



**HAL**  
open science

# Safe Control of Robotic Manipulators in Dynamic Contexts

Anis Meguenani

► **To cite this version:**

Anis Meguenani. Safe Control of Robotic Manipulators in Dynamic Contexts. Robotics [cs.RO]. Université Pierre & Marie Curie - Paris 6; Institut des Systèmes Intelligents et de Robotique, 2017. English. NNT: . tel-01762465v1

**HAL Id: tel-01762465**

**<https://hal.science/tel-01762465v1>**

Submitted on 20 Oct 2017 (v1), last revised 10 Apr 2018 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A dissertation submitted in partial fulfilment of the requirements for the degree of

# Doctor of Philosophy

*of the*

**Pierre & Marie Curie University**

*Sorbonnes Universités, Université Pierre & Marie Curie Paris VI*

*Doctoral school of Mechanical, Acoustic, Electronic and Robotic Sciences of Paris  
École Doctorale de Sciences Mécanique, Acoustique, Électronique et Robotique de Paris*

*in*

**Robotics**

presented by

**Anis Meguenani**

---

## Safe Control of Robotic Manipulators in Dynamic Contexts

---

*Defended on the 13<sup>th</sup> of October, 2017*

### Committee

M. Olivier STASSE	Senior Researcher at LAAS-CNRS, Toulouse	Referee
M. Sami HADDADIN	Research director at Leibniz Universität, Hannover	Referee
M. Guillaume MOREL	Professor at Université Pierre & Marie Curie	Examiner
M. Andrea DEL PRETE	Post Doc at LAAS-CNRS, Toulouse	Examiner
M. Yvan MEASSON	CEO at ISYBOT, Gif-Sur-Yvette	Guest member
M. Philippe BIDAUD	Professor at Université Pierre & Marie Curie	Adviser
M. Vincent PADOIS	Associate professor at Université Pierre & Marie Curie	Adviser



# Acknowledgements

---

Foremost, I would like to express my sincere gratitude to my supervisors Prof. Philippe Bidaud and Prof. Vincent Padois for giving me the opportunity to be part of the *Institut des systèmes intelligents et de robotique* (ISIR), for their continuous support during my Ph.D study and research, for their patience, motivation, enthusiasm, trust and immense knowledge. Their guidance made it possible to achieve the goals fixed for my thesis and also to write this report.

Additionally, i would like to thank my committee members Dr. Olivier Stasse, Prof. Sami Haddadin, Prof. Guillaume Morel, Dr. Andrea Del Prete and Dr. Yvan Measson for their interest in my work.

I am also thankful to my colleagues Zhanat Kappassov, Ziad Zamzami, Mohammed Fnadi, Pauline Maurice, Ryan Lober, Mingxing Liu, Darwin Lau, Aurlien Ibanez, Philipp Schlehuner and Jorhabib Eljaik Gomez for their friendship, support, and for creating a cordial working environment. I would like also to thank Sovannara Hak, Antoine Hoarau and Jimmy Da Silva for their valuable help and technical assistance for the experimentations performed on the KUKA LWR4 robot.

Finally, in the pursuit of this Ph.D, the members of my family have been of great importance to me. More specifically, i would like to thank my mother Assia Meguenani and father Saadoun Meguenani for their unconditional love and moral and emotional support throughout my life and for making of me the person i am today.





# Abstract

---

The intended goal of this thesis is to bring new insights for developing robotic systems capable of safely sharing their workspace with human-operators. Within this context, the presented work focuses on the control problem. The following questions are tackled:

- for reactive control laws, i.e., control problems where the task to be performed is not known in advance but discovered *on-line*, how is it possible to guarantee for every time-step the existence of a solution to the control problem? This solution should allow the robot to accomplish at best its prescribed task and at the same time to strictly comply with existing constraints, among which, constraints related to the physical limitations of its actuators and joints.
- How to integrate the human-operator in the control loop of the robot so that physical contact can safely be engaged and de-engaged?

Regarding the first point, our work arises as the continuity of previous results developed by Sébastien Rubrecht during his PhD thesis. Sébastien Rubrecht introduced the concept of *constraints incompatibility* for robots reactively controlled at the kinematic-level. The problem of *constraints incompatibility* appears for example when the formulation of the constraint on an articular position of a robot does not account for the amount of deceleration producible by its actuator. In such case, if the articular position constraint is activated tardively, the system may not have sufficient time to cope with the imposed joint position limit considering its bounded dynamic capabilities.

In our work, the problem of *constraints incompatibility* is tackled for robots controlled at the dynamic-level (torque control). New mathematical formulations for the *articular position* and *joint velocity* constraints that account for both the deceleration and jerk capabilities of the actuators of the robot are introduced. The *optimal* time needed for the system to cope with the different limits of these constraints is computed. When included in a QP formulation of the control problem, the introduced new expressions trigger the activation of the constraints at the right time considering the dynamic capabilities of the robot. The existence of a constraint compliant and optimal solution as joint position, velocity, torque/deceleration and also jerk limits are all respected at the same time and at every time-step is guaranteed. Consequently the *viability* of the state of the robot is also ensured.

Regarding the second point, for a given shape of the robot, two main parameters are usually considered for safety: the impact force generated at collision and the contact force generated after the establishment of physical contact. The most generic way to consider and express these parameters is to use an energetic formulation of the safety problem. Indeed, energy is a universal quantity that can describe all the safety related physical phenomena occurring during human-robot interaction. An impact force for example is directly proportional to the amount of kinetic energy dissipated at collision. The contact force on the other hand derives from the amount of potential energy that accumulates in the controller of the robot during physical contact. In our work, safety during human-robot interaction is handled by modulating the amount of energy instantaneously displayed by the robot. The main contributions are as follows:

- the formulation of kinetic and potential energy based safety indicators that represent the degree of danger of the robot at collision and during physical contact.

- The formulation of energy based safety criteria that bound the dynamics of the robot during different interaction phases: at the approach of the human-operator, at collision, during physical contact and finally when contact is released.
- The expression of the introduced indicators and criteria as inequality constraints related to the torque control input of a robotic manipulator.
- The introduction of the concept of “task energy profile” to track the amount of energy used to accomplish a considered repetitive task in the most optimal way. This profile is then used to formulate a constraint on the instantaneous amount of energy the robot is allowed to exhibit at every time-step. When implemented in the control scheme, the “task energy profile” related constraint makes the robot comply to any deliberate or non-deliberate contact with its environment.

For the validation of the proposed algorithms, beyond the theoretical contributions, simulation and experimental applications are performed on a KUKA LWR4 robotic manipulator.

**Keywords:** *Redundant Robots, Dynamic Control, Safe Human-Robot Interaction, Kinetic energy, Potential Energy, Energy-Based Constraints.*

# Contents

<b>1</b>	<b>Introduction, State-of-the-art and proposed contributions</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Safe mechanical design . . . . .	2
1.2.1	Flexible actuators . . . . .	3
1.2.2	Lightweight Robotic Systems . . . . .	4
1.3	Control . . . . .	5
1.4	Safety standards . . . . .	7
1.5	Discussion and proposed contributions . . . . .	8
1.5.1	Considerations when dealing with safety during human-robot interaction . . . . .	8
1.5.2	Considerations when dealing with constraints . . . . .	9
1.6	Related Publications . . . . .	12
<b>2</b>	<b>Constraints incompatibility</b>	<b>13</b>
2.1	Introduction . . . . .	13
2.1.1	The general problem . . . . .	14
2.1.2	Different types of constraints . . . . .	18
2.1.3	Corresponding literature . . . . .	19
2.1.4	Contributions . . . . .	21
2.2	The QP form and the different constraints incompatibility cases . . . . .	22
2.2.1	Task formulation . . . . .	22
2.2.2	Controller formulation . . . . .	22
2.2.3	Articular constraints: naive formulations . . . . .	23
2.3	Test case scenario for simulation . . . . .	24
2.4	Articular constraints: new formulations . . . . .	25
2.4.1	Joint velocity constraint incompatibility with jerk limits . . . . .	26
2.4.1.1	Illustration 1 . . . . .	29
2.4.2	Joint position constraint incompatibility with deceleration and/or jerk limits	30
2.4.2.1	Joint position constraint incompatibility with deceleration limits	35
2.4.2.1.1	Illustration 2 . . . . .	37
2.4.2.2	Joint position constraint incompatibility with jerk limits 1 . . . . .	40
2.4.2.2.1	Illustration 3 . . . . .	42
2.4.2.3	Joint position constraint incompatibility with jerk limits 2 . . . . .	45
2.4.2.3.1	Illustration 4 . . . . .	47
2.4.2.4	Joint position constraint incompatibility with both deceleration and jerk limits 1 . . . . .	48
2.4.2.4.1	Illustration 5 . . . . .	53
2.4.2.5	Joint position constraint incompatibility with both jerk and deceleration limits 2 . . . . .	55

2.4.2.5.1	Illustration 6 . . . . .	60
2.5	Final bounds on the acceleration control variable . . . . .	60
2.5.1	Illustration 7 . . . . .	61
2.6	Conclusion . . . . .	65
<b>3</b>	<b>Energy based safe control</b>	<b>67</b>
3.1	Introduction . . . . .	67
3.2	Interaction forces and safety indicators . . . . .	69
3.2.1	Kinetic energy . . . . .	69
3.2.2	Potential energy . . . . .	71
3.2.3	Relation between kinetic and potential energies . . . . .	73
3.3	Safety limit values . . . . .	75
3.3.1	Safety limit value for the pre-collision safety indicator . . . . .	75
3.3.1.1	Pre-collision safety criterion extension . . . . .	78
3.3.2	Safety limit value for the physical-contact safety indicator . . . . .	79
3.4	Energy constraints implementation . . . . .	80
3.5	Task energy profile . . . . .	80
3.6	Safe dynamic controller . . . . .	81
3.6.1	Task formulation . . . . .	81
3.6.2	Constraints formulation . . . . .	82
3.6.2.1	Articular constraints . . . . .	82
3.6.2.2	Energy related constraints . . . . .	83
3.6.3	Controller formulation . . . . .	86
3.7	Simulation . . . . .	86
3.7.1	Test case scenario . . . . .	86
3.7.2	Scenario 1: obstacle intersecting with the trajectory of the robot and no constraints on its energy . . . . .	89
3.7.3	Scenario 2: nearby obstacle not intersecting with the trajectory of the robot and constraint on its kinetic energy . . . . .	91
3.7.4	Scenario 3: obstacle intersecting with the trajectory of the robot and constraint on its kinetic energy . . . . .	93
3.7.5	Scenario 4: obstacle intersecting the trajectory of the robot and constraints on its kinetic and <i>potential</i> energies . . . . .	96
3.7.6	Scenario 5: obstacle intersecting with the trajectory of the robot and constraint on its task energy profile . . . . .	99
3.8	Conclusion . . . . .	102
<b>4</b>	<b>Safe human-robot interaction: experimental results</b>	<b>105</b>
4.1	Introduction . . . . .	105
4.2	Experimental setup . . . . .	105
4.3	Test case scenario . . . . .	106
4.4	Human-robot interaction, scenario 1 . . . . .	108
4.5	Human-robot interaction, scenario 2 . . . . .	109
4.6	Human-robot interaction, scenario 3 . . . . .	109
4.7	Human-robot interaction, scenario 4 . . . . .	112
4.8	Conclusion . . . . .	113

<b>5</b>	<b>Conclusions and outlook</b>	<b>119</b>
5.1	Contributions . . . . .	119
5.2	Perspectives . . . . .	121
5.2.1	Extension of the work on the constraints incompatibility problem . . . . .	121
5.2.2	Control of collaborative robots for safe human-robot interaction . . . . .	121
<b>A</b>	<b>Appendix</b>	<b>123</b>
	<b>Bibliography</b>	<b>125</b>



# Chapter 1

## Introduction, State-of-the-art and proposed contributions

The present thesis is partially supported by the RTE company through the RTE/UPMC Chair of “*Intervention Robotics*” held by Vincent Padois. The intended goal of this chair is to bring new insights for the development of intervention robotic systems and assistant robots capable of safely interacting with humans. Within this context, the work presented hereby focuses especially on the control problem.

### 1.1 Motivation

Despite their matured technology, robotic solutions are still underused in many industrial contexts. These contexts include for example the aeronautics and shipbuilding industries, the construction and energy production domains, and the small and medium-sized enterprises (SMBs) operating in a flexible craft mode rather than in a standardized manufacturing mode. Several reasons, when combined, can explain the low presence of robots in these areas:

1. the complex nature of the tasks to be performed, which require a technical and gestural expertise of one or several human-operators;
2. the complexity and dynamics of partially or fully unstructured environments in which it is difficult, if not impossible, to place a robot and still be able to guarantee safety for nearby humans;
3. the cost of dedicated robotic solutions.

An efficient way of introducing robots in these domains is to consider the use of robots as *collaborative* tools which tasks are defined *on the fly* and can be adapted online through simple interaction with humans. For example, it is possible to combine the gestural expertise of an operator and his skills for solving problems and taking decisions with the payload, precision and repeatability performances of a robot. This so-called “*collaborative*” robotics is a subject of growing attention in the fields of industrial and academic robotics. Nevertheless, it poses fundamental questions that remain so far open:

- A) how to guarantee on-line the existence of a solution to the control problem when the task is not defined a priori by planning or when it is adapted in real-time as a reaction to a dynamic environment? Indeed, in conventional applications of robotics, tasks are defined a priori and off-line planning can ensure that all the mechanical and actuation



related constraints inherent to the physical capabilities of the robot are not violated by the achieved movements. This is no longer the case when the task is not defined off-line and it becomes necessary to ensure that the retained command decision at each time-step in order to perform the task in the most optimal way will not ultimately lead to inevitable constraints violation.

- B) How to ensure the safety of the environment in cases where: i) Geometric collision avoidance cannot be guaranteed (the robot has a fixed base and cannot escape to avoid collisions). ii) Unanticipated physical interaction is an acceptable scenario for short periods of time (e.g., when a human intersects with the trajectory of a robot in motion)? This topic is referred to in the literature as “Physical Human-Robot Interaction (pHRI)” and it is the first branch of the broader topic of “*Human-Robot Interaction*” [Albu-Schaffer 2005].
- C) Beyond the implementation of robust interactive control laws and the problem of safety that must be handled during physical interaction, how is it possible to simplify/facilitate cognitively the sharing of tasks between a robot and a human? Indeed, providing robots with the capability of predicting humans behaviour by estimating their affective state can considerably improve the quality of human-robot interaction. Acquired data like verbal and non verbal signals for example facial expressions and the “body language” can be used to adapt the robot actions and make its interaction with humans more effective and intuitive [Kulic 2007], [Mavridis 2015]. This field that studies the “social and Cognitive aspects of Human-Robot Interactions (cHRI)” is the second branch of the “*Human-Robot interaction*” topic.

The present thesis provides new insights regarding the problems described in A) and B). First, it addresses a problem referred to in the literature as *constraints incompatibility*. This problem occurs for reactive controllers, i.e., control problems where the task to perform is not known in advance but discovered in real-time, when the formulation of the constraints (e.g., constraints related to the physical limitations of a robot actuators) the robot must cope with does not take into account the dynamic capabilities of its actuators. The solvability of the control problem in such cases cannot be ensured for the present and future states of the robot. The problem of *constraints incompatibility* in the joint space is solved in the first part of the work presented in this thesis. The second part focuses on the peculiar aspects of “physical” human-robot interaction with safety as a main criteria for developing a new control strategy. The proposed controller uses the amount of energy deployed by the robot to asses and limit its degree of danger towards a nearby human-operator during different interaction phases: at the establishment/during physical contact, in case of non-deliberate collisions and when physical contact is released.

The rest of this chapter is organized as follows: first, state-of-the-art mechanical designs, control strategies and safety standards used to handle safety during human-robot interaction are briefly outlined and used to position the presented work. Then, the problems tackled in each chapter are furthermore detailed and finally, the different contributions are summarized.

## 1.2 Safety oriented mechanical design

Traditionally in robots, stiff actuators governed by the principle of “the stiffer, the better” are used for tasks requiring tracking of a desired trajectory with high accuracy [Salisbury 1991]. When these are perfect for such use, they pose however various fundamental problems for applications requiring interaction with unknown and dynamic environments including humans. These robots cannot just blindly move along computed trajectories without concern of the induced forces caused by contacts with the environment. They must be able to properly and safely

react to such perturbations. Challenges related to the interaction of a robot with its environment can partially be tackled from a control point of view, but to reach human like compliance capabilities without sacrificing much of the machines performances, the entire robotic design must be reconsidered. Indeed, researchers have quickly realized that in humans not only the brain creates the intelligence of the body, but also the morphology and biomechanics of the muscles play a critical role in the way humans think and move [Pfeifer 2006]. Therefore, besides the proprioceptive sensors, such as Cartesian force/torque sensors, articular torque sensors and tactile sensitive skins [Robotics 2017a], back-drivable motors can be used to passively react to external forces [Townsend 1993] and subsequently, intrinsically flexible actuators inspired from the biological spring-like properties of human muscles became a subject of interest [Burdet 2001].

