

UNIVERSITÀ DI PISA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE



CORSO DI LAUREA MAGISTRALE IN INGEGNERIA ROBOTICA ED AUTOMAZIONE

Progetto del corso di Controllo dei Robot

Hybrid Force Position control of a KUKA LWR4+

Candidati:

Nicola Piga Matr.: 481729 e-mail: *nicolapiga@gmail.com*
Giulio Romualdi Matr.: 479395 e-mail: *giulio.romualdi@gmail.com*

Supervisori:

Prof. Antonio Bicchi *Ing. Manuel Bonilla*

A.A. 2016-2017

Contents

| | |
|-----------------------------------------------------------------|-----------|
| 1 Description of the task | 4 |
| 1.1 Typical set-up | 4 |
| 1.2 Division of the task in phases | 5 |
| 1.3 Frames of reference | 6 |
| 1.4 Notation and convention | 7 |
| 2 Hybrid force position controller | 8 |
| 2.1 Desired dynamics of the error | 8 |
| 2.2 Definition of a state vector | 9 |
| 2.3 Hybrid Impedance control strategy | 9 |
| 2.4 Frequency domain impedances | 10 |
| 2.5 The duality principle | 11 |
| 2.5.1 Position control | 12 |
| 2.5.2 Force control | 13 |
| 2.6 Design of the control law | 14 |
| 3 Inverse Dynamics Inner Loop | 17 |
| 3.1 Design of the inverse dynamics controller | 17 |
| 3.2 Control of redundant manipulators | 18 |
| 3.2.1 Design of the null operational wrench command | 20 |
| 4 Measure of the contact forces and torques | 22 |
| 4.1 Newton-Euler equations | 22 |
| 4.2 Software induced offset | 23 |
| 4.3 Estimation of parameters and offsets | 24 |
| 4.3.1 Example of estimation | 25 |
| 5 Approaching phase controller | 27 |
| 5.1 Design of a joint space point to point controller | 27 |
| 6 Experimental setup and results | 29 |
| 6.1 Experimental setup | 29 |
| 6.1.1 Force Torque sensor | 30 |
| 6.1.2 Notes about controllers implementation | 30 |
| 6.2 Experiment 1 | 31 |
| 6.3 Experiment 2 | 32 |
| 6.4 Experiment 3 | 34 |

Introduction

In Robotics one of the most interesting task is the manipulation of objects. In general it can be distinguished between the task of restraining objects, called grasping, and the task of manipulating objects with fingers called dexterous manipulation.

The Pisa/IIT Hand [1] is one example of end-effector recently coined as soft hands. It is simple because driven by one motor only but at the same time robust because it adapts itself to the shape of grasped objects.

Altough very versatile the simplicity of the hand make the grasp of thin objects, such as a credit card or a sheet of paper, a very hard task because such objects do not provide enough contact constraints to shape the hand during the grasp execution.

Recent studies [2] propose to face this problem taking into account and exploiting hand-environment interaction. For example an object placed on a table can be dragged until it reaches the edge of the table and sticks out of it. Then a standard grasp can be performed as usual. However in order to make this solution workable the problem of making the dragging phase *safe* must be faced. Using a *pure position* control strategy the *uncontrolled* contact forces and torques arising between the hand and the object and between the object and the surface of the table could damage the object or the hand itself.

In this project a *hybrid force position* controller was implemented and tested on the scenario described above to show that a force feedback solution allows to accomplish the task safely.

The implementation was done using the ROS Control architecture and the robot used in the experimental setup was a KUKA LWR 4+ manipulator.

Contents description

In section one the task of interest is briefly recalled and a number of reference frames are introduced. They will be useful to identify the quantities involved in the design of the controllers.

In section two the main controller which realizes the hybrid force position strategy is designed by choosing appropriate “robot impedances” for each degree of freedom within the so-called *Hybrid Impedance (HIC)* framework proposed by Spong in the 80s.

In section three the synthesis of an inner loop inverse dynamics controller, which is required by the HIC, is discussed. Also the issues of controlling a kinematically redundant robot, like the LWR 4+ in the case of the task described, in the operational space are discussed and solved using a *dynamically consistent generalized inverse* of the Jacobian from Khatib.

In section four the Newton-Euler equations are written for a rigid body attached to a common force/torque sensor in order to understand how to obtain the measure of the contact forces and torques from the raw signal of the sensor. This measure is required as a feedback signal within the HIC framework. Also a simple least squares like approach is proposed to estimate the mass of the rigid body, its center of mass and some software induced offset introduced by the sensor itself.

In section five a joint inverse dynamics controller with inverse kinematics is developed. This controller is used to move the hand near the object of interest when the starting configuration of the hand is too far from the object and the HIC controller can not be used due to the singularities in the attitude parametrization used in HIC and due to the joints limits that are not handled in HIC.

In the ending section the experimental setup is presented together with the results of several experiments.

1 Description of the task

This section serves as a preliminary one for the rest of the report. The main task of interest is briefly recalled and several frames of reference are introduced within the typical setup in which the task is performed. These frames of reference, along with some notations and conventions, will be useful to easily identify the quantities that belong to the state of the controllers developed in the next sections of this report.

1.1 Typical set-up

The typical setup, as shown in figure 1, consists of a 7-joints KUKA LWR 4+ anthropomorphic manipulator equipped with an hand-like end-effector, a table and an object that the hand is not able to grasp using its primitives. Then the task to be performed consists in dragging the object on the table,

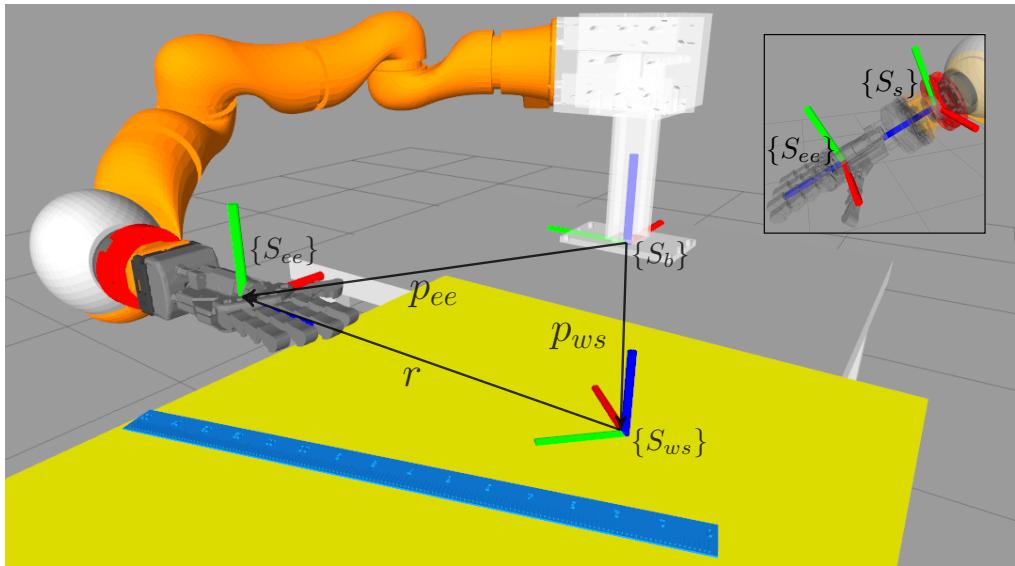


Figure 1: Typical set-up with reference frames

hence exploiting the environmental constraints offered by the surface of the table, until the object sticks out of the border of the table and a standard grasp is feasible. Also the dragging phase should be accomplished without damaging the object or the hand because of too high contact forces between the object and the surface of the table or between the object and the hand. For this reason the manipulator is equipped with a force/torque sensor (fig.

2) which is rigidly attached to the last link of the robot and measures forces and torques exchanged at the wrist of the robot. The hand is connected to the sensor using a mounting plate and a clamp.

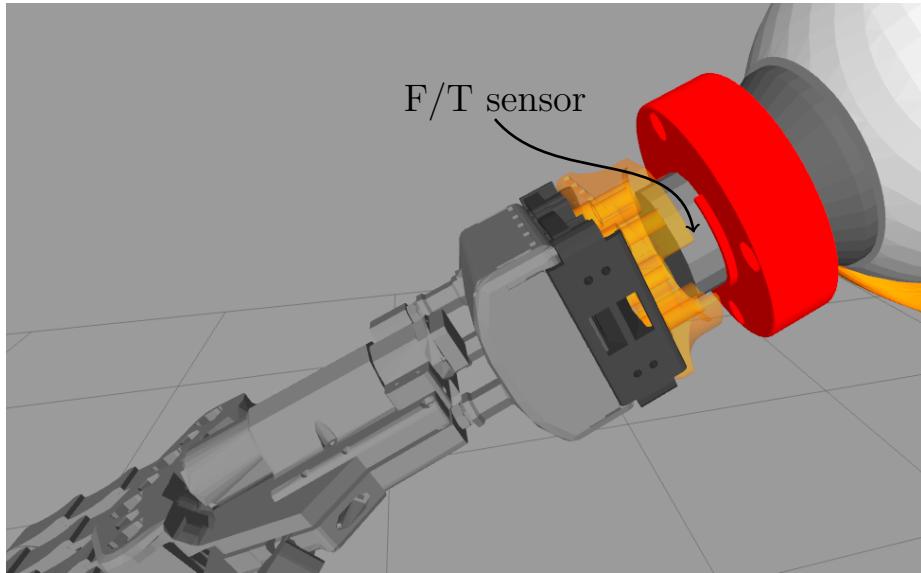


Figure 2: Set-up of the force/torque sensor.

