

# Hybrid Force Position Control of a Kuka LWR4

Corso di LM in Ingegneria Robotica ed Automazione  
Controllo dei Robot

---

Students:

Nicola Piga  
Giulio Romualdi

17/05/2017

Università di Pisa

Supervisors:

Prof. Antonio Bicchi  
Ing. Manuel Bonilla

## Project description

---

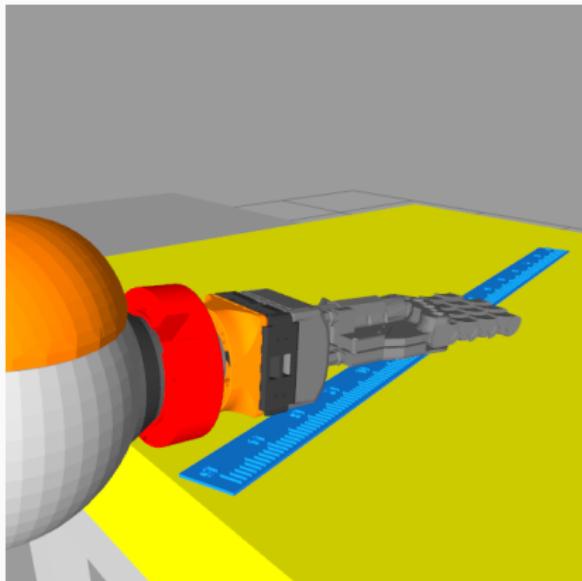
# Grasp of thin objects

## Problem

Performing the grasp of thin objects could be a hard task because not enough contact constraints are provided by the object

# A solution to the problem

Use of environmental constraints to arrange an easier grasp and then perform the grasp



## Example

1. the hand is placed on the object
2. the hand drags the object until it **sticks out of the border** of the table
3. the hand grasps the object

# How to perform a safe dragging phase

The dragging phase could **damage** the object due to uncontrolled contact forces between the hand and the object.

An hybrid position force control strategy allows to avoid this issue.

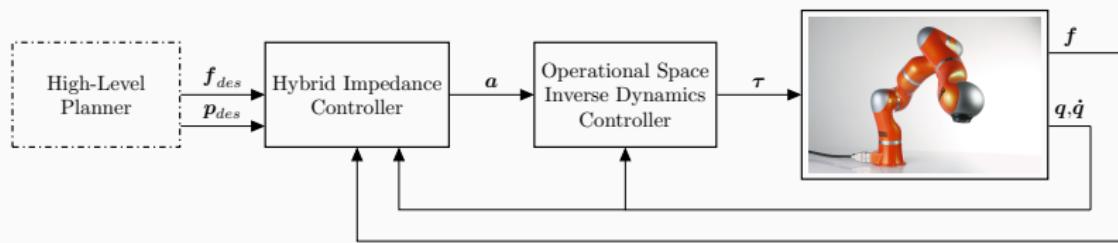
## Aim of this project

Develop an hybrid force position control that allows to move the hand and at the same time to regulate the contact force using a force feedback signal.

# Control architecture

The hybrid force position control was obtained using the **Hybrid Impedance Approach (HIC)** proposed by Anderson and Spong in the 1988.

This approach requires the existence of an **inverse dynamics** control **inner** loop to cancel the nonlinearities of the manipulator



The inner loop applied to the manipulator results in a double integrator system  $\ddot{x} = a$  where  $x$  is a Cartesian description of the end-effector and  $a$  is chosen using the HIC approach.

# Contents

The contents presented are

- the HIC approach
- the development of a HIC based outer control loop
- the inverse dynamics inner loop
- how to correctly use a force torque sensor that is required to close the outer loop
- the results obtained in simulation and in a real scenario
- notes about the software implementation
- future works

## Hybrid impedance control

---

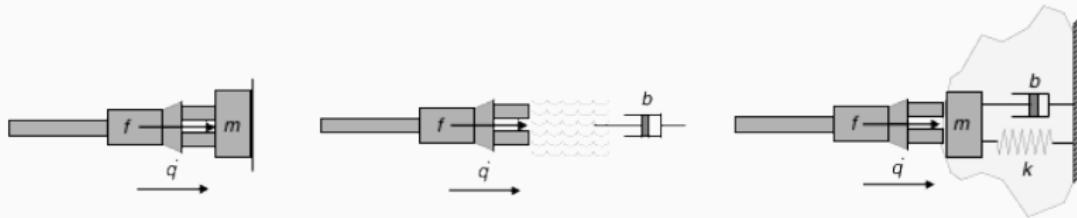
## Hybrid impedance control [Anderson, Spong, 1988]

The Hybrid Impedance Control approach features a more general concept of impedance than that used in typical PD position controllers with tuned apparent impedances of the form

$$(\ddot{\mathbf{x}}^{des} - \ddot{\mathbf{x}}) + K_{damping}(\dot{\mathbf{x}}^{des} - \dot{\mathbf{x}}) + K_{stiffness}(\mathbf{x}^{des} - \mathbf{x}) = \mathbf{F}$$

It allows to synthesize for each DoF of the manipulator both a position or a direct force controller by matching a given “environment impedance” with the appropriate “manipulator impedance”

# HIC - Type of impedances



- for each cartesian DoF the manipulator and the environment can be described using impedances  $Z_m$  and  $Z_e$
- the environment is defined to be any element connected to or contacting the robot anywhere **past the wrist** force sensor
- $Z(\omega) = R(\omega) + jX(\omega)$
- type of impedances
  - inertial iff  $|Z(0)| = 0$
  - resistive iff  $|Z(0)| = c \in (0, \infty)$
  - capacitive iff  $|Z(0)| \rightarrow \infty$