### 1.2.1 Flexible actuators

Despite the tradition of making the interface between an actuator and its load as stiff as possible, reducing this stiffness has a number of advantages that can bring the actuation closer to human capabilities. This can result into: greater shock tolerance, lower reflected inertia by decoupling the motor side from the link-side inertia, more stable and accurate force control in addition to the capacity for storing and releasing energy [Garabini 2011], [Chen 2013]. Depending on how their stiffness and damping are achieved, intrinsically flexible actuators can be divided into two main categories [Vanderborght 2012]:

- Actuators with emulated compliance, where the impedance (stiffness and/or damping) of the actuator can be adjusted using software control and joint level torque sensing [Loughlin 2007]. The main advantage of such actuators is the possibility to tune the desired elasticity parameters online so the mechanical behaviour can be adapted for different tasks. On the other hand, as no physical compliant element is present, energy cannot passively be stored in the system. Active compliance has been pioneered by the DLR and is today commercialized in Kuka robotic systems [Bischoff 2010]. This technology has also been shown in hydraulic actuators on systems such as Sarcos [Hyon 2007] and the hydraulically actuated quadruped robot HyQ [Boaventura 2012].
- Actuators with inherent compliance that contains a passive flexible element where the effective joint impedance is altered via active control. Such designs started with earlier VIA (Variable Impedance Actuators) such as Mckibben pneumatic actuators [Caldwell 1995] and the famous series elastic actuator (SEA) [Pratt 1995], [Laffranchi 2011], [Vanderborght 2013] (see Fig. 1.1). This category can be divided into two main design approaches: i) Mechanisms where the mechanical compliance of the element cannot be changed (fixed compliance). For example the Tsagarakis et al. [Tsagarakis 2009] ComPact soft actuator for the iCub robot that uses 6 linear springs and the HypoSEA [Ivar 2011] designed to stretch a linear spring in a non-linear way. ii) Adaptable compliance systems where the stiffness is controlled by mechanical reconfiguration. In this case, usually two motors are used: one to control the equilibrium position of the joint and a second to control its compliance. A famous example of such design is the DLR Floating Spring Joint (FSJ) [Wolf 2011] designed for the first 4 axes of the anthropomorphic DLR Hand Arm System [Griebenstein 2011].

To help tuning passive elasticity for actuators, Kashiri et al in [Kashiri 2013] introduced a method for the selection of passive stiffness of compliant actuated arms. The method is based on the effects of passive compliance on key parameters including natural frequency, damping ratio, Cartesian stiffness and energy storage capacity.

More details about VIAs and their various design technologies can be found in [Vanderborght 2013] and among the VIATORS project [VIA 2017] related publications.

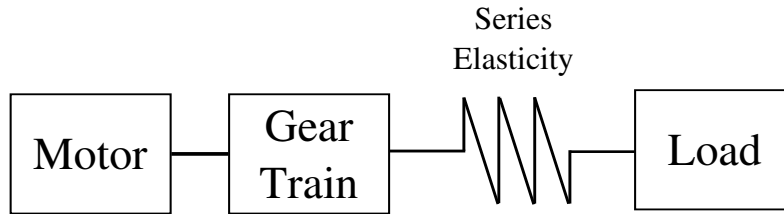


Figure 1.1: Bloc diagram of Series Elastic actuators.

### 1.2.2 Lightweight Robotic Systems

Using the aforementioned actuation technology, robots based on lightweight and highly integrated mechatronics designs that are more suitable for physical interaction with humans have been developed. Prominent state-of-the-art robots from this category are depicted in Fig. 1.2: The KUKA LBR iiwa [KUKA 2017] is a fully torque controlled robot designed as a continuation

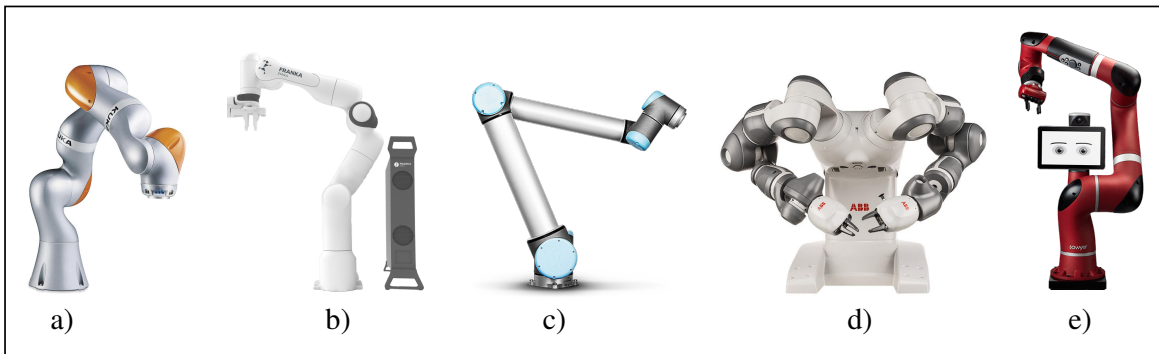


Figure 1.2: (a) KUKA LBR iiwa, (b) Franka Emika arm, (c) Universal Robots UR10, (d) ABB YuMi and (e) Rethink Robotics Sawyer.

of the DLR lightweight robotic arms lineup. Based on the DLR LWR-III, it has 7 degrees of freedom, weighs 14 *kg* and is able to achieve a unitary payload-to-weight ratio. In addition to its torque sensors at each joint, the robot is equipped with redundant joint position sensors on both actuation and link sides.

Franka Emika [EMIKA 2017] is a highly cost competitive robotic arm also aimed for human-robot collaboration. Launched in 2017, despite its low cost, it has similar features to other more expensive systems. The robot is equipped with torque sensors in all its 7 joints and can handle payloads up to 3 *kg*.

UR10 from UNIVERSAL ROBOTS [ROBOTS 2017] is a commercially available lightweight redundant arm with a payload of 10 *kg*. With only 6 degrees of freedom, it has however a great working range with joints that can perform complete  $\pm 360$  rotations.

ABB Yumi is a collaborative dual-arm robot with 14 axes of freedom (7 in each arm) [ABB 2017]. It includes flexible hands, parts feeding systems and embedded cameras. With a payload of 0.5 *kg*, this robot is specifically designed for small parts assembly.

Sawyer is the latest robot model from Rethink Robotics. Based on the two armed Baxter, with 7 degrees of freedom, it exhibits better accuracy and repeatability performances [Robotics 2017b].

With embedded vision, the robot is able to handle loads up to 4 *kg* and thanks to its torque sensors, it can also be fully torque controlled.

While other robot providers have recently started to propose collaborative versions of their standard robots, the five examples cited hereby remain forerunners of this domain.

## 1.3 Control

For robots that physically interact with humans, the interaction force should imperatively be considered as an important safety quantifier [ISO 2016]. It is thus necessary to control not only the position but also the interaction force [Ikuta 2001]. To design an interaction oriented controller, the idea of force control has been an active research topic with initial work in [Whitney 1977] and [Mason 1981]. It can be divided into two main approaches [Siciliano 2016, Chapter 9]:

- direct force control that uses an explicit closure of a force feedback loop. A widely adopted strategy from this category is hybrid force/position control that allows the user to simultaneously specify a desired motion along the unconstrained task directions and a desired contact force/moment along the constrained task directions [Raibert 1981], [Khatib 1987]. A major concern with this approach is that for good performances, an accurate model of the contact properties of the environment of the robot is needed.
- Indirect force control without explicit closure of a force feedback loop. Force control in this case is achieved via motion control. The most widely used scheme from this category is probably impedance control. The idea of extending human-like impedance control to robots aimed for physical interaction started with the pioneering work of Hogan [Hogan 1984]. The basic objective of such controllers is the achievement of a desired dynamical relation between external forces applied to the robot and its resulting movement. In many robotic applications this dynamical behaviour is specified in terms of equivalent mass, stiffness and damping matrices with respect to some Cartesian coordinates [Ott 2010]:

$$F_{ext} = M^* \ddot{e} + D^* \dot{e} + K^* e, \quad (1.1)$$

The terms  $e$ ,  $\dot{e}$  and  $\ddot{e}$  are not explicitly defined here because they are representation dependent (see [Siciliano 2008]). They respectively correspond to the position and orientation errors of the end-effector of the robot with respect to a desired pose  $x^*$  and their first and second derivatives.  $F_{ext} \in \mathbb{R}^6$  is the external wrench (see Fig. 1.3).  $M^*, K^*, D^* \in \mathbb{R}^{6 \times 6}$

are the desired Cartesian inertia, stiffness and damping matrices.

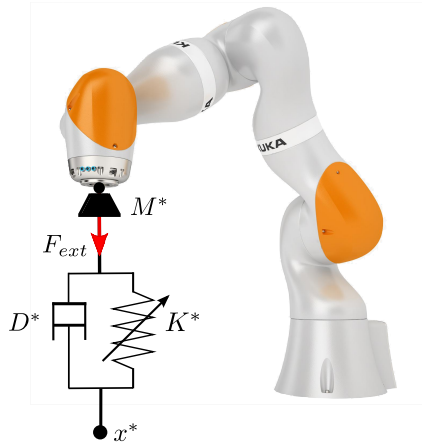


Figure 1.3: Desired mechanical behaviour expressed by a mass-spring-damper like system.

Impedance control laws provide accurate trajectory tracking in free motion while displaying passive properties (mass-spring-damper like system) in response to unexpected collisions. To overcome position tracking accuracy and stability problems when implementing impedance control for flexible joint robots, adapted schemes that take into account the robot's joints elasticity have been proposed [Zollo 2005], [Albu-Schäffer 2007], [Ott 2008b], [Ott 2008a].

Two interesting implementations are the schemes proposed by Schindlbeck et al. and Ficuciello et al. In [Schindlbeck 2015], a novel Cartesian passivity-based controller that combines both force tracking and impedance control is presented. To ensure the stability of the controller and its robustness with respect to contact loss, the concept of Task-energy tanks is used [Ferraguti 2013]. Basically, the *energy tank* dynamically cancels out the non-passive terms which arise during contact such that the overall system with its combined force/impedance controller is passive. In [Ficuciello 2015], a Cartesian impedance controller which parameters are modulated to enhance the comfort perceived by humans during physical interaction is presented. Better performances in terms of compromise between accuracy and execution time are obtained.

Apart from the cited strategies, various other control approaches using internal and external force/torque sensors have been developed to handle safety for human-robot interaction during pre and post impact/contact phases [Ebert 2002], [Lumelsky 1993], [Ikuta 2003]. Haddadin in [Haddadin 2008] and De Luca in [De Luca 2006] present different strategies to reduce the effect of undesired impacts. A collision detection parameter based on the estimated external torque is introduced and used with joint torque feedback to scale down the link inertia obtaining a “lighter” robot that “flees” from the collision area. An other strategy is to use the disturbance input (sensed external torque) to slow the robot until zero velocity is reached then pushing it back along its original path. Heizmann and Zelinsky in [Heizmann 2003] propose a safety criterion based on the *potential impact force* to filter the control torque of the system. The introduced controller allows one to consider two potential contact points at the same time for a real-time implementation. As the degree of potential injury is directly related to the mass and velocity of the colliding objects, the controller proposed in [Haddadin 2012a] takes into account the reflected robot inertia along a collision direction to decide about the maximum operational point velocity. The bounds on this velocity are based on experimental results relating mass, velocity,

geometry and medically observable soft tissue injury by systematic drop-testing experiments with pig abdominal wall sample. By making use of the redundancy property of a KUKA/DLR lightweight arm, [De Luca 2008] proposes a physical interaction strategy that is able to react safely to collisions while continuing to execute as much possible of the original task. Finally, if physical interaction between the robot and its environment is prohibited, obstacle avoidance techniques such as presented in [Flacco 2012] and [De Luca 2012] can be used. Considering the limited dynamic capabilities of robots, these techniques to avoid physical contact can however fail in case of highly dynamic environments.

## 1.4 Safety standards for human-robot interaction

The original safety standard for traditional industrial robots is the ISO 10218 initially edited in 1992. During its last revision, the decision was taken to separate it into two parts: ISO 10218-1 and ISO 10218-2 lately revised in 2011 [Fryman 2014]. On the one hand, ISO 10218-1 [ISO 2011a] is particularly dedicated to manufacturers; It describes dangerous phenomena related to robots and gives recommendations and guidelines for protective measures and safe design indications in line with the machinery directive 2006/42/EC. Technical specifications in this part concern, in a non-exhaustive way, definitions and requirements related to: singularity, protection against hazards that can be caused by mechanical transmission, requisites in case of loss of power, performances of safety control circuits, conditions relative to control modes and limitations of power and strength plus the description of emergency stop functions. On the other hand, ISO 10218-2 is drafted to give more diverse safety requirements and guidance to integrators for the installation of industrial robots and industrial robot systems “robot+cell(s)”. ISO 10218-2 [ISO 2011b] describes how to properly conduct a task-based risk assessment to eliminate or reduce risks associated with hazards to personnel: hazards engendered by the design, implementation and use of these systems. The newest and most exciting utility introduced in ISO 10218-1 and -2 is the direct interaction between robotic systems and human-operators. Indeed, historically, for safety reasons, industrial robots are usually separated from humans by the mean of physical segregation, i.e., cages. ISO 10218-1 and -2 give general guidelines regarding safety requirements for human-robot interaction but do not provide the needed metrics for the design of collaborative robots, tasks and control strategies. In these standards, four collaborative techniques are considered:

- Safety-rated monitored stop: the system does a monitored stop so the human can enter its workspace to perform an assigned task. The robot system resumes its autonomous task once the operator is out.
- Hand-guiding: based on a safety-rated monitored stop, a human-operator can take control of the end-effector to move the robot. Once motion is complete, a monitored stop is issued again as the person exists the workspace.
- Speed separation monitoring: the robot arm and the human maintain a safe distance from each other. Depending on this distance, the speed of the robot is reduced.
- Power and force limiting: Power and force of the robot are limited by design or control during physical interaction.

The latest document from the International Standardization Organization (ISO) fully dedicated to human-robot collaboration is the ISO/TS 15066 [ISO 2016]. Issued in 2016, this Technical Specification (TS) is a game changer for the robotics industry as it gives specific and detailed



metric-based safety guidance needed to assess and control risks during human-robot interaction. Pressure, force and even transferred energy limit values based on injury and pain sensitivity thresholds for various areas of the human body are provided. It states for example the following limits for allowing human-robot interaction at the hands level: maximum transferred energy  $E = 0.49 J$  and maximum permissible force  $F = 140 N$ .

## 1.5 Discussion and proposed contributions

### 1.5.1 Considerations when dealing with safety during human-robot interaction

Although the presented state-of-the-art gives various propositions on how to handle safety during human-robot interaction, there are still numerous issues to be addressed, specifically in terms of satisfying the safety limit values recommended by the recent ISO/TS 15066 standard [ISO 2016] and the discontinuity problems that may occur when switching between control modes for different interaction phases, e.g., at collisions and at contact establishment and release. In this sense, the work presented in this thesis aims at closing the gap and defining a generic framework to control a robotic arm during different interaction phases with a human-operator. Safety in the controller is therefore included as a constraint.

The following aspects are considered for the formulation of safety constraints and a safe human-robot interaction controller:

- Safety indicators  $I$  related to the safety limit values provided in ISO/TS 15066 [ISO 2016] that can be expressed at any point of a multi-body robot must be formulated. These indicators should reflect the amount of danger the robot exhibits towards a nearby human-operator during different interaction phases: as the human enters the workspace of the robot, after the establishment of a deliberate or non-deliberate physical contact and as contact is released.
- The formulation of safety criteria  $I_{limit}$  that represent the maximum values allowed for the safety indicators and that consequently bound the dynamics of the robot for each interaction phase. These safety criteria should also be based on the safety metrics given by the ISO/TS 15066 standard [ISO 2016].
- Finally using the safety indicators and safety criteria, the formulation of safety constraints that depend on the joint control input  $\tau$ :  $I(\tau) \leq I_{limit}$ . These constraints can then easily be included in an optimization control scheme.

The different safety indicators from the literature are usually only useful during one interaction phase. Velocity of the end-effector of the robot for example is significant only during the movements of the robot and therefore cannot be used as a safety indicator during physical contact. Contact forces on the other hand can only be evaluated during physical interaction. More generic indicators should therefore be proposed.

For a given shape of the contact surface, two parameters are source of danger during human-robot interaction: the impact force created at collision and the contact force generated after the establishment of a physical contact. The most generic way to include and consider these parameters for the formulation of safety indicators is to use energy. Indeed, energy is a universal quantity that can describe most of the physical phenomena occurring during human-robot interaction. The impact force for example is directly related to the amount of kinetic energy dissipated at collision and the contact forces mostly derive from the amount of *potential energy*

that accumulates in controller of the robot during physical contact. Velocity, inertia and also the resulting position error during physical contact are all part of the mathematical expression of energy. Safety during human-robot interaction can therefore be ensured by modulating the amount of energy instantaneously deployed by the robot. The main contributions presented in **Chapter 3** regarding safety during human-robot interaction are as follows:

- the formulation of kinetic and *potential energy*<sup>1</sup> based safety indicators that represent the degree of danger of a robotic arm at collision and during physical contact.
- The formulation of safety criteria that bound the dynamics of the robot during different interaction phases: at the approach of a human-operator, at collision, during physical contact and finally as contact is released.
- The expression of these indicators and criteria as inequality constraints related to the torque control input of the robot.
- The introduction of the concept of “task energy profile” to track the amount of energy used to accomplish a considered repetitive task in the most optimal way. This profile is then used to constrain the instantaneous amount of energy the robot is allowed to exhibit at every time-step. The resulting controller renders the robot compliant to any deliberate or non-deliberate contact with its environment.

For the validation of the proposed algorithms, beyond theoretical contributions, simulation results conducted with a KUKA LWR4 serial robot that physically interacts with its environment are also presented in **Chapter 3**. **Chapter 4** on the other hand presents the results of experimental applications on a real KUKA LWR4 whose energy is modulated as a human-operator enters its workspace, physically interacts with it, release contact then safely leave the workspace. In this scenario, the amount of kinetic energy deployed by the KUKA LWR4 during the approach of the human is modulated according to the operator-robot real-time distance. This distance is computed with an algorithm that uses point clouds acquired from the scene using a set of depth sensors (Kinects).

### 1.5.2 Considerations when dealing with constraints

The formulation of safety related constraints usually depend on sensory input that gives *online* information about the dynamic environment the robot interacts with. The controller in this case *reactively* cope with these constraints as it is impossible to perform a pre-optimization of *safe movements* for the robot without accurate data about the present and future dynamics of its environment.

Reactive controllers are usually formulated and solved to convert the operational inputs (user specified objectives and constraints) into joint inputs and this, at each time-step. The impact of the retained solution on the *solvability* of future control problems is rarely considered and therefore, issues of *constraints incompatibility* may occur. Consequently, this can render the control problem impossible to solve in the future without violating the imposed constraints.