1.2 Division of the task in phases

The dragging phase described above can be considered as the *final* part of a more complex task which should be divided, at least, in three parts:

- approaching phase in which the robot moves the hand near the object of interest;
- contact phase in which the contact between the hand and the object takes place;
- dragging phase.

In the rest of the this report more attention is given to the dragging phase since it is the phase where the hybrid force position strategy is used. As regards the contact phase the experimental results, presented in the last section, show that it can be achieved using the same controller used in the dragging phase. The approaching phase is also described in a separate section.

1.3 Frames of reference

In order to develop a control system that is able to regulate the position of the object and the force exerted on it, i.e. an hybrid force position control system, while the hand is dragging the object on the table a natural choice for a basis in which express the commanded positions and forces is that of an inertial reference frame with the origin W on a point of the table and the x and y axes parallel to the surface of the table

$$\{S_{ws}\} = \{W; x_{ws}, y_{ws}, z_{ws}\}$$

where ws stands for “workspace” since the table represents the intended workspace for the robot. This choice allows to specify the commanded position as 2-dimensional vector with components along the directions \hat{i}_{ws} and \hat{j}_{ws} and the commanded force as a positive scalar along the direction $-\hat{k}_{ws}$. A quantity related to this reference frame is the vector \mathbf{r} going from the origin W to the center of the palm of the hand.

A second reference frame to be introduced is an inertial frame fixed to the main base of the robot

$$\{S_b\} = \{B; x_b, y_b, z_b\}$$

Similarly to \mathbf{r} the vector \mathbf{p}_{ee} goes from B to the center of the palm of the hand. It should be noted that frames $\{S_{ws}\}$ and $\{S_b\}$ have no relative orientation and that the vector \mathbf{p}_{ws} going from B to W is constant. As a consequence the vector $\mathbf{r}(t)$ can be express as

$$\mathbf{r}(t) = \mathbf{p}_{ee}(t) - \mathbf{p}_{ws}$$

where \mathbf{p}_{ws} is known and $\mathbf{p}_{ee}(t)$ is given by forward kinematics numerical routines such as those offered by the library KDL used in this project. The frame $\{S_b\}$ is also the frame in which the library KDL expresses quantities like the vector \mathbf{p}_{ee} , the Jacobians and their derivatives.

Other reference of frames of interest are those fixed to the end-effector

$$\begin{aligned} \{S_s\} &= \{S; x_s, y_s, z_s\} \\ \{S_{ee}\} &= \{E; x_{ee}, y_{ee}, z_{ee}\} \end{aligned}$$

where S corresponds to the center of the F/T sensor and E is in the center of the palm of the hand. These frames, that have no relative orientation, are of interest mainly because the signal produced by the force/torque sensor are expressed in these frames.

1.4 Notation and convention

In the rest of this report the following notation will be used to express any quantity X encountered:

$${}^b X_p$$

where b specifies the reference frame $\{S_b\}$ in which X is expressed and, whenever X is a wrench or a Jacobian J such that $X\dot{q}$ is a twist, p specify the reference point of that wrench or that twist. In case X is an analytical Jacobian or X is a vector containing *also* angular displacement or angular rates it should be clear that the specification of a basis has no meaning and it applies only to a part of the vector.

This notation is very general and sometimes the prescript b and/or the subscript p will be missing when they are clear from the context.

2 Hybrid force position controller

In this section the hybrid force position controller is presented. With reference to the quantities introduced in the section 1.3 this controller should allow to regulate the position and the attitude of the hand with respect to the frame $\{S_{ws}\}$ and at the same time the contact force between the hand and the object expressed along the z axis of the same frame.

2.1 Desired dynamics of the error

In order to express the desired dynamics of the controlled system more precisely the following error quantities are introduced

$$\begin{aligned} e_x(t) &= {}^{ws}r_{x,des}(t) - {}^{ws}r_x(t) \\ e_y(t) &= {}^{ws}r_{y,des}(t) - {}^{ws}r_y(t) \\ \mathbf{e}_\Phi(t) &= \boldsymbol{\Phi}_{des}(t) - \boldsymbol{\Phi}(t) \end{aligned}$$

where \mathbf{r} is the vector from the origin W of the frame $\{S_{ws}\}$ to the center of the palm of the hand (see section 1.3) and $\boldsymbol{\Phi}$ is a vector containing three angles that represent the orientation of the hand with respect to the frame $\{S_{ws}\}$. Also an error e_z is defined as

$$e_z(t) = {}^{ws}F_{z,des}(t) - {}^{ws}F_z(t)$$

where ${}^{ws}F_z$ is the force exerted by the hand on the object expressed along the $\hat{\mathbf{k}}_{ws}$ direction.

Using these definitions the desired dynamics of the error is expressed as

$$\begin{aligned} \ddot{e}_x + B_x \dot{e}_x + K_x e_x &= \mathbf{0} \\ \ddot{e}_y + B_y \dot{e}_y + K_y e_y &= \mathbf{0} \\ \ddot{\mathbf{e}}_\Phi + B_\Phi \dot{\mathbf{e}}_\Phi + K_\Phi \mathbf{e}_\Phi &= \mathbf{0} \\ e_z &\xrightarrow[t \rightarrow \infty]{} 0 \end{aligned}$$

The first two equations express the fact that the hand should rigidly follow a given trajectory on a plane parallel to the surface of the table in a second order damped fashion (K and B represent the proportional and damping terms). The same reasoning applies for the attitude (third equation). Finally the fourth equation only requires that the steady state force error should be zero.

2.2 Definition of a state vector

In order to develop a controller with the features described above an *operational* state space vector is defined as

$${}^{ws}\boldsymbol{x} = \begin{bmatrix} {}^{ws}r_x & {}^{ws}r_y & {}^{ws}r_z & \psi & \theta & \phi \end{bmatrix}^T \quad (1)$$

where the angles ψ , θ and ϕ are the Euler ZYZ parametrization of the rotation matrix ${}^{ws}R_{ee}$ between the frames $\{S_{ws}\}$ and $\{S_{ee}\}$

$${}^{ws}R_{ee} = R_{ZYZ}(\psi, \theta, \phi) = R_{ZYZ}(\boldsymbol{\Phi})$$

Even though the force ${}^{ws}F_z$ does not appear directly in the state vector by making the hypothesis that the second derivative of the state can be commanded arbitrarily

$${}^{ws}\ddot{\boldsymbol{x}} = \boldsymbol{a}_{cmd} \quad (2)$$

it will be shown that force can be regulated by an appropriate choice of term \boldsymbol{a}_{cmd} . The hypothesis made above is a strong one and will be relaxed in the section 3 where an inverse dynamics inner loop controller will be developed for the manipulator. This kind of controller allows to transform the non-linear dynamics of the manipulator in a double integrator system of the form (2).

2.3 Hybrid Impedance control strategy

Among the various works on force control strategies available in the robot control literature the strategy called *Hybrid Impedance Control* proposed by Anderson and Spong [3] was chosen due to its generality.

The Hybrid impedance controller (HIC) combines the hybrid force/position control with an impedance based control approach.

Hybrid force/position control was first proposed by Raibert and Craig [4] and allows to assign different control strategies to each Cartesian degree of freedom using some selection matrices. However it neglects the importance of the manipulator impedance as seen by the environment during the interaction with it.

The concept of impedance proposed by Anderson and Spong is more general with respect to the common impedance approach in which a PD position controller is implemented with position and velocity feedback gains adjusted to obtain different apparent impedances. First of all the impedance control is performed in operational space so that an apparent impedance can be imposed regardless of the configuration of the manipulator. Moreover

the concept of impedance proposed allows to synthesize *direct force* control schemes by matching a given “environment impedance” with the appropriate “manipulator impedance”.

In the following the HIC approach is explained in more details and a control law \mathbf{a}_{cmd} is designed in order to fulfill the control objectives described in 2.1.

2.4 Frequency domain impedances

A central concept in the HIC framework is that of frequency domain impedances. A complex number of the form

$$Z(\omega) = R(\omega) + jX(\omega) \quad (3)$$

with real part $R(\omega)$ and imaginary part $X(\omega)$, which resembles the Laplace transform between a generalized effort and flow for a linear system, is used to model both the environment and the manipulator *for each* degree of freedom available. It should be noted that in this framework the environment is defined to be any element connected to or contacting the robot anywhere past the wrist force sensor.

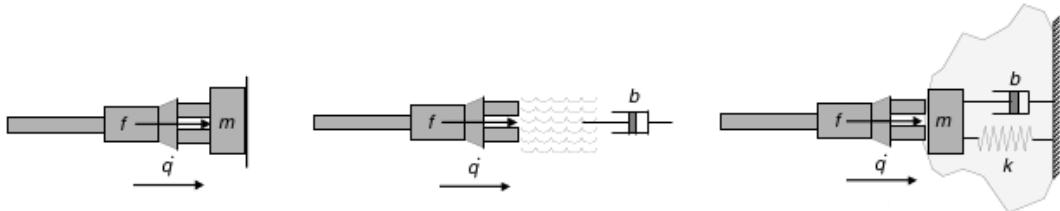


Figure 3: Type of impedances.

Impedances can be classified depending upon their *low-frequency* behavior. As ω approaches zero, one of three things can happen to the magnitude of the environment’s and robot’s impedance. It can approach infinity, it can approach a non-zero finite number or it can approach zero. The following classification is now introduced:

- A system with impedance given by (3) is *inertial* IFF $\lim_{\omega \rightarrow 0} |Z(\omega)| = 0$ (Fig. 3.a)
- A system with impedance given by (3) is *resistive* IFF $\lim_{\omega \rightarrow 0} |Z(\omega)| = c$ where $0 < c < \infty$ (Fig. 3.b)
- A system with impedance given by (3) is *capacitive* IFF $\lim_{\omega \rightarrow 0} |Z(\omega)| = \infty$ (Fig. 3.c)

Capacitive, inertial and resistive systems are usually described in terms of Norton and Thèvenin equivalents. A capacitive system is represented by an impedance in parallel with a flow source (Norton). An inertial system is represented by impedance in series with an effort source (Thévenin). Finally a resistive system can be either represented by a Thèvenin or Norton equivalent.

