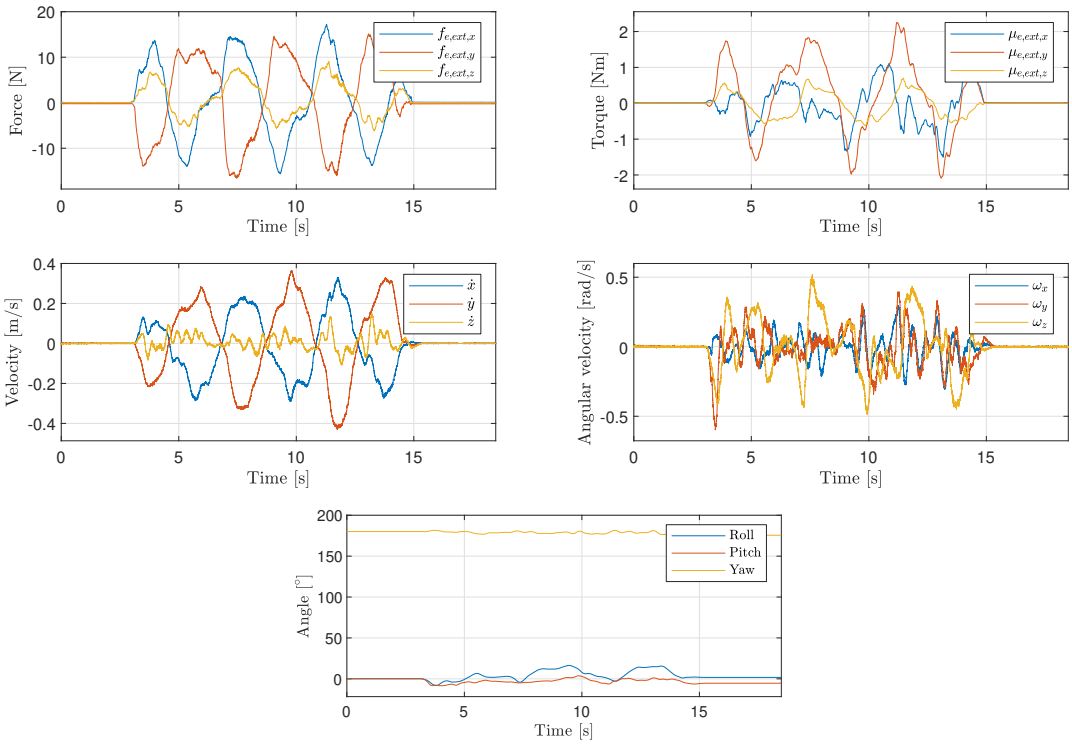


FIGURE A.10: Diagonal motion test with translational impedance control, with low inertia and damping parameters: end effector forces and torques, linear and angular velocities and roll pitch and yaw angles.

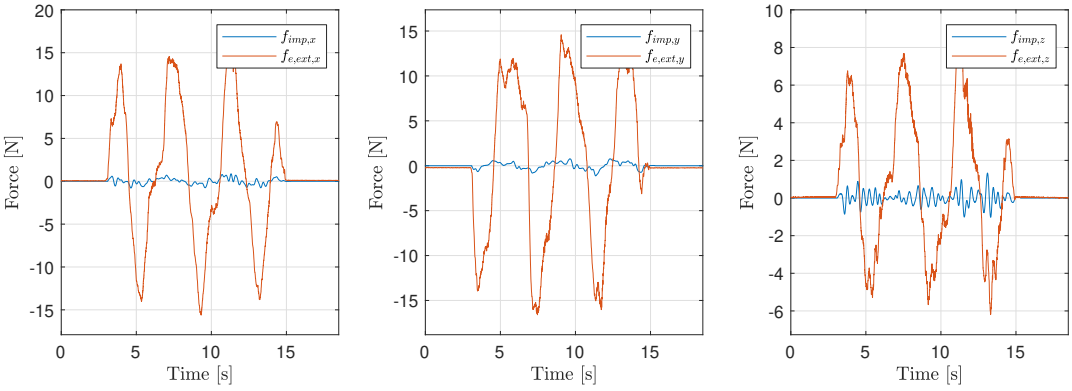


FIGURE A.11: Diagonal motion test with translational impedance control, with low inertia and damping parameters: comparison of expected and measured forces.

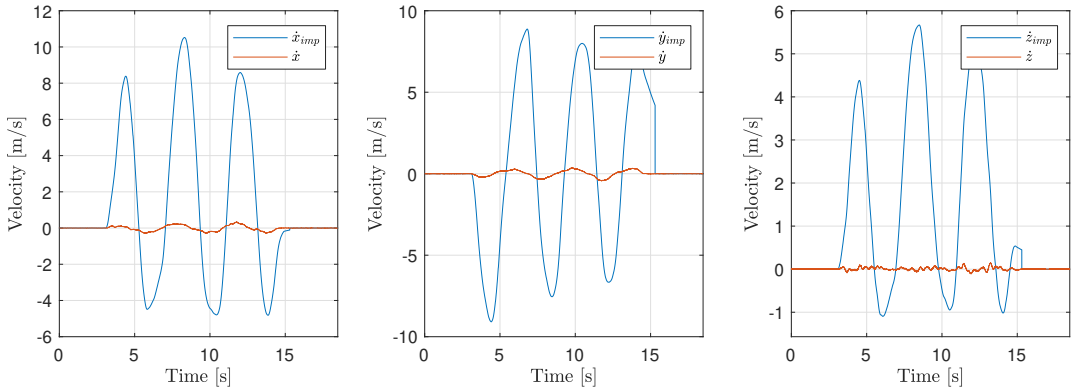


FIGURE A.12: Diagonal motion test with translational impedance control, with low inertia and damping parameters: comparison of expected and measured linear velocities.

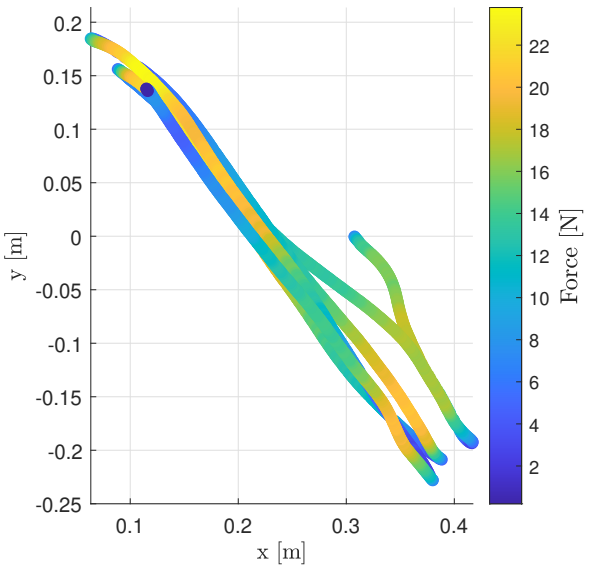


FIGURE A.13: Diagonal motion test with translational impedance control, with low inertia and damping parameters: norm of the end effector force vector along the trajectory in the x-y plane.

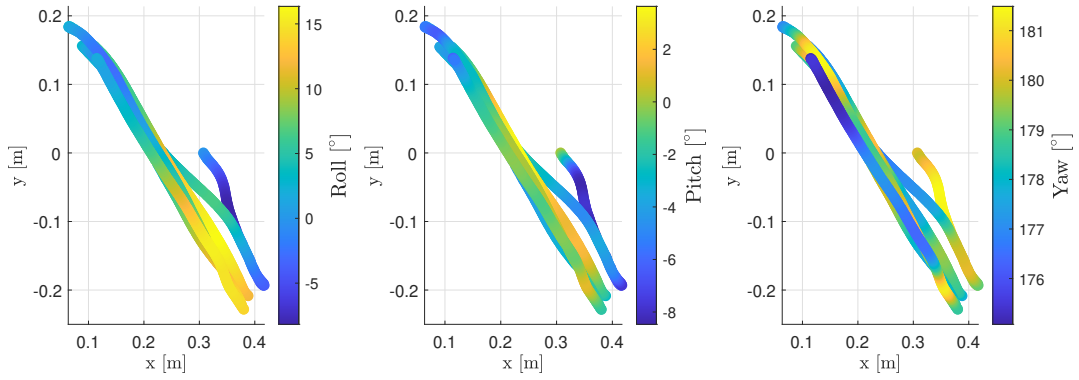


FIGURE A.14: Diagonal motion test with translational impedance control, with low inertia and damping parameters: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

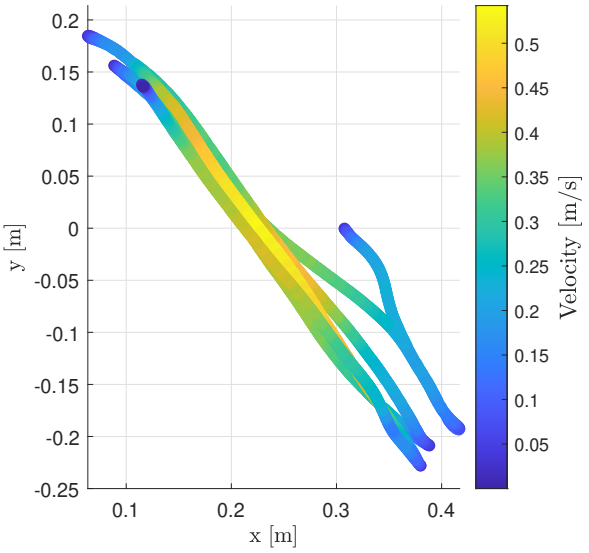


FIGURE A.15: Diagonal motion test with translational impedance control, with low inertia and damping parameters: norm of the end effector velocity vector along the trajectory in the *x*-*y* plane.

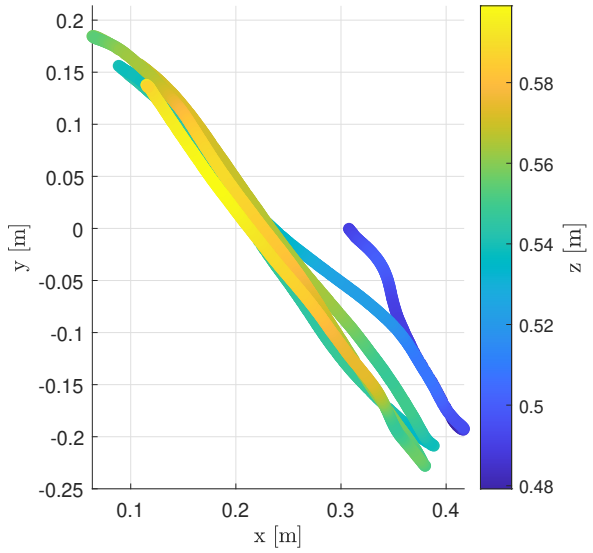


FIGURE A.16: Diagonal motion test with translational impedance control, with low inertia and damping parameters: height on the z axis along the trajectory in the x-y plane.

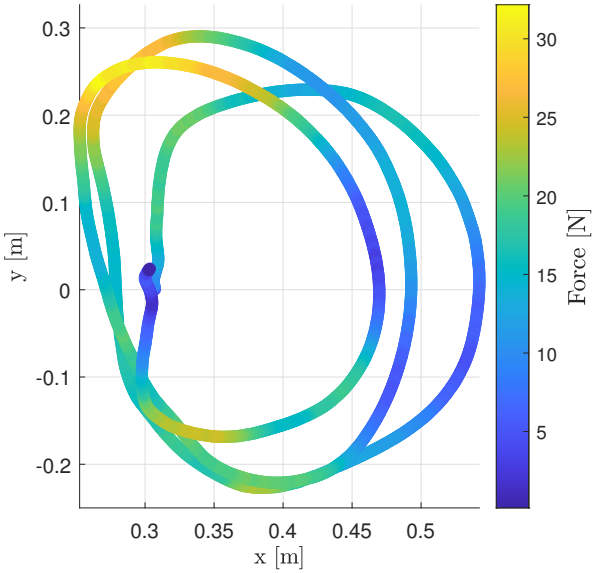


FIGURE A.17: Circular motion test on the x-y plane with translational impedance control, with low inertia and damping parameters: norm of the end effector force vector along the trajectory in the x-y plane.