To better understand the problem of *constraints incompatibility*, the following scenario in which a robot interacts with its dynamic environment is considered  $\ll$  a moving telemanipulated robotic arm must keep a safe distance  $d_s$  from an obstacle  $O$  entering its workspace  $\gg$  (see Fig. 1.4)

<sup>1</sup>The term “*potential energy*” here refers to the amount of energy loaded in the controller of the robot during both *free* movement and physical contact phases.



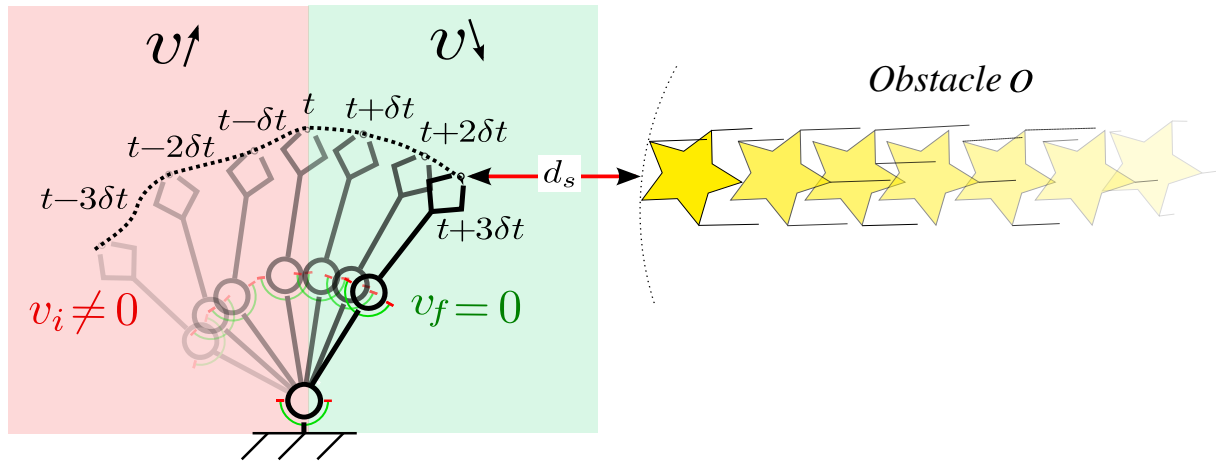


Figure 1.4: A telemanipulated robotic arm coping with a *safety distance* constraint between its end-effector and an obstacle  $O$  entering its workspace.

To prevent getting the system into a state in which its control problem cannot be solved, various specifications must be considered for the formulation of the *safe distance* constraint and the reactive controller for the robot in such situation: the present and future dynamics of obstacle  $O$  must be known so that the reaction of the robot can be triggered at the right time “ $t$ ” considering its dynamic capabilities in Cartesian space, i.e., producible deceleration and jerk. These reaction capabilities for a multi-body robot are configuration dependent and are function of the deceleration and jerk capabilities of its actuators in joint space. Moreover, minimum values for these dynamic capabilities in operational space cannot be guaranteed without knowing the future configurations of the robot.

These exposed considerations make the control of the robot in such scenario a complex problem to solve. However, if the present and future dynamics of both the robotic arm and obstacle  $O$  are predictable, a **formal optimal** solution can be computed as follows: considering the dynamics of obstacle  $O$  and minimum guaranteed deceleration and jerk capabilities of the end-effector of the robot in Cartesian space, the optimal time “ $t$ ” at which the robot must start **braking** using its **maximum producible deceleration and jerk** so it can stop exactly when the distance between its end-effector and the moving obstacle  $O$  is equal to  $d_s$ , can be computed. The robot can then keep this distance by moving in the opposite direction as it is chased by obstacle  $O$ . Note that in this scenario, if the reaction of the robot is triggered after the optimally computed instant “ $t$ ”, the robot will never be able to cope with the *safe distance* constraint considering its limited reaction capabilities. A *constraints incompatibility* problem between the *safe distance* constraint and the bounded maximum producible deceleration and/or jerk will inevitably occur as it becomes impossible to cope with both constraints at the same time during the movement of the robot.

In addition to these various challenges related to the *safe distance* constraint, the controller of the robot in the proposed scenario must also take into account the very own actuation limitations of the system’s actuators, i.e., the articular velocity and position limits in addition to the constraints on the max/min producible articular acceleration and jerk. From a closer point of view, the articular constraints related to these physical limitations are actually a *simplified* version of the *safe distance* constraint. Analogously, the instantaneous distance to the moving obstacle in Fig. 1.4 relates to the static articular velocity and position limits, the position of the moving end-effector corresponds to the actual position or velocity of the constrained joints that increase towards their respective boundaries; and the deceleration and jerk capabilities in

Cartesian space equate to the dynamical capabilities of the robot in joint space. We recall that in this scenario, the robot is reactively controlled and trajectories for its joints cannot be pre-planned. Consequently, the constraints related to its articular velocity and position boundaries are also handled reactively by the controller. To avoid *constraints incompatibility* problems regarding these two constraints, their classic formulations usually used in the literature are modified in **Chapter 2** to include the dynamic capabilities of the robot in joint space, namely, the maximum producible articular deceleration and jerk:

- the formulation of the constraint on articular velocity is modified to include the maximum producible articular jerk used to bring the joint's acceleration to 0 at the activation of the constraint and;
- the formulation of the constraint on articular position is modified to include the maximum producible articular deceleration and jerk used to bring both the articular acceleration and velocity to 0 at the activation of the constraint.

The problem of *constraints incompatibility* relates to the notion of *safety at the control level* about which, Fraichard in 2007 proposed 3 main criteria [Fraichard 2007]:

- to decide its future motion, a robotic system should consider its own dynamics;
- to decide its future motion, a robotic system should consider the future behaviour of its environment;
- to decide its future motion, a robotic system should reason over an infinite time-horizon.

In his work, Fraichard assesses different control approaches for single body mobile robots regarding the definition of safety he proposed. Based on these criteria, our work in **Chapter 2** arises as the continuity of previous results developed by Sébastien Rubrecht [Rubrecht 2010]. In his PhD thesis [Rubrecht 2011], Rubrecht introduced the concept of *constraints incompatibility* for the constraints related to the articular limitations of a robotic manipulator. This problem appears for example when the formulation of a constraint on articular position does not take into account the amount of deceleration producible by the actuators of the robot. In such case, if this constraint is activated **tardively**, no sufficient time may be available for the actuator of the joint to stop its motion before exceeding the allowed position limit. In his work, Rubrecht solved this problem for a serial robot reactively controlled at the kinematic-level. Our work presented in **Chapter 2** on the other hand, tackles the problem of *constraints incompatibility* for robotic manipulators controlled at the dynamic-level (torque control) and an optimal solution to this problem regarding the articular velocity and position constraints is proposed.

For each of these two constraints, the new proposed formulation is derived as follows:

- For the constraint on articular velocity, first, the joint's discrete state during the *braking phase* implicitly induced by the actuator of the joint to bring its acceleration to 0 when coping with an articular velocity limit is described. Second, the *general form* of this state is derived and used to formulate the new expression of the constraint. Finally, the number  $n$  of time-steps, that corresponds to the duration of the induced *braking phase*, needed to optimally cope with the articular velocity limit is computed. **The constraint on the articular velocity and the one on the articular jerk are made compatible.**
- For the joint position constraint, first, the joint's discrete state during the *braking phase* implicitly induced by the actuator of the joint to completely stop its motion when coping with an articular position limit is described. Second, the *general form* of this state is

derived and used to formulate the new expression of the constraint. Finally, the number  $n$  of time-steps, that corresponds to the duration of the induced *braking phase*, needed to optimally cope with the joint position constraint is computed. **The articular position constraint together with the constraints on articular deceleration and jerk are made compatible.**

When included in a QP formulation of the control problem, the introduced new formulations of the constraints related to the articular physical limitations of the robot guarantee the existence of a constraint compliant and optimal solution as joint position, velocity, torque/deceleration and also jerk limits are all simultaneously satisfied during the movement of the robot. The *viability* of the state of the robot is guaranteed for every control time-step.

## 1.6 Related Publications

Some of the novel developments introduced in this work have been peer-reviewed and validated with the publications listed hereafter:

- Meguenani, A., Padois, V., & Bidaud, P. (2015, September). Control of robots sharing their workspace with humans: an energetic approach to safety. In Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on (pp. 4678-4684). IEEE.
- Meguenani, A., Padois, V., Da Silva, J., Hoarau, A., & Bidaud, P. (2017). Energy based control for safe human-robot physical interaction. In 2016 International Symposium on Experimental Robotics (Vol. 1). Springer.

## Chapter 2

# Constraints incompatibility

For reactive control laws, input data for the operational task are usually not known a priori but discovered every time-step. In this case, constraints such as joint limits cannot be handled beforehand through planning and are therefore included in the controller as inequalities. Depending on how these constraints are expressed, incompatibility issues (i.e., the impossibility of satisfying two or several constraints at the same time) that can render the control problem impossible to solve or lead to constraints violation may appear. Indeed, coping for example with an articular position or velocity limit within a small amount of time (e.g., 1 *ms*) may be impossible considering the limited reaction capabilities (i.e., producible torque/deceleration and jerk) the actuators of a robot can produce. In this chapter, the problem of constraints incompatibility related to the articular physical limitations of a serial robot is studied. Incompatibility cases between constraints are exposed then resolved by proposing new mathematical formulations for the joint position and velocity constraints. The new formulations take into account the dynamic capabilities of the actuators of the robot. The constraints are implemented into a QP (quadratic program) scheme to solve the control problem of general fixed-based serial manipulators, and illustrated using a KUKA LWR4 at the dynamic control level (torque control). The resulting new expressions allow the activation of the considered constraints, at the right time to comply with the joint position, velocity, torque/acceleration and jerk limits, all at the same time. A solution for the control problem is available at every time-step, smoother control torques are produced and the “*viability*” of the state of the robot is guaranteed over an infinite horizon of time.

### 2.1 Introduction

In this section, first, the general problem of *constraints incompatibility* that appears when using reactive control laws is furthermore introduced and defined. The different types of constraints a robotic manipulator could face are enumerated and the various considerations regarding the *constraints incompatibility problem* when coping with constraints from each type are highlighted. A brief overview of the *constraints incompatibility problem* state-of-the regarding specifically the constraints related to the articular limitations of robotic manipulators is presented. Finally, the problem specifically tackled in this chapter and the contributions are summarized and the outline of the chapter is described.

### 2.1.1 The general problem

In practical application, the control approach for a robotic system is usually formulated depending on the nature of the operational task to perform. Two cases are distinguished: proactive controllers when the objective of the operational task is already known, e.g., moving the end-effector from point A to point B; and reactive controllers in case input data for the operational task are not known a-priori but discovered at every time-step, for example: tele-manipulation, co-manipulation applications, sensory-based control...etc.

In both control cases, compliance to constraints and more importantly to the physical limitations of system's actuators is a must for any robot. This is particularly true in cases where constraints violation is highly critical, for example: surgery, de-mining, nuclear plants decommissioning...etc.

To ensure strict compliance, for proactive controllers, constraints and tasks can be handled offline. For example, to move the end-effector of a robot from point A to point B in Cartesian space, trajectories can be generated directly at the joint level. In this context, Katzschmann et. al in [Katzschmann 2013] present a trajectory generating algorithm that takes into account the max/min position, velocity, torque and even jerk capabilities of the actuators of a robot. On the other hand, when tasks or constraints are not known in advance (reactive control cases), *trajectories cannot be generated* and it becomes impossible to use the same approach to cope with constraints; that are therefore included *the hard way* in the controller. The work presented hereby focuses on robotic manipulators reactively controlled at the dynamic-level. Which actuation capabilities related constraints are reactively dealt with.

In reactive cases, the control problem can either be formulated at the kinematic-level as the inversion of the following relation:

$$\dot{\mathbf{X}} = J(\mathbf{q})\dot{\mathbf{q}}, \quad (2.1)$$

or at the dynamic-level when dynamic capabilities like contact forces control are needed:

$$\ddot{\mathbf{X}} = \dot{J}(\mathbf{q})\dot{\mathbf{q}} + J\ddot{\mathbf{q}}. \quad (2.2)$$

$J(\mathbf{q})$ ,  $\dot{\mathbf{X}}$ ,  $\ddot{\mathbf{X}}$ ,  $\dot{\mathbf{q}}$ ,  $\ddot{\mathbf{q}}$  are respectively the operational task Jacobian matrix, operational velocity and acceleration vectors and the joint velocity and acceleration vectors.

From the literature, several techniques have been developed to include constraints in the formulation of control problems using analytical schemes. In [Baerlocher 2004], a *clamping* method is introduced to cope with the articular position limits. This technique is based on removing the participation of the saturated joint  $i$  in the completion of the operational task by zeroing its Jacobian column  $\mathbf{J}_i$  and its  $i$  diagonal term from a projection matrix in the kernel of the Jacobian. Flacco et al. in [Flacco 2015] present a *saturation in the null space* similar method that uses a scaling factor for eventually scaling the task when it is necessary to cope with a joint limit. An other method to deal with constraints on articular velocity and position is presented in [Ngo 2005],[Ngo 2006]. It uses a Lyapunov function:  $V(\mathbf{q}, \dot{\mathbf{q}}) = \frac{\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{(\mathbf{q}_{max} - \mathbf{q})(\dot{\mathbf{q}}_{max} - \dot{\mathbf{q}})}$ , that becomes asymptotically infinite when a joint gets close to one of its limits, to derive the control law for a robotic arm.

Although it is possible to implement constraints using such analytical schemes, the most general approach consists in formulating the control problem as a convex optimization one. Constraints are then expressed through both equalities and inequalities [Chen 1994], [Ma 2002]. Independently from the type of the controller, one question remains so far open: for the current and every future time-step, is it always possible to find a solution to the control problem? Considering the

*naive*<sup>1</sup> classic expressions of constraints, the answer is clearly: “No”; and this, because of the *constraints incompatibility* problem. To better illustrate this issue, let's consider the following scenarios.

Scenario 1: On a straight line, a car is cruising towards a wall. Its current velocity is  $v = 10 \text{ m/s}$ , its distance to the wall is  $d = 10 \text{ m}$  and its deceleration capability is  $a_m = -5 \text{ m/s}^2$ . As shown in Fig. 2.1, the velocity of the car is brought to 0 within two time-steps and the car is stopped exactly at its position limit just before hitting the wall. In this case, it is said that: both constraints on the deceleration and the position of the car are compatible as a solution is always available for the control problem.

Scenario 2: On a straight line, a second car is also cruising towards the wall. Its current

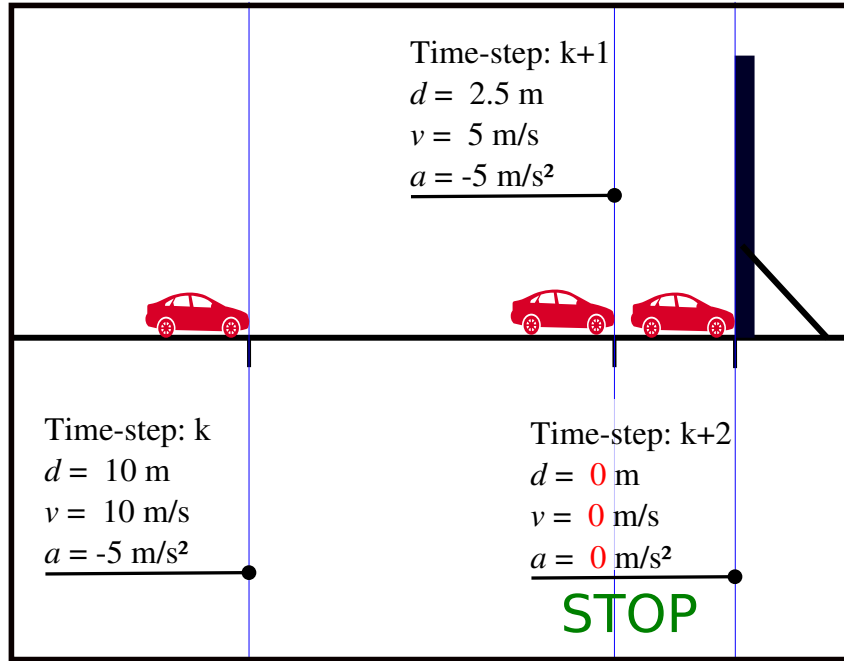


Figure 2.1: Braking phase for the car in scenario 1 as it stops before hitting the wall.

velocity is  $v = 10 \text{ m/s}$ , its distance to the wall is  $d = 10 \text{ m}$  and its deceleration capability is  $a_m = -1 \text{ m/s}^2$ . In its current state, none of the constraints<sup>2</sup> of the car are violated. However, even if the vehicle starts *braking* from time-step<sup>3</sup>  $k$  using its full deceleration capability  $a = a_m = -1 \text{ m/s}^2$  (control input), collision with the wall is inevitable. Indeed, as shown in Fig. 2.2, during its braking phase, the car hits the wall between time-steps  $k+1$  and  $k+2$  and this, before bringing its velocity to 0. In this case, it is said that: the constraint related to the deceleration capability of the car and the one on its position are *incompatible*. At time-step  $k+1$ , there is no solution for the control problem that does not violate the car's constraints at the next time-step. It is clear why there is no issue of constraints incompatibility for the vehicle in the first scenario as the amount of deceleration it can produce is larger than for the car in scenario 2,  $-5 \text{ m/s}^2$  compared to  $-1 \text{ m/s}^2$ . A [solution](#) to ensure compatibility for the constraints of the vehicle in scenario 2 can be as follows: considering its low deceleration capability ( $-1 \text{ m/s}^2$ ), the black car in Fig. 2.2 should start *braking* several time-steps in advance. Precisely, as shown in Fig. 2.1, 10

<sup>1</sup>Naive because they do not take into account the reaction capabilities of the actuators of the robot, i.e., the max/min producible deceleration and jerk.