2.5 The duality principle

Once the environment has been properly modelled, the desired manipulator response may be determined. A fundamental goal for designing a controller is zero steady-state error to a step input. This will be obtained if the following duality principle is applied.

Duality Principle. *The manipulator should be controlled to respond as the dual of the environment.*

This principle is most easily described in terms of Norton and Thèvenin equivalents. An inertial environment, represented using a Thèvenin equivalent (Fig. 4), is controlled by a manipulator represented by a non-inertial impedance. Using the superposition principle the current (velocity) v can be

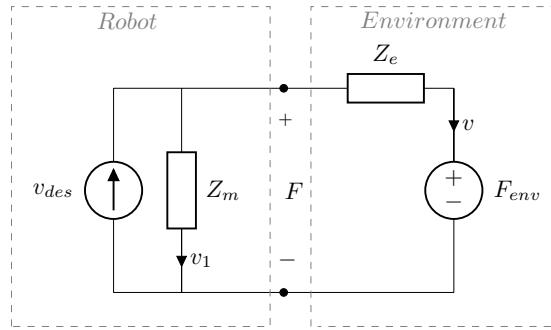


Figure 4: Inertial environment.

evaluated

$$v = \frac{Z_m(s)}{Z_m(s) + Z_e(s)} v_{des} - \frac{F_{env}}{Z_e + Z_m} \quad (4)$$

and the steady state error, assuming no environmental input ($F_{env} \equiv 0$) is equal to

$$e_{ss} \Big|_{F_{env}(t) \equiv 0} = \lim_{s \rightarrow 0} (v - v_{des}) = \frac{-Z_e(0)}{Z_m(0) + Z_e(0)}$$

Since the environment is inertial by hypothesis ($|Z_e(0)| = 0$) a choice of a non-inertial manipulator impedance ($Z_m(0) \neq 0$) assures that

$$e_{ss} \Big|_{F_{env}(t) \equiv 0} = 0$$

So inertial environments are position controlled with a non-inertial manipulator impedances.

A capacitive environment, represented using a Norton equivalent (Fig. 5), is controlled by a manipulator represented by a non-capacitive impedance. Using the superposition principle the voltage (force or torque) F can be

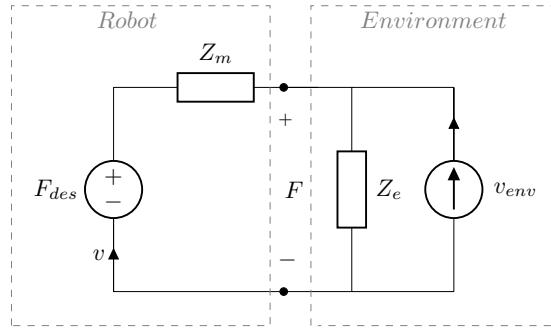


Figure 5: Capacitive environment.

evaluated

$$F = \frac{Z_e(s)}{Z_m(s) + Z_e(s)} F_{des} + \frac{Z_e Z_m}{Z_m + Z_e} V_{env} \quad (5)$$

and the steady state error, assuming no environmental input ($v_{env} \equiv 0$) is equal to

$$e_{ss} \Big|_{v_{env}(t) \equiv 0} = \lim_{s \rightarrow 0} (F - F_{des}) = \frac{-Z_m(0)}{Z_m(0) + Z_e(0)}$$

Since the environment is capacitive by hypothesis ($Z_e(0) \rightarrow \infty$) a choice of a non-capacitive manipulator impedance ($Z_m(0) < \infty$) assures that

$$e_{ss} \Big|_{v_{env}(t) \equiv 0} = 0$$

So capacitive environments are force controlled with non-capacitive manipulator impedances.

2.5.1 Position control

The transfer function for the position-controlled circuit given in the Equation (4) can be realized as a control scheme where the contact force is fed back.

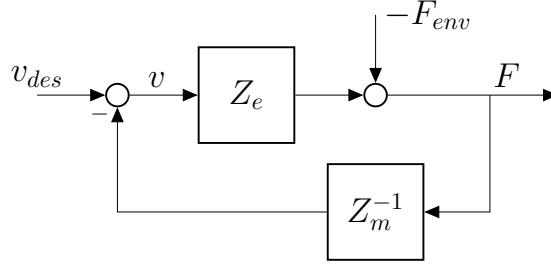


Figure 6: Position control feedback.

Figure 6 shows a block diagram of the position-control implementation . The commanded acceleration can be obtained from the velocity v as

$$a = \frac{d}{dt} \left(v_{des} - \frac{F}{Z_m} \right)$$

In practice, the control a can be obtained without use of differentiators using only the measured force F , the end-effector position x and velocity \dot{x} . This is possible only if the manipulator impedance is expressed in a special form

$$Z_m = Ms + \tilde{Z}_m$$

where M is a tuning parameter.

Finally the position control law evaluates to

$$a = \dot{v}_{des} + (v_{des} - v) \frac{\tilde{Z}_m}{M} - \frac{F}{M} \quad (6)$$

2.5.2 Force control

The transfer function for the force-controlled circuit given in Equation (5) can also be realized as a force feedback scheme. Figure 7 shows a block diagram of the force-control implementation. Again the commanded acceleration can be obtained from the velocity v as

$$a = \frac{d}{dt} \left(\frac{F - F_{des}}{Z_m} \right)$$

As done before for the position-controlled DoF a can be obtained without differentiators if the manipulator impedance assume the form

$$Z_m = Ms + \tilde{Z}_m$$

The force control law can then be written as

$$a = \frac{1}{M}(F - F_{des}) - \frac{1}{M}(\tilde{Z}_m v) \quad (7)$$

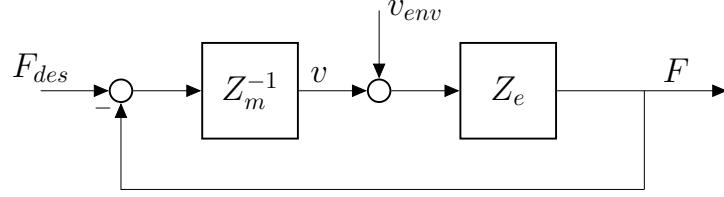


Figure 7: Force control feedback.

2.6 Design of the control law

In order to fulfill the control objectives described in 2.1 the manipulator impedances were chosen as follow.

For the position and attitude-controlled DoFs they are

$$Z_{m,p} = M_p s + \tilde{Z}_{m,p} = M_p s + B_p + \frac{K_p}{s}$$

$$Z_{m,a} = M_a s + \tilde{Z}_{m,a} = M_a s + B_a + \frac{K_a}{s}$$

which result in the following acceleration commands

$$a_p = a_{des,p} + \frac{B_p}{M_p}(v_{des,p} - v_p) + \frac{K_p}{M_p}(x_{des,p} - x_p) - \frac{F_p}{M_p}$$

$$a_a = a_{des,a} + \frac{B_a}{M_a}(v_{des,a} - v_a) + \frac{K_a}{M_a}(x_{des,a} - x_a) - \frac{F_a}{M_a}$$

For the force-controlled DoF the impedance is given by

$$Z_{m,f} = M_f s + \tilde{Z}_{m,f} = M_f s + B_f$$

which results in the control law

$$a_f = M_f^{-1}((f_{des} - f) - B_f v_z)$$

In principle a position control law and a force control law could be assigned for each DoF. Since this behavior is not possible physically a selection matrix S is used to separate the force-controlled and position-controlled reciprocal subspaces

$$\mathbf{a} = S \begin{bmatrix} \mathbf{a}_p \\ \mathbf{a}_a \end{bmatrix} + (I - S)\mathbf{a}_f$$

where the vector \mathbf{a}_p and \mathbf{a}_a are the position and attitude control laws

$$\begin{aligned}\mathbf{a}_p &= \begin{bmatrix} a_{p,x} & a_{p,y} & a_{p,z} \end{bmatrix}^T \\ \mathbf{a}_a &= \begin{bmatrix} a_{a,\psi} & a_{a,\theta} & a_{a,\phi} \end{bmatrix}^T\end{aligned}$$

the vector \mathbf{a}_f is the force/torque control law

$$\mathbf{a}_f = \begin{bmatrix} a_{f,x} & a_{f,y} & a_{f,z} & a_{f,\psi} & a_{f,\theta} & a_{f,\phi} \end{bmatrix}^T$$

and S is the selection matrix that for the set-up described in section 1 assumes the form

$$S = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The resulting control laws are

$$\begin{cases} {}^{ws}a_1 = a_x = \ddot{r}_{x,des} + B_x(\dot{r}_{x,des} - \dot{r}_x) + K_x(r_{x,des} - r_x) - F_x \\ {}^{ws}a_2 = a_y = \ddot{r}_{y,des} + B_y(\dot{r}_{y,des} - \dot{r}_y) + K_y(r_{y,des} - r_y) - F_y \\ {}^{ws}a_3 = a_z = -B_f \dot{r}_z + K_f(F_{z,des} - F_z) \\ {}^{ws}a_4 = a_\psi = \ddot{\psi}_{des} + B_\psi(\dot{\psi}_{des} - \dot{\psi}) + K_\psi(\psi_{des} - \psi) \\ {}^{ws}a_5 = a_\theta = \ddot{\theta}_{des} + B_\theta(\dot{\theta}_{des} - \dot{\theta}) + K_\theta(\theta_{des} - \theta) \\ {}^{ws}a_6 = a_\phi = \ddot{\phi}_{des} + B_\phi(\dot{\phi}_{des} - \dot{\phi}) + K_\phi(\phi_{des} - \phi) \end{cases} \quad (8)$$

where the ratios of the form $\frac{K}{M}$ or $\frac{B}{M}$ are substituted with K and B respectively.