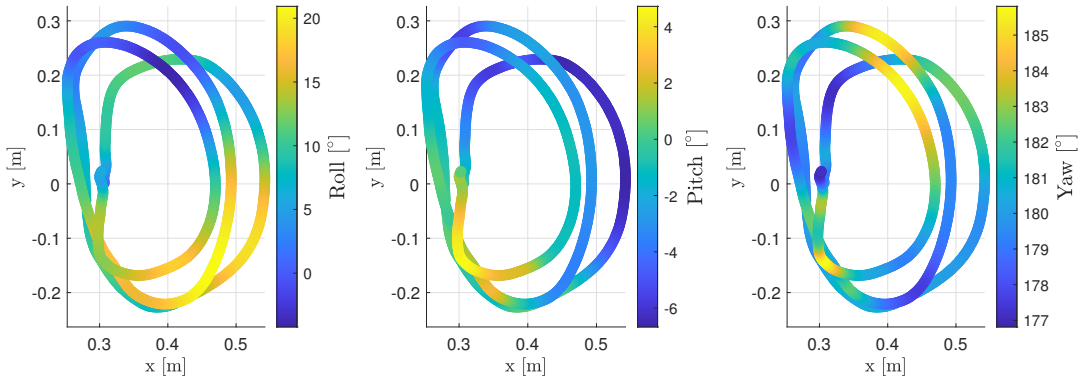


FIGURE A.18: Circular motion test on the *x*-*y* plane with translational impedance control, with low inertia and damping parameters: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

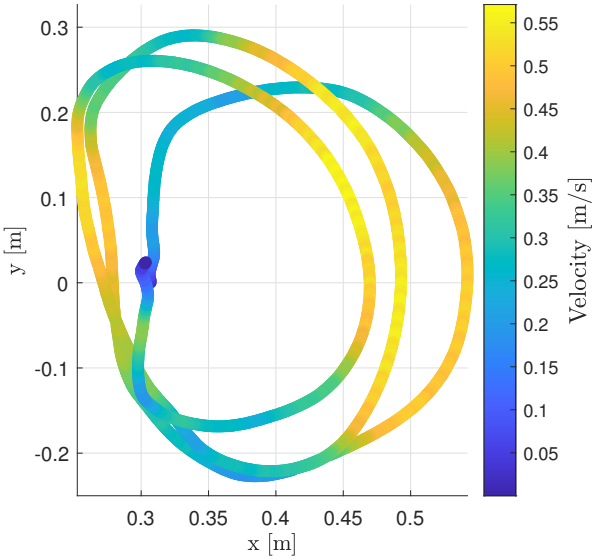


FIGURE A.19: Circular motion test on the *x*-*y* plane with translational impedance control, with low inertia and damping parameters: norm of the end effector velocity vector along the trajectory in the *x*-*y* plane.

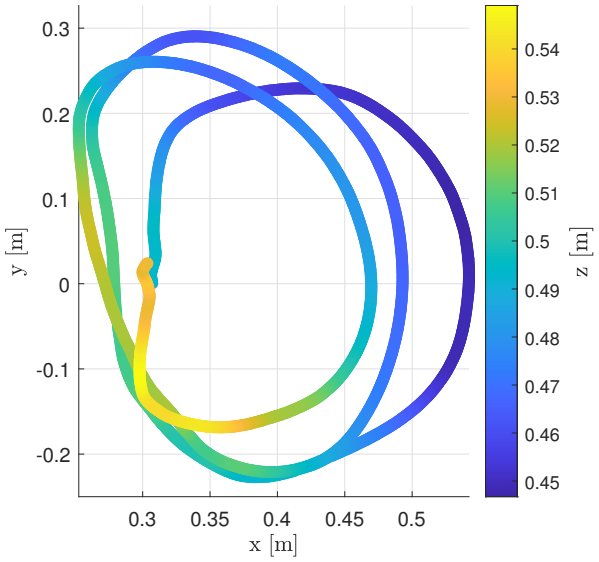


FIGURE A.20: Circular motion test on the x-y plane with translational impedance control, with low inertia and damping parameters: height on the z axis along the trajectory in the x-y plane.

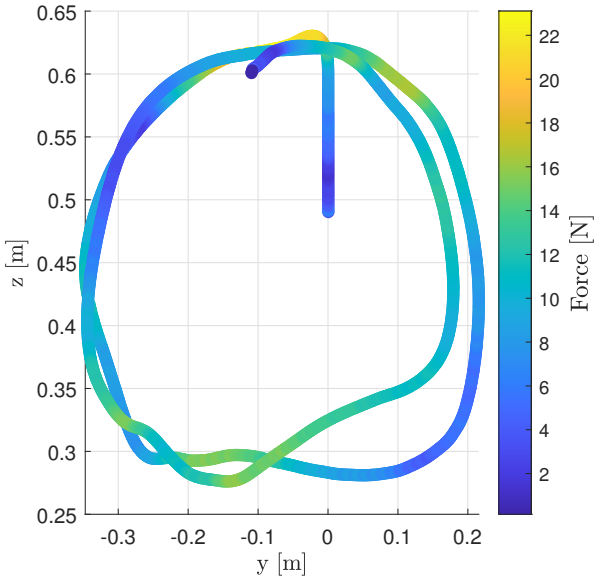


FIGURE A.21: Circular motion test on the y-z plane with translational impedance control, with low inertia and damping parameters: norm of the end effector force vector along the trajectory in the y-z plane.

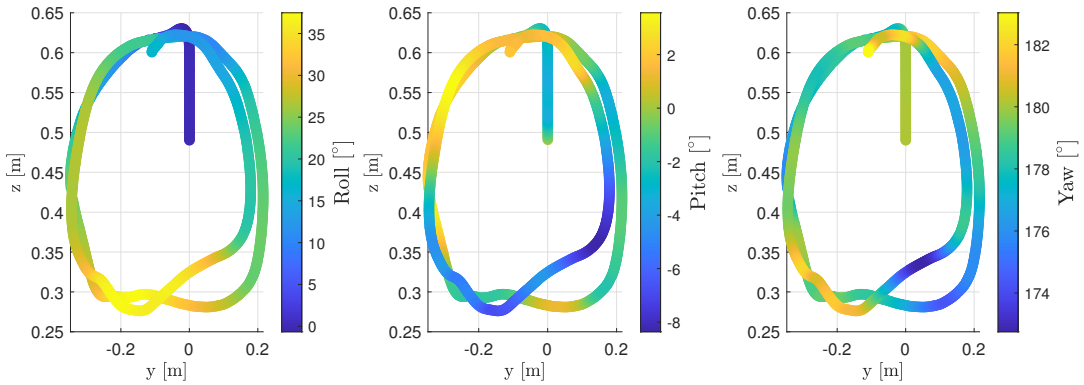


FIGURE A.22: Circular motion test on the y-z plane with translational impedance control, with low inertia and damping parameters: roll, pitch and yaw angles of the end effector along the trajectory in the y-z plane.

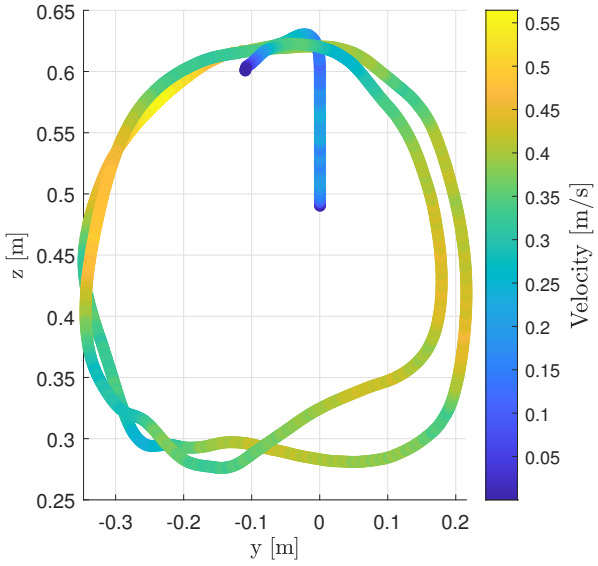


FIGURE A.23: Circular motion test on the y-z plane with translational impedance control, with low inertia and damping parameters: norm of the end effector velocity vector along the trajectory in the y-z plane.

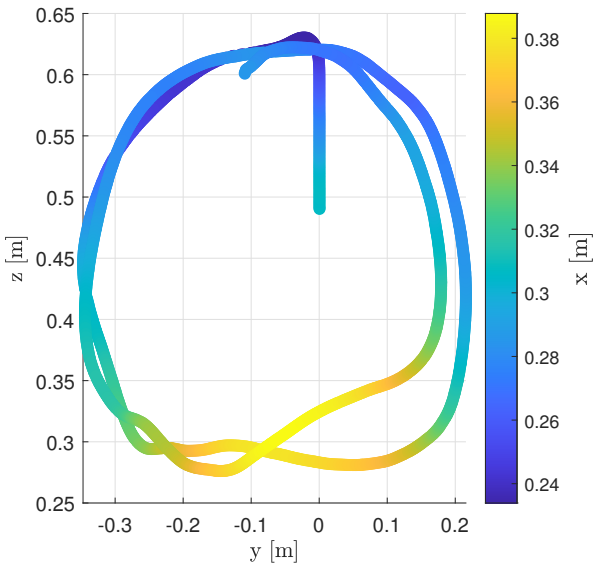


FIGURE A.24: Circular motion test on the y-z plane with translational impedance control, with low inertia and damping parameters: height on the x axis along the trajectory in the y-z plane.

Appendix B

Additional figures of the helping control tests

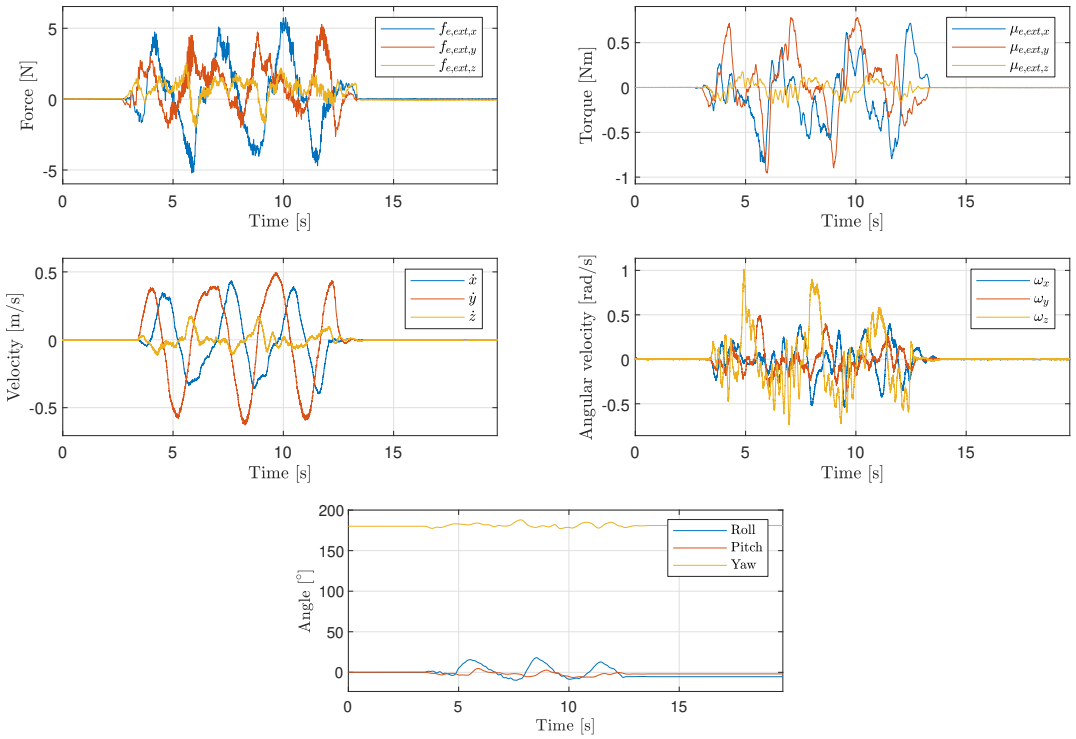


FIGURE B.1: Circular motion test on the x-y plane with helping control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

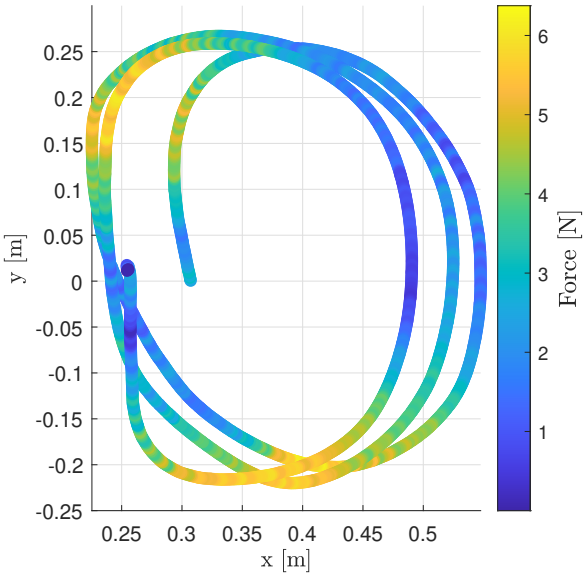


FIGURE B.2: Circular motion test on the *x*-*y* plane with helping control: norm of the end effector force vector along the trajectory in the *x*-*y* plane.