<sup>2</sup>The constraint on its position ( $d \geq 0$ ) and the constraint related to its deceleration capability ( $a_m \leq a \leq a_M$ ).

<sup>3</sup>For this example, the time-step duration is 1 s.

time-steps before reaching the wall and 50  $m$  from it. These are the needed time and distance to completely stop the vehicle considering an initial velocity of  $v = 10 \text{ m/s}$ . The only way to implement this solution is to reformulate the naive expression of the position related constraint to take into account the available deceleration capability  $a_m$  in addition to the instantaneous state of the car plus the considered position limit:  $f(d, v, a_m) \geq 0$ . With such a formulation, every time the vehicle reaches an *extreme* state<sup>4</sup>, the reflected constraint on the control variable is automatically activated, a *braking phase* is implicitly engaged and a *viable* [Aubin 1991] solution is available for the control problem as the car copes with its position limit.

We recall the concept of *viability* [Aubin 2011] that is often used when dealing with the balance issue for humanoid robots. It can naturally be extended to the *constraints incompatibility* problem [Fraichard 2007] as follows: *A state is viable if starting from this state there exists over an infinite horizon of time a sequence of control inputs that satisfies all constraints in the future.* Therefore, considering the notion of *viability*, unlike the state of the black car at time-step  $k$  in scenario 1, the state of the red one in scenario 2 at the same time-step and for all the following time-steps is *viable*.

---

<sup>4</sup>An extreme state for the car is a state from which, considering its deceleration capability, if it starts braking using this maximum deceleration, it will stop just before hitting the wall.

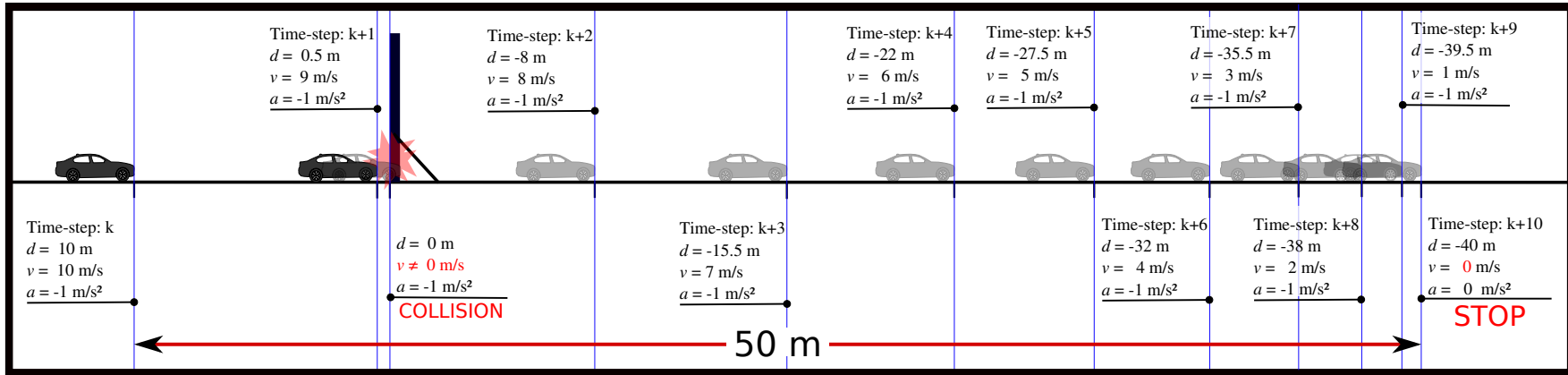


Figure 2.2: Braking phase for the car in scenario 2 as it collides with the wall. Transparent frames show the continuation of the braking phase in case the wall is removed.

Articular space constrains			Operational space constraints		
Type 1: static constraints $S \leq S_M$	Dynamic constraints $S \leq \tilde{S}_M$		Type 4: Static constraints $w(S) \leq w_M$	Dynamic constraints $w(S) \leq \tilde{w}_M$	
	Type 2: Predictable limit $\tilde{S}_M$	Type 3: Non predictable limit $\tilde{S}_M$		Type 5: Predictable limit $\tilde{w}_M$	Type 6: Non predictable limit $\tilde{w}_M$

Table 2.1: Types of constraints encountered when using a reactive controller.  $S = \{\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \ddot{\ddot{\mathbf{q}}}\}$  is the extended state of the robot in articular space.  $w(S) = \{\mathbf{X}, \dot{\mathbf{X}}, \ddot{\mathbf{X}}, \ddot{\ddot{\mathbf{X}}}\}$  is the mapping of this state in operational space.



### 2.1.2 Different types of constraints

For reactive control schemes, types of constraints that can be encountered are summarized in table 2.1. We distinguish two groups of constraints: 1) articular space constraints that concern directly the extended state of the robot  $S = \{\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \overset{\cdot\cdot\cdot}{\mathbf{q}}\}$ . For example, the constraints related to an actuator's position and velocity limits. And 2) operational space constraints usually expressed as non-linear functions of  $S$  and that relate to the movements of the robot in operational space. For example, a constraint of keeping a safe distance  $d_{safe}$  between the robot and an obstacle within its workspace. Such constraint can, independently from other constraints, be expressed:

$$\underbrace{(J_c(\mathbf{q})\dot{\mathbf{q}})}_{\dot{X}} \vec{n} \delta t + d \geq d_{safe}, \quad (2.3)$$

where  $J_c$  is the Jacobian matrix associated to the closest part from the structure of the robot to the obstacle,  $\vec{n}$  is the vector associated to the shortest distance between the obstacle and the robot,  $\dot{\mathbf{q}}$  is the generalised articular velocity,  $\delta t$  is the control sample-time and  $d$ : the instantaneous closest distance between the robot and the considered obstacle.

As shown in table 2.1, constraints from the two groups are either static or dynamic. A static constraint is a constraint which limit is constant and does not variate over time, for example, a physical position limit for an actuator. On the other hand, dynamic constraints are formulated with limits that can change in time, for example based on sensory acquired data. Variations of these limits can be either: 1) predictable, for example, a constraint of keeping a safe distance between the robot and an obstacle entering its workspace with a *constant* velocity, 2) non-predictable, like the movement of a human-operator nearby the robot. The variation of the distance between the robot and the person cannot easily be predicted.

To prevent a robot from getting into *non-viable* states, all its constraints must be reformulated to include the reaction capabilities of its actuators, namely: the max/min producible articular deceleration/torque and jerk. Analogue to the cars in scenarios 1 and 2, coping with a constraint implies a braking phase that is implicitly induced by the actuators of the robot. The span of such braking phase depends on the system's dynamic capabilities. Therefore, each constraint in table 2.1, expressed with its naive formulation, can be compatible with the reaction capabilities of the robot in case of actuators that can produce infinite torque/deceleration and jerk. In such case, the duration of the implicitly induced braking phases can be reduced to **one** control sample-time.

The naive expressions of constraints in table 2.1 can be modified as follows:

For type 1 constraints, just as the suggested [solution](#) for the black car in scenario 2, considering constant articular deceleration and jerk capabilities<sup>5</sup> and a static limit  $S_M$ , a *formal* solution to the control problem that preserves *viability* for the state of the robot can be computed. For this, the formulation of this type of constraints must be modified to take into account the system's articular reaction capabilities, namely, producible articular deceleration  $[\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M]$  and jerk  $[\overset{\cdot\cdot\cdot}{\mathbf{q}}_m, \overset{\cdot\cdot\cdot}{\mathbf{q}}_M]$ . The same can also be done in case of constraints with dynamic predictable limits  $\tilde{S}_M$  (type 2). Considering an accurate prediction, such bounds can be handled as static limits. For type 3 constraints, a straight forward solution is to widely bound the variations of the dynamic limit  $\tilde{S}_M$ , which can however substantially alter the optimality of the control solution regarding the task to perform. A type 3 constraint can be for example a constraint on the kinetic energy of the robot expressed in articular space. Such constraint can be formulated:  $\frac{1}{2}M(\mathbf{q})\dot{\mathbf{q}}^2 \leq E_c^{art}$ , with  $E_c^{art}$  the maximum allowed kinetic energy in articular space. When reflected on  $\dot{\mathbf{q}}$ , the constraint is written:  $\dot{\mathbf{q}} \leq \sqrt{2M(\mathbf{q})^{-1}E_c^{art}}$ . The problem with constraints of

<sup>5</sup>Which is not exactly the case for multi-body robots.

type 4 is, even if the reaction capabilities of the robot in joint space can be considered constant, the non-linear and configuration dependent mapping<sup>6</sup> of these performances into operational space renders the estimation of the braking capabilities in Cartesian space impossible without predicting the future movements of the robot. Consequently, coping *properly* with a static limit in Cartesian space is more complex. And it is even more for dynamic constraints (type 5 and type 6). A type 4 constraint can be for example a constraint on the operational velocity of the end-effector of a robotic manipulator. Such constraint can be expressed:  $\dot{\mathbf{X}} = J(\mathbf{q})\dot{\mathbf{q}} \leq \dot{\mathbf{X}}_M$ , with  $\dot{\mathbf{X}}_M$  the maximum allowed velocity for the end-effector in Cartesian space. A type 6 constraint can be for example a constraint on the operational velocity of end-effector of the robot but with a maximum allowed limit that depends on the instantaneous distance to a considered approaching obstacle.

### 2.1.3 Corresponding literature

The problem of *constraints incompatibility* in this chapter is tackled for constraints related to the articular position and velocity limitations of a robotic manipulator. Therefore, the brief state-of-the-art presented hereby is exclusive to the same type of constraints: (type 1 in table 2.1).

The issue of *constraints incompatibility* is studied in the literature since a long time and remains an open one. In [Park 1998], the problem of high peak of torque and chattering phenomena on the movement of the actuator is highlighted when a joint angle limit of a serial robot is reached. This results from decelerating the limiting joint in order to stop its motion in one sampling time. The following problems appear:

1. collision with the joint angle limit when no sufficient control torque and/or jerk capabilities are available.
2. vibrations by exciting hidden resonant modes of the manipulator which can lead to instability.

Park et al. in [Park 1998] use a method called *P-Step-Ahead Predictor (PSAP)*. Articular position and velocity limits are detected far earlier and the braking phases are activated  $p$  time-steps in advance to reduce the required amount of torque. In addition, the modulation of a weighting matrix related to the manipulability measure [Yoshikawa 1985] is used simultaneously for the same purpose. The problem with this proposed approach is the inevitable heuristic trial and error method to fix the acting parameters (e.g.,  $p$ ). Decré et al. in [Decré 2009] were the first to formulate a mathematical solution to directly integrate the deceleration capabilities of a serial robot into the expression of a Cartesian position constraint aimed to keep the end-effector of a robot at a specific distance from the considered obstacle. Which results into a smoother articular velocity control input. Rubrecht et al. in [Rubrecht 2010] extend the idea to the constraints related to the articular limitations of a robotic manipulator. The deceleration capabilities are included in the formulation of the articular position constraint, that is then automatically activated  $n$  time-steps before the joint position boundaries. However, only kinematic-level control is considered in their work. Lange in [Lange 2014] and [Lange 2015] use a scaling method to generate an immediate Path-Accurate and Jerk-Limited stopping motion for industrial robots. This method can at first glance be transposed and used when coping with articular limitations related constraints. The optimal time for triggering the stopping motion needs however to be computed.

Besides these formal methods that have been proposed to solve the problem of constraints incompatibility, other techniques that are based on Optimal Control approaches seem well suited

<sup>6</sup> $[\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M], [\dot{\mathbf{q}}_m, \dot{\mathbf{q}}_M] \rightarrow [\ddot{\mathbf{X}}_m, \ddot{\mathbf{X}}_M], [\dot{\mathbf{X}}_m, \dot{\mathbf{X}}_M]$ .

for handling that problem. Indeed, Optimal Control provides a systematic approach to optimally compute the control input so as to best track the control objectives while accounting for the current and future constraints on the control inputs and states of a system. As a matter of fact, to be able to guarantee the viability<sup>7</sup> of the state of the robot with respect to its articular constraints but also with respect to operational ones, the control problem should not be solved reactively but over an infinite horizon of time under these strict and explicit constraints.

Nevertheless, reasoning and solving the control problem over an infinite horizon of time does not make much sense in very dynamic environments and contexts where the tasks to be performed are defined and adapted on-line. This advocates for online Optimal Control approaches where the horizon of control input is recomputed at each control time step. However this comes at a very high price as it requires to employ a model of the system's dynamics in order to preview its behaviour over time and compute an optimal horizon of control inputs that maintain the system into a valid state space. It therefore requires a time-integration of the local system model which generally leads to costly non-linear and non-convex optimisation problems [Padois 2016]. In practice, to address this complexity problem, the time horizon can be reduced to some appropriate value. This falls in the category of Model Predictive Control where the Optimal Control problem is solved for a given receding horizon. While this reduces the computation cost, the latter remains high and one way to simplify it further is to use a reduced model of the system's dynamics in order to render the MPC problem simpler to solve. This approach has seen a lot of developments in the control of balance for humanoid robots where the dynamics of balancing are reduced to some more or less complex inverted pendulum control problem. This simplified model is used to compute, over a receding horizon, the desired acceleration of the center of mass of the robot given some path to follow and some balance constraints [Wieber 2006], [Ibanez 2015]. While this model reduction approach leads to great results, it also implies a suboptimal use of the dynamics of the robot. Furthermore, the reduced model cannot not incorporate joint space variables which makes it impossible to actually account for very low-level constraints such as joint position limits or input torque saturation.

MPC approaches are therefore gradually extended to control with the consideration of models capturing the system state at the joint level. Computational limitations nevertheless requires the setup of simplifications of the control problem. A significant improvement in this direction is performed by Y. Tassa et al. in [Tassa 2012]. The authors address computational issues in two ways in this work. First, the resolution of the MPC optimisation problem is considered over the whole duration of the activity. That is, the control problem solely aims at continuously driving the system towards an all-time minimum, rather than finding an optimum at each control step. To this aim a suboptimal solution solely is demanded at each control step, generally obtained through a single iteration of the optimisation algorithm. It therefore allows the consideration of much more complex optimisation problems. In this particular work, an iterative Linear Quadratic Gaussian optimiser, a variant of Differential Dynamic Programming, is setup to work over the joint actuation space. Computational issues nevertheless arise from the needs to update the model rapidly for an iteration of the optimiser. To meet this challenge, the second contribution of this work is thus to propose a fast dynamics integrator. However this controller still requires a large time slowdown with optimisation iterations of 140ms, thus forbidding a direct real-time implementation. The MPC problem is employed as a trajectory generator running slower than the rate of the lower, tracking control loop. This cannot be considered as satisfactory in terms of constraint compatibility but still provides an interesting step toward a general approach to this problem.

---

<sup>7</sup>As redefined as the very end of section 2.1.1.

In this work, a different type of solution is explored and the choice is explicitly made to follow a formal approach, trying to exploit as much as possible formal compatibilisation of joint level imbricated constraints. The clear goal of such an approach is clearly to provide a very reactive way of keeping track of viability thus ensuring an optimal use of the system's dynamic capabilities.

#### 2.1.4 Contributions

In this work, the problem of *constraints incompatibility* is solved for constraints of type 1 and precisely for robots controlled at the dynamic-level. The constraints on articular positions, velocities, accelerations and jerk are expressed as inequalities within a QP formulation of a reactive control problem. The main contribution concerns the mathematical reformulation of the following constraints:

- no violation of joint position bounds;
- no violation of joint velocity bounds.

To avoid incompatibility problems when reactively coping with such constraints, the new proposed formulations that are expressed at the dynamic-level take into account the reaction capabilities, i.e., the articular deceleration and jerk producible by the actuators of the robot. One kind of incompatibility can occur for example when a braking phase is implicitly induced by the actuators of the robot to stop the movement of a joint at a static position limit within a short period of time, for e.g., one control time-step. This requires high deceleration and jerk capabilities the robot may not be able to produce. Indeed, to be optimal, a robotic system must take the highest advantage from its dynamic capabilities within its limiting ranges; for example, to be as close as possible to a maximum allowed position with high articular acceleration and velocity and still be able to stop in time and not violate the considered limit. This can be achieved through a braking phase triggered  $n$  time-steps before reaching the limit and that takes into account the system's limited available deceleration capabilities, i.e., the producible articular torque/deceleration and jerk. Every time an actuator reaches an *extreme* state, with its new formulation, the constraint is activated and a braking phase is implicitly engaged at the right moment. In addition to the new formulations of the articular position and velocity constraints, that take into account the dynamic capabilities of the robot actuators. A method to compute the exact number  $n$  of time-steps needed for the implicitly induced braking phases, triggered at the activation of the considered constraints, is introduced. The *viability* of the state of the robot regarding the joint's position, velocity, acceleration/torque and jerk can then be guaranteed over an infinite horizon of time.

The present chapter is organized as follows: In section 2, the reactive controller used to compare different formulations of the constraints on articular velocity and position is introduced. First, the naive formulations of these constraints are presented then, by studying the relation between the extended state of the robot  $S = \{\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \dddot{\mathbf{q}}\}$  and the system's physical limitations:  $[\mathbf{q}_m, \mathbf{q}_M]$ ,  $[\dot{\mathbf{q}}_m, \dot{\mathbf{q}}_M]$ ,  $[\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M]$ ,  $[\dddot{\mathbf{q}}_m, \dddot{\mathbf{q}}_M]$ , the different incompatibility cases that may occur are exposed. In section 3, the incompatibility cases are resolved and the new formulations of the joint velocity and position constraints that take into account the dynamic capabilities of the actuators of the robot are introduced. In section 4, the final bounds reflected on the dynamic control variable that allow coping with all a robotic manipulator's articular limitations at the same time are formulated. In section 5, the contributions are summarized, the encountered difficulties and the limitations of the proposed new formulation of constraints are highlighted.