The desired trajectories for the position and attitude-controlled DoFs were chosen as 5th order polynomials of the form

$$s(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

with boundary conditions

$$\begin{aligned}s(0) &= s_0 & \dot{s}(0) &= 0 & \ddot{s}(0) &= 0 \\ s(t_f) &= s_f & \dot{s}(t_f) &= 0 & \ddot{s}(t_f) &= 0\end{aligned}$$

while the reference force trajectory was chosen as a 3th order polynomial

$$s(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

with boundary conditions

$$\begin{aligned}s(0) &= s_0 & \dot{s}(0) &= 0 \\ s(t_f) &= s_f & \dot{s}(t_f) &= 0\end{aligned}$$

3 Inverse Dynamics Inner Loop

In the previous section the hybrid force position control scheme was described with the assumption that an inverse dynamics inner loop was available. In this section the inverse dynamics controller is described and the issue of controlling a kinematically redundant manipulator is discussed.

3.1 Design of the inverse dynamics controller

The development of the controller starts from the dynamic model of the 7-joint manipulator

$$B(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} - {}^b J_S^T {}^b \mathbf{w}_S \quad (9)$$

where \mathbf{w}_S is the wrench exerted by the hand on the environment with reference point in the centre of the sensor (S) and J_S is the geometric Jacobian of the robot. The gravity term is canceled because it is handled internally by the robot itself.

The first step towards the exact dynamics inversion is to cancel the Coriolis term $C\dot{\mathbf{q}}$ and the term containing the wrench \mathbf{w}_S which is supposed to be available as a feedback signal. The resulting control torque is then

$$\boldsymbol{\tau} = C\dot{\mathbf{q}} + {}^b J_S^T ({}^b \boldsymbol{\gamma} + {}^b \mathbf{w}_S)$$

where an additional control wrench $\boldsymbol{\gamma}$ is introduced. Substituting the control torque and solving for the joints accelerations $\ddot{\mathbf{q}}$ leads to

$$\ddot{\mathbf{q}} = B^{-1}({}^b J_S^T) {}^b \boldsymbol{\gamma} \quad (10)$$

The link between the joint space description and the operational space description is given by the following formula

$${}^{ws} \ddot{\mathbf{x}} = {}^{ws} J_{A,E} \ddot{\mathbf{q}} + {}^{ws} \dot{J}_{A,E} \dot{\mathbf{q}} \quad (11)$$

where ${}^{ws} \mathbf{x}$ is the previously introduced (eq 1) state in operational space expressed in workspace frame and $J_{A,E}$ is the analytical Jacobian of the robot with reference point in the palm of the hand. The analytical Jacobian, which takes into account the conversion between the angular velocity $\boldsymbol{\omega}$ and the vector of angular rates $\dot{\boldsymbol{\Phi}}$, can be written as

$${}^{ws} J_{A,E} = \begin{bmatrix} I & 0 \\ 0 & T^{-1}(\boldsymbol{\Phi}) \end{bmatrix} {}^{ws} J_E \quad {}^{ws} \boldsymbol{\omega} = T(\boldsymbol{\Phi}) \dot{\boldsymbol{\Phi}}$$

with J_E the geometric Jacobian with reference point int the palm of the hand. It should be noted that the inverse of the matrix T exists if and only if the sine of the angle θ (part of the state \mathbf{x}) is not zero. The derivative of the analytical Jacobian evaluates to

$${}^{ws} \dot{J}_{A,E} = \begin{bmatrix} 0 & 0 \\ 0 & -T^{-1} \dot{T} T^{-1} \end{bmatrix} {}^{ws} J_E + \begin{bmatrix} I & 0 \\ 0 & T^{-1} \end{bmatrix} {}^{ws} \dot{J}_E$$

where the derivative of the Jacobian \dot{J}_E can be evaluated numerically using software libraries like KDL (Kinematics and Dynamics Library).

By substituing equation (10) in equation (11) one obtains

$${}^{ws} \ddot{\mathbf{x}} = \underbrace{{}^{ws} J_{A,E} B^{-1} ({}^b J_S^T) {}^b \boldsymbol{\gamma}}_{\Lambda_A^{-1}} + {}^{ws} \ddot{J}_{A,E} \dot{\mathbf{q}}$$

where the kinetic psuedo-kinetic energy matrix Λ_A was introduced. Multipling both sides of the equation by Λ_A leads to

$$\Lambda_A {}^{ws} \ddot{\mathbf{x}} = {}^b \boldsymbol{\gamma} + \Lambda_A {}^{ws} \dot{J}_{A,E} \dot{\mathbf{q}}$$

where the control wrench $\boldsymbol{\gamma}$ can now be easily chosen as

$$\boldsymbol{\gamma} = \Lambda_A \mathbf{a}_{cmd} - \Lambda_A {}^{ws} \dot{J}_{A,E} \dot{\mathbf{q}}$$

resulting in

$${}^{ws} \ddot{\mathbf{x}} = \mathbf{a}_{cmd}$$

where \mathbf{a}_{cmd} represents the outer control expressed in equation (8). Taking into account the command wrench $\boldsymbol{\gamma}$ the final commanded torque is

$$\boldsymbol{\tau} = C \dot{\mathbf{q}} + {}^b J_S^T (B_A \mathbf{a} - B_A {}^{ws} \dot{J}_{A,E} \dot{\mathbf{q}} + {}^b \mathbf{w}_S) \quad (12)$$

3.2 Control of redundant manipulators

The control law presented in the previous subsection does not take into account the fact that the controlled manipulator is kinematically redundant. In fact it has 7 joints while the commanded state in the operational space is 6-dimensional. A consequence of this fact is that [5] the configuration of the robot, which describes the end-effector position and orientation, does not constitute a generalized coordinate system for the whole redundant manipulator and the dynamic behavior of the entire redundant system cannot be represented by a dynamic model in coordinates only of the end-effector configuration. Indeed it turns out that the command torques specified in equation

(12), although they specify univocally the behavior of the end-effector, they are none other than one of the possible torques producing a given operational command wrench γ on the end-effector, i.e, the expression

$$\tau = J^T \gamma \quad (13)$$

represents just one of these torques. Clearly different torques lead to completely different behaviors of the additional degree of freedom of a kinematically redundant manipulator, some of which may be undesired or even dangerous. For this reason a complete characterization of the torques producing a given command wrench at the end-effector is necessary to avoid this issue.

Independently of the specific choice for the operational space description of the robot the second derivative with respect to time of the state can be written in the form

$$\ddot{x} = J(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \quad (14)$$

for some Jacobian matrix J and some function of the joints state and velocity \mathbf{h} . By combining equations (9), in the case of zero contact forces, (13) and (14) the equations of motion of the end-effector are obtained

$$\Lambda \ddot{x} + \boldsymbol{\mu}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{p} = \gamma \quad (15)$$

where

$$\Lambda = (JB^{-1}J^T)^{-1}$$

$$\boldsymbol{\mu} = \bar{J}^T C \dot{\mathbf{q}} - \Lambda \mathbf{h}$$

$$\mathbf{p} = \bar{J}^T \mathbf{G}$$

are the already seen pseudo-kinetic energy matrix, the centrifugal and Coriolis forces acting on the end-effector and the gravity force acting on the end-effector respectively. The matrix $\bar{J} = B^{-1}J^T\Lambda$ is a generalized inverse of the Jacobian matrix which solves the problem of finding the joints velocities that produce a desired twist while minimizing the manipulator's instantaneous kinetic energy.

Interestingly equation (15) can also be written in the form

$$\bar{J}^T(B\ddot{\mathbf{q}} + C\dot{\mathbf{q}} + \mathbf{G}) = \gamma$$

Since the joint space dynamics term that is multiplied by \bar{J}^T is equal to the commanded torque τ one finds that

$$\gamma = \bar{J}^T \tau \quad (16)$$

which is an important result since it allows to design a command torque $\boldsymbol{\tau}$ of the form

$$\boldsymbol{\tau} = J^T \boldsymbol{\gamma} + (I_7 - J^T \bar{J}^T) \boldsymbol{\gamma}_0 \quad (17)$$

where the command wrench $\boldsymbol{\gamma}_0$ is projected in the null space of the generalized inverse \bar{J}^T hence, as results by direct substitution of equation (17) in equation (16), it corresponds to a *null* operational wrench command. Clearly the term $\boldsymbol{\gamma}_0$ can be used to avoid undesired motion of the additional degree of freedom of the manipulator without interfering with the desired motion of the end-effector.

It should be noted that the expression in equation (17) is very general, i.e., every commanded torque $\boldsymbol{\tau}$ can always be expressed in that form. Also due to its filtering capabilities the matrix \bar{J} is called *dynamically consistent generalized inverse* of J in the sense that the projector in the null of \bar{J}^T is able to filter out commanded wrenches in a way that is consistent with the dynamics of both the whole manipulator and the end-effector.