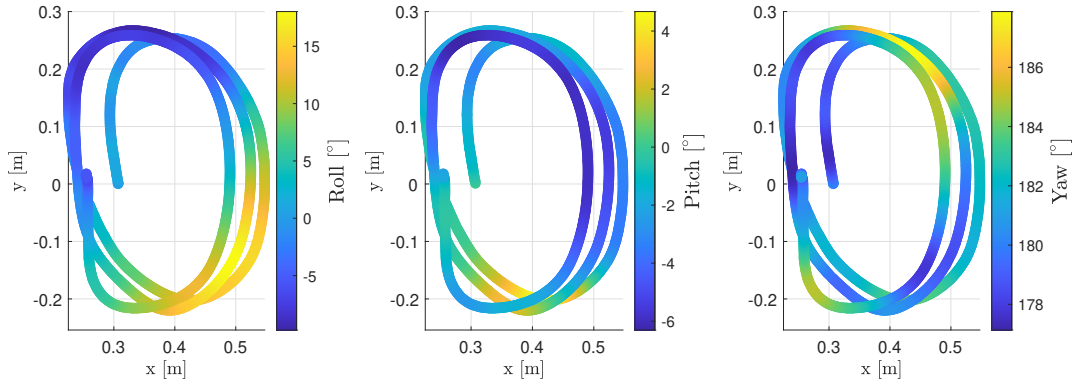


FIGURE B.3: Circular motion test on the *x*-*y* plane with helping control: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

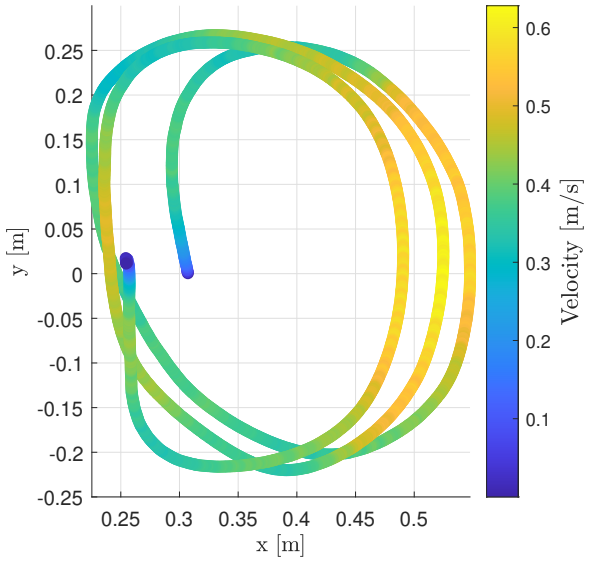


FIGURE B.4: Circular motion test on the x-y plane with helping control: norm of the end effector velocity vector along the trajectory in the x-y plane.

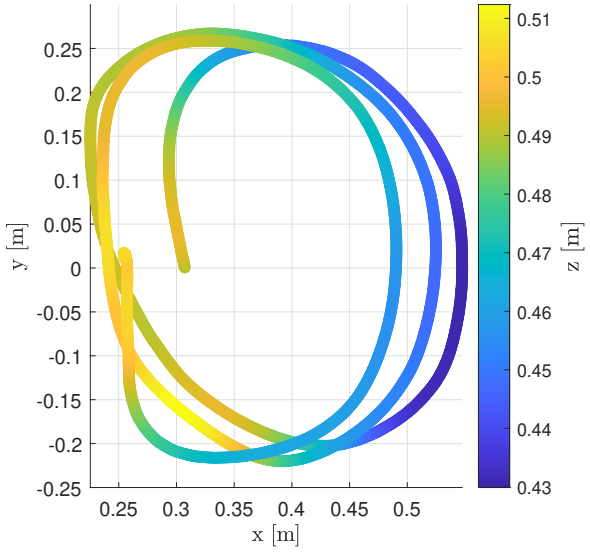


FIGURE B.5: Circular motion test on the x-y plane with helping control: height on the z axis along the trajectory in the x-y plane.

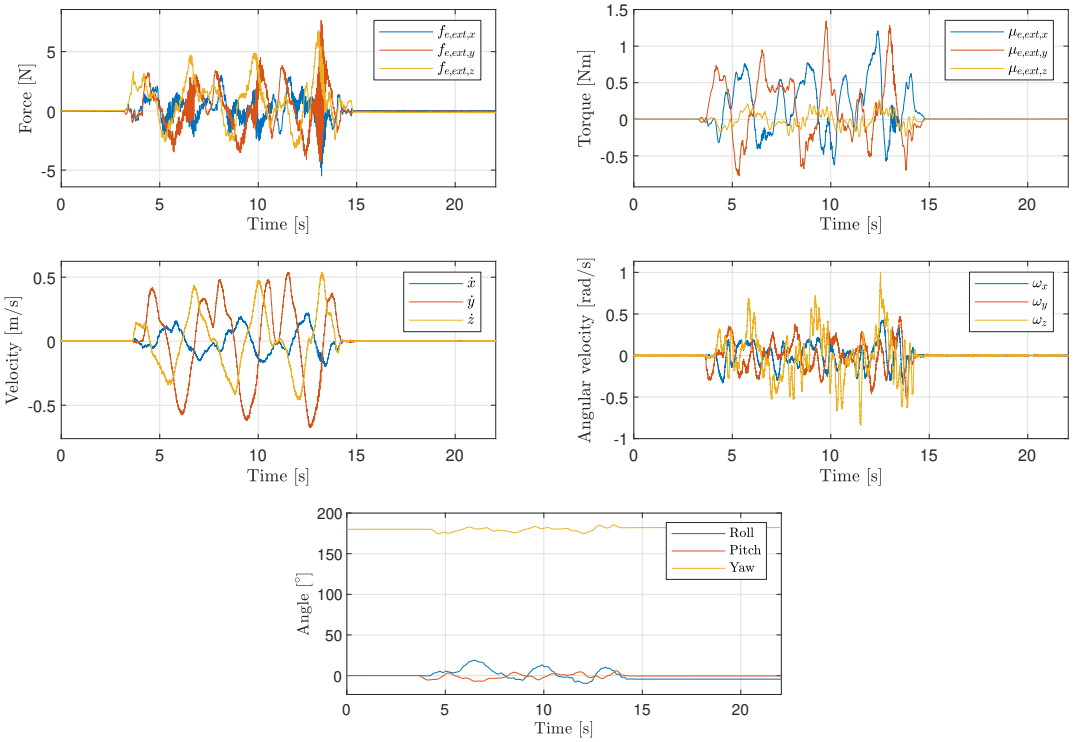


FIGURE B.6: Circular motion test on the *y*-*z* plane with helping control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

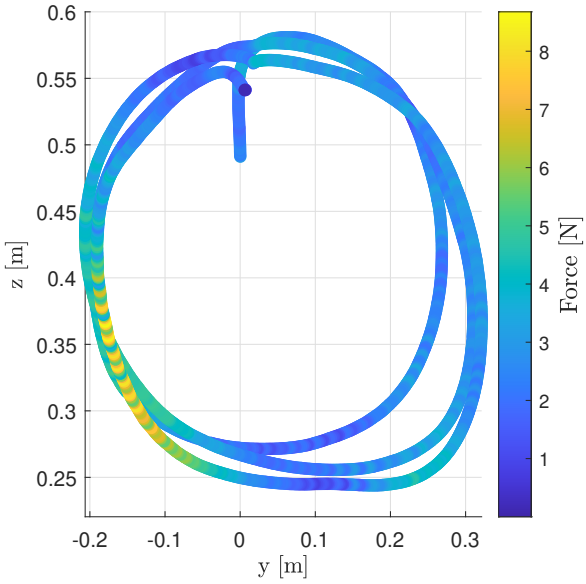


FIGURE B.7: Circular motion test on the *y*-*z* plane with helping control: norm of the end effector force vector along the trajectory in the *y*-*z* plane.

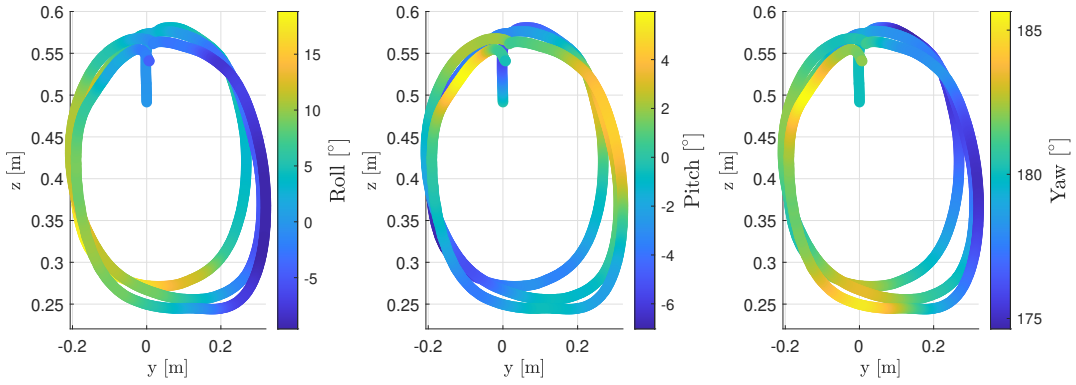


FIGURE B.8: Circular motion test on the y-z plane with helping control: roll, pitch and yaw angles of the end effector along the trajectory in the y-z plane.

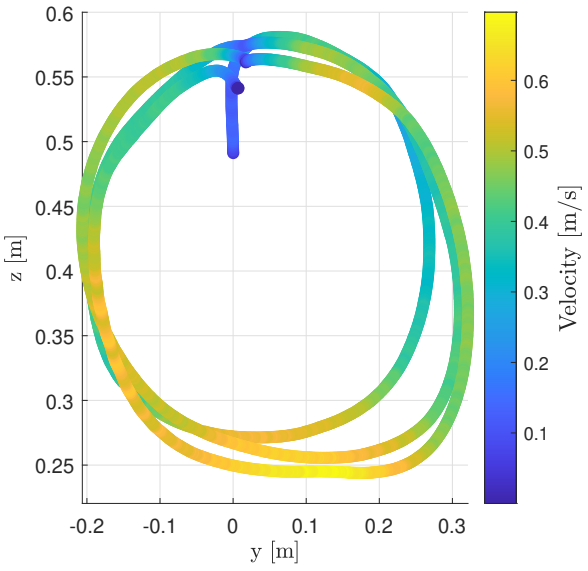


FIGURE B.9: Circular motion test on the y-z plane with helping control: norm of the end effector velocity vector along the trajectory in the y-z plane.

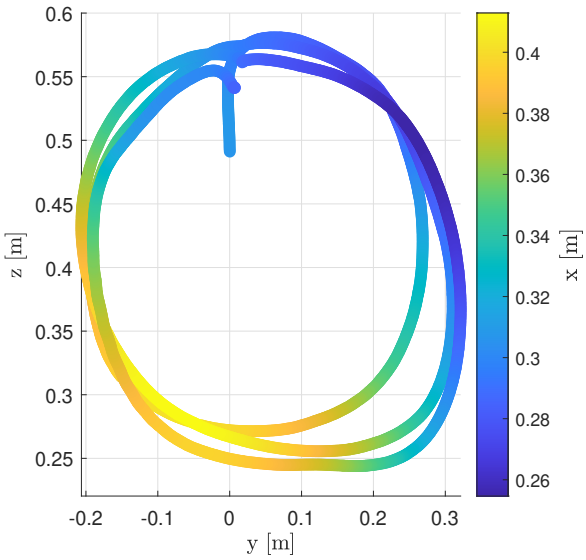


FIGURE B.10: Circular motion test on the *y*-*z* plane with helping control: height on the *x* axis along the trajectory in the *y*-*z* plane.