## 2.2 The QP form and the different constraints incompatibility cases

This section aims at establishing the general formulation of the constrained, redundant dynamic reactive control scheme. The objective is to compute every time-step the control torque  $\boldsymbol{\tau}_{|k}^c$  in order to perform a trajectory tracking task in joint space while coping at the same time with the articular position, velocity, torque/acceleration and jerk constraints. We highlight the fact that, desired trajectories for the joints of the robot are not known in advance but *discovered at every time-step* (tele-operation analogue situation). Incompatibility cases related to the naive way of expressing the articular constraints are exposed and discussed.

### 2.2.1 Task formulation

The objective function of the controller is defined as an error to be minimized. A joint-space acceleration task is considered and the error is between the desired articular acceleration  $\ddot{\mathbf{q}}^{des}$  and the expected acceleration  $\ddot{\mathbf{q}}^c$ ;  $\ddot{\mathbf{q}}^{des}$  is computed:

$$\ddot{\mathbf{q}}_{|k}^{des} = K_p(\mathbf{q}_{|k}^* - \mathbf{q}_{|k}) - K_d\dot{\mathbf{q}}_{|k}^*, \quad (2.4)$$

where  $K_p, K_d \in \mathbb{R}^+$  are the proportional and derivative gains. The acceleration task function to be minimized is written:

$$\mathbf{g}(\ddot{\mathbf{q}}_{|k}^c) = \ddot{\mathbf{q}}_{|k}^{des} - \ddot{\mathbf{q}}_{|k}^c. \quad (2.5)$$

### 2.2.2 Controller formulation

The proposed control strategy computes the control torque by minimizing the norm of the articular acceleration task function expressed in the following quadratic form:

$$\arg \min_{\boldsymbol{\tau}_{|k}^c, \ddot{\mathbf{q}}_{|k}^c} \left\| \mathbf{g} \left( \ddot{\mathbf{q}}_{|k}^c \right) \right\|_{Q_t}^2 + \epsilon \|\boldsymbol{\tau}_{|k}^c\|_{Q_r}^2, \quad (2.6)$$

subject to:

$$M(\mathbf{q}_{|k})\ddot{\mathbf{q}}_{|k}^c + \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}) = \boldsymbol{\tau}_{|k}^c, \quad (2.7)$$

with  $\ddot{\mathbf{q}}_{|k}^c$  and  $\boldsymbol{\tau}_{|k}^c$ : the optimization control variables.  $\mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k})$  are the non linear terms, namely gravity, Coriolis and centrifugal induced generalized forces.  $M(\mathbf{q}_{|k})$  is the joint space inertia matrix of the robot.  $Q_t$  and  $Q_r$  are positive semi-definite weighting matrices and  $\|\mathbf{a}\|_Q$  is the  $Q$ -weighted euclidean norm of  $\mathbf{a}$ .  $\epsilon \|\boldsymbol{\tau}_{|k}^c\|_{Q_r}^2$  with  $\epsilon \ll 1$  serves as a regularization task in order to ensure the uniqueness of the control solution and to minimize the norm of the computed control torque.

It can be shown that the quadratic forms composing the tasks can be written as functions of positive semi-definite matrices. This LQP optimization problem is therefore convex and admits a unique global solution [Boyd 2004].

In addition to the equality constraints corresponding to the dynamic equation (2.7), the problem is also subject to inequality constraints described in the upcoming sections.

### 2.2.3 Articular constraints: naive formulations

First, we introduce some notations that will be used throughout the manuscript:

- $t \in \mathbb{R}^+$  denotes time;
- $k \in \mathbb{N}$  denotes the current discrete time-step;
- $\delta t$  is the time-step duration of the discrete-time controller;
- $S_{|k} = \{\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}, \ddot{\mathbf{q}}_{|k}, \dddot{\mathbf{q}}_{|k}\}$  is the extended state of the system at the current time-step  $k$  describing its joint position, velocity, acceleration and jerk;
- $[\mathbf{q}_M, \mathbf{q}_m]$  are the max/min joint position boundaries;
- $[\dot{\mathbf{q}}_m, \dot{\mathbf{q}}_M], [\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M], [\dddot{\mathbf{q}}_m, \dddot{\mathbf{q}}_M]$  are the joint velocity, acceleration and jerk boundaries.

A naive satisfaction of the constraints related to these boundaries can be stated as: the computed control input  $\tau_{|k}^c$  at instant  $k$  must be such that the articular limits are not violated at the next time-step  $k + 1$ . These constraints can naturally be expressed in form of inequalities:

$$\begin{cases} \mathbf{q}_m \leq \mathbf{q}_{|k+1} \leq \mathbf{q}_M, & (2.8a) \\ \dot{\mathbf{q}}_m \leq \dot{\mathbf{q}}_{|k+1} \leq \dot{\mathbf{q}}_M, & (2.8b) \\ \ddot{\mathbf{q}}_m \leq \ddot{\mathbf{q}}_{|k}^c \leq \ddot{\mathbf{q}}_M, & (2.8c) \\ \tau_m \leq \tau_{|k}^c \leq \tau_M, & (2.8d) \\ \ddot{\mathbf{q}}_m \leq \ddot{\mathbf{q}}_{|k+1} \leq \ddot{\mathbf{q}}_M. & (2.8e) \end{cases}$$

With  $\ddot{\mathbf{q}}_m$  and  $\ddot{\mathbf{q}}_M$  that can be computed as in Appendix A.

To be easily accounted for, these constraints can be expressed in function of the control variable  $\ddot{\mathbf{q}}_{|k}^c$ . Considering the literature [Rubrecht 2010], [Decré 2009], the classic approach to do that is based on the extended state of the system at instant  $k$  and a local discrete linear approximation of its behaviour within a  $\delta t$  time-step duration:

$$\begin{cases} \mathbf{q}_{|k+1} = \mathbf{q}_{|k} + \delta t \dot{\mathbf{q}}_{|k} + \frac{\delta t^2}{2} \ddot{\mathbf{q}}_{|k}, & (2.9a) \\ \dot{\mathbf{q}}_{|k+1} = \dot{\mathbf{q}}_{|k} + \delta t \ddot{\mathbf{q}}_{|k}, & (2.9b) \\ \ddot{\mathbf{q}}_{|k+1} = \frac{1}{\delta t} (\ddot{\mathbf{q}}_{|k}^c - \ddot{\mathbf{q}}_{|k}). & (2.9c) \end{cases}$$

Note: for a torque control input  $\tau_{|k}^c$  computed for the current time-step  $k$  and equivalent to an acceleration command  $\ddot{\mathbf{q}}_{|k}^c$  (2.7), the expected articular acceleration at the next time-step  $k + 1$  can be written:

$$\ddot{\mathbf{q}}_{|k}^c(\text{input}) = \ddot{\mathbf{q}}_{|k+1}(\text{output}). \quad (2.10)$$

Using (2.9), in case of a control at the dynamic-level (our case), constraints in (2.8) can be reflected on the acceleration control variable  $\ddot{\mathbf{q}}_{|k}^c$  as:

$$\begin{cases} \frac{2}{\delta t^2} (\mathbf{q}_m - \mathbf{q}_{|k} - \delta t \dot{\mathbf{q}}_{|k}) \leq \ddot{\mathbf{q}}_{|k}^c \leq \frac{2}{\delta t^2} (\mathbf{q}_M - \mathbf{q}_{|k} - \delta t \dot{\mathbf{q}}_{|k}), & (2.11a) \\ \frac{1}{\delta t} (\dot{\mathbf{q}}_m - \dot{\mathbf{q}}_{|k}) \leq \ddot{\mathbf{q}}_{|k}^c \leq \frac{1}{\delta t} (\dot{\mathbf{q}}_M - \dot{\mathbf{q}}_{|k}), & (2.11b) \\ \ddot{\mathbf{q}}_m \leq \ddot{\mathbf{q}}_{|k}^c \leq \ddot{\mathbf{q}}_M, & (2.11c) \\ \ddot{\mathbf{q}}_m \delta t + \ddot{\mathbf{q}}_{|k} \leq \ddot{\mathbf{q}}_{|k}^c \leq \ddot{\mathbf{q}}_M \delta t + \ddot{\mathbf{q}}_{|k}. & (2.11d) \end{cases}$$

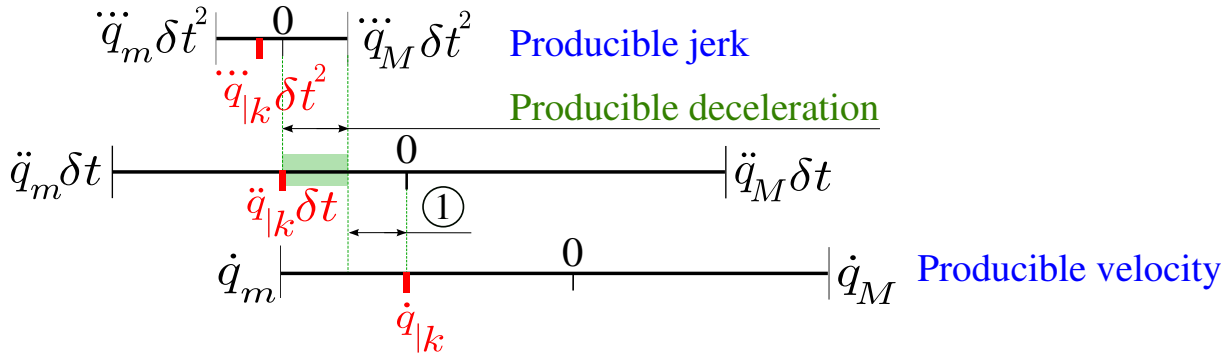


Figure 2.3: Extended state  $S_{|k}$  of an articular joint moving towards its lower position limit  $\mathbf{q}_m$  and one time-step before reaching its velocity limit  $\dot{\mathbf{q}}_m$ . The producible jerk is limiting the range from which the control variable  $\ddot{\mathbf{q}}_{|k}^c$  can be picked:  $\ddot{\mathbf{q}}_{|k}^c$  cannot be chosen equal to zero. Therefore, at the next time-step, at which  $\dot{\mathbf{q}}_{|k}$  is equal to  $\dot{\mathbf{q}}_m$ , the articular acceleration cannot be brought to zero. The joint will continue to accelerate and the velocity limit will inevitably be violated. Because of the incompatibility with the producible jerk, it is impossible in this case to cope with the constraint on articular velocity. The incompatibility disappears if an actuator that can produce a larger amount of jerk is used: gape ① disappears.

When these classic formulations are used, the reflected constraints (2.11) on  $\ddot{\mathbf{q}}_{|k}^c$  are activated only one time-step before reaching the considered limits (i.e.,  $[\mathbf{q}_M, \mathbf{q}_m]$ ,  $[\dot{\mathbf{q}}_m, \dot{\mathbf{q}}_M]$ ,  $[\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M]$  and  $[\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M]$ ). In such case, no sufficient time may be available for the actuators to react and cope with these boundaries. The following **incompatibility cases** can appear and the control problem may become impossible to solve:

1. **Incompatibility case related to the constraint on articular velocity:** Coping with a joint velocity constraint automatically implies bringing the articular acceleration  $\ddot{\mathbf{q}}_{|k}$  to zero when hitting a max/min velocity limit  $[\dot{\mathbf{q}}_m, \dot{\mathbf{q}}_M]$ . In such case, an incompatibility problem may arise if no sufficient articular jerk can be produced by the actuator of the robot during the implicitly induced braking phase that brings the articular acceleration to zero. Coping at the same time with the articular velocity limit and the articular jerk limit may become impossible (see Fig. 2.3).
2. **Incompatibility cases related to the joint position constraint:** On the other hand, coping with a joint position constraint implies bringing both the articular velocity  $\dot{\mathbf{q}}_{|k}$  and acceleration  $\ddot{\mathbf{q}}_{|k}$  to zero when hitting the max/min position boundaries  $[\mathbf{q}_M, \mathbf{q}_m]$ . In this case, two incompatibility cases may arise:
  - A The first is when no sufficient articular deceleration is available during the implicitly induced braking phase (that lasts 1 control time-step) even if sufficient braking jerk can be produced (see Fig. 2.4).
  - B The second is when no sufficient jerk capabilities are available even if enough deceleration can be generated by the considered actuator (see Fig. 2.5).

## 2.3 Test case scenario for simulation

In this section, the test case scenario used as a basis for testing all the different formulations of the constraints on articular velocity and position is presented. A comparison between the



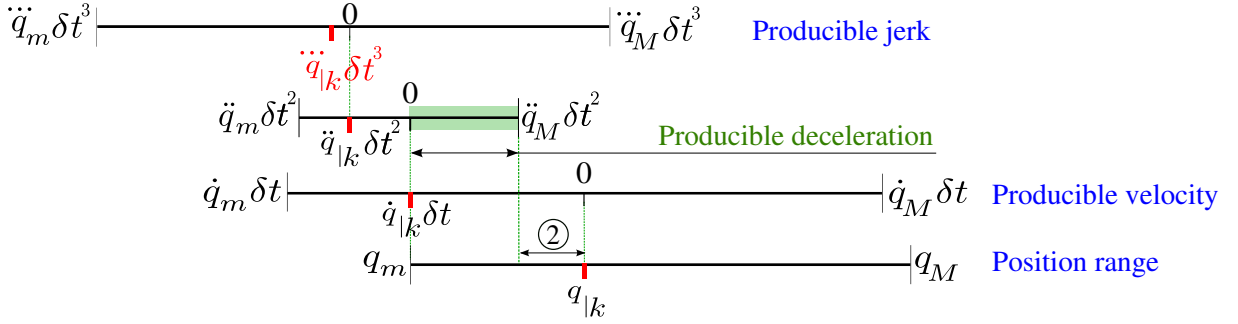


Figure 2.4: Extended state  $S|_k$  of an articular joint moving towards and one time-step before reaching its lower position limit  $q_m$ . The range of producible deceleration  $\ddot{q}|_k^c$  is not sufficient to allow forcing the articular velocity to zero at the next time-step, at which  $q|_k$  becomes equal to  $q_m$ . Because of the incompatibility with the deceleration capability, it is impossible in this case to satisfy the joint position constraint. This incompatibility disappears if an actuator that can produce a larger amount of deceleration is used: gap ② disappears.

naive and new formulations will then be performed in Section 2.4. The controller described in Section 2.2.2 is implemented as a C++ Orocos component [Soetens 2017] on a virtual model of the KUKA LWR4 serial robot using XDE, a robotics-oriented physics simulation engine [Merlhiot 2012].

As a main activity, the robot performs a trajectory tracking task where its joints track desired articular positions (*discovered at every time-step*) (see Fig. 2.6). During its movement, the system is pushed to its physical limits (max/min articular positions, velocities, accelerations and jerks). The LQP is solved in real-time at a period of 1 *ms* using Gurobi, a commercial optimization software [Gurobi Optimization Inc. 2015]. Values used for the gains of the controller during simulation are:  $K_p = 400$ ,  $K_d = 2\sqrt{K_p}$ .  $Q_t$  and  $Q_r$  in this case are identity matrices. For demonstration purposes, only the physical capabilities of the first joint of the robot (joint 0) are constrained.

Fig. 2.7 shows the extended state of joint 0 as it reaches its desired position  $q_0^* = 3.79 \text{ rad}^8$  within 3.5 *s*. During its movement, joint 0 reaches a maximum velocity of  $2.57 \text{ rad.s}^{-1}$ , a maximum acceleration of  $3.15 \text{ rad.s}^{-2}$  and a maximum jerk of  $-8.05 \text{ rad.s}^{-3}$ .

## 2.4 Articular constraints: new formulations

In this section, the incompatibility cases previously exposed for the articular velocity and position constraints are resolved. The classic (naive) formulation of the articular velocity constraint is modified to include the actuators jerk capabilities. The naive expression of the joint position constraint on the other hand, is reformulated to take into account both the amounts of deceleration and jerk producible by the actuators of the robot. Considering these reaction capabilities and compared to the cases where the naive formulations of the constraints are used, the braking phases *implicitly induced* on the movements of a constrained articulation start earlier, and at the right time to cope with the different joint's physical limitations. Each incompatibility case is resolved as follows: first, the extended state  $S$  of the joint during the *implicitly induced* braking phase<sup>9</sup> is described then second, based on this description, the new formulations of the

<sup>8</sup>The position range for joint 0 on a real KUKA LWR4 is  $[q_{m_0}, q_{M_0}] = [-2.97, 2.97] \text{ rad}$ . A wider domain is considered in simulation.

<sup>9</sup>Induced at the activation of the constraint.



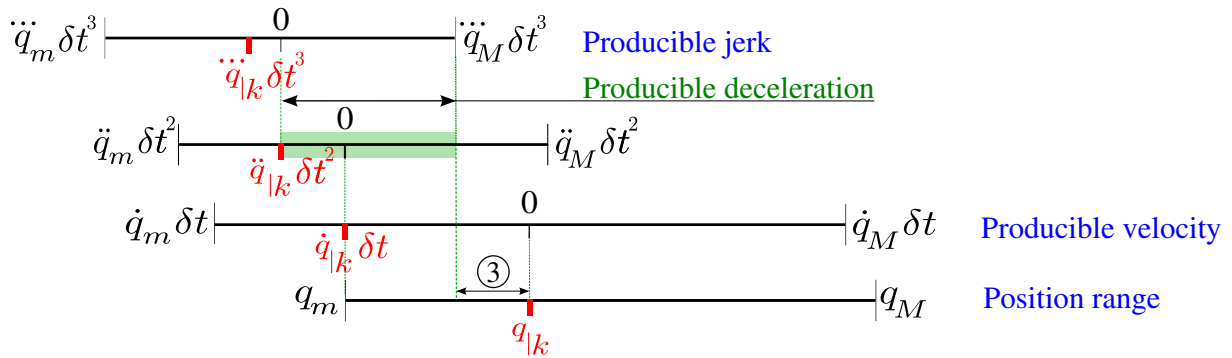


Figure 2.5: Extended state  $S|_k$  of an articular joint moving towards and one time-step before reaching its lower position limit  $q_m$ . The producible jerk is limiting the range from which the control variable  $\ddot{q}|_k^c$  can be picked:  $\ddot{q}|_k^c$  cannot be chosen equal to zero. Therefore, it is not possible to force the articular velocity to zero for the next time-step, at which  $q|_k$  becomes equal to  $q_m$ . Because of the incompatibility with the producible jerk, it is impossible in this case to cope with the joint position constraint. The incompatibility disappears if an actuator that can produce a larger amount of jerk is used: gape ③ disappears.