Taking into account the development presented in this section the commanded torques $\boldsymbol{\tau}$ in equation (12) are updated to

$$\boldsymbol{\tau} = C\dot{\boldsymbol{q}} + {}^bJ_S^T(\Lambda_A \boldsymbol{a} - \Lambda_A^{ws} \dot{J}_{A,E} \dot{\boldsymbol{q}} + {}^b\boldsymbol{w}_S) + (I_7 - ({}^bJ_S^T)({}^b\bar{J}_S^T)) \boldsymbol{\gamma}_0 \quad (18)$$

where

$${}^b\bar{J}_S = B^{-1}({}^bJ_S^T)({}^b\Lambda_S)$$

and

$${}^b\Lambda_S = ({}^bJ_S B^{-1}({}^bJ_S^T))^{-1}$$

3.2.1 Design of the null operational wrench command

Design of the control law assigned to the term $\boldsymbol{\gamma}_0$ was guided by the simulation of the task described in section 1. Extensive simulations revealed that the uncontrolled internal motions of the robot, while not affecting the desired attitude of the hand, caused the 5-th and 7-th links to rotate cooperatively and reach their limits soon. Another issue with internal motions is that they could cause, in some situations, collisions between the 4-th link and the table which is part of the workspace of the robot.

In order to handle the issues described the control $\boldsymbol{\gamma}_0$ was chosen as

$$\boldsymbol{\gamma}_0 = J_{im}^T(K_p(\boldsymbol{x}_{im,des} - \boldsymbol{x}_{im}) - K_d \dot{\boldsymbol{x}}_{im})$$

where im stands for “internal motion”, the state \boldsymbol{x}_{im}

$$\boldsymbol{x}_{im} = \begin{bmatrix} {}^b x_{l4} & {}^b y_{l4} & {}^b z_{l4} & \psi_{l5} & \theta_{l5} & \phi_{l5} \end{bmatrix}^T = \begin{bmatrix} \boldsymbol{p}_{l4}^T & \boldsymbol{\Phi}_{l5}^T \end{bmatrix}^T$$

describes the position, with respect to the base of the robot, of the 4-th link and the attitude of the 5-th link using a ZYZ parametrization and the Jacobian J_{im} is such that

$$\begin{bmatrix} \dot{\mathbf{p}}_{l4} \\ \dot{\boldsymbol{\Phi}}_{l5} \end{bmatrix} = J_{im} \dot{\mathbf{q}}$$

The gains K_p and K_d and the desired state $\mathbf{x}_{im,des}$ were chosen as

$$K_p = \text{diag}(0, 0, k_{p,z}^{im}, 0, 0, k_{p,att}^{im}) \quad K_d = k_d^{im} I_6$$

$$\mathbf{p}_{l4} = \begin{bmatrix} * & * & {}^b z_{ee} + off_z \end{bmatrix}^T \quad \boldsymbol{\Phi}_{l5} = \begin{bmatrix} * & * & 0 \end{bmatrix}^T$$

where ${}^b z_{ee}$ is the altitude of the hand with respect to the base of the robot and $k_{p,z}^{im}$, $k_{p,att}^{im}$ and k_d^{in} are suitable gains. Such a choice allows to fix the orientation of the 5-th link to a value that is comfortable during the execution of task described in the section 1 and to regulate the altitude between the 4-th link and the hand so as to avoid collisions between the elbow of the robot and the workspace.

4 Measure of the contact forces and torques

In the previous sections the hybrid force-position control architecture was described and the assumption was made that the wrench \mathbf{w}_S (see eq. 9) exerted by the hand on the environment was available as a feedback signal.

In this section it is explained how such a wrench can be obtained starting from the *raw* signal coming from a common force/torque sensor and what are the main problems encountered in doing that.

4.1 Newton-Euler equations

In order to understand what are the actual quantities measured by a force torque sensor it is useful to describe, by means of the Newton-Euler approach, the motion of a rigid body, attached to the sensor through a mounting plate, due to external forces and torques.

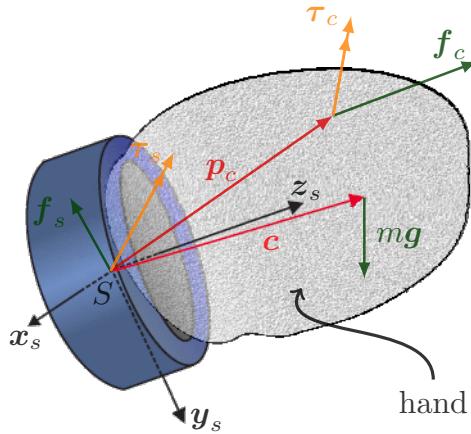


Figure 8: F/t sensor with hand attached

The resulting equations are the following

$${}^s \mathbf{f}_s = - {}^s \mathbf{f}_{pl} - {}^s \mathbf{f}_c - m {}^s \mathbf{g} + m {}^s \mathbf{a}_{cm} \quad (19)$$

$$\begin{aligned} {}^s \boldsymbol{\tau}_s = & - {}^s \boldsymbol{\tau}_{pl} - {}^s \boldsymbol{\tau}_c - {}^s \tilde{\mathbf{p}}_c {}^s \mathbf{f}_c + {}^s \tilde{\mathbf{g}} m {}^s \mathbf{c} \\ & - {}^s \tilde{\mathbf{a}}_{cm} m {}^s \mathbf{c} + {}^s I_{cm} {}^s \boldsymbol{\alpha} + {}^s \tilde{\boldsymbol{\omega}} {}^s I_{cm} {}^s \boldsymbol{\omega} \end{aligned} \quad (20)$$

where \mathbf{f}_s and $\boldsymbol{\tau}_s$ are the forces and torques exerted by the sensor on the rigid body, \mathbf{f}_{pl} and $\boldsymbol{\tau}_{pl}$ are due to the mounting plate, \mathbf{f}_c and $\boldsymbol{\tau}_c$ are the *external* forces and torques exerted on the rigid body in the interaction with

the environment, \mathbf{c} is the vector from the center of the sensor to the CoM of the rigid body, \mathbf{p}_c is the vector from the center of the sensor to the contact point with the environment, I_{cm} is the inertia matrix of the rigid body with respect to the CoM, \mathbf{g} is the gravity vector, \mathbf{a}_{cm} is the linear acceleration of the CoM of the rigid body, $\boldsymbol{\omega}$ is the angular velocity of the rigid body and $\boldsymbol{\alpha}$ is the angular acceleration of the rigid body. It should be noted that the wrench \mathbf{w}_S is given by

$$\mathbf{w}_S = - \begin{bmatrix} \mathbf{f}_c \\ \boldsymbol{\tau}_c + \tilde{\mathbf{p}}_c \mathbf{f}_c \end{bmatrix}$$

Since in this project the force/torque sensor is primarily used to control the contact force when the end-effector is still or when it moves along straight lines the angular velocity and the angular acceleration of the end-effector attached to the sensor can be neglected leading to the equations

$$\mathbf{f}_s = -\mathbf{f}_{pl} - \mathbf{f}_c - m\mathbf{g} + m\mathbf{a}_{cm} \quad (21)$$

$$\boldsymbol{\tau}_s = -\boldsymbol{\tau}_{pl} - \boldsymbol{\tau}_c - \tilde{\mathbf{p}}_c \mathbf{f}_c + \tilde{\mathbf{g}} m \mathbf{c} - \tilde{\mathbf{a}}_{cm} m \mathbf{c} \quad (22)$$

4.2 Software induced offset

The hardware of the force/torque sensor is built in a such way that the forces and torques exchanged with the rigid body attached to it are measured. With reference to the equations (21) and (22) the signal produced by the sensor can be written as

$$\begin{aligned} \mathbf{f}_m &= -\mathbf{f}_s + \mathbf{f}_{sw,off} \\ \boldsymbol{\tau}_s &= -\boldsymbol{\tau}_s + \boldsymbol{\tau}_{sw,off} \end{aligned}$$

where $\mathbf{f}_{sw,off}$ and $\boldsymbol{\tau}_{sw,off}$ represent the offset introduced by the sensor when the user activates the *calibration* of the sensor. This operation is performed when the sensor is *still* and not in contact with the environment and the resulting offset is chosen such that $\mathbf{f}_{m,0} = \boldsymbol{\tau}_{m,0} = \mathbf{0}$ where the zero subscript represents the calibration condition. Simple substitutions show that

$$\begin{aligned} \mathbf{f}_{sw,off} &= \mathbf{f}_{s,0} = -\mathbf{f}_{pl} - m^s \mathbf{g}_0 \\ \boldsymbol{\tau}_{sw,off} &= \boldsymbol{\tau}_{s,0} = -\boldsymbol{\tau}_{pl} + {}^s \tilde{\mathbf{g}}_0 m \mathbf{c} \end{aligned}$$

where \mathbf{g}_0 is the value assumed by the gravity expressed in the sensor frame S at the moment of the calibration.

Taking into account the expressions of the offset the measured forces and torques in absence of angular motions can be written as

$$\mathbf{f}_m = \mathbf{f}_c + m\mathbf{g} - m^s\mathbf{g}_0 - m\mathbf{a}_{cm} \quad (23)$$

$$\boldsymbol{\tau}_m = \boldsymbol{\tau}_c + \tilde{\mathbf{p}}_c \mathbf{f}_c - \tilde{\mathbf{g}} m \mathbf{c} + {}^s\tilde{\mathbf{g}}_0 m \mathbf{c} + \tilde{\mathbf{a}}_{cm} m \mathbf{c} \quad (24)$$

As suggested by the previous formulas in order to obtain the part of the measure related to the contact with the environment it is required to estimate the mass of the rigid body m , the position of the centre of mass with respect to center of the sensor \mathbf{c} and the offsets $-m^s\mathbf{g}_0$ and ${}^s\tilde{\mathbf{g}}_0 m \mathbf{c}$. The gravity ${}^s\mathbf{g}$ expressed in sensor frame can be obtained using the direct kinematics software facilities developed for the project while the linear acceleration \mathbf{a}_{cm} can be obtained using an appropriate jacobian, its derivative and an estimate of the angular velocities and accelerations of the joints of the robot.