## Duality principle

The manipulator should be controlled to respond as the dual of the environment

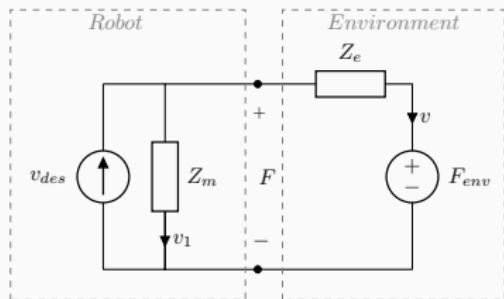
This principle is most easily described in terms of Norton and Thèvenin equivalents

- an inertial environment is represented using a Thèvenin equivalent
- a capacitive environment is represented using a Norton equivalent
- a resistive environment is represented using either a Thèvenin or a Norton equivalent

# HIC - Duality principle for position control

Manipulator impedance chosen as the dual of the environment impedance in order to obtain zero steady state error to a step input

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



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

as long as  $Z_m(0) \neq 0$  and  $Z_e(0) = 0$

## Rule of thumb

inertial environments are position controlled with a noninertial manipulator impedance (the actual value of  $Z_e(0)$  should not be known)

# HIC - Position controlled subsystem

The electrical circuit can be seen as a control feedback scheme where  $a$  is the outer loop acceleration corresponding to the desired velocity  $v_{des}$

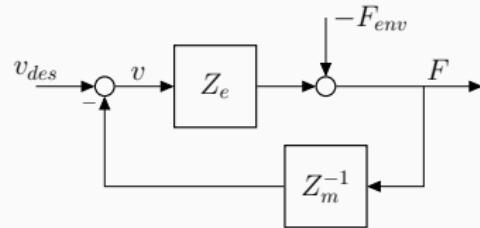
$$v_{des} = v + v_1$$

$$v_1 = \frac{F}{Z_m}$$

$$F = Z_e v + F_{env}$$

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

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



The acceleration can be written without derivatives

$$a = \frac{d}{dt} \left( v_{des} - \frac{F}{Ms + \tilde{Z}_m} \right) = \dot{v}_{des} - \frac{Fs}{Ms + \tilde{Z}_m} = \dot{v}_{des} - sv_1$$

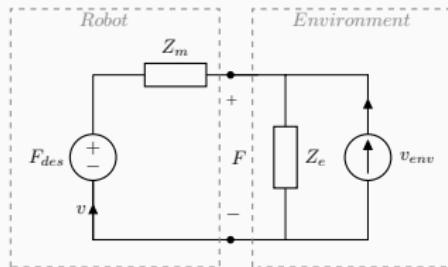
$$F = v_1(Ms + \tilde{Z}_m) \quad v_1 = \frac{F - (v_{des} - v)\tilde{Z}_m}{Ms}$$

$$a = \dot{v}_{des} - sv_1 = \dot{v}_{des} - s \left( \frac{F - (v_{des} - v)\tilde{Z}_m}{Ms} \right) = \dot{v}_{des} + \frac{(v_{des} - v)\tilde{Z}_m}{M} - \frac{F}{M}$$

# HIC - Duality principle for force control

Manipulator impedance chosen as the dual of the environment impedance in order to obtain zero steady state error to a step input

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



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

as long as  $Z_m(0) < \infty$  and  $Z_e(0) \rightarrow \infty$

## Rule of thumb

capacitive environments are force controlled with a noncapacitive manipulator impedance (the actual value of  $Z_e(0)$  should not be known)

# HIC - Force controlled subsystem

The electrical circuit can be seen as a control feedback scheme where  $a$  is the outer loop acceleration of the DoF corresponding to the desired force  $F_{des}$

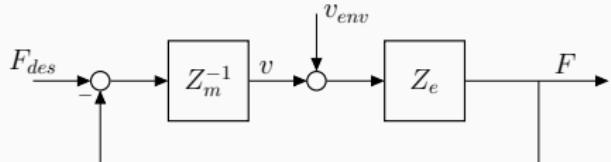
$$F = F_{des} + Z_m v$$

$$v = \frac{F - F_{des}}{Z_m}$$

$$F = Z_e(v + v_{env})$$

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

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



The acceleration can be written without derivatives

$$a = \frac{d}{dt} \left( \frac{F - F_{des}}{Ms + \tilde{Z}_m} \right) = \left( \frac{s(F - F_{des})}{Ms + \tilde{Z}_m} \right)$$

$$vMs + v\tilde{Z}_m = F - F_{des}$$

$$v = \frac{1}{Ms}(F - F_{des} - v\tilde{Z}_m)$$

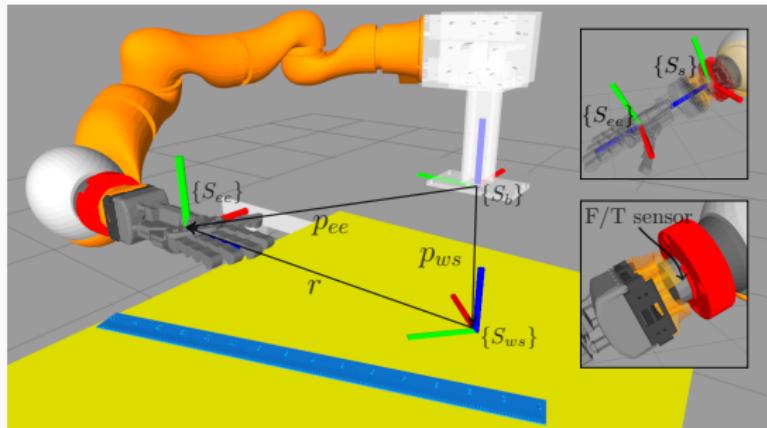
$$a = \dot{v} = \frac{\$}{Ms}(F - F_{des}) - \frac{\$}{Ms}(\tilde{Z}_m v) = \frac{1}{M}(F - F_{des}) - \frac{1}{M}(\tilde{Z}_m v)$$