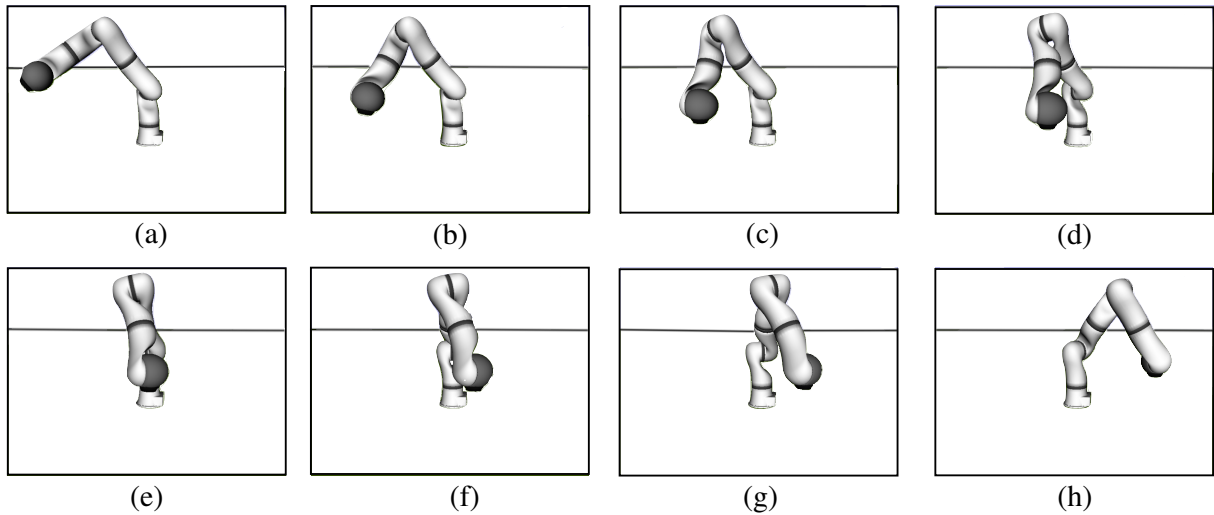


Figure 2.6: Screenshots of the test case scenario simulation with the KUKA LWR4 robot.

constraints on articular velocity and position are derived. Finally, the behaviour of the robot with the new formulations of the constraints is compared to its behaviour when the naive ones are used.

#### 2.4.1 Joint velocity constraint incompatibility with jerk limits

As previously explained, the constraint on the articular velocity of the robot (2.11b) and the one on its producible articular jerk (2.11d) may become incompatible if a joint gets close to one of its velocity limits too fast (see Fig. 2.3). In this case, admissible<sup>10</sup> jerk may not be sufficient to allow a quick variation of the articular torque and it becomes impossible to bring the articular acceleration to zero within only one control time-step. The only solution to preserve

<sup>10</sup>Regarding either the dynamic capabilities of the actuator or the task to be performed by the robot.

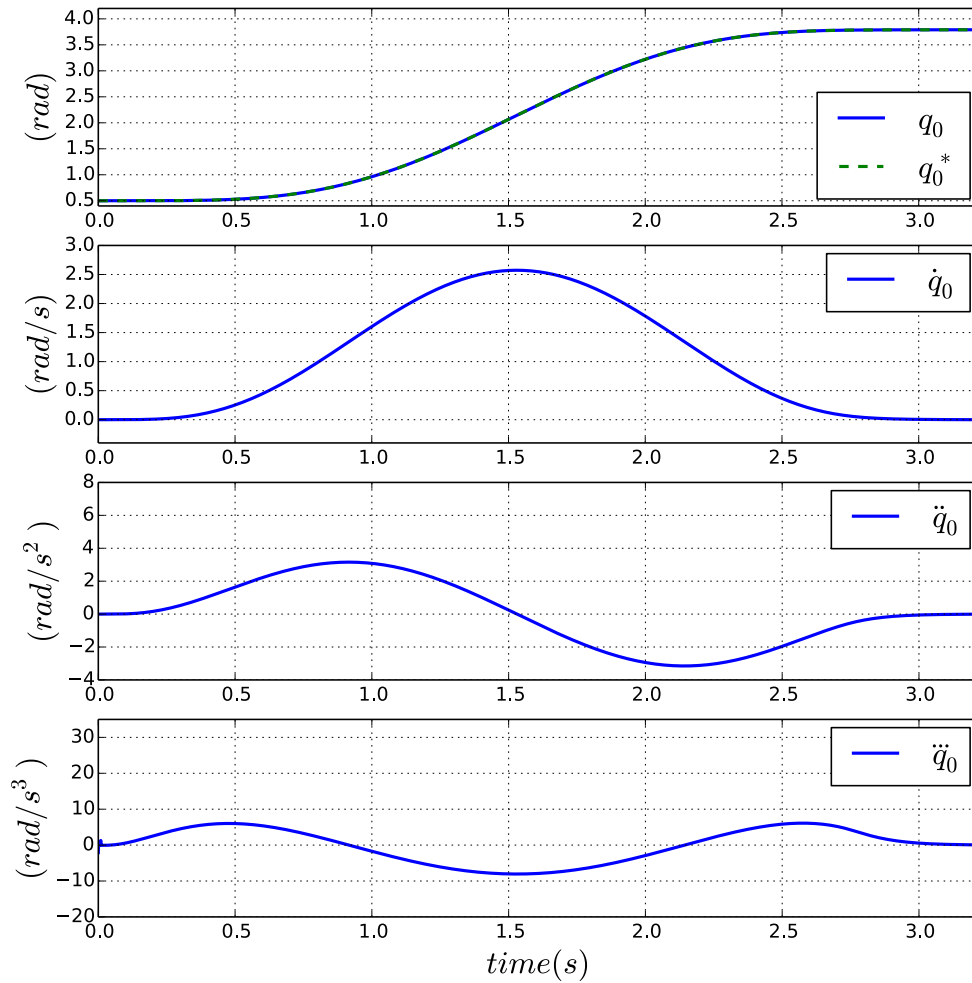


Figure 2.7: Extended state  $S$  of joint 0 during the test case scenario. The movement of the joint is not constrained. Top to bottom: position, velocity, acceleration and jerk.

compatibility is to *induce an implicit* braking phase with maximum jerk, several time-steps (not just one) before reaching the velocity boundary. The braking phase lasts therefore ( $n_1 \in \mathbb{N} \geq 1$ ) control time-steps.

Lets consider the braking phase for a joint moving towards and reaching its maximum velocity limit  $\dot{q}_M$ . During this phase, the actuator brakes with constant jerk  $\ddot{q}_m \leq 0$  until maximum velocity  $\dot{q}_M \geq 0$  is attained. The extended state  $S$  of the system during such phase can be described (Note that the upcoming calculations are performed for each joint of the robot.  $q, \dot{q}$ ,

$\ddot{q}$ ,  $\ddot{\boldsymbol{q}}$  respectively represent one of the components of  $\boldsymbol{q}$ ,  $\dot{\boldsymbol{q}}$ ,  $\ddot{\boldsymbol{q}}$ ,  $\ddot{\boldsymbol{q}}$  (in bold)):

$$\begin{aligned}
 S_{|k+1} & \begin{cases} \dot{q}_{|k+1} = \dot{q}_{|k} + \ddot{q}_{|k}\delta t, \\ \ddot{q}_{|k+1} = \ddot{q}_{|k} + \ddot{\boldsymbol{q}}_m\delta t; \end{cases} \\
 S_{|k+2} & \begin{cases} \dot{q}_{|k+2} = \dot{q}_{|k+1} + \ddot{q}_{|k+1}\delta t, \\ \ddot{q}_{|k+2} = \ddot{q}_{|k+1} + \ddot{\boldsymbol{q}}_m\delta t; \end{cases} \\
 & \quad \vdots \quad \quad \quad \vdots \\
 S_{|k+n_1} & \begin{cases} \dot{q}_{|k+n_1} = \dot{q}_{|k+n_1-1} + \ddot{q}_{|k+n_1-1}\delta t, \\ \ddot{q}_{|k+n_1} = \ddot{q}_{|k+n_1-1} + \ddot{\boldsymbol{q}}_m\delta t. \end{cases}
 \end{aligned} \tag{2.12}$$

With:  $\dot{q}_{|k} \geq 0$ ,  $\ddot{q}_{|k} \geq 0$  and  $\ddot{\boldsymbol{q}}_m \leq 0$ . The joint velocity evolution in  $n_1$  iterations is equal to the general form<sup>11</sup> of the numerical sequence (2.12):

$$\dot{q}_{|k+n_1} = \dot{q}_{|k} + n_1\ddot{q}_{|k}\delta t + \frac{(n_1^2 - n_1)}{2}\ddot{\boldsymbol{q}}_m\delta t^2. \tag{2.13}$$

In case of a control at the dynamic-level, considering (2.10), the condition  $\dot{q}_{|k+n_1} \leq \dot{q}_M$  for all integer  $n_1$  leads to:

$$\ddot{q}_{|k}^c \leq \frac{(\dot{q}_M - \dot{q}_{|k})}{n_1\delta t} - \frac{(n_1 - 1)}{2}\ddot{\boldsymbol{q}}_m\delta t. \tag{2.14}$$

With  $n_1$ , the integer minimizing the right-hand side of (2.14). By differentiating this expression w.r.t  $n_1$ :

$$\left. \begin{aligned} & n_1 \geq 1 \\ & \frac{-(\dot{q}_M - \dot{q}_{|k})}{n_1^2\delta t} - \frac{\delta t}{2}\ddot{\boldsymbol{q}}_m = 0 \end{aligned} \right\} \Rightarrow n_1 = -\frac{\sqrt{-2\ddot{\boldsymbol{q}}_m(\dot{q}_M - \dot{q}_{|k})}}{\ddot{\boldsymbol{q}}_m\delta t}. \tag{2.15}$$

Following the same reasoning for the lower velocity limit, the condition  $\dot{q}_{|k+n_2} \geq \dot{q}_m$  for all integer  $n_2 \geq 1$  becomes:

$$\ddot{q}_{|k}^c \geq \frac{(\dot{q}_m - \dot{q}_{|k})}{n_2\delta t} - \frac{(n_2 - 1)}{2}\ddot{\boldsymbol{q}}_M\delta t. \tag{2.16}$$

With:

$$n_2 = \frac{\sqrt{-2\ddot{\boldsymbol{q}}_M(\dot{q}_m - \dot{q}_{|k})}}{\ddot{\boldsymbol{q}}_M\delta t}, \tag{2.17}$$

maximizing the right-hand side of (2.16).

$n_1$  is the number of time-steps corresponding to the span of a braking phase implicitly induced by an articulation moving towards its upper velocity limit  $\dot{q}_M$ , and during which its positive acceleration is brought to zero by jerking negatively with  $\ddot{\boldsymbol{q}}_m$ ; and  $n_1$  is the number of time-steps corresponding to the span of a braking phase implicitly induced by an articulation moving

<sup>11</sup>Computed using Maple [Maplesoft Inc. 2016].

towards its lower velocity limit  $\dot{q}_m$ , and during which its negative acceleration is brought to zero by jerking positively using  $\ddot{q}_M$ .

Finally, the reflected constraint on the acceleration control variable  $\ddot{q}_{|k}^c$  is of the form:

$$f_\beta(\dot{q}_{|k}, \dot{q}_m, \ddot{q}_M, n_2) \leq \ddot{q}_{|k}^c \leq f_\alpha(\dot{q}_{|k}, \dot{q}_M, \ddot{q}_m, n_1), \quad (2.18)$$

with  $f_\beta$  and  $f_\alpha$  as in (2.14) and (2.16).

### 2.4.1.1 Illustration 1

To illustrate this first theoretical result, in this simulation, using the test case scenario, joint 0 moves towards its upper velocity limit  $\dot{q}_{M_0}$ . Only the classic (naive) formulation of the joint velocity constraint (2.11b) is implemented with the controller: (2.6), (2.7). The constraint on articular jerk (2.11d) is not considered. Fig. 2.8 shows the extended state of joint 0 during a

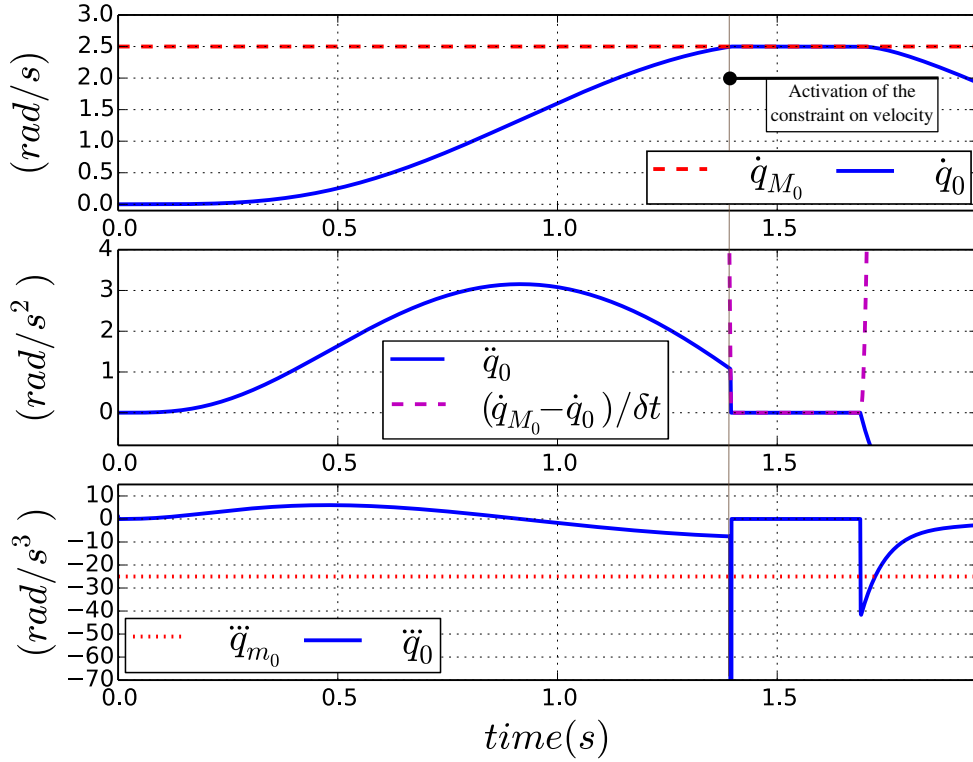


Figure 2.8: Extended state  $S$  of joint 0 during the braking phase (that lasts 1 control time-step = 1  $ms$ ) implicitly induced to cope with the maximum velocity limit  $\dot{q}_{M_0}$ . The naive formulation of the articular velocity constraint (2.11b) is used. Top to bottom: velocity, acceleration and jerk.

braking phase implicitly induced only one control time-step (1  $ms$ ) before reaching  $\dot{q}_{M_0}$ . A high peak of jerk is induced. In this case, if the constraint on articular jerk (2.11d) is implemented in the controller's configuration, the control problem becomes impossible to solve. It is then clear that for a real system with limited jerk capabilities (e.g.,  $\ddot{q}_{M_0} = 25 \text{ rad.s}^{-3}$ ), coping with a velocity limit using the naive formulation of the articular velocity constraint is simply impossible.

On the other hand, when the new formulation of this constraint (2.14) is used, the system starts braking  $n_1$  time-steps before reaching the velocity limit  $\dot{q}_{M_0}$  (see Fig. 2.9). Joint 0 is capable

of coping at the same time with both its velocity and jerk limits (see Fig. 2.10). In this case, including the constraint on articular jerk (2.11d) in the configuration of the controller will not result into an infeasible control problem. Therefore, the incompatibility between the articular velocity and jerk constraints is resolved.