4.3 Estimation of parameters and offsets

Since all the quantities to be estimated are revealed even when the sensor is still it is possible to estimate them by acquiring a certain number of measures each when the robot, hence the sensor, assumes a static pose. The estimation can be obtained from the measures using a *linear* least squares approach since there exist a linear relationship between the measurements and the parameters

$$\begin{aligned} \begin{bmatrix} \mathbf{f}_m \\ \boldsymbol{\tau}_m \end{bmatrix} &= \begin{bmatrix} \mathbf{g}m + (-m^s\mathbf{g}_0) \\ -\tilde{\mathbf{g}} m \mathbf{c} + ({}^s\tilde{\mathbf{g}}_0 m \mathbf{c}) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{g} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \\ \mathbf{0} & -\tilde{\mathbf{g}} & 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix} \begin{bmatrix} m \\ m\mathbf{c} \\ -m^s\mathbf{g}_0 \\ {}^s\tilde{\mathbf{g}}_0 m \mathbf{c} \end{bmatrix} \\ &= H({}^s\mathbf{g})\boldsymbol{\theta} \end{aligned}$$

where ${}^s\mathbf{g} = {}^s\mathbf{g}(\mathbf{q})$ depends on the configuration \mathbf{q} of the robot.

Given n configurations of the robot the estimate is given by

$$\hat{\boldsymbol{\theta}} = \begin{bmatrix} H(\mathbf{q}^1) \\ \vdots \\ H(\mathbf{q}^n) \end{bmatrix}^+ \begin{bmatrix} \mathbf{f}_m^1 \\ \boldsymbol{\tau}_m^1 \\ \vdots \\ \mathbf{f}_m^n \\ \boldsymbol{\tau}_m^n \end{bmatrix} = H_n^+ \begin{bmatrix} \mathbf{f}_m^1 \\ \boldsymbol{\tau}_m^1 \\ \vdots \\ \mathbf{f}_m^n \\ \boldsymbol{\tau}_m^n \end{bmatrix}$$

where it can be shown that for $n \geq 3$ the matrix H_n is always full column rank so that $H_n^+ = H_n(H_n^T H_n)^{-1}$. Once the estimate is computed contact forces and torques can be evaluated as

$$\begin{aligned}\hat{\mathbf{f}}_c &= \mathbf{f}_m - \hat{\theta}_1 \mathbf{g} - \hat{\theta}_{5:7} + \hat{\theta}_1 \hat{\mathbf{a}}_{cm}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \\ \hat{\boldsymbol{\tau}}_c + \tilde{\mathbf{p}}_c \hat{\mathbf{f}}_c &= \boldsymbol{\tau}_m + \tilde{\mathbf{g}} \hat{\theta}_{2:4} - \hat{\theta}_{8:10} - \tilde{\mathbf{a}}_{cm}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \hat{\theta}_{2:4}\end{aligned}$$

where $\hat{\mathbf{a}}_{cm}$ is an estimate of the linear acceleration of the centre of mass of the rigid body.

4.3.1 Example of estimation

An example of estimation, with $n = 60$, is shown in the figures below where each step represents a new row of the matrix H_{60} . The estimated parameters are the following:

- $\hat{m} = 0.6848 \text{ kg}$ (see Fig. 1a)
- $\hat{\mathbf{c}} = [0.32 \ 0.28 \ 11.20]^T \text{ cm}$ (see Fig. 1b)
- $-\hat{m}^s \tilde{\mathbf{g}}_0 = [0.8577 \ -6.0316 \ 0.8438]^T \text{ N}$ (see Fig. 2a)
- ${}^s \tilde{\mathbf{g}}_0 \hat{m} \hat{\mathbf{c}} = [0.8299 \ 0.0772 \ -0.0149]^T \text{ Nm}$ (see Fig. 2b)

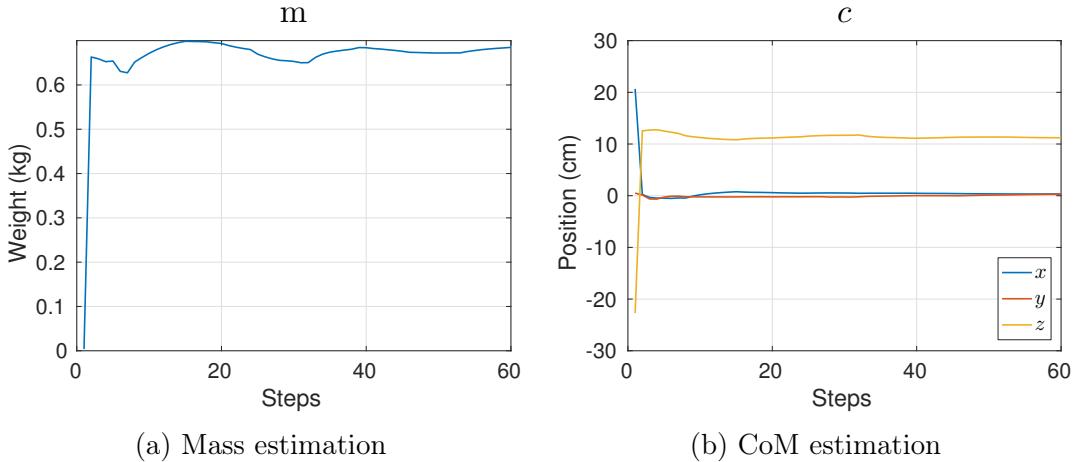


Table 1: Estimation results

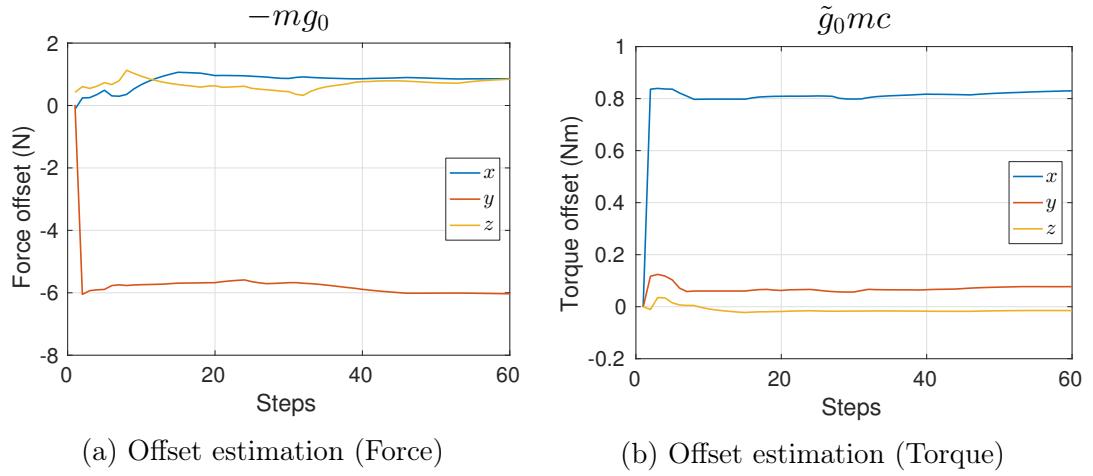


Table 2: Estimation results

It can be verified that

$$\frac{\|\hat{m}\hat{\mathbf{g}}_0\|}{9.81} = 0.6270 \approx 0.6848 = \hat{m}$$

as expected.

5 Approaching phase controller

As explained in the section 1.2 before the execution of the dragging phase a preliminar approaching phase is required to move the hand near the object of interest.

In principle the HIC controller could also be used to perform this task but in practice some issues due to the singularity introduced by the Euler ZYZ parametrization and due to the joints limits make that controller useful during the dragging phase only where short movements are executed and the attitude is carefully commanded to avoid the singularity issue. So another controller was developed using a joint space approach so that large movements can be performed.

5.1 Design of a joint space point to point controller

The alternative controller was developed combining an inverse dynamics controller in joint space with an inverse kinematics solver.

By substituting a command torque of the form

$$\boldsymbol{\tau} = C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \mathbf{G}(\boldsymbol{q}) + B(\boldsymbol{q})\boldsymbol{a}$$

in the joint space dynamics of the manipulator

$$B(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \mathbf{G}(\boldsymbol{q}) = \boldsymbol{\tau}$$

the following results

$$\ddot{\boldsymbol{q}} = \boldsymbol{a}_{p2p}$$

where \boldsymbol{a}_{p2p} is the new input vector.

In order to move the hand from the previous configuration \boldsymbol{q}_{prev} to a new one expressed as a position in cartesian coordinates \boldsymbol{p}_{ee} and an orientation ${}^bR_{ee}$ a joint space trajectory was generated

$$q_{des}^i(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5$$

with boundary conditions

$$\begin{aligned} q_{des}^i(0) &= q_{prev}^i & \dot{q}_{des}^i(0) &= 0 & \ddot{q}_{des}^i(0) &= 0 \\ q_{des}^i(t_f) &= q_f^i & \dot{q}_{des}^i(t_f) &= 0 & \ddot{q}_{des}^i(t_f) &= 0 \end{aligned}$$

where \boldsymbol{q}_f is given by an inverse kinematics solver

$$\boldsymbol{q}_f = FK^{-1}(\boldsymbol{p}_{ee,des}, {}^bR_{ee,des})$$

and is discarded whenever its components exceed the joints limits.

In order to execute the trajectory an input vector of the form

$$\boldsymbol{a}_{p2p} = \ddot{\boldsymbol{q}}_{des} + K_d(\dot{\boldsymbol{q}}_{des} - \dot{\boldsymbol{q}}) + K_p(\boldsymbol{q}_{des} - \boldsymbol{q})$$

was used where K_p and K_d are positive definite matrices.