## **HIC based control architecture**

---

# References frames

Before delving into an HIC based controller let us introduce some useful reference frames and notation



$b$ : base

$s$ : sensor

$ee$ : end-effector  
(hand)

$ws$ : work-space  
(table)

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

$$\{S_s\} = \{S; x_s, y_s, z_s\}$$

$$\{S_{ee}\} = \{E; x_{ee}, y_{ee}, z_{ee}\}$$

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

# Notation

## Space state vector definition

A natural choice for a basis in which the commanded positions and forces are **expressed in ws**

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

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

## Convention

Convention used for any quantity  $X$  encountered,  ${}^bX_p$

- ${}^b.$  reference frame
- $.p$  reference point (for wrench and Jacobian only)

## Control aims

Suppose that the second derivative of the state can be chosen arbitrarily

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

Find  $\mathbf{a}_{cmd}$  using HIC such that

- $r_x$ ,  $r_y$  and  $\Phi$  are rigidly controlled i.e.

$$\text{i. } \ddot{e}_x + B_x \dot{e}_x + K_x e_x = 0 \quad e_x(t) = {}^{ws}r_{x,des}(t) - {}^{ws}r_x(t)$$

$$\text{ii. } \ddot{e}_y + B_y \dot{e}_y + K_y e_y = 0 \quad e_y(t) = {}^{ws}r_{y,des}(t) - {}^{ws}r_y(t)$$

$$\text{iii. } \ddot{e}_\Phi + B_\Phi \dot{e}_\Phi + K_\Phi e_\Phi = 0 \quad e_\Phi(t) = \Phi_{des}(t) - \Phi(t)$$

- the DoF along  $z_{ws}$  is force controlled i.e.

$$\text{i. } e_z \xrightarrow[t \rightarrow \infty]{} 0 \quad e_z(t) = {}^{ws}F_{z,des}(t) - {}^{ws}F_z(t)$$

where  ${}^{ws}F_z(t)$  is the force exerted by the hand to the object expressed in  $ws$

# Resulting HIC based controller

## Position ( ${}^{ws}r_x, {}^{ws}r_y$ )

- inertial environment supposed, i.e., manipulator moving a payload along given axis
- $Z_{m,p} = M_p s + \tilde{Z}_{m,p} = M_p s + B_p + \frac{K_p}{s} = s + B_p + \frac{K_p}{s}$

$${}^{ws}a_{cmd,x} = \ddot{r}_{x,des} + B_x(\dot{r}_{x,des} - \dot{r}_x) + K_x(r_{x,des} - r_x) - F_x$$

$${}^{ws}a_{cmd,y} = \ddot{r}_{y,des} + B_y(\dot{r}_{y,des} - \dot{r}_y) + K_y(r_{y,des} - r_y) - F_y$$

## Attitude ( $\psi, \theta, \phi$ )

- inertial environment supposed, i.e., manipulator rotating a payload about given axis
- $Z_{m,a} = M_a s + \tilde{Z}_{m,a} = M_a s + B_a + \frac{K_a}{s} = s + B_a + \frac{K_a}{s}$

$${}^{ws}a_{cmd,\Phi} = \ddot{\Phi}_{des} + B_\Phi(\dot{\Phi}_{des} - \dot{\Phi}) + K_\Phi(\Phi_{des} - \Phi)$$

## Force ( ${}^{ws}F_z$ )

- capacitive environment supposed
- $Z_{m,f} = M_f s + \tilde{Z}_{m,f} = M_f s + \tilde{B}_f = \frac{s}{K_f} + \frac{B_f}{K_f}$

$${}^{ws}a_{cmd,z} = -B_f \dot{r}_z + K_f(F_{z,des} - F_z)$$

## **Inner loop inverse dynamics**

---

## Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

## Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

Joint Space description

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

## Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

Joint Space description

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

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

## Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

Joint Space description

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

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

HIC requires operational space description

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

# Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

Joint Space description

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

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

HIC requires operational space description

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

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

$$\det(T) = -\sin(\theta) \neq 0$$

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

# Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

Joint Space description

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

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

HIC requires operational space description

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

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

$$\det(T) = -\sin(\theta) \neq 0$$

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

Substitute  $\ddot{\mathbf{q}}$  in the operational space description

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

# Inverse dynamics controller

In order to assign the second derivative of the state  $\dot{\boldsymbol{x}}$  arbitrarily an inverse dynamics controller is required

$$B(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{\tau} - {}^b J_S^T {}^b \boldsymbol{w}_S \longrightarrow {}^{ws} \ddot{\boldsymbol{x}} = \boldsymbol{a}$$

Joint Space description

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

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

HIC requires operational space description

$${}^{ws} \ddot{\boldsymbol{x}} = {}^{ws} J_{A,E} \ddot{\boldsymbol{q}} + {}^{ws} \dot{J}_{A,E} \dot{\boldsymbol{q}}$$

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

$$\det(T) = -\sin(\theta) \neq 0$$

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

Substitute  $\ddot{\boldsymbol{q}}$  in the operational space description

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

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

# Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

Joint Space description

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

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

HIC requires operational space description

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

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

$$\det(T) = -\sin(\theta) \neq 0$$

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

Substitute  $\ddot{\mathbf{q}}$  in the operational space description

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

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

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

# Inverse dynamics controller

In order to assign the second derivative of the state  $\mathbf{x}$  arbitrarily an inverse dynamics controller is required

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

Joint Space description

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

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

HIC requires operational space description

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

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

$$\det(T) = -\sin(\theta) \neq 0$$

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

Substitute  $\ddot{\mathbf{q}}$  in the operational space description

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

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

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

$$\boldsymbol{\tau} = C\dot{\mathbf{q}} + \mathbf{G} + {}^b J_S^T(\Lambda_A \mathbf{a}_{cmd} - \Lambda_A {}^{ws} \dot{J}_{A,E} \dot{\mathbf{q}} + {}^b \mathbf{w}_S) \implies {}^{ws} \ddot{\mathbf{x}} = \mathbf{a}$$

## Issues with internal motions

In the case of the task described the Kuka LWR 4+ is a kinematically redundant manipulator.