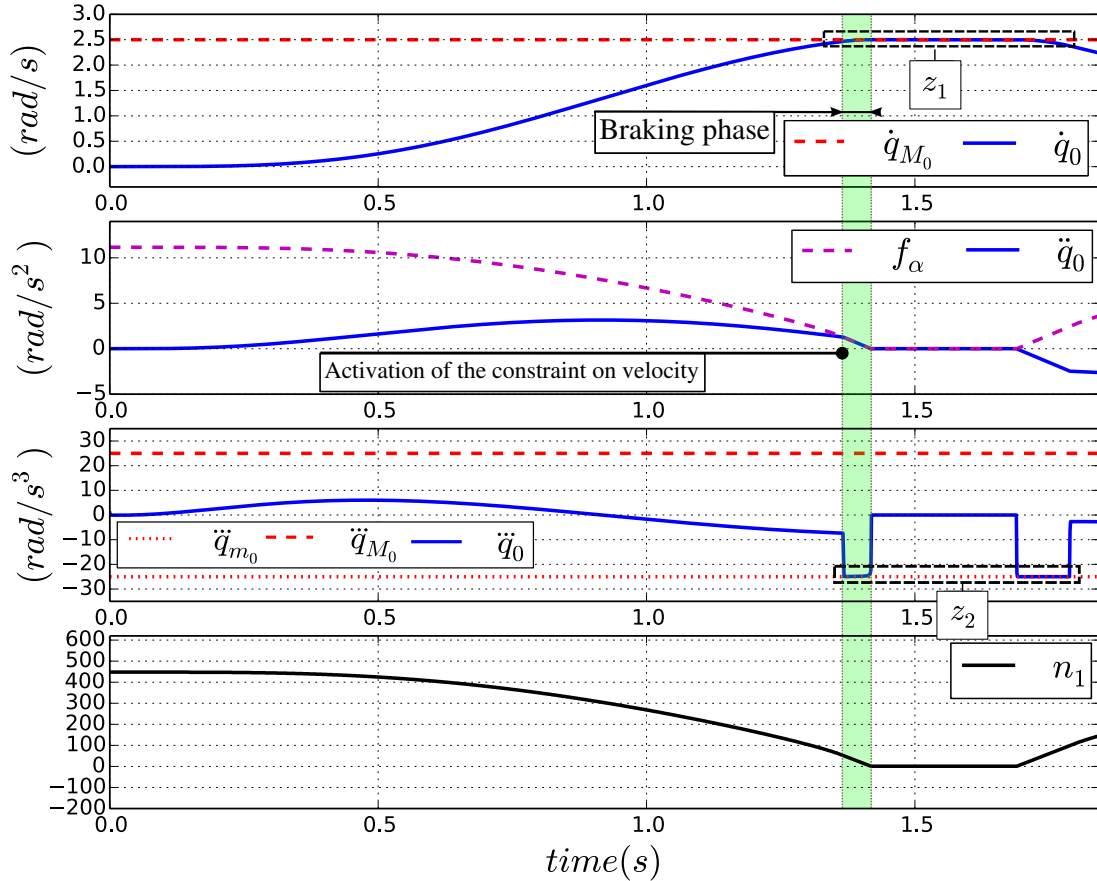


Figure 2.9: Extended state  $S$  of joint 0 during the braking phase implicitly induced to cope with a maximum velocity limit  $\dot{q}_{M_0}$ . In this case, the new formulation of the constraint on articular velocity (2.14) that takes into account the actuator's maximum producible jerk ( $\ddot{q}_{m_0} = 25 \text{ rad}\cdot\text{s}^{-3}$ ) is used. Note that  $n_1$ , is physically meaningful only during the span of the implicitly induced braking phase. Top to bottom: velocity, acceleration, jerk and  $n_1$ .  $z_1$  and  $z_2$  are in Fig. 2.10

#### 2.4.2 Joint position constraint incompatibility with deceleration and/or jerk limits

With its *naive*<sup>12</sup> formulation, the constraint on articular position (2.11a) may become incompatible with the constraints on articular acceleration (2.11c) and/or jerk (2.11d) if a joint gets close to one of its position limits too fast (see Fig. 2.4 and Fig. 2.5). In such case, *admissible* deceleration and/or jerk may not be sufficient to allow a fast variation of the control torque, and it becomes impossible to bring both the articular acceleration and velocity to zero within

<sup>12</sup>Naive because it does not take into account the max/min amount of deceleration and jerk producible by the actuators of the robot.

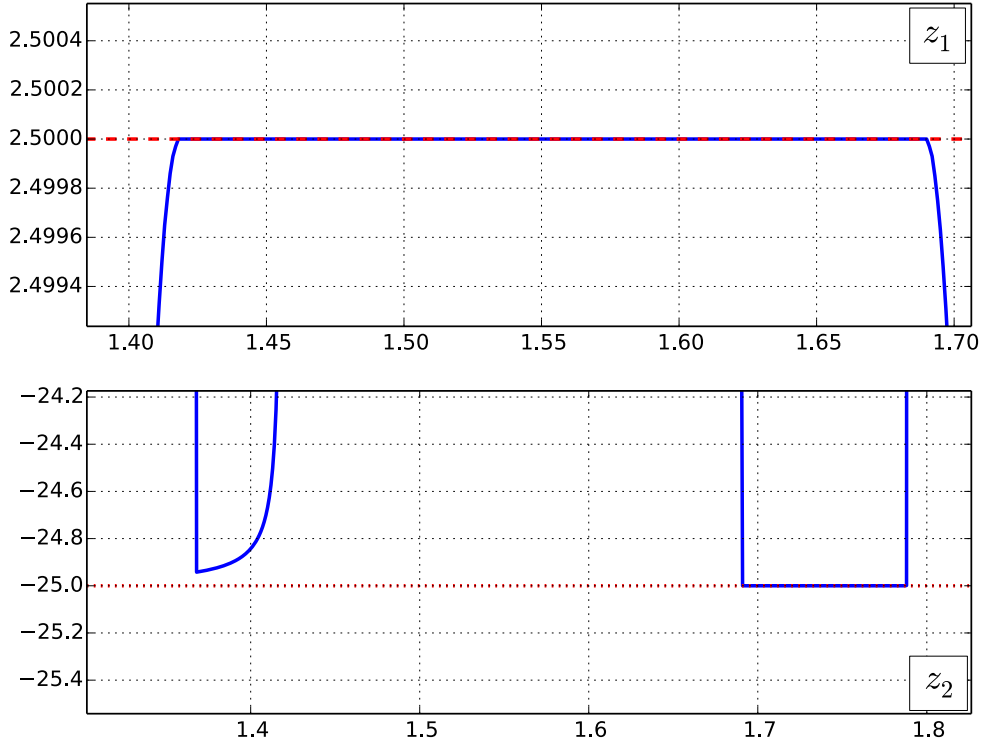


Figure 2.10: Zooms corresponding to Fig. 2.9.

only one control time-step. For example, the complete braking phase needed to be induced by an actuator to *properly* cope with its upper position limit  $q_M$  is as follows:

1. at the activation of the constraint, the actuator starts jerking with maximum negative producible jerk ( $\ddot{q} = \ddot{q}_m$ ) (sub-phase ① in Fig. 2.11).
2. if the maximum producible deceleration  $\ddot{q}_m$  is reached, braking jerk  $\ddot{q}$  is brought to zero and the “charged” deceleration ( $\dot{q} = \dot{q}_m$ ) is applied during several time-steps (sub-phase ② in Fig. 2.11).
3. the amount of deceleration in the joint is finally “de-charged” and brought to zero by jerking positively with maximum producible jerk ( $\ddot{q} = \ddot{q}_M$ ) (sub-phase ③ in Fig. 2.11). By the end, the joint is stopped exactly at its upper position limit  $q_M$ .

As explained, coping with a joint position limit may fail because of incompatibilities with two other constraints: articular deceleration limits (2.8c) and producible articular jerk (2.8d). The resolution of these incompatibilities will be **methodically** performed as follows:

1. **first**, independently from jerk capabilities, the naive expression of the position constraint (2.11a) is modified to include solely the deceleration capabilities: ( $\ddot{q}_m \leq 0$ ) considering the upper position limit ( $q_M \geq 0$ ), and ( $\ddot{q}_M \geq 0$ ) considering the lower position limit ( $q_m \leq 0$ ). With the new formulation, when coping for example with an upper articular position limit, the joint implicitly starts braking using its maximum deceleration capability  $n_3 \in \mathbb{N}$  iterations before reaching  $q_M$  (see Fig. 2.12). Using this formulation, both constraints on the articular position (2.8a) and the articular acceleration (2.8c) become compatible. The reflected bounds on the control variable  $\ddot{q}_k^c$  will be of the form:

$$f_\xi(q_m, \ddot{q}_M, n_4) \leq \ddot{q}_k^c \leq f_\gamma(q_M, \ddot{q}_m, n_3). \quad (2.19)$$

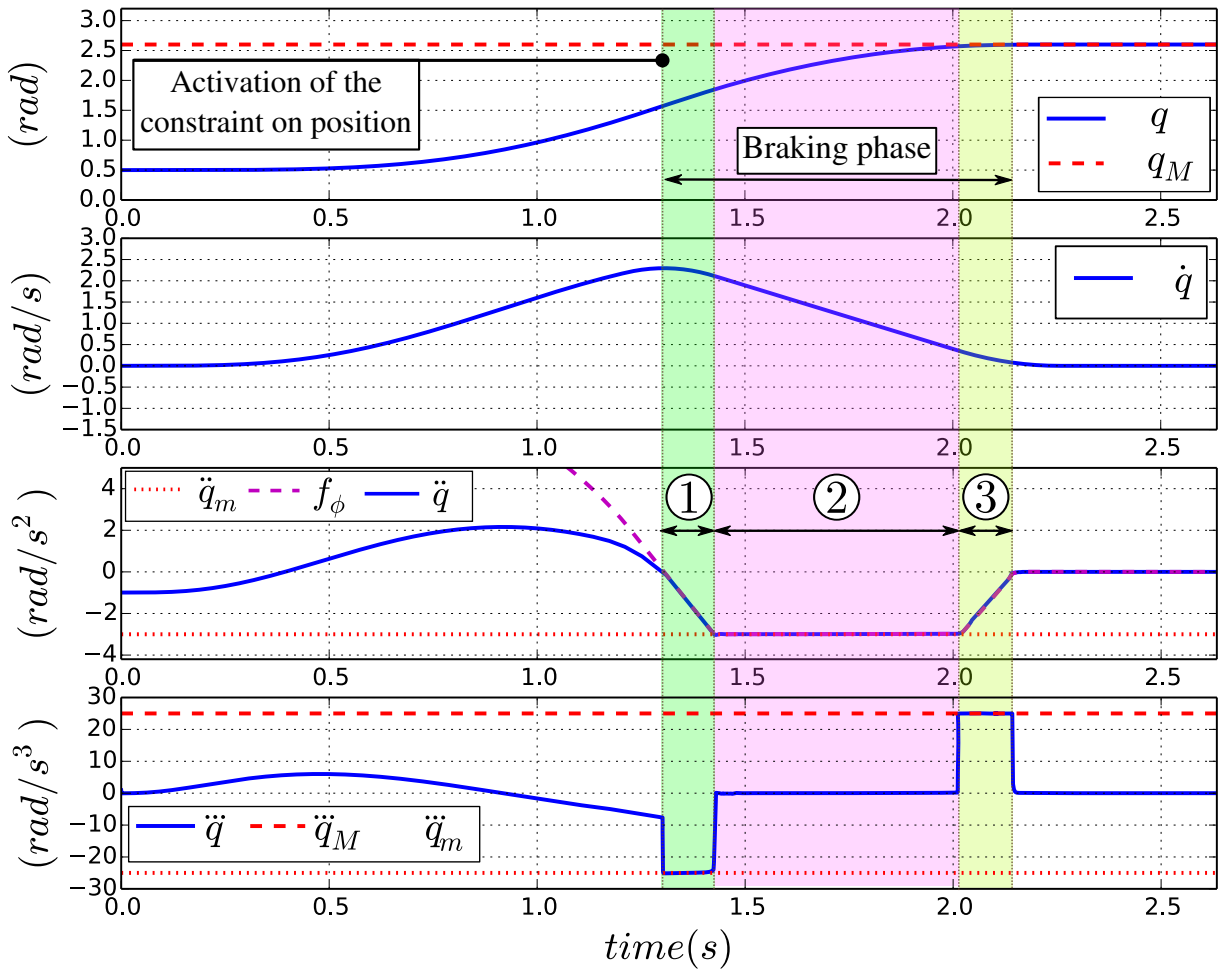


Figure 2.11: Extended state  $S$  of a joint during the induced *complete* braking phase needed to **properly** cope with an upper position limit  $q_M$ . The braking phase is composed of three sub-phases that each lasts for a number of time-steps: sub-phase 1 lasts  $n_{15}$  control time-steps; sub-phase 2 lasts  $n_{17}$  control time-steps and sub-phase 3 lasts  $n_{19}$  control time-steps. Note that this illustration is not the result of a simulation. It represents the state of an actuator in case the *complete* braking phase is used to cope with its upper position limit. Top to bottom: position, velocity, acceleration and jerk.

The proper methods to compute  $n_3$ ,  $n_4$ ,  $f_\gamma$  and  $f_\xi$  are fully defined in Section 2.4.2.1.

2. **Second**, independently from the articular deceleration capabilities, the naive expression of the constraint on articular position (2.11a) is modified to include exclusively the actuator's jerk capabilities: ( $\ddot{q}_m \leq 0$ ) considering an upper position limit ( $q_M \geq 0$ ), and ( $\ddot{q}_M \geq 0$ ) when dealing with the lower position limit ( $q_m \leq 0$ ). With the resulting formulation, when coping for example with an upper position limit  $q_M$ , a braking movement is implicitly induced on the considered joint for a duration of  $n_5 \in \mathbb{N}$  control time-steps. Maximum producible negative jerk  $\ddot{q}_m$  is applied by the actuator during this braking phase until the joint reaches and stops at upper position limit  $q_M$ . With such braking phase, both constraints on the articular position (2.8a) and jerk (2.8c) become compatible. In such

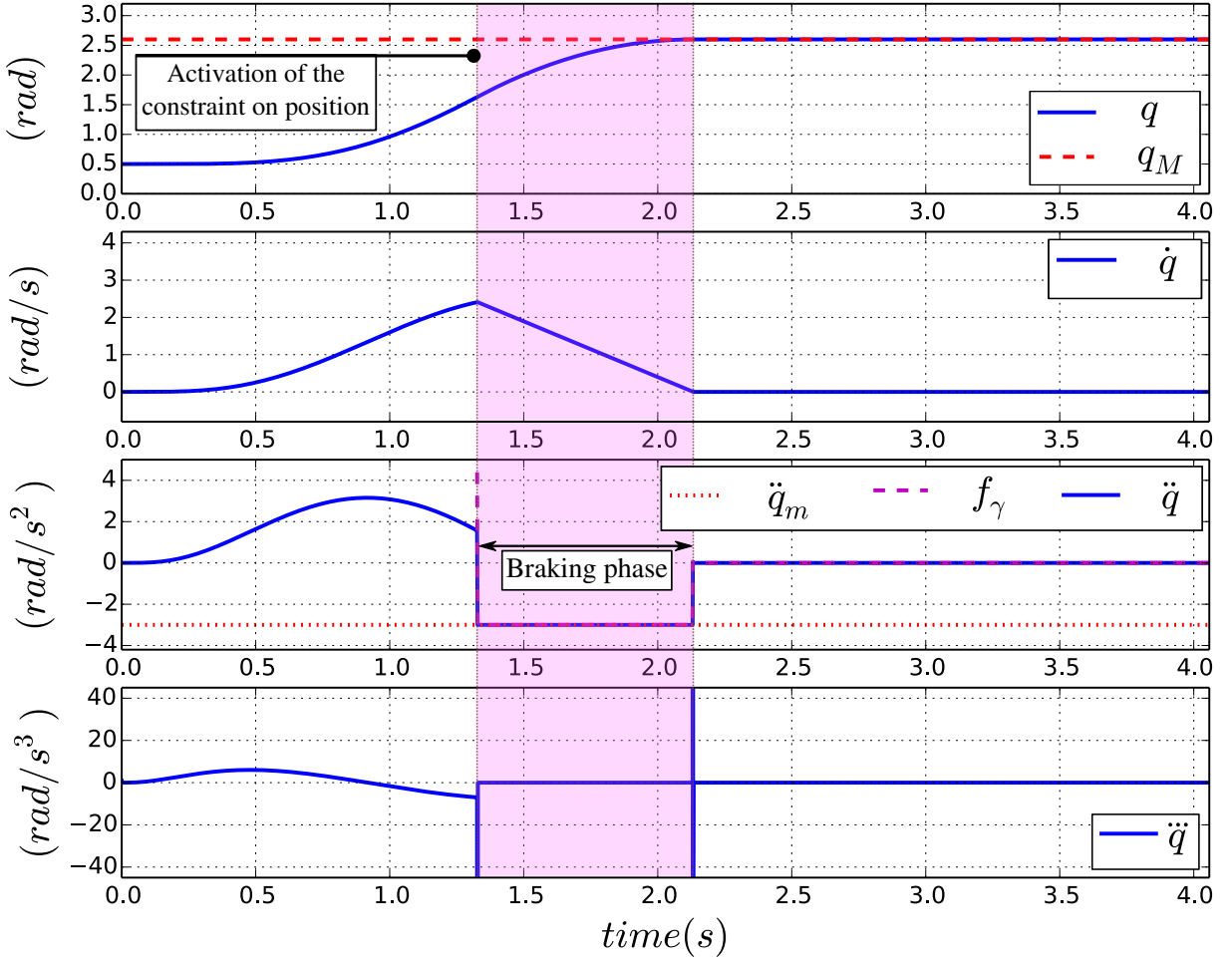


Figure 2.12: Extended state  $S$  of a joint during the induced braking phase needed to cope with an upper position limit  $q_M$ . Lasting  $n_3$  control time-steps, the braking phase in this case takes into account only the articular deceleration capability  $\ddot{q}_m$ . Jerk limits are not considered. Top to bottom: position, velocity, acceleration and jerk.

case, the reflected bounds on the acceleration control variable will be of the form:

$$f_\theta(q_m, \ddot{q}_M, n_6) \leq \ddot{q}_k \leq f_\eta(q_M, \ddot{q}_m, n_5). \quad (2.20)$$

The proper methods to compute  $n_5$ ,  $n_6$ ,  $f_\eta$  and  $f_\theta$  are fully defined in Section 2.4.2.2.

3. **Thirdly**, because of undesired oscillations (more details about this problem in subsection 2.4.2.2) that may appear in the movement of the constrained joint when using (2.20) to cope with a position limit, this formulation is modified to include at the same time both the negative ( $\ddot{q}_m \leq 0$ ) and positive ( $\ddot{q}_M \geq 0$ ) maximum producible articular jerk in both its left-hand and right-hand sides. When coping for example with an upper position limit, the actuator should start braking with negative maximum producible jerk  $\ddot{q}_m$  for  $n_7$  control time-steps. As a consequence of this first braking sub-phase (phase ① in Fig. 2.13), deceleration is “loaded” in the articular joint. The actuator should then jerk positively with maximum producible jerk  $\ddot{q}_M$  to bring the “loaded” deceleration to zero (phase ② in Fig. 2.13). The second sub-phase of the induced braking phase lasts for  $n_9$  control time-steps and by its end, the actuator reaches and stops at its upper position



limit  $q_M$ . With such braking phase, the new formulation of the constraint on articular position will be of the form:

$$f_\mu(q_m, \ddot{q}_M, \ddot{q}_m, n_8, n_{10}) \leq \ddot{q}_k^c \leq f_\lambda(q_M, \ddot{q}_m, \ddot{q}_M, n_7, n_9). \quad (2.21)$$

The proper methods to compute  $n_7$ ,  $n_9$ ,  $n_8$ ,  $n_{10}$ ,  $f_\lambda$  and  $f_\mu$  are fully defined in Section 2.4.2.3.