6 Experimental setup and results

In this section the experimental setup is described and the results of three experiments are presented. In the first one the robot is supposed to press an emergency button with variables force references. In the second and in the third one the robot drags an object on a table while the contact force is regulated until it reaches the edge of the table and sticks out of it.

6.1 Experimental setup

The test-bed for this project is a Kuka LWR4+, a 7-joints industrial manipulator, which is provided with an internal software layer called Fast Research Interface (FRI).



Figure 9: KUKA LWR4+ manipulator.

In order to send the control laws, described in the previous sections, to the robot one of the internal modes of operation, called Joints Specific Impedance, was used

$$\boldsymbol{\tau}_{cmd} = K_j(\boldsymbol{q}_{FRI} - \boldsymbol{q}_{msr}) + D(d_j) + \boldsymbol{\tau}_{FRI} + \mathbf{f}_{dynamics}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})$$

where $\boldsymbol{\tau}_{cmd}$ is the effective torque sent to the robot, \boldsymbol{q}_{FRI} is the vector of the desired angular positions of the joints, \boldsymbol{q}_{msr} is the vector of the angular positions of the joints, K_j and $D(d_j)$ are gains that shape the dynamics of the impedance control mode, $\boldsymbol{\tau}_{FRI}$ is an additional torque provided by the user and $\mathbf{f}_{dynamics}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})$ is a gravity compensation term provided by the producer.

Since the control laws developed are pure torque laws only the term τ_{FRI} was used and the others were set as:

- $K_j = 0;$
- $d_j = 0;$
- $\mathbf{q}_{FRI} = \mathbf{q}_{msr}$

6.1.1 Force Torque sensor

The force/torque sensor used in this work is an ATI Mini45 [6]. In all the experiments the measured signal was filtered using a built-in LP filter with cutoff frequency of 8 Hz.

6.1.2 Notes about controllers implementation

In order to implement the control laws

$$\begin{aligned}\boldsymbol{\tau} &= C\dot{\mathbf{q}} + B\mathbf{a}_{p2p} \\ \boldsymbol{\tau} &= C\dot{\mathbf{q}} + {}^bJ_F^T(\Lambda_A \mathbf{a}_{cmd} - \Lambda_A^{ws} \dot{J}_{A,E} \dot{\mathbf{q}} + {}^b\mathbf{w}_F) + (I_7 - ({}^bJ_F^T)({}^b\bar{J}_F^T))\boldsymbol{\gamma}_0\end{aligned}$$

several quantities are required among which

- angular position, velocity and acceleration of the joints;
- Jacobians;
- inertia matrix;
- Coriolis matrix;
- direct and inverse kinematics facilities;
- Euler ZYZ kinematical matrix.

The angular position are provided by the FRI software layer. The angular velocities and the angular accelerations are not provided by the producer so they were estimated using an exponential smoothing filter of the form

$$\begin{cases} \dot{\mathbf{q}}_0 = \mathbf{0} \\ \dot{\mathbf{q}}_k = (1 - \alpha)\dot{\mathbf{q}}_{k-1} + \alpha \frac{\mathbf{q}_k - \mathbf{q}_{k-1}}{t_s} \end{cases}$$

$$\begin{cases} \ddot{\mathbf{q}}_0 = \mathbf{0} \\ \ddot{\mathbf{q}}_k = (1 - \alpha)\ddot{\mathbf{q}}_{k-1} + \alpha \frac{\dot{\mathbf{q}}_k - \dot{\mathbf{q}}_{k-1}}{t_s} \end{cases}$$

where the constant $\alpha \in [0, 1]$ is the smoothing factor and t_s is the sampling time. In all the experiments $\alpha = 0.2$ and $t_s = 3\text{ ms}$.

All the others quantities were evaluated numerically using the facilities offered by the KDL library which was integrated into the already existing KUKA LWR4+ software stack provided to the students

6.2 Experiment 1

In this experiment (fig. 10) the robot is supposed to press an emergency button.

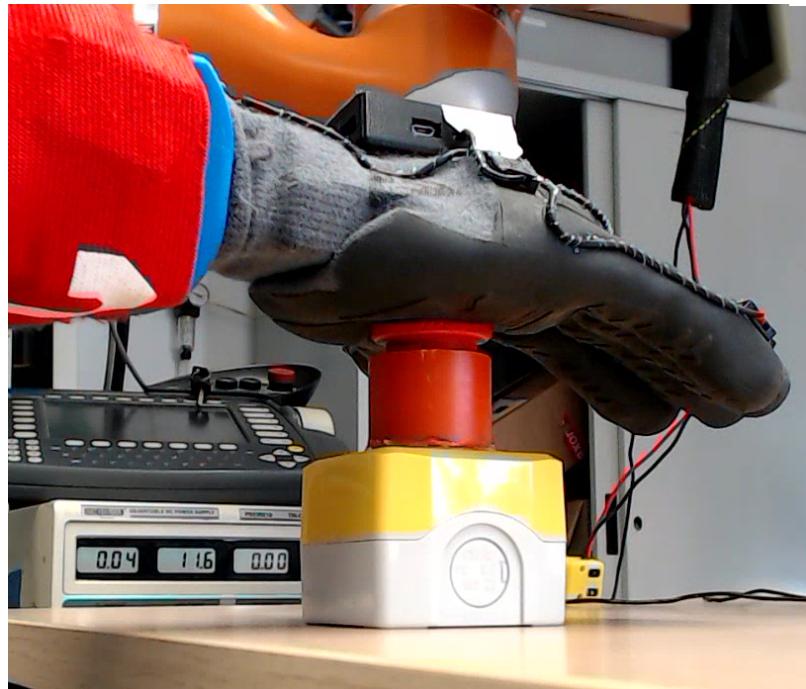
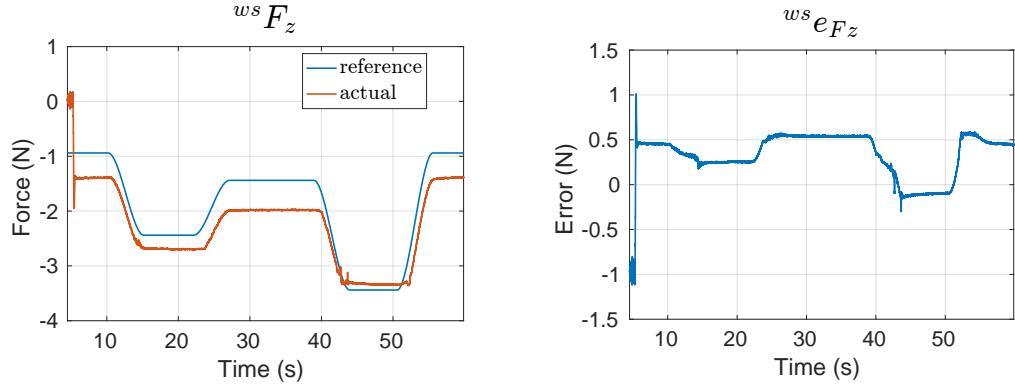


Figure 10: The hand while it presses the emergency button.

Initially a reference command of approximately 1 N is sent to the robot until the contact takes place. Then several set points are sent to the robot:

- $F_{z,des} = -2.5\text{ N}$
- $F_{z,des} = -1.5\text{ N}$
- $F_{z,des} = -3.5\text{ N}$
- $F_{z,des} = -1\text{ N}$



(a) Reference and actual force.

(b) Error.

Table 3: Experiment 1, the hand presses a button.

$$K_f = 2 \text{ and } B_f = 45$$

The results, presented in the Table 3, show that the error after the contact phase is bounded between -0.35 N and 0.5 N and it decreases as the requested force increases. The contact phase is visible as a spike in the measured force (see Table 3a).

6.3 Experiment 2

In this experiment (Fig. 11) a banana is dragged on a table while the contact force between the hand and the fruit is regulated. The object is dragged in

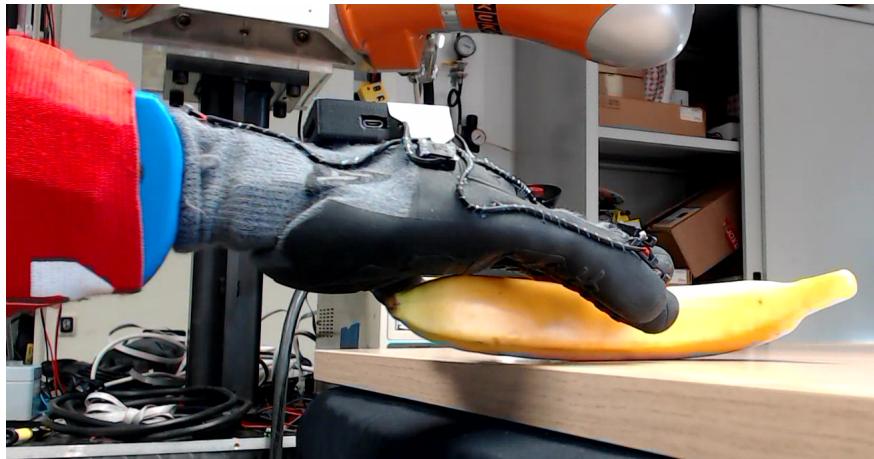


Figure 11: The hand while it drags the banana.