Extensive simulations revealed that the uncontrolled internal motions of the robot, while not affecting the desired attitude of the hand, cause 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.

These issues are solved using a dynamically consistent generalized inverse of the Jacobian [Khatib, 1987]

## Control of internal motions [Khatib, 1987]

The standard operational space dynamics can be written by substituting (1) in (2) resulting in (3)

$$\begin{aligned} B(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) &= \boldsymbol{\tau} = J^T \boldsymbol{\gamma} & (1) \quad \Lambda &= (JB^{-1}J^T)^{-1} \\ \ddot{\boldsymbol{x}} &= J(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) & (2) \quad \mu &= \bar{J}^T C \dot{\boldsymbol{q}} - \Lambda \boldsymbol{h} \\ \Lambda \ddot{\boldsymbol{x}} + \mu(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{p} &= \boldsymbol{\gamma} & (3) \quad \boldsymbol{p} &= \bar{J}^T \boldsymbol{G} \\ & & & \bar{J} = B^{-1}J^T\Lambda \end{aligned}$$

The equation (3) can also be written in the form

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

resulting in

$$\boldsymbol{\gamma} = \bar{J}^T \boldsymbol{\tau}$$

i.e.  $\bar{J} = B^{-1}J^T\Lambda$  is a dynamically consistent generalized inverse of the Jacobian matrix

## Control of internal motions [Khatib, 1987]

In order to control the undesired internal motions a command torque  $\tau$  of the form

$$\tau = J^T \gamma + (I_7 - J^T \bar{J}^T) \gamma_0$$

can be used where  $\gamma_0$  is projected in the null space of  $\bar{J}^T$  hence not affecting the command wrench seen by the hand

The final control law is

$$\tau = C\dot{\mathbf{q}} + \mathbf{G} + {}^b J_S^T (\Lambda_A \mathbf{a}_{cmd} - \Lambda_A^{ws} \dot{J}_{A,E} \dot{\mathbf{q}} + {}^b \mathbf{w}_S) + (I_7 - ({}^b J_S^T)({}^b \bar{J}_S^T)) \gamma_0$$

where

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

## Design of the null operational wrench command $\gamma_0$

To fix the orientation of the 5-th link and to regulate the altitude between the 4-th link and the hand  $\gamma_0$  is chosen

$$\gamma_0 = J_{im}^T(K_p(\mathbf{x}_{im,des} - \mathbf{x}_{im}) - K_d \dot{\mathbf{x}}_{im})$$

$$\mathbf{x}_{im} = \begin{bmatrix} {}^b x_{I4} & {}^b y_{I4} & {}^b z_{I4} & \psi_{I5} & \theta_{I5} & \phi_{I5} \end{bmatrix}^T = \begin{bmatrix} \mathbf{p}_{I4}^T & \Phi_{I5}^T \end{bmatrix}^T$$

$$\begin{bmatrix} \dot{\mathbf{p}}_{I4} \\ \dot{\Phi}_{I5} \end{bmatrix} = J_{im} \dot{\mathbf{q}}$$

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

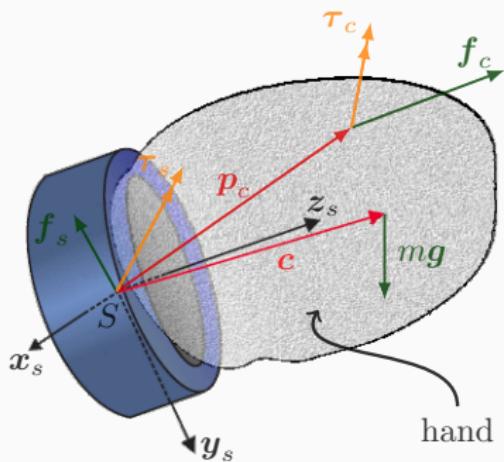
$$\mathbf{p}_{I4} = \begin{bmatrix} * & * & {}^b p_{ee_z} + off_z \end{bmatrix}^T \quad \Phi_{I5} = \begin{bmatrix} * & * & 0 \end{bmatrix}^T$$

**Measure of forces/torques  
exchanged with the object**

---

## Force/torque sensor with hand attached

The wrench  $\mathbf{w}_S$  exerted by the hand on the object is required to implement the controllers properly



$-\mathbf{f}_s, -\boldsymbol{\tau}_s$  := measured forces

$-\mathbf{f}_c, -\boldsymbol{\tau}_c$  := contact forces

$\mathbf{f}_{pl}, \boldsymbol{\tau}_{pl}$  := due to mounting plate <sup>1</sup>

$\mathbf{c}$  := CoM of the hand

$\mathbf{p}_c$  := vector from \$S\$ to contact point

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

<sup>1</sup>Not shown in the picture

## Newton-Euler equations for a rigid body attached to the sensor

The Newton-Euler equations show how to obtain  $\mathbf{w}_S$  from  $\mathbf{f}_s$  and  $\boldsymbol{\tau}_s$

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

$$\boldsymbol{\tau}_s = -\boldsymbol{\tau}_{pl} - \boldsymbol{\tau}_c - \tilde{\mathbf{p}}_c \mathbf{f}_c + \tilde{\mathbf{g}} m \mathbf{c} - \tilde{\mathbf{a}}_{cm} m \mathbf{c}^2$$