Appendix C

Additional figures of the admittance control tests

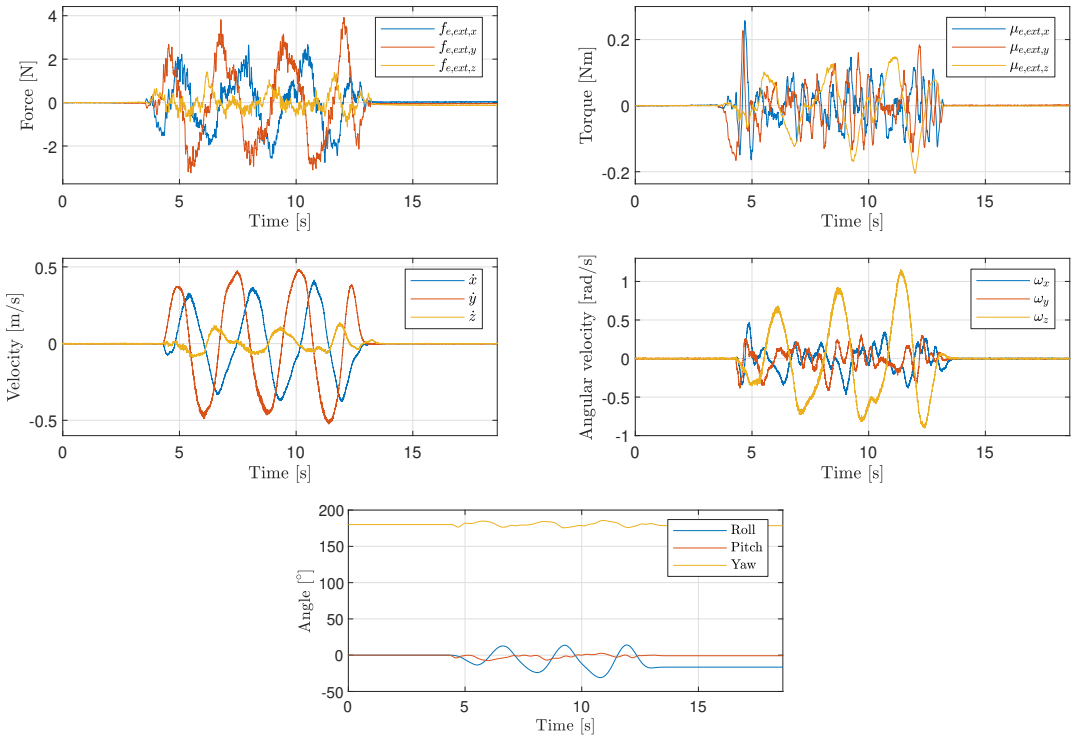


FIGURE C.1: Circular motion test on the x-y plane with admittance control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

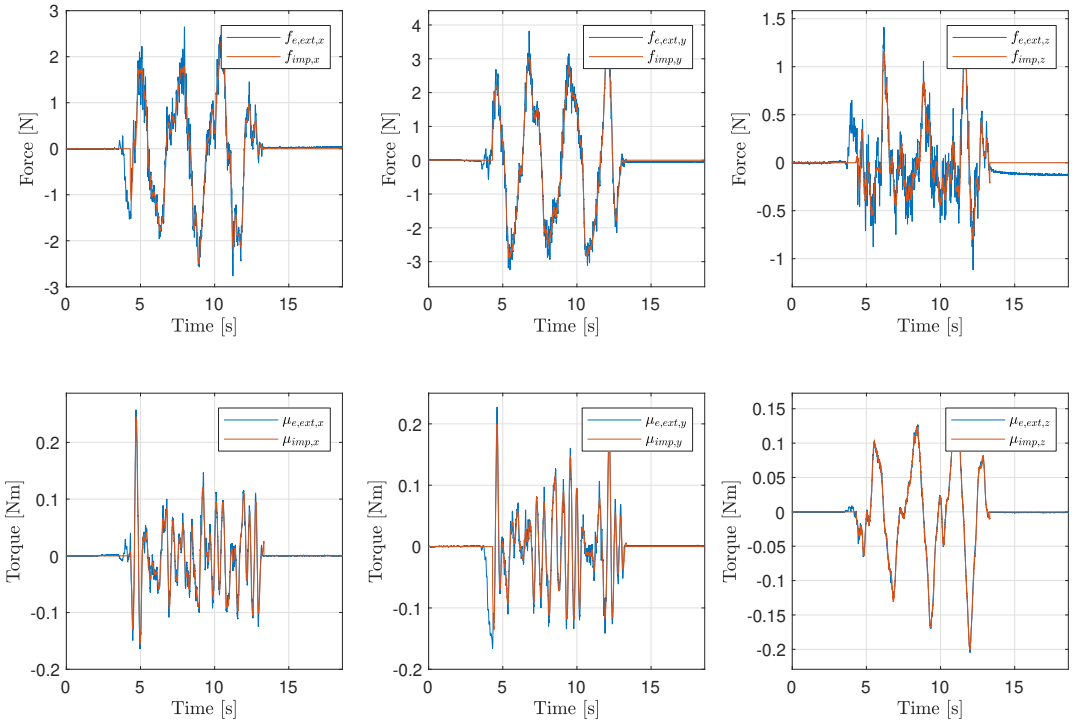


FIGURE C.2: Circular motion test on the *x*-*y* plane with admittance control: comparison of expected and measured forces and torques.

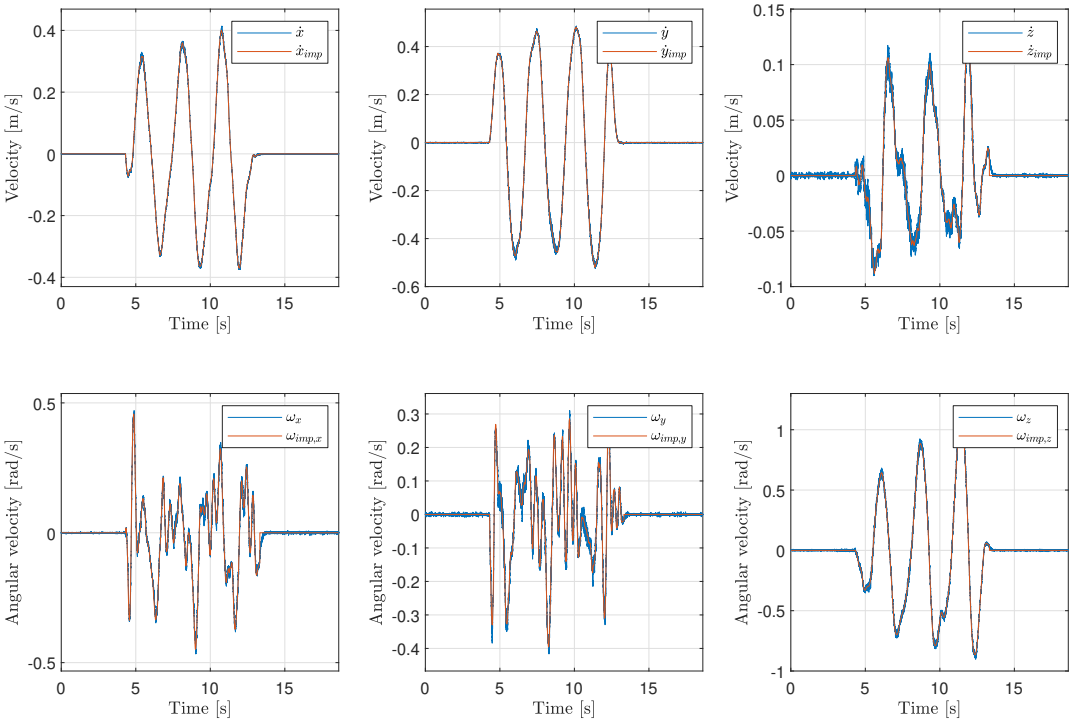


FIGURE C.3: Circular motion test on the *x*-*y* plane with admittance control: comparison of expected and measured linear and angular velocities.

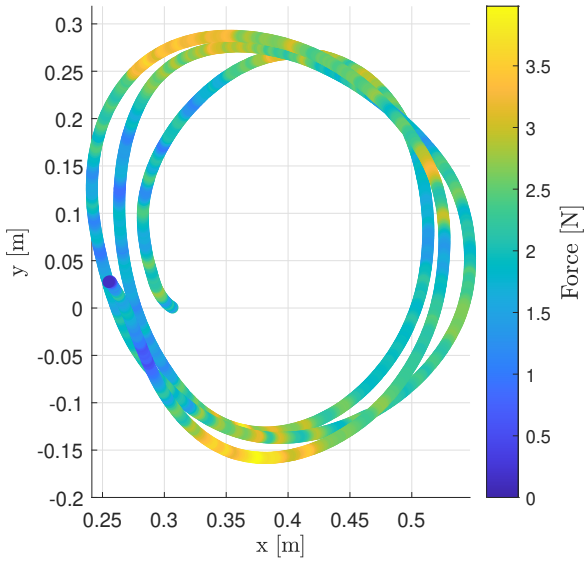


FIGURE C.4: Circular motion test on the *x*-*y* plane with admittance control: norm of the end effector force vector along the trajectory in the *x*-*y* plane.

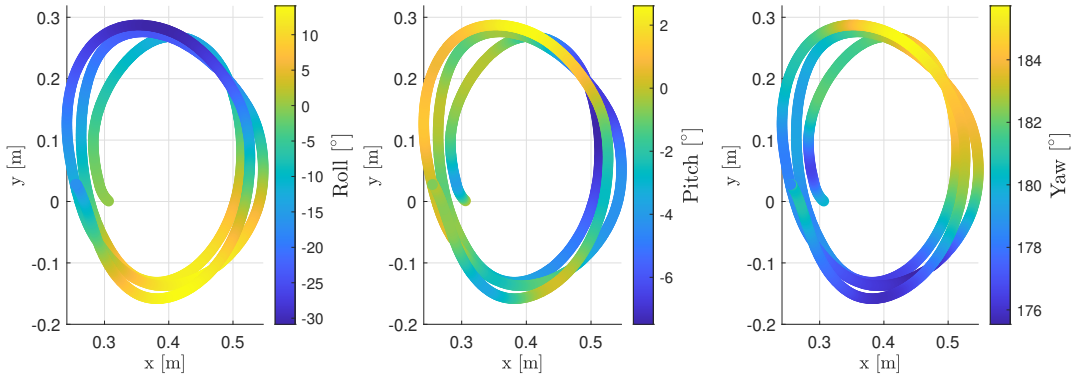


FIGURE C.5: Circular motion test on the *x*-*y* plane with admittance control: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

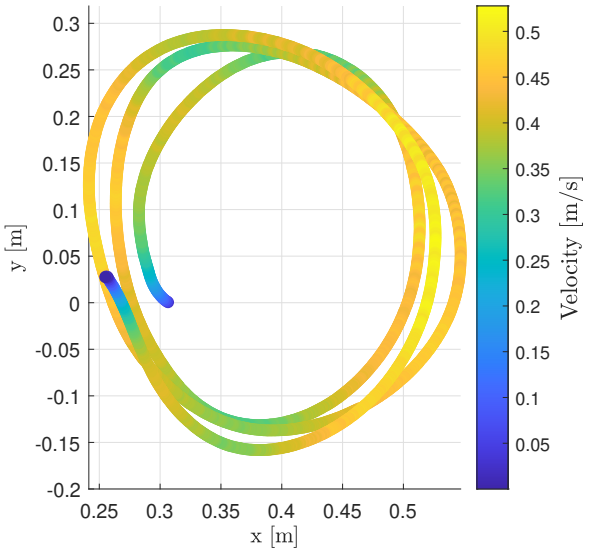


FIGURE C.6: Circular motion test on the *x*-*y* plane with admittance control: norm of the end effector velocity vector along the trajectory in the *x*-*y* plane.

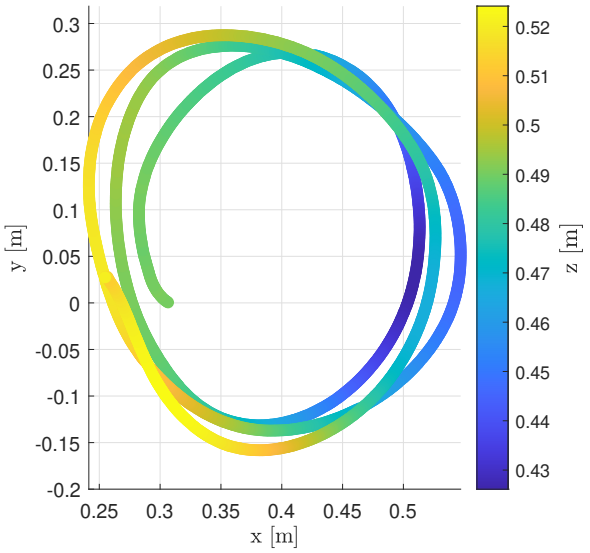


FIGURE C.7: Circular motion test on the *x*-*y* plane with admittance control: height on the *z* axis along the trajectory in the *x*-*y* plane.