5 s for a distance of 12 cm and the requested force is 3 N. In the Table 4 only the results of the dragging phase are shown. Although the force error is

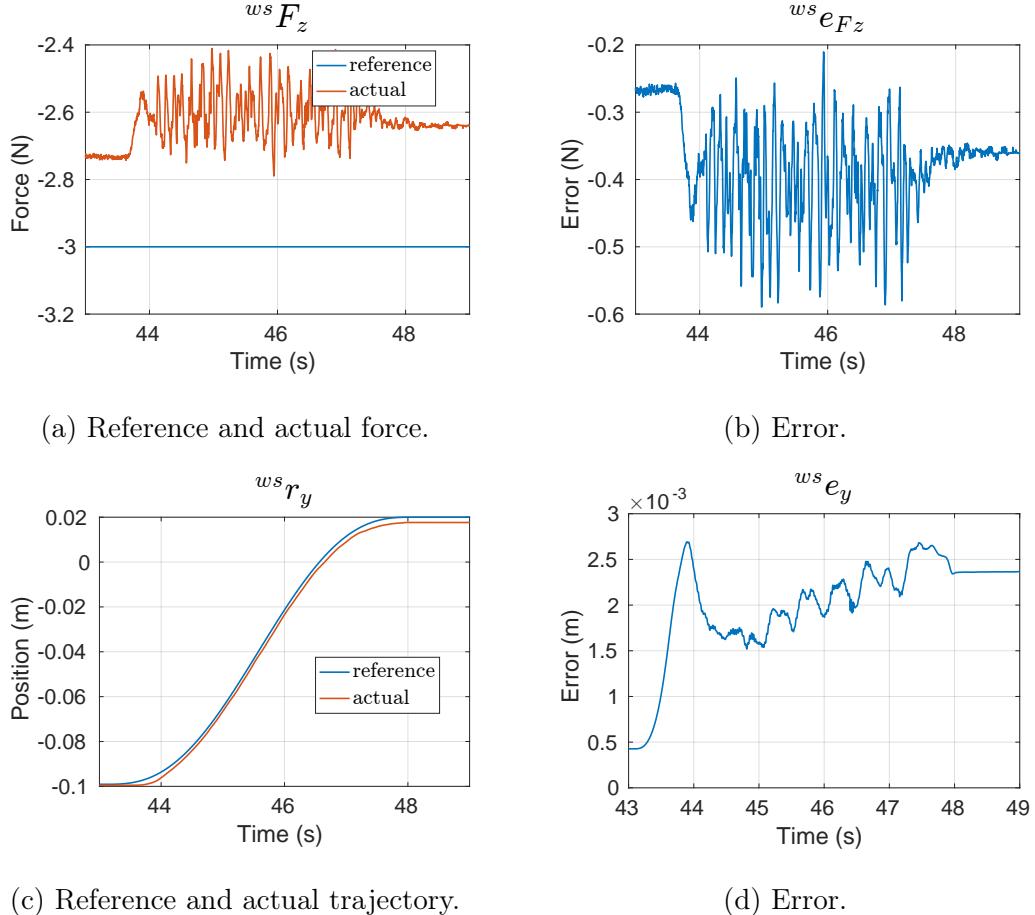


Table 4: Experiment 2, the hand drags a banana.

$$K_f = 2.5 \text{ and } B_f = 85.$$

bounded between -0.6 N and -0.25 N undesired spikes are present because of the vibrations due to the movement.

6.4 Experiment 3

In this experiment (Fig. 12) a ruler is dragged on a table while the contact force between the hand and the ruler is regulated. The object is dragged in



Figure 12: The hand while it drags the ruler.

5 s for a distance of 15 cm and the requested force is about 3 N. In the Tables 5 and 6 only the results of the dragging phase are shown.

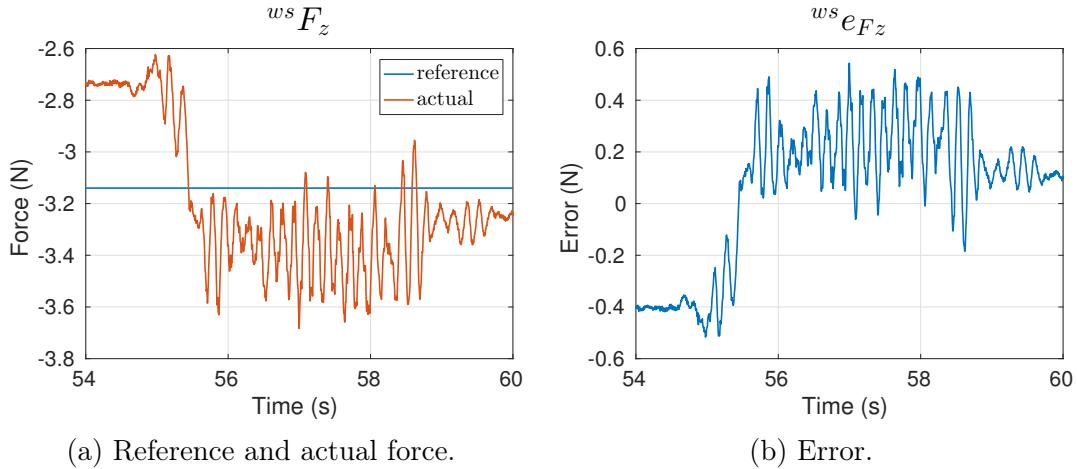


Table 5: Experiment 3, the hand drags a ruler.

$$K_f = 2.5 \text{ and } B_f = 45.$$

Although the force error is bounded between -0.6 N and 0.6 N undesired spikes are present because of the vibrations due to the movement.

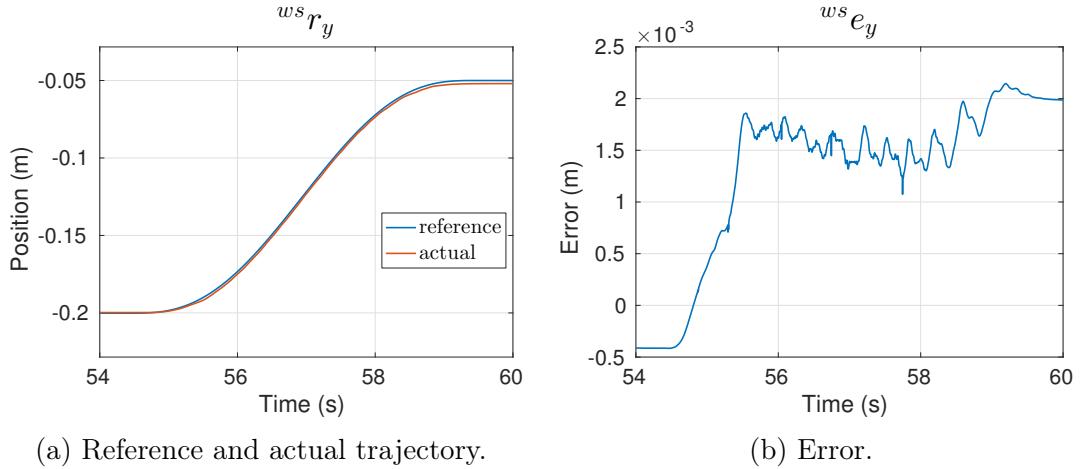


Table 6: Experiment 3, the hand drags a ruler.

$$K_f = 2.5 \text{ and } B_f = 45.$$

After the ruler sticks out of the border of the table the hand is able to grasp it properly.

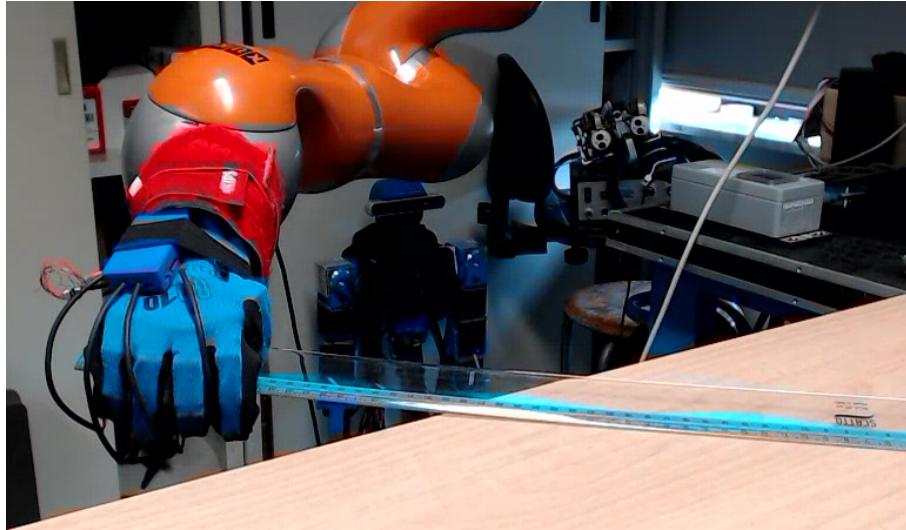


Figure 13: The hand while it grasp the ruler.

References

- [1] M. G. Catalano, G. Grioli, E. Farnioli, A. Serio, C. Piazza, and A. Bicchi, “Adaptive Synergies for the Design and Control of the Pisa / IIT SoftHand,” *International Journal of Robotics Research*, vol. 33, no. 5, pp. 768–782, 2014.
- [2] M. Bonilla, E. Farnioli, C. Piazza, M. Catalano, G. Grioli, M. Garabini, M. Gabiccini, and A. Bicchi, “Grasping with Soft Hands,” in *IEEE-RAS International Conference on Humanoid Robots*, vol. 2015-February, pp. 581–587, 2015.
- [3] R. Anderson and M. Spong, “Hybrid impedance control of robotic manipulators,” *IEEE Journal on Robotics and Automation*, vol. 4, no. 5, pp. 549–556, 1988.
- [4] M. H. Raibert and J. J. Craig, “Hybrid Position/Force Control of Manipulators,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 103, no. 2, p. 126, 1981.
- [5] O. Khatib, “A unified approach for motion and force control of robot manipulators: The operational space formulation,” *Robotics and Automation, IEEE Journal of*, vol. 3, no. 1, pp. 43–53, 1987.
- [6] ATI Industrial Automation, “ATI Industrial Automation: F/T Sensor Mini45,” 2017.