---

<sup>2</sup>Angular velocities and accelerations neglected because the f/t sensor is primarily used when the hand is still or when it moves along straight lines

# Software induced offset

The signal produced by the f/t sensor is

$$\mathbf{f}_{meas} = -\mathbf{f}_s + \mathbf{f}_{sw,off}$$

$$\boldsymbol{\tau}_{meas} = -\boldsymbol{\tau}_s + \boldsymbol{\tau}_{sw,off}$$

Offsets  $\mathbf{f}_{sw,off}$ ,  $\boldsymbol{\tau}_{sw,off}$  are set when the f/t sensor is calibrated such that

$$\mathbf{f}_{m,0} = \boldsymbol{\tau}_{m,0} = \mathbf{0}^3$$

$$\mathbf{f}_{sw,off} = \mathbf{f}_{s,0} = -\boldsymbol{\tau}_{pl} - m^s \mathbf{g}_0$$

$$\boldsymbol{\tau}_{sw,off} = \boldsymbol{\tau}_{s,0} = -\boldsymbol{\tau}_{pl} + {}^s \tilde{\mathbf{g}}_0 \mathbf{mc}$$

The resulting signal is

$$\mathbf{f}_{meas} = \mathbf{f}_c + m\mathbf{g} - m^s \mathbf{g}_0 - m\mathbf{a}_{cm}$$

$$\boldsymbol{\tau}_{meas} = \boldsymbol{\tau}_c + \tilde{\mathbf{p}}_c \mathbf{f}_c - \tilde{\mathbf{g}} \mathbf{mc} + {}^s \tilde{\mathbf{g}}_0 \mathbf{mc} + \tilde{\mathbf{a}}_{cm} \mathbf{mc}$$

---

<sup>3</sup>The zero subscript represents the calibration condition performed when the manipulator is still

# Estimation of unknowns quantities

In order to obtain  $\mathbf{f}_c$  and  $\boldsymbol{\tau}_c$  the unknowns are estimated

$$\mathbf{f}_{meas} = \mathbf{f}_c + m\mathbf{g} + (-m^s\mathbf{g}_0) - ma_{cm}$$

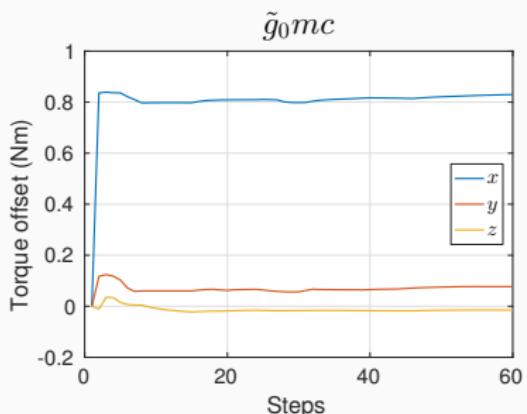
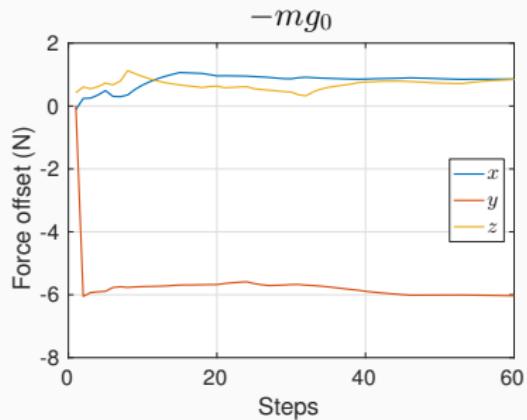
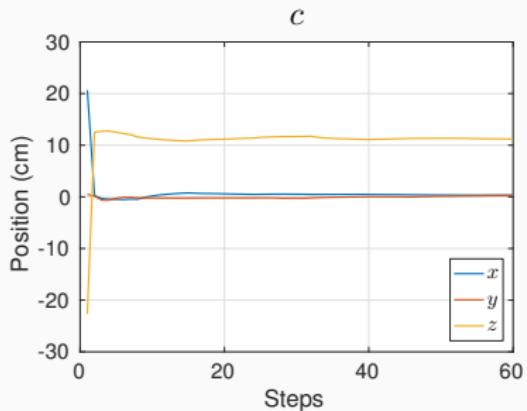
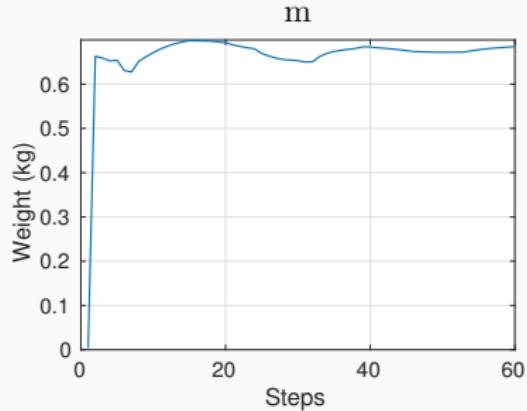
$$\boldsymbol{\tau}_{meas} = \boldsymbol{\tau}_c + \tilde{\mathbf{p}}_c \mathbf{f}_c - \tilde{\mathbf{g}} \mathbf{mc} + {}^s\tilde{\mathbf{g}}_0 \mathbf{mc} + \tilde{\mathbf{a}}_{cm} \mathbf{mc}$$

The estimation is obtained from  $n$  measurements collected when the manipulator assumes a static pose

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

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

# Estimation of unknowns quantities - An example



## Approaching phase

---

## Approaching phase

Before using HIC the robot is moved in a given position above the surface of the table using a joint space inverse dynamics **to avoid singularity** introduced by Euler ZYZ parametrization