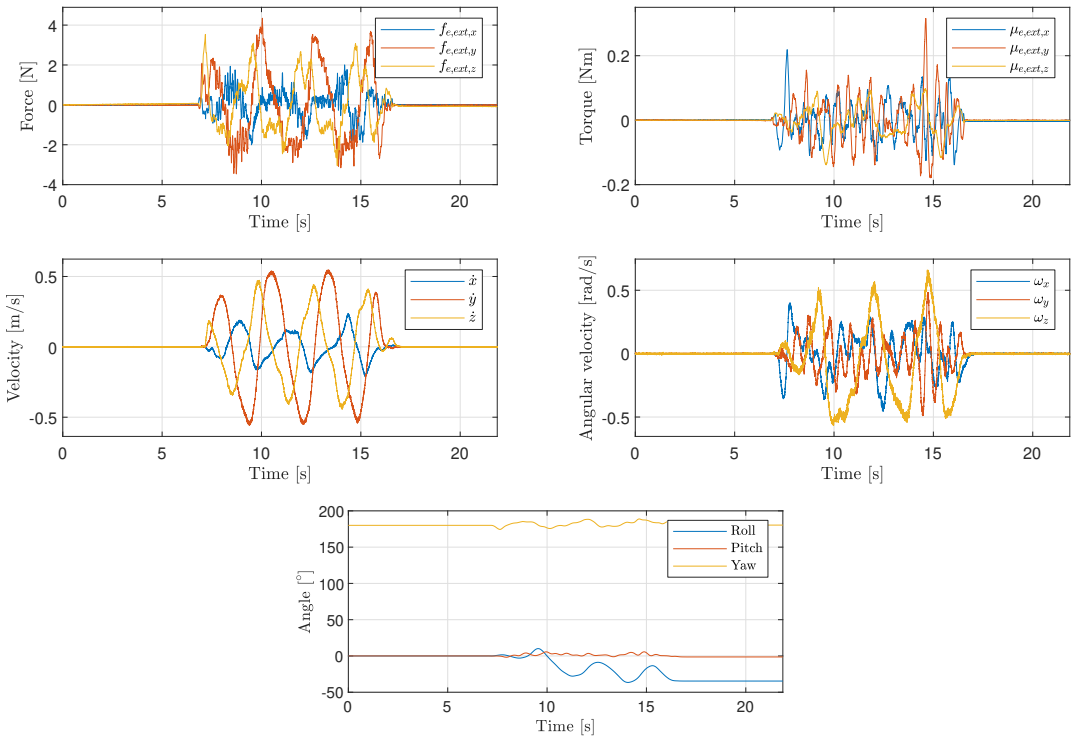


FIGURE C.8: Circular motion test on the *y*-*z* plane with admittance control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

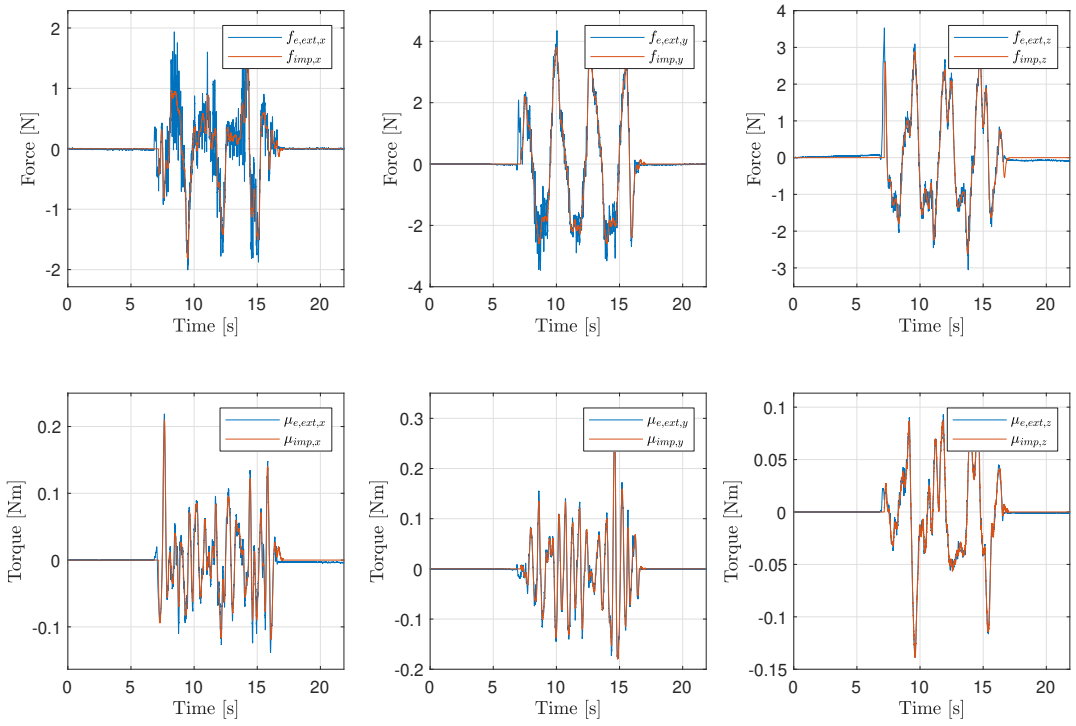


FIGURE C.9: Circular motion test on the *y*-*z* plane with admittance control: comparison of expected and measured forces and torques.

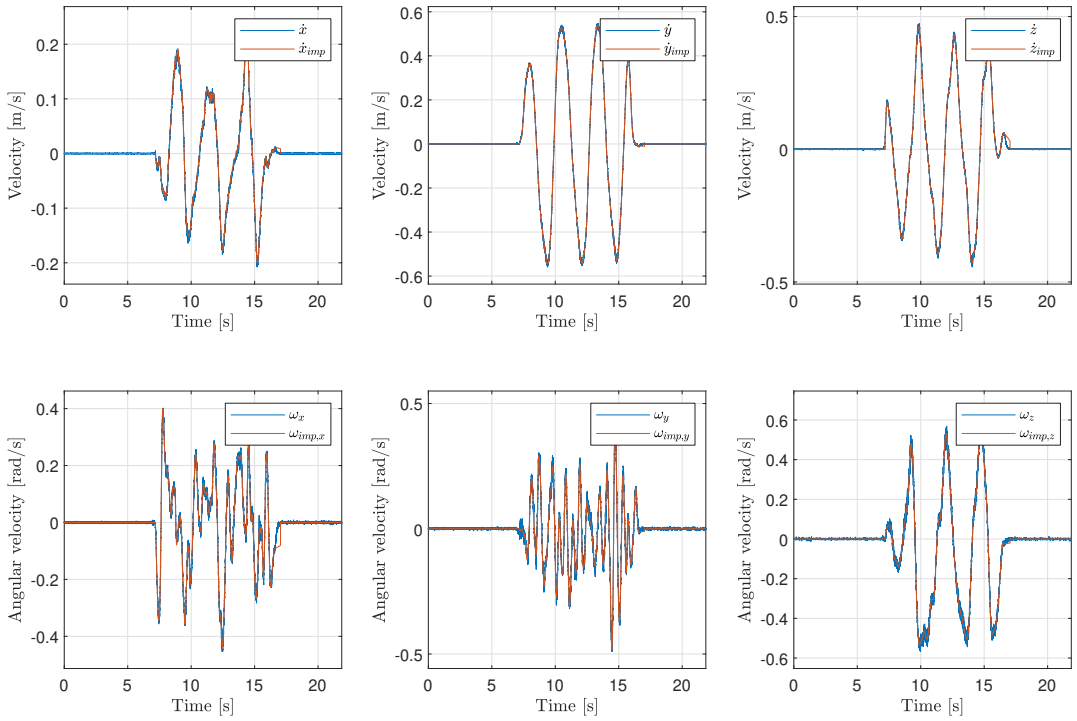


FIGURE C.10: Circular motion test on the *y*-*z* plane with admittance control: comparison of expected and measured linear and angular velocities.

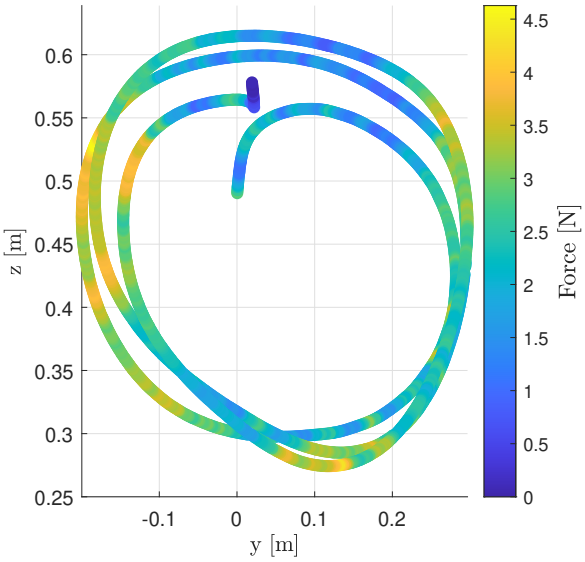


FIGURE C.11: Circular motion test on the *y*-*z* plane with admittance control: norm of the end effector force vector along the trajectory in the *y*-*z* plane.

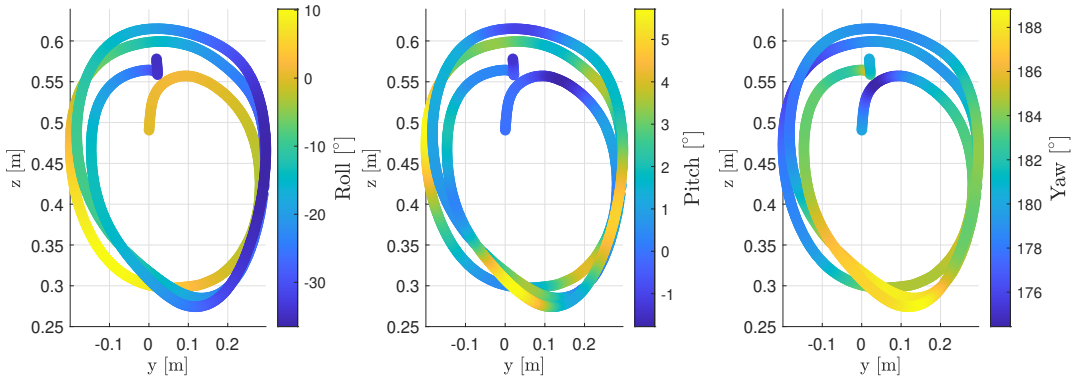


FIGURE C.12: Circular motion test on the *y*-*z* plane with admittance control: roll, pitch and yaw angles of the end effector along the trajectory in the *y*-*z* plane.

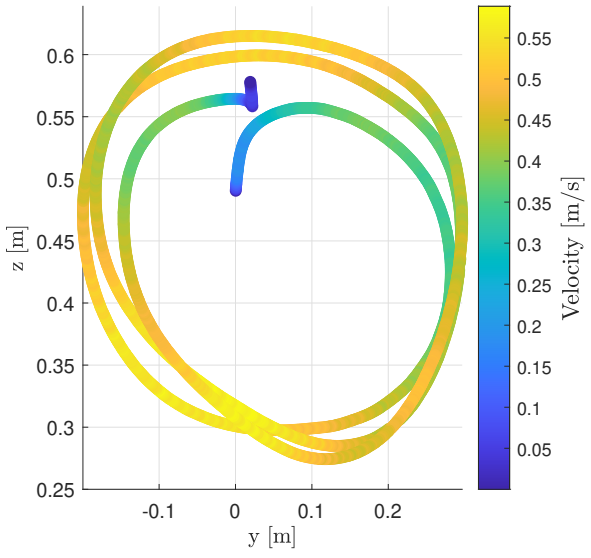


FIGURE C.13: Circular motion test on the *y*-*z* plane with admittance control: norm of the end effector velocity vector along the trajectory in the *y*-*z* plane.

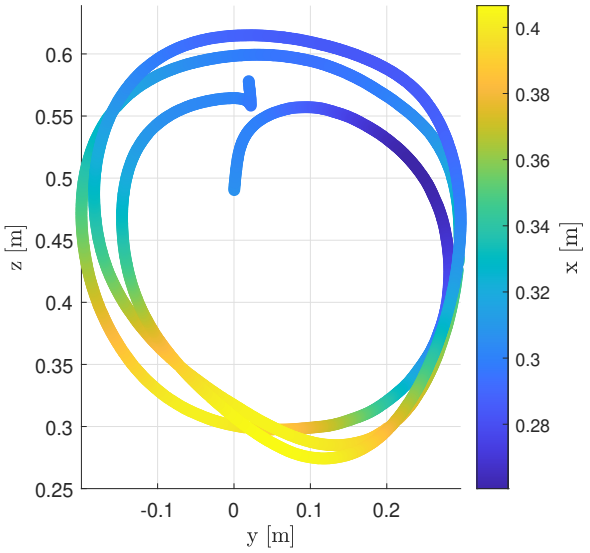


FIGURE C.14: Circular motion test on the y-z plane with admittance control: height on the z axis along the trajectory in the y-z plane.