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

$$\boldsymbol{\tau} = C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + B(\mathbf{q})\mathbf{a}_{p2p}$$

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

$$\mathbf{q}_f = F\mathbf{K}^{-1}(\mathbf{x}_{des})$$

where  $\mathbf{q}_f$  is **discarded whenever its component exceeds the joints limits**

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

$$q_{des,i}(0) = q_{prev,i} \quad \dot{q}_{des,i}(0) = 0 \quad \ddot{q}_{des,i}(0) = 0$$

$$q_{des,i}(t_f) = q_{f,i} \quad \dot{q}_{des,i}(t_f) = 0 \quad \ddot{q}_{des,i}(t_f) = 0$$

## Results of simulation

---

## Simulation setup

- emulated Kuka FRI Joint specific impedance control mode

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

- $K_j = 0$
- $\mathbf{q}_{FRI} = \mathbf{q}_{msr}$
- $d_j = 0$
- $\mathbf{f}_{dynamics}$  supposed to be  $\mathbf{G}$  evaluated using KDL library <sup>4</sup>
- $\boldsymbol{\tau}_{FRI}$  used as commanded torque

---

<sup>4</sup>The Kinematics and Dynamics Library (KDL) develops an application independent framework for modelling and computation of kinematic chains

## Simulation setup (continued)

$$\tau_{FRI} = C\dot{\boldsymbol{q}} + \mathcal{G} + B\mathbf{a}_{p2p}$$

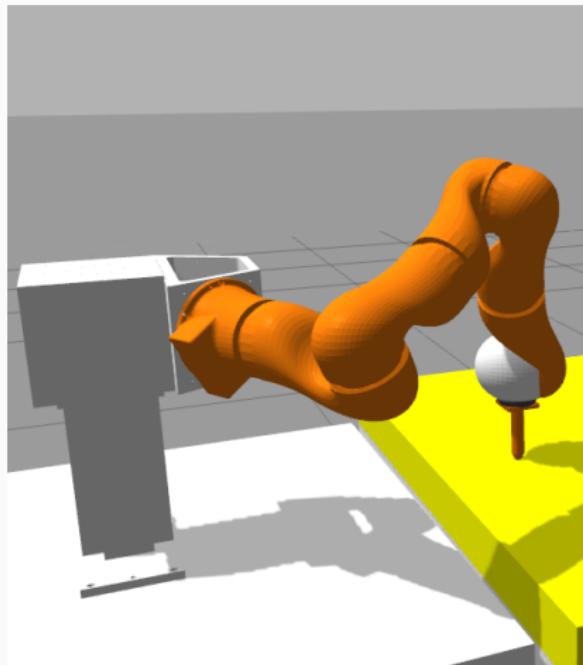
$$\tau_{FRI} = C\dot{\boldsymbol{q}} + \mathcal{G} + {}^bJ_S^T(B_A\mathbf{a}_{cmd} - B_A{}^{ws}J_{A,E}\dot{\boldsymbol{q}} + {}^b\mathbf{w}_S) + (I_7 - ({}^bJ_S^T)({}^bJ_S^T))\gamma_0$$

- controllers running @ 1kHz
- $G$  canceled because the gravity is compensated by the robot
- $B$  given by KDL
- $C\dot{\boldsymbol{q}}$  given by KDL
- only  $\boldsymbol{q}$  is available (as in the real scenario)
  - $\dot{\boldsymbol{q}}$  estimated using an exponential smoothing  $\dot{\boldsymbol{q}}_k = (1 - \alpha)\dot{\boldsymbol{q}}_{k-1} + \alpha \frac{\boldsymbol{q}_k - \boldsymbol{q}_{k-1}}{t_s}$
- Jacobians are evaluated using KDL library
- $\mathbf{w}_F$  given by force/torque sensor software plugin
  - corrupted signal when the end-effector is mounted on the sensor → massless tool
- joints friction and contact friction neglected for ease of simulation
- generic end-effector instead of the hand for ease of simulation

## Description of the simulated scenarios

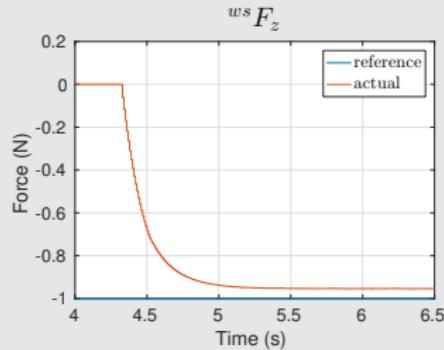
The following scenarios were simulated in the Gazebo robot simulator

- contact phase
- force regulation
- simultaneous force regulation and dragging

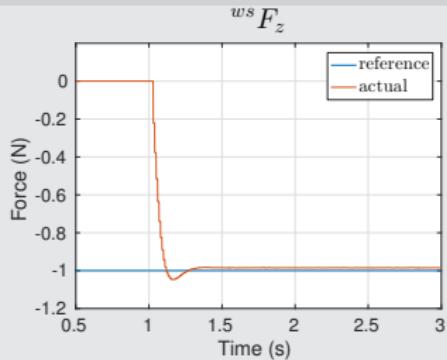


# Results - Force regulation

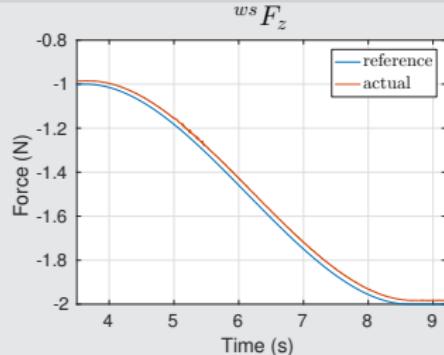
## Contact phase



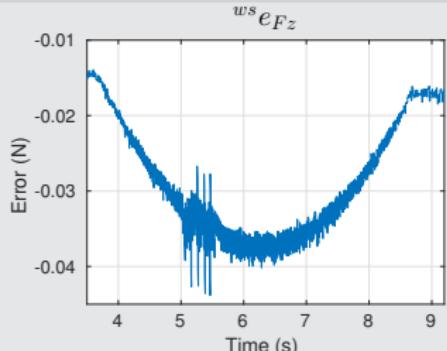
$$K_f \in \{2, 3\}, K_d = 25$$



## Force setpoint

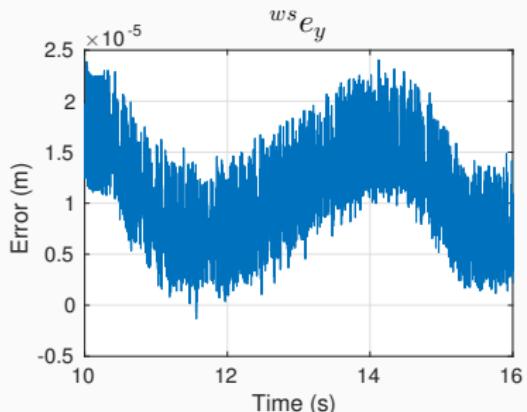
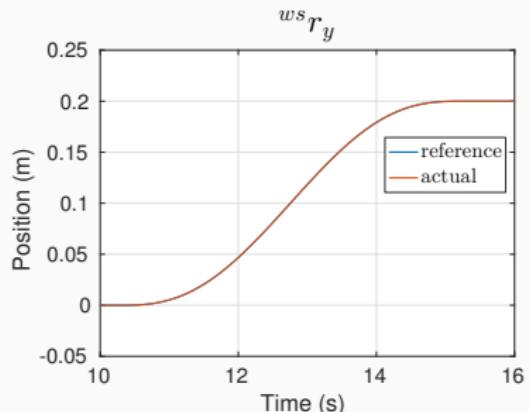
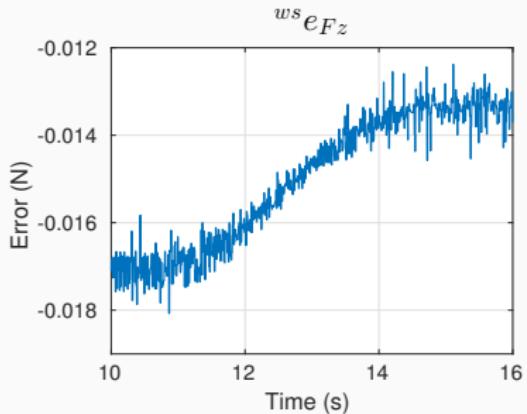
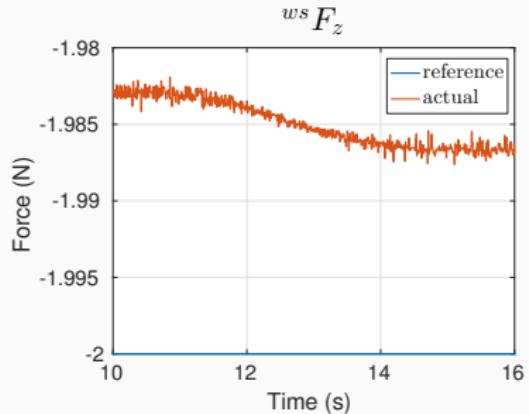


$$K_f = 3, B_f = 45$$



## Results - Dragging and Force regulation

( $K_f = 3, B_f = 45$ )



## Results of experiments

---

## Experimental setup

- Kuka FRI Joint specific impedance control mode

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

- $K_j = 0$
- $\mathbf{q}_{FRI} = \mathbf{q}_{msr}$
- $d_j = 0$
- $\mathbf{f}_{dynamics}$  supposed to be  $\mathbf{G}$
- $\tau_{FRI}$  used as commanded torque

## Experimental setup (continued)

$$\tau_{FRI} = C\dot{\boldsymbol{q}} + \cancel{G} + B\boldsymbol{a}_{p2p}$$

$$\tau_{FRI} = C\dot{\boldsymbol{q}} + \cancel{G} + {}^bJ_S^T(B_A\boldsymbol{a}_{cmd} - B_A{}^{ws}\dot{J}_{A,E}\dot{\boldsymbol{q}} + {}^b\boldsymbol{w}_S) + (I_7 - ({}^bJ_S^T)({}^b\bar{J}_S^T))\gamma_0$$

- controllers running @ 300Hz
- $G$  canceled because the gravity is compensated by the robot
- $B$  given by KDL
- $C\dot{\boldsymbol{q}}$  given by KDL
- only  $\boldsymbol{q}$  is available (as in the real scenario)
  - $\dot{\boldsymbol{q}}$  estimated using an exponential smoothing  $\dot{\boldsymbol{q}}_k = (1 - \alpha)\dot{\boldsymbol{q}}_{k-1} + \alpha \frac{\boldsymbol{q}_k - \boldsymbol{q}_{k-1}}{t_s}$
- Jacobians are evaluated using KDL library
- $\boldsymbol{w}_S$  given by force/torque sensor

## Experiment n.1 - Force regulation

## Experiment n.2 - Force regulation

## Experiment n.3 - Dragging with force regulation and grasp

## **Software implementation**

---

## Some notes about implementation

- ROS based software
- already existing (Centro Piaggio) KUKA LWR4+ software stack (model, FRI, F/T sensor interface, ROS controllers)
- built upon KinematicChainControllerBase (KDL facilities)
- extends KinematicChainControllerBase by providing an Operational Space Inverse Dynamics *abstract* controller
  - arbitrary commanded acceleration in operational space
  - state (with derivatives) ready to use and projected in ws
  - jacobian extended to take into account the length of the tool
  - null projection available
- Hybrid Impedance Controller based on Inverse Dynamics Controller

## Future works

---

## Combine HIC with a High-Level planner

A natural extension of this work would be to combine the HIC controller with a high-level planner that provide the reference forces and positions to perform complex tasks where the interaction with the environment is required.