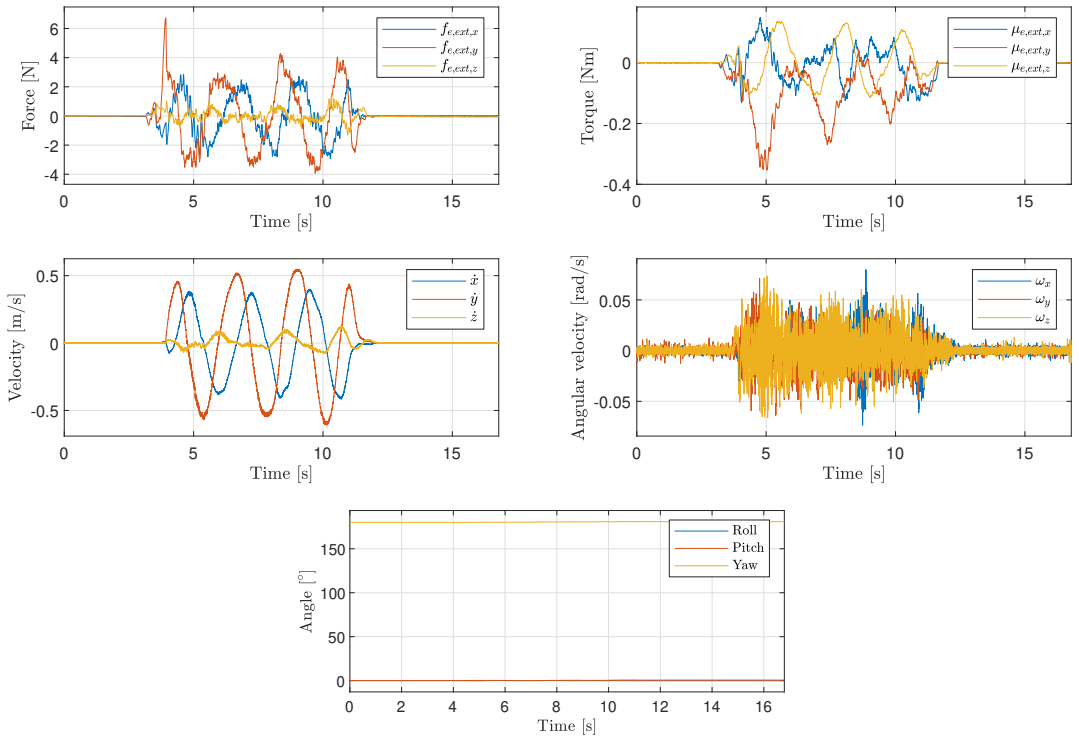


FIGURE C.15: Circular motion test on the x-y plane with translational admittance control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

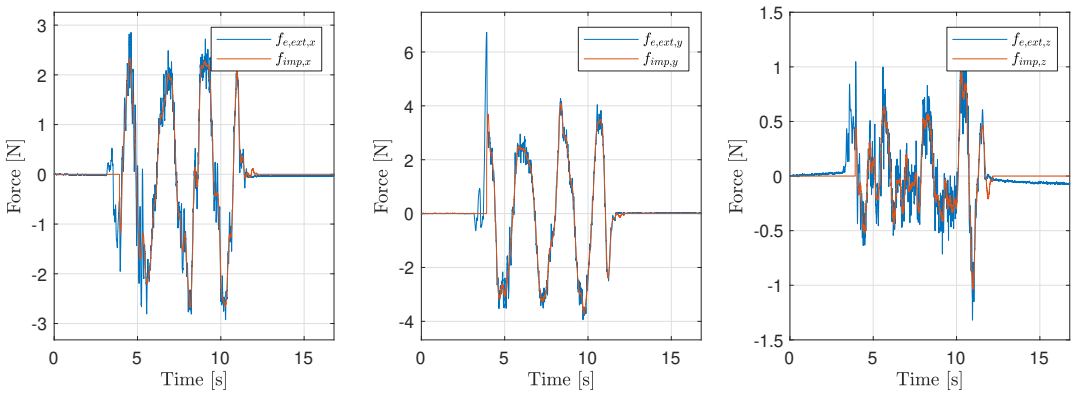


FIGURE C.16: Circular motion test on the *x*-*y* plane with translational admittance control: comparison of expected and measured forces.

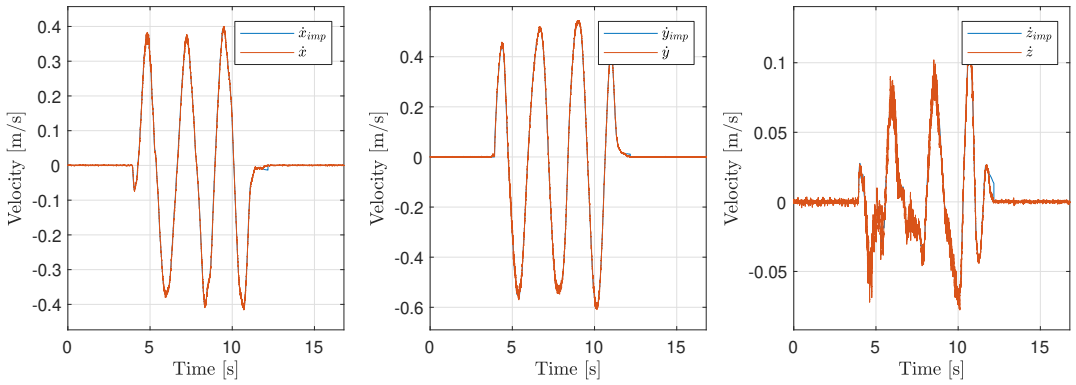


FIGURE C.17: Circular motion test on the *x*-*y* plane with translational admittance control: comparison of expected and measured linear velocities.

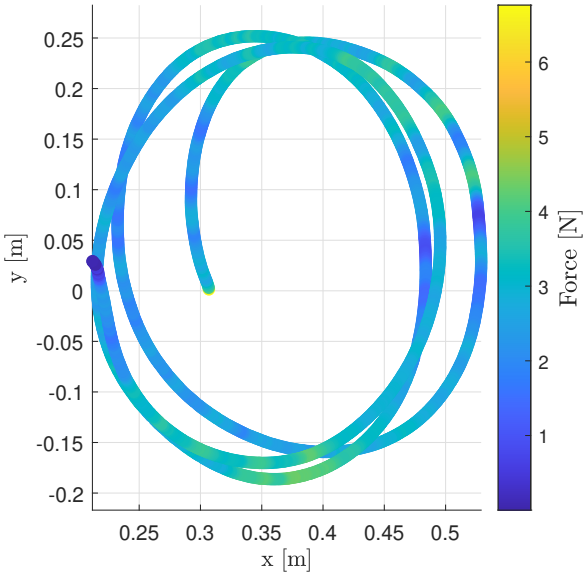


FIGURE C.18: Circular motion test on the *x*-*y* plane with translational admittance control: norm of the end effector force vector along the trajectory in the *x*-*y* plane.

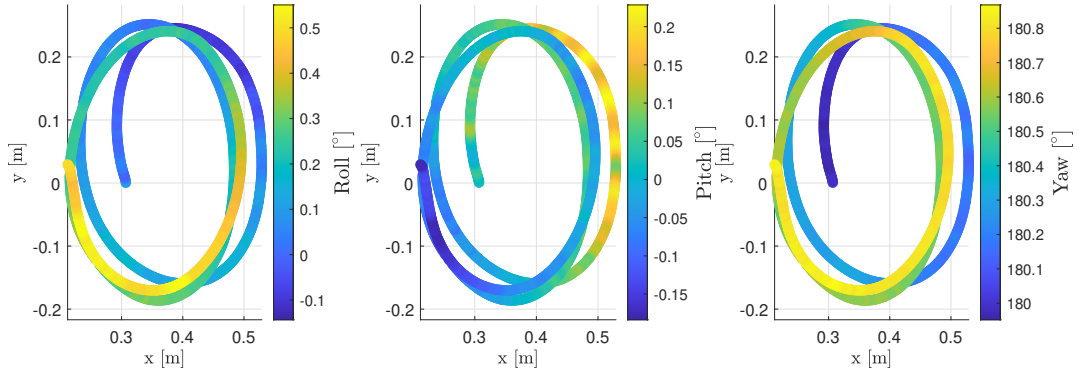


FIGURE C.19: Circular motion test on the *x*-*y* plane with translational admittance control: roll, pitch and yaw angles of the end effector along the trajectory in the *x*-*y* plane.

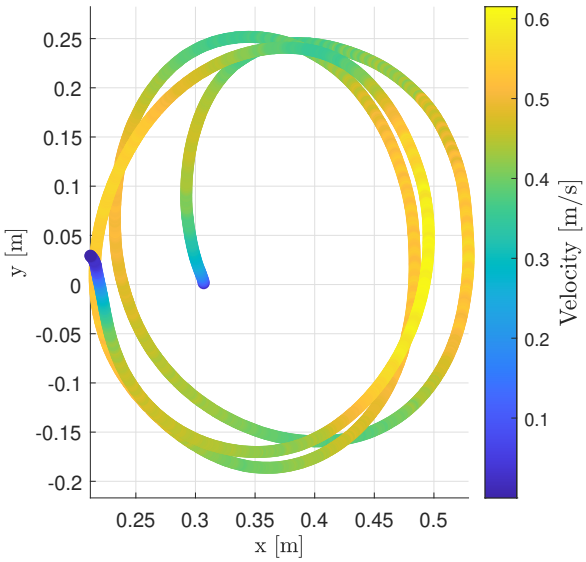


FIGURE C.20: Circular motion test on the *x*-*y* plane with translational admittance control: norm of the end effector velocity vector along the trajectory in the *x*-*y* plane.

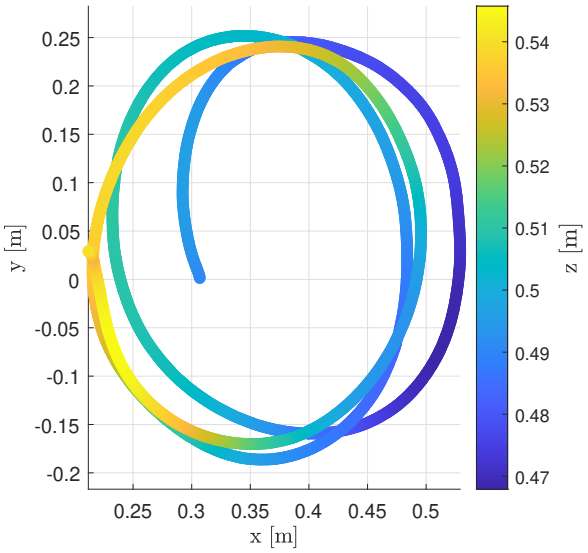


FIGURE C.21: Circular motion test on the *x*-*y* plane with translational admittance control: height on the *z* axis along the trajectory in the *x*-*y* plane.