## Appendix: Kinesthetic Filtering

---

## How to write $\mathbf{a}_{cmd}$ using a selection matrix $S$

A selection matrix  $S$  **may** be used to separate the force-controlled and position-controlled *reciprocal* subspaces

$${}^{ws}\mathbf{a}_{cmd} = S \begin{bmatrix} a_{cmd,x} \\ a_{cmd,y} \\ * \\ \mathbf{a}_{cmd,\Phi} \end{bmatrix} + (I - S) \begin{bmatrix} * \\ * \\ a_{cmd,z} \\ * \\ * \\ * \end{bmatrix} \quad S = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

## A different interpretation of the selection matrix

Position controlled and force controlled subspaces can be seen in terms of *natural* and *artificial* constraints:

- natural: directions along which the end effector can not move and exert forces
- artificial: directions along which the end effector can move and exert forces

Artificial constraints directions belong to the span of appropriate matrices

## An example

Consider admissible twists and wrenches projected in  $ws$

$$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad a = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$
$$\overset{ws}{\xi}_{adm} \in \mathcal{R}(B)$$
$$\overset{ws}{w}_{adm} \in \mathcal{R}(a)$$

The most general motion is that of a translation ( $x$ - $y$  direction of  $ws$ ) and/or a rotation about some axis that pass through a point  $P$ , e.g. the center of the palm of the hand.

Force can only be exerted along the  $z$  direction of  $ws$ . No torques are allowed.

## An example (continued)

In order to filter out undesired twists and wrenches the following projection matrices are used (Kinesthetic filter)

$$P_B = B(B^T B)^{-1}B^T$$

$$P_a = a(a^T a)^{-1}a^T$$

$$P_B \mathbf{e}_i = \mathbf{e}_i \quad i \in \{1, 2, 4, 5, 6\}$$

$$P_a \mathbf{e}_i = \mathbf{0} \quad i \in \{1, 2, 4, 5, 6\}$$

$$P_B \mathbf{e}_3 = \mathbf{0}$$

$$P_a \mathbf{e}_3 = \mathbf{e}_3$$

The projectors lead to the already seen selection matrices  $S = P_B$  and  $I - S = P_a$

## A more general example

In general twists/wrenches can be commanded with respect to a different reference point  $E'$  for example  $EE' = [\bar{x} \ 0 \ 0]^T$

$${}^{ws}\xi' = K(\bar{x})({}^{ws}\xi)$$

$$B' = KB = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & x \\ 0 & 0 & 0 & -x & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{ws}\mathbf{w}' = G(\bar{x})({}^{ws}\mathbf{w})$$

$$a' = Ga = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ x \\ 0 \end{bmatrix}$$

However using the standard projector gives for example

$$P'_B K e_3 = [0 \ 0 \ \frac{x^2}{x^2+1} \ 0 \ -\frac{x}{x^2+1} \ 0]^T$$

$$P'_a G e_5 = [0 \ 0 \ \frac{x}{x^2+1} \ 0 \ \frac{x^2}{x^2+1} \ 0]^T$$

# An invariant filter [Marescotti, Bonivento and Melchiorri, 1990]

An invariant filter behaves well even when the reference point changes

$$P''_B = K P_B K^{-1}$$

If  $P_B({}^{ws}\xi) = {}^{ws}\xi$

$$\begin{aligned} \text{then } P''_B({}^{ws}\xi') &= P''_B K({}^{ws}\xi) \\ &= K P_B K^{-1} K({}^{ws}\xi) \\ &= K P_B({}^{ws}\xi) \\ &= K({}^{ws}\xi) = {}^{ws}\xi' \end{aligned}$$

If  $P_B({}^{ws}\xi) = \mathbf{0}$

$$\begin{aligned} \text{then } P''_B({}^{ws}\xi') &= K P_B({}^{ws}\xi) \\ &= \mathbf{0} \end{aligned}$$

$$P''_a = G P_a G^{-1}$$

If  $P_a({}^{ws}\mathbf{w}) = {}^{ws}\mathbf{w}$

$$\begin{aligned} \text{then } P''_a({}^{ws}\mathbf{w}') &= P''_a G({}^{ws}\mathbf{w}) \\ &= G P_a G^{-1} G({}^{ws}\mathbf{w}) \\ &= G P_a({}^{ws}\mathbf{w}) \\ &= G({}^{ws}\mathbf{w}) = {}^{ws}\mathbf{w}' \end{aligned}$$

If  $P_a({}^{ws}\mathbf{w}) = \mathbf{0}$

$$\begin{aligned} \text{then } P''_a({}^{ws}\mathbf{w}') &= G P_a({}^{ws}\mathbf{w}) \\ &= \mathbf{0} \end{aligned}$$

# Kinesthetic filtering

Since the kinesthetic filtering is applied to twists and wrenches it is not possible to apply it directly to the commanded accelerations described above.

Anyway, as observed in [Marescotti, Bonivento and Melchiorri, 1990], there is no necessity to use the alternative projectors  $P_B''$  and  $P_a''$  in the control algorithm since twists and wrenches expressed with respect to another reference point can always be transformed back to their original form using the inverse transformations  $K^{-1}$  and  $G^{-1}$ .

In that case the projection matrices always reduce to the selection matrices  $S$  and  $I - S$  that are also suitable to filter undesired commanded accelerations.