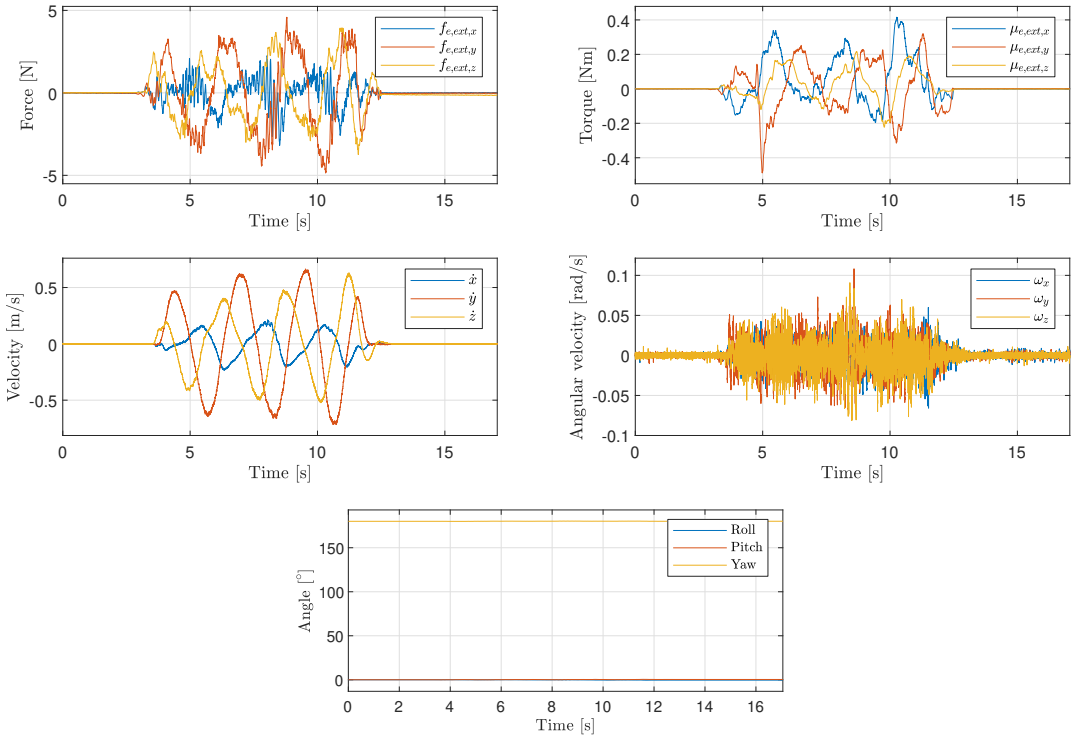


FIGURE C.22: Circular motion test on the *y*-*z* plane with translational admittance control: forces and torques, linear and angular velocities and roll pitch and yaw angles.

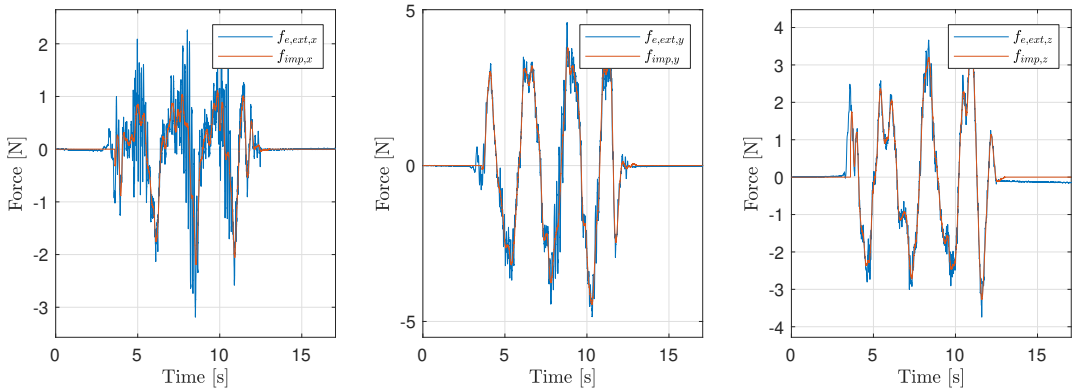


FIGURE C.23: Circular motion test on the y-z plane with translational admittance control: comparison of expected and measured forces.

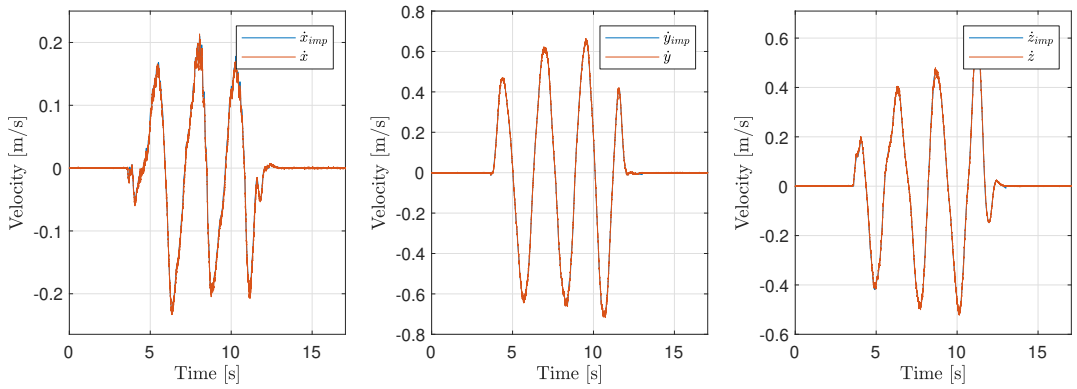


FIGURE C.24: Circular motion test on the y-z plane with translational admittance control: comparison of expected and measured linear velocities.

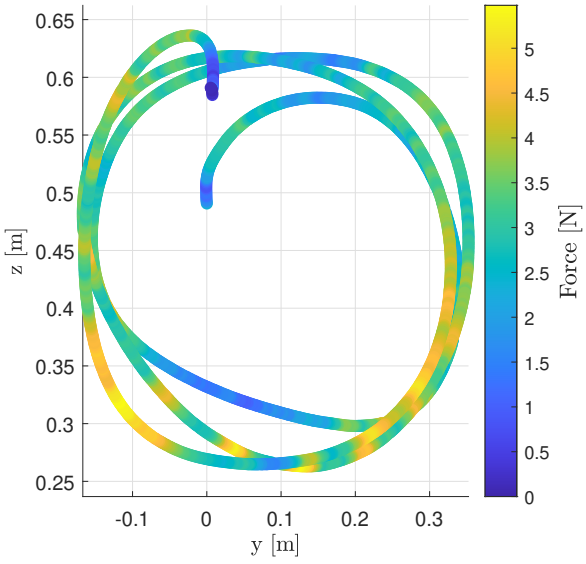


FIGURE C.25: Circular motion test on the y-z plane with translational admittance control: norm of the end effector force vector along the trajectory in the y-z plane.

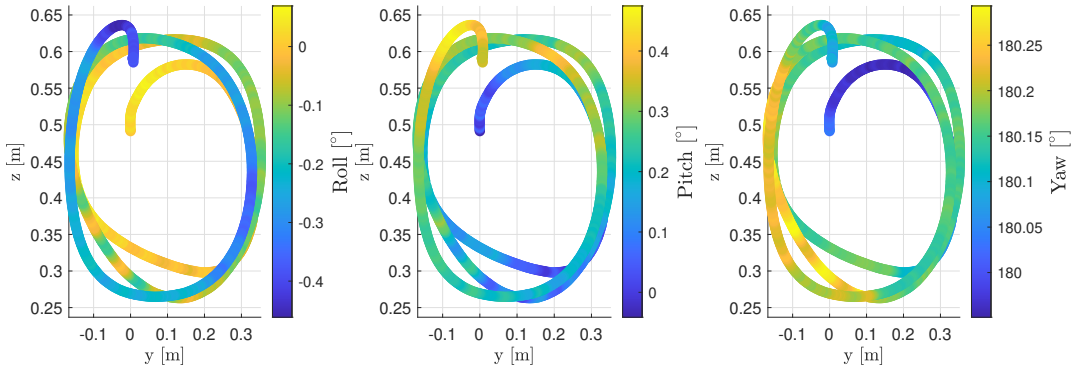


FIGURE C.26: Circular motion test on the y-z plane with translational admittance control: roll, pitch and yaw angles of the end effector along the trajectory in the y-z plane.

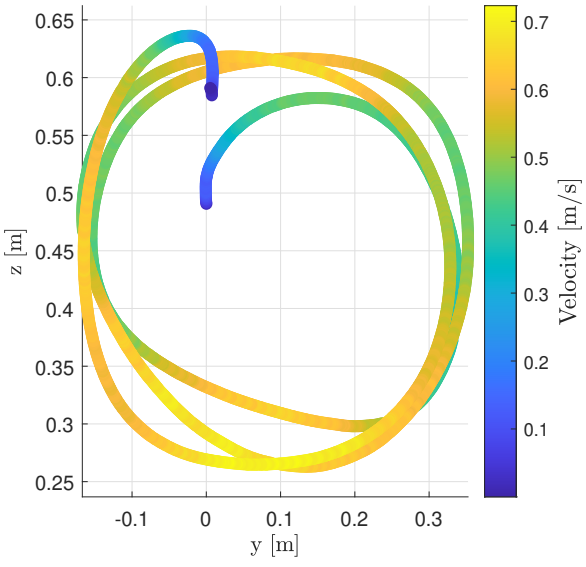


FIGURE C.27: Circular motion test on the y-z plane with translational admittance control: norm of the end effector velocity vector along the trajectory in the y-z plane.

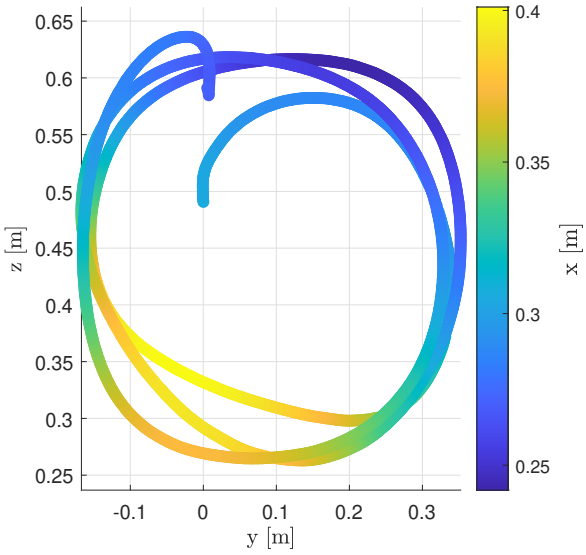


FIGURE C.28: Circular motion test on the y-z plane with translational admittance control: height on the z axis along the trajectory in the y-z plane.

Bibliography

[1] Alin Albu-Schäffer et al. "The DLR lightweight robot: design and control concepts for robots in human environments". In: *Industrial Robot: an international journal* 34.5 (2007), pp. 376–385.

[2] James Arnold and Hyunglae Lee. "Variable impedance control for phri: Impact on stability, agility, and human effort in controlling a wearable ankle robot". In: *IEEE Robotics and Automation Letters* 6.2 (2021), pp. 2429–2436.

[3] Giulia Avallone et al. "A COBOT-IMU hand-guiding system with online collision avoidance in null space". In: *Symposium on Robot Design, Dynamics and Control*. Springer. 2022, pp. 151–159.

[4] Jangho Bae et al. "Variable admittance control with virtual stiffness guidance for human–robot collaboration". In: *IEEE Access* 8 (2020), pp. 117335–117346.

[5] Lisanne Bainbridge. "Ironies of automation". In: *Analysis, design and evaluation of man–machine systems*. Elsevier, 1983, pp. 129–135.

[6] Stephen Bevan. "Economic impact of musculoskeletal disorders (MSDs) on work in Europe". In: *Best Practice & Research Clinical Rheumatology* 29.3 (2015), pp. 356–373.

[7] KS Bibby et al. "Man's role in control systems". In: *IFAC Proceedings Volumes* 8.1 (1975), pp. 664–683.

[8] Paolo Biscari et al. *Meccanica razionale*. Springer Nature, 2022.

[9] Sergey Bochkanov. *ALGLIB*. (2022). URL: https://www.alglib.net/.

[10] Etienne Burdet et al. "The central nervous system stabilizes unstable dynamics by learning optimal impedance". In: *Nature* 414.6862 (2001), pp. 446–449.

[11] Tan Fung Chan and Rajiv V Dubey. "A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators". In: *IEEE transactions on Robotics and Automation* 11.2 (1995), pp. 286–292.

[12] Su Il Choi and Byung Kook Kim. "Obstacle avoidance control for redundant manipulators using collidability measure". In: *Robotica* 18.2 (2000), pp. 143–151.

[13] Jan De Kok et al. *Work-related musculoskeletal disorders: prevalence, costs and demographics in the EU*. Tech. rep. European Agency for Safety and Health at Work, 2019.

[14] Michiel P De Looze et al. "Exoskeletons for industrial application and their potential effects on physical work load". In: *Ergonomics* 59.5 (2016), pp. 671–681.

[15] Andrea Del Prete et al. "Prioritized motion–force control of constrained fully-actuated robots:"Task Space Inverse Dynamics"". In: *Robotics and Autonomous Systems* 63 (2015), pp. 150–157.

[16] Fotios Dimeas and Nikos Aspragathos. "Reinforcement learning of variable admittance control for human-robot co-manipulation". In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2015, pp. 1011–1016.

[17] Rajiv V Dubey, James A Euler, and Scott M Babcock. "An efficient gradient projection optimization scheme for a seven-degree-of-freedom redundant robot with spherical wrist". In: *Proceedings. 1988 IEEE International Conference on Robotics and Automation*. IEEE. 1988, pp. 28–36.

[18] James A Euler et al. "A comparison of two real-time control schemes for redundant manipulators with bounded joint velocities". In: *Proceedings, 1989 International Conference on Robotics and Automation*. IEEE. 1989, pp. 106–112.

[19] Åsa Fast-Berglund et al. "Evaluating cobots for final assembly". In: *Procedia CIRP* 44 (2016), pp. 175–180.

[20] Bernard Faverjon and Pierre Tournassoud. "A local based approach for path planning of manipulators with a high number of degrees of freedom". In: *Proceedings. 1987 IEEE international conference on robotics and automation*. Vol. 4. IEEE. 1987, pp. 1152–1159.

[21] Federica Ferraguti et al. "A variable admittance control strategy for stable physical human–robot interaction". In: *The International Journal of Robotics Research* 38.6 (2019), pp. 747–765.

[22] Fanny Ficuciello, Luigi Villani, and Bruno Siciliano. "Variable impedance control of redundant manipulators for intuitive human–robot physical interaction". In: *IEEE Transactions on Robotics* 31.4 (2015), pp. 850–863.

[23] Fabrizio Flacco, Alessandro De Luca, and Oussama Khatib. "Control of redundant robots under hard joint constraints: Saturation in the null space". In: *IEEE Transactions on Robotics* 31.3 (2015), pp. 637–654.

[24] *fmincon function*. URL: https://www.mathworks.com/help/optim/ug/fmincon.html.

[25] *Franka Control Interface Documentation*. (2022). URL: https://frankaemika.github.io/docs/.

[26] Juan M Gandarias et al. "Enhancing flexibility and adaptability in conjoined human-robot industrial tasks with a minimalist physical interface". In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE. 2022, pp. 8061–8067.

[27] Claudio Gaz, Fabrizio Flacco, and Alessandro De Luca. "Identifying the dynamic model used by the KUKA LWR: A reverse engineering approach". In: *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE. 2014, pp. 1386–1392.

[28] Claudio Gaz et al. "Dynamic identification of the franka emika panda robot with retrieval of feasible parameters using penalty-based optimization". In: *IEEE Robotics and Automation Letters* 4.4 (2019), pp. 4147–4154.

[29] Dongsheng Guo and Yunong Zhang. "Acceleration-level inequality-based MAN scheme for obstacle avoidance of redundant robot manipulators". In: *IEEE Transactions on Industrial Electronics* 61.12 (2014), pp. 6903–6914.

[30] Yida Guo, Yang Tian, and Haoping Wang. "Learning-based Adaptive Optimal Impedance Control to Enhance Physical Human-robot Interaction Performance". In: *International Journal of Control, Automation and Systems* 20.9 (2022), pp. 3053–3062.

[31] Blake Hannaford and Jee-Hwan Ryu. "Time-domain passivity control of haptic interfaces". In: *IEEE transactions on Robotics and Automation* 18.1 (2002), pp. 1–10.

[32] Wei He et al. "Admittance-based controller design for physical human–robot interaction in the constrained task space". In: *IEEE Transactions on Automation Science and Engineering* 17.4 (2020), pp. 1937–1949.

[33] Neville Hogan. "Impedance control: An approach to manipulation: Part II: Implementation". In: (1985).

[34] Yingxin Huo et al. "Intention-driven variable impedance control for physical human-robot interaction". In: *2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE. 2021, pp. 1220–1225.

[35] Oussama Kanoun, Florent Lamiraux, and Pierre-Brice Wieber. "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task". In: *IEEE Transactions on Robotics* 27.4 (2011), pp. 785–792.

[36] Zsolt Kemeny. "Redundancy resolution in robots using parameterization through null space". In: *IEEE Trans. on Ind. Electronics* 50.4 (2003), pp. 777–783.

[37] Wisama Khalil and Etienne Dombre. *Modeling identification and control of robots*. CRC Press, 2002.

[38] Oussama Khatib. "A unified approach for motion and force control of robot manipulators: The operational space formulation". In: *IEEE Journal on Robotics and Automation* 3.1 (1987), pp. 43–53.

[39] Charles A Klein and Bruce E Blaho. "Dexterity measures for the design and control of kinematically redundant manipulators". In: *Int. J. Rob. Res.* 6.2 (1987), pp. 72–83.

[40] Jessica Lanini et al. "Human intention detection as a multiclass classification problem: Application in physical human–robot interaction while walking". In: *IEEE Robotics and Automation Letters* 3.4 (2018), pp. 4171–4178.

[41] Alain Liegeois. "Automatic supervisory control of the configuration and behavior of multibody mechanisms". In: *IEEE Trans. Sys., Man and Cyber.* 7.12 (1977), pp. 868–871.

[42] Anthony A Maciejewski and Charles A Klein. "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments". In: *Int. J. Rob. Res.* 4.3 (1985), pp. 109–117.

[43] Norman H Mackworth. "Researches on the measurement of human performance." In: *Researches on the Measurement of Human Performance.* 268 (1950).

[44] Ali Ahmad Malik and Arne Bilberg. "Developing a reference model for human–robot interaction". In: *International Journal on Interactive Design and Manufacturing (IJIDeM)* 13 (2019), pp. 1541–1547.

[45] Ali Ahmad Malik and Arne Bilberg. "Framework to implement collaborative robots in manual assembly: a lean automation approach". In: *28th DAAAM International Symposium on Intelligent Manufacturing & Automation*. DAAAM International. 2017, pp. 1151–1160.

[46] Eleonora Mariotti, Emanuele Magrini, and Alessandro De Luca. "Admittance control for human-robot interaction using an industrial robot equipped with a F/T sensor". In: *2019 International Conference on Robotics and Automation (ICRA)*. IEEE. 2019, pp. 6130–6136.

[47] Chowarit Mitsantisuk, Kiyoshi Ohishi, and Seiichiro Katsura. "Variable mechanical stiffness control based on human stiffness estimation". In: *2011 IEEE International Conference on Mechatronics*. IEEE. 2011, pp. 731–736.

[48] Eugenio Monari, Yi Chen, and Rocco Vertechy. "On locally optimal redundancy resolution using the basis of the null space". In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 9665–9671.

[49] Eugenio Monari et al. "Physical Ergonomics Monitoring in Human–Robot Collaboration: A Standard-Based Approach for Hand-Guiding Applications". In: *Machines* 12.4 (2024), p. 231.

[50] Martín Mujica et al. "Robust variable admittance control for human–robot co-manipulation of objects with unknown load". In: *Robotics and Computer-Integrated Manufacturing* 79 (2023), p. 102408.

[51] Ehsan Noohi, Miloš Žefran, and James L Patton. "A model for human–human collaborative object manipulation and its application to human–robot interaction". In: *IEEE transactions on robotics* 32.4 (2016), pp. 880–896.

[52] Christian Ott. *Cartesian impedance control of redundant and flexible-joint robots*. Springer, 2008.

[53] Christian Ott et al. "On the passivity-based impedance control of flexible joint robots". In: *IEEE Transactions on Robotics* 24.2 (2008), pp. 416–429.

[54] Michael Peshkin and J Edward Colgate. "Cobots". In: *Industrial Robot: An International Journal* 26.5 (1999), pp. 335–341.

[55] François G Pin and Faithlyn A Tulloch. "Resolving kinematic redundancy with constraints using the FSP (full space parameterization) approach". In: *Proc. of IEEE Int. Conf. Rob. and Autom.* Vol. 1. IEEE. 1996, pp. 468–473.

[56] Ron P Podhorodeski, Andrew A Goldenberg, and Robert G Fenton. "Resolving redundant manipulator joint rates and identifying special arm configurations using Jacobian null-space bases". In: *IEEE Trans. Rob and Autom.* 7.5 (1991), pp. 607–618.

[57] Isura Ranatunga et al. "Intent aware adaptive admittance control for physical human-robot interaction". In: *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE. 2015, pp. 5635–5640.

[58] Kyle B Reed and Michael A Peshkin. "Physical collaboration of human-human and human-robot teams". In: *IEEE transactions on haptics* 1.2 (2008), pp. 108–120.

[59] Alexander Reiter, Hubert Gattringer, and Andreas Müller. "Redundancy resolution in minimum-time path tracking of robotic manipulators". In: *Int. Conf. Informatics in Control, Autom. and Rob.* Vol. 3. SCITEPRESS. 2016, pp. 61–68.

[60] Alexander Reiter, Andreas Müller, and Hubert Gattringer. "On higher order inverse kinematics methods in time-optimal trajectory planning for kinematically redundant manipulators". In: *IEEE Trans. Ind. Informatics* 14.4 (2018), pp. 1681–1690.

[61] Loris Roveda et al. "Model-based reinforcement learning variable impedance control for human-robot collaboration". In: *Journal of Intelligent & Robotic Systems* 100.2 (2020), pp. 417–433.

[62] Mohammad Safeea, Richard Bearee, and Pedro Neto. "End-effector precise hand-guiding for collaborative robots". In: *ROBOT 2017: Third Iberian Robotics Conference: Volume 2*. Springer. 2018, pp. 595–605.

[63] Conrad Sanderson and Ryan Curtin. "Armadillo: a template-based C++ library for linear algebra". In: *J. Open Source Soft.* 1.2 (2016), p. 26.

[64] Lorenzo Sciavicco and Bruno Siciliano. "A solution algorithm to the inverse kinematic problem for redundant manipulators". In: *IEEE J. Rob. and Autom.* 4.4 (1988), pp. 403–410.

[65] Abdel-Nasser Sharkawy, Panagiotis N Koustournpardis, and Nikos Aspragathos. "Variable admittance control for human-robot collaboration based on online neural network training". In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 1334–1339.

[66] Bruno Siciliano, Oussama Khatib, and Torsten Kröger. *Springer handbook of robotics*. Vol. 200. Springer, 2008.

[67] Bruno Siciliano et al. *Robotics: Modelling, Planning and Control*. Springer, 2009.

[68] Internation Organization for Standardization. *ISO 10218–1:2011: Robots and robotic devices – Safety requirements for industrial robots – Part 1: Robots*. 2011.

[69] Internation Organization for Standardization. *ISO 10218–2:2011: Robots and robotic devices – Safety requirements for industrial robots – Part 1: Robot systems and integration*. 2011.

[70] Internation Organization for Standardization. *ISO 11228-1:2021, Ergonomics – Manual handling – Part 1: Lifting, lowering and carrying*. 2021.

[71] Internation Organization for Standardization. *ISO 11228-2:2021, Ergonomics – Manual handling – Part 2: Pushing and Pulling*. 2021.

[72] Internation Organization for Standardization. *ISO 11228-3:2021, Ergonomics – Manual handling – Part 3: Handling of low load at high frequency*. 2021.

[73] Internation Organization for Standardization. *ISO 15066:2016: Robots and robotic devices – Collaborative robots*. 2016.

[74] Barry Strauch. "Ironies of automation: Still unresolved after all these years". In: *IEEE Transactions on Human-Machine Systems* 48.5 (2017), pp. 419–433.

[75] Tairen Sun et al. "Repetitive Impedance Learning-Based Physically Human–Robot Interactive Control". In: *IEEE Transactions on Neural Networks and Learning Systems* (2023).

[76] Keng Peng Tee et al. "Concurrent adaptation of force and impedance in the redundant muscle system". In: *Biological cybernetics* 102 (2010), pp. 31–44.

[77] Toru Tsumugiwa, Ryuichi Yokogawa, and Kei Hara. "Variable impedance control based on estimation of human arm stiffness for human-robot cooperative calligraphic task". In: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*. Vol. 1. IEEE. 2002, pp. 644–650.

[78] Federico Vicentini. "Collaborative robotics: a survey". In: *Journal of Mechanical Design* 143.4 (2021), p. 040802.

[79] Tsuneo Yoshikawa. "Manipulability of robotic mechanisms". In: *Int. J. Rob. Res.* 4.2 (1985), pp. 3–9.

[80] Xinbo Yu et al. "Adaptive-constrained impedance control for human–robot co-transportation". In: *IEEE transactions on cybernetics* 52.12 (2021), pp. 13237–13249.

[81] Hajer Zghal, Rajiv V Dubey, and James A Euler. "Efficient gradient projection optimization for manipulators with multiple degrees of redundancy". In: *Proc. IEEE Int. Conf. Rob. and Autom.* IEEE. 1990, pp. 1006–1011.

[82] Liqin Zhu et al. "Determining null-space motion to satisfy both task constraints and obstacle avoidance". In: *IEEE Int. Sym. Assembly and Manufact.* IEEE. 2016, pp. 112–119.