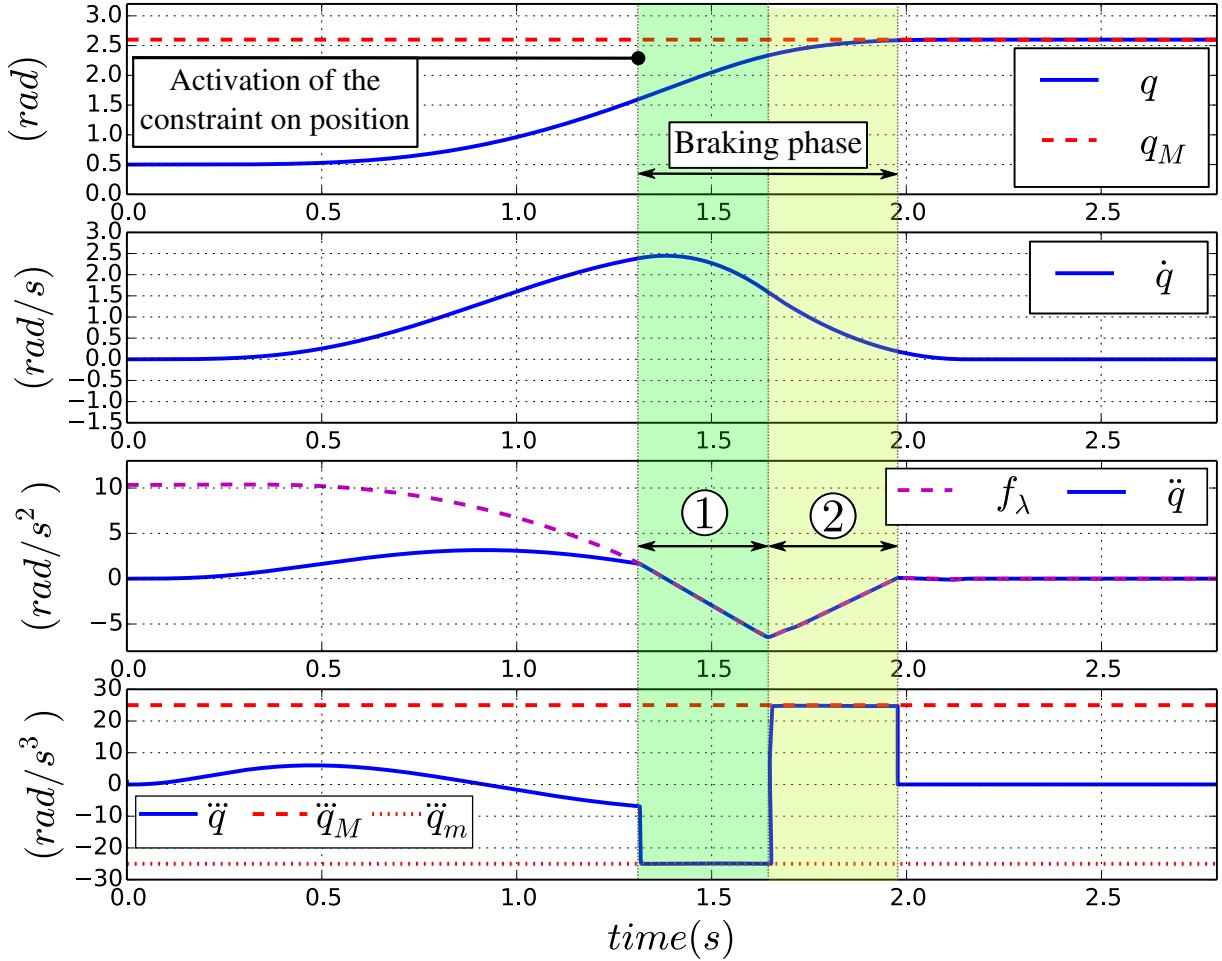


Figure 2.13: Extended state  $S$  of a joint during the induced braking phase needed to cope with an upper position limit  $q_M$ . The braking phase in this case is composed of two sub-phases: sub-phase 1 that lasts  $n_7$  control time-steps and sub-phase 2 that lasts  $n_9$  iterations. Note that, this illustration of the state of the joint is not the result of a simulation. It is a representation of how should be a braking phase induced using (2.21) when coping with an upper position limit. Top to bottom: position, velocity, acceleration and jerk.

4. **Fourthly**, an expression of the constraint on articular position that takes into account at the same time both the jerk and deceleration capabilities is formulated. When included in the controller, this formulation renders the constraints on articular acceleration (2.8c) and jerk (2.8d) both compatible with the constraint on articular position (2.8a). Reflected bounds on the acceleration control variable  $\ddot{q}_k^c$  will be of the form:

$$f_\rho(q_m, \ddot{q}_M, \ddot{q}_m, n_{12}, n_{14}) \leq \ddot{q}_k^c \leq f_\pi(q_M, \ddot{q}_m, \ddot{q}_M, n_{11}, n_{13}). \quad (2.22)$$

The proper methods to compute  $n_{11}$ ,  $n_{13}$ ,  $n_{12}$ ,  $n_{14}$ ,  $f_\pi$  and  $f_\rho$  are fully defined in Section 2.4.2.4.

5. **Finally**, based on all the introduced forms of the various expressions proposed for the articular position constraint, a formulation that is based on the **complete** description of the braking phase used to **properly** cope with a joint's upper/lower position limit is proposed. This includes the deceleration “*de-charging*” sub-phase (sub-phase ③ in Fig. 2.11) that is necessary to prevent possible oscillations on the movement of the actuator during the activation of the constraint. This new and complete formulation takes into account both the negative and positive jerk capabilities ( $\ddot{q}_m, \ddot{q}_M$ ) together with the producible deceleration: ( $\ddot{q}_m \leq 0$ ) when coping with an upper position limit ( $q_M \geq 0$ ), and ( $\ddot{q}_M \geq 0$ ) when coping with a lower position limit ( $q_m \leq 0$ ). Without any drawbacks<sup>13</sup>; both constraints on articular deceleration (2.8c) and jerk (2.8d) become compatible with the constraint on articular position (2.8a). With such formulation of the constraint on articular position derived from the **complete** description of the braking phase, the reflected bounds on the acceleration control variable are of the form:

$$f_\chi(q_m, \ddot{q}_M, \ddot{q}_m, \ddot{q}_m, n_{16}, n_{18}, n_{20}) \leq \ddot{q}_{|k}^c \leq f_\phi(q_M, \ddot{q}_m, \ddot{q}_m, \ddot{q}_M, n_{15}, n_{17}, n_{19}). \quad (2.23)$$

The proper methods to compute  $n_{15}$ ,  $n_{17}$ ,  $n_{19}$ ,  $n_{16}$ ,  $n_{18}$ ,  $n_{20}$ ,  $f_\phi$  and  $f_\chi$  are fully defined in Section 2.4.2.5.

**Note:** for a reactively controlled robot to optimally and *properly* cope with a constraint on its articular position, considering both its max/min deceleration and jerk capabilities, the only formulation of the joint position constraint that should be implemented in its optimization control scheme is (2.23). The reader can directly check Section 2.4.2.5 for the results on the behaviour of the robot when this formulation is used. The other formulations: (2.19), (2.20), (2.21) and (2.22), besides being presented for pedagogical reasons, their intermediary results are used for the formulation of (2.23) and taken individually, they can be used in specific cases where only the max/min producible articular deceleration OR jerk are of interest. Plus, due to the computation load linked to the  $n_i$  parameters, these simpler formulations are faster to compute compared to the complete and proper formulation of the articular position constraint (2.23).

In the upcoming sections, the methods used to compute the introduced new formulations of the joint position constraint are furthermore detailed. In Section 2.5, the final bounds on the optimization control variable  $\ddot{q}_{|k}^c$  that take into account the constraints on the articular position, velocity, acceleration and jerk all at the same time are formulated.

#### 2.4.2.1 Joint position constraint incompatibility with deceleration limits

In this section, the braking phase implicitly induced by a joint's actuator moving towards its upper position limit  $q_M$  is considered. During this phase, the actuator's maximum producible deceleration is  $\ddot{q}_m$ . Its jerk capabilities [ $\ddot{q}_m, \ddot{q}_M$ ] are not considered. The extended state of the system during such braking phase, that makes the actuator stop at its position limit  $q_M$ , can be

<sup>13</sup>No oscillations in the movement of the joint.

described as:

$$\begin{aligned}
 S_{|k+1} & \begin{cases} q_{|k+1} = q_{|k} + \dot{q}_{|k}\delta t, \\ \dot{q}_{|k+1} = \dot{q}_{|k} + \ddot{q}_m\delta t; \end{cases} \\
 S_{|k+2} & \begin{cases} q_{|k+2} = q_{|k+1} + \dot{q}_{|k+1}\delta t, \\ \dot{q}_{|k+2} = \dot{q}_{|k+1} + \ddot{q}_m\delta t; \end{cases} \\
 & \quad \vdots \quad \quad \quad \vdots \\
 S_{|k+n_3} & \begin{cases} q_{|k+n_3} = q_{|k+n_3-1} + \dot{q}_{|k+n_3-1}\delta t, \\ \dot{q}_{|k+n_3} = \dot{q}_{|k+n_3-1} + \ddot{q}_m\delta t. \end{cases}
 \end{aligned} \tag{2.24}$$

With:  $q_{|k} \geq 0$ ,  $\dot{q}_{|k} \geq 0$  and  $\ddot{q}_m \leq 0$  (braking phase). The joint position evolution in  $n_3 \geq 1$  iterations is equal to the general form<sup>14</sup> of the numerical sequence (2.24):

$$q_{|k+n_3} = q_{|k} + n_3\dot{q}_{|k}\delta t + \frac{(n_3^2 - n_3)}{2}\ddot{q}_m\delta t^2. \tag{2.25}$$

The condition  $q_{|k+n_3} \leq q_M$  for all integer  $n_3$  leads to:

$$\dot{q}_{|k}^c \leq \underbrace{\frac{(q_M - q_{|k})}{n_3\delta t} - \frac{(n_3 - 1)}{2}\ddot{q}_m\delta t}_{f_{\gamma_{vel}}}. \tag{2.26}$$

With  $\dot{q}_{|k}^c$ : the velocity control input in case the robot is controlled at the kinematic-level. The equivalent constraint on the acceleration control variable  $\ddot{q}_{|k}^c$ , in case the robot is controlled at the dynamic-level, that generates the same braking profile can be written:

$$\ddot{q}_{|k}^c \leq \frac{(q_M - q_{|k})}{n_3\delta t^2} - \frac{\dot{q}_{|k}}{\delta t} - \frac{(n_3 - 1)}{2}\ddot{q}_m. \tag{2.27}$$

With:

$$n_3 = \frac{-\sqrt{-2\ddot{q}_m(q_M - q_{|k})}}{\ddot{q}_m\delta t}, \tag{2.28}$$

the integer minimizing the right-hand side of (2.27).

The transition from (2.26) to (2.27) can easily be proven as follows: in case of a velocity control input  $\dot{q}_{|k}^c$ , the resulting articular velocity at the next time-step  $k + 1$  is:  $\dot{q}_{|k+1}(output) = \dot{q}_{|k}^c(input)$ . On the other hand, in case of a control at the dynamic-level:  $\dot{q}_{|k+1} = \dot{q}_{|k} + \ddot{q}_{|k}^c\delta t$ , with  $\ddot{q}_{|k}^c$ : the control variable linked to the control torque input  $\tau_{|k}^c$  and  $\dot{q}_{|k+1}$  the resulting articular velocity. The relation between the kinematic-level and the dynamic-level control inputs that results into the same articular movement can then be derived:  $\dot{q}_{|k} + \ddot{q}_{|k}^c\delta t = \dot{q}_{|k}^c$ .

<sup>14</sup>Computed using Maple [Maplesoft Inc. 2016].

Following the same reasoning for the lower articular position limit, when reflected on the acceleration control variable, the condition  $q_{|k+n_4} \geq q_m$  for all integer  $n_4 \geq 1$  leads to:

$$\ddot{q}_{|k}^c \geq \frac{(q_m - q_{|k})}{n_4 \delta t^2} - \frac{\dot{q}_{|k}}{\delta t} - \frac{(n_4 - 1)}{2} \ddot{q}_M. \quad (2.29)$$

With:

$$n_4 = \frac{\sqrt{-2\ddot{q}_M(q_m - q_{|k})}}{\ddot{q}_M \delta t}, \quad (2.30)$$

maximizing the right-hand side of (2.29). We recall that:  $n_3$  is the number of control time-steps corresponding to the duration of a braking phase implicitly induced by an actuator moving towards its upper position limit  $q_M$ , and during which both its acceleration and velocity are brought to zero by decelerating with maximum negative producible acceleration  $\ddot{q}_m$ ; and  $n_4$  is the number of control time-steps corresponding to the duration of a braking phase implicitly induced by an actuator moving towards its lower position limit  $q_m$ , and during which both its acceleration and velocity are brought to zero by accelerating positively with maximum producible acceleration  $\ddot{q}_M$ .

$f_\xi$  and  $f_\gamma$  are respectively equivalent to the right-hand sides of (2.29) and (2.27).

#### 2.4.2.1.1 Illustration 2

For this simulation, using the test case scenario, the controller is implemented with the new formulation of the constraint on articular position (2.27), that takes into account the available deceleration capability ( $\ddot{q}_{m_0} = -100 \text{ rad.s}^{-2}$ ). In this case, just like the test case scenario, joint 0 moves towards its maximum allowed position  $q_{M_0}$ .

Fig. 2.14 shows the extended state of joint 0 as an implicitly induced braking phase is started at the right time and during which the actuator uses its maximum producible deceleration  $\ddot{q}_{m_0}$  to cope with its upper position limit  $q_{M_0}$ . As the articular jerk limits  $[\ddot{\dot{q}}_{m_0}, \ddot{\dot{q}}_{M_0}]$  are not considered in this formulation of the constraint on articular position, a jerk peak is induced at the beginning of the braking phase at the activation of the constraint. When this new formulation of the articular position constraint is used, including the naive acceleration/deceleration constraint (2.11c) in the configuration of the controller will not result into an infeasible control problem. The incompatibility between the articular position and articular deceleration constraint is resolved.

On the other hand, it has been impossible to cope with the same upper position limit using the naive formulation of the constraint on articular position (2.11a). Much larger deceleration capabilities (2.11c) are needed for a braking phase that is induced only one time-step before joint 0 reaches  $q_{M_0}$ .

Fig. 2.16 illustrates how the starting time of the braking phase is automatically adapted to the amount of available deceleration. For example, when  $\ddot{q}_{m_0}$  is reduced to  $-3 \text{ rad.s}^{-2}$ , the braking phase starts earlier and lasts longer.

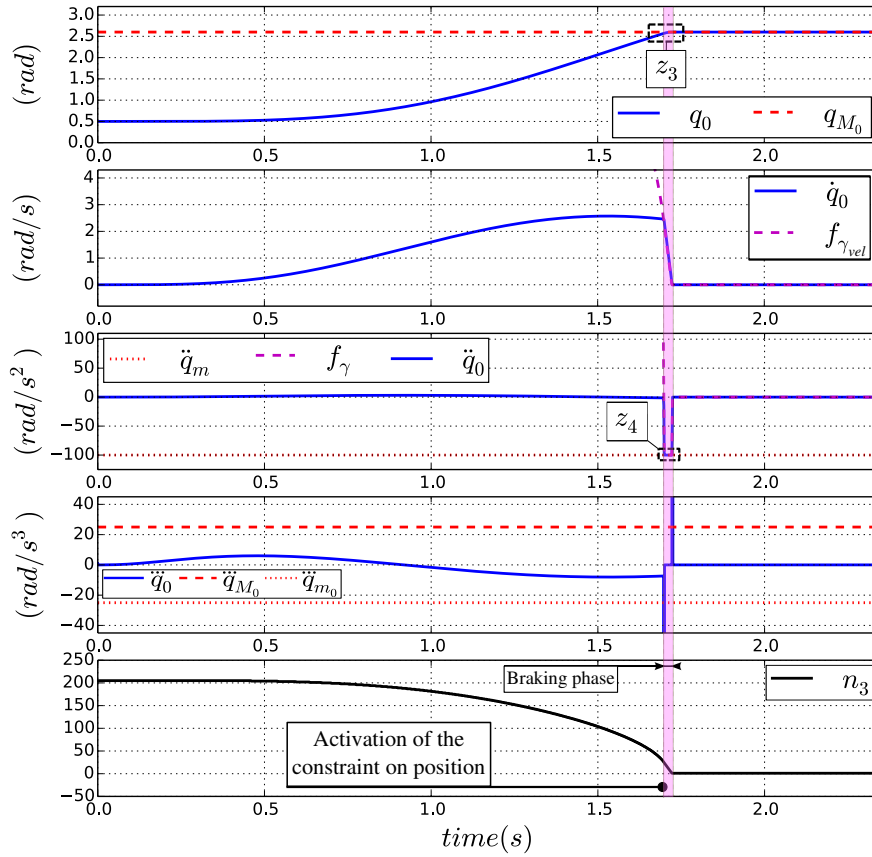


Figure 2.14: Extended state  $S$  of joint 0 during the braking phase implicitly induced by the joint’s actuator to cope with the upper position limit  $q_{M_0}$ . The new formulation of the constraint on articular position (2.27) that takes into account the deceleration capability  $\ddot{q}_{m_0} = -100 \text{ rad.s}^{-2}$  is used. Top to bottom: position, velocity, acceleration, jerk and  $n_3$ . See  $z_3$  and  $z_4$  in Fig. 2.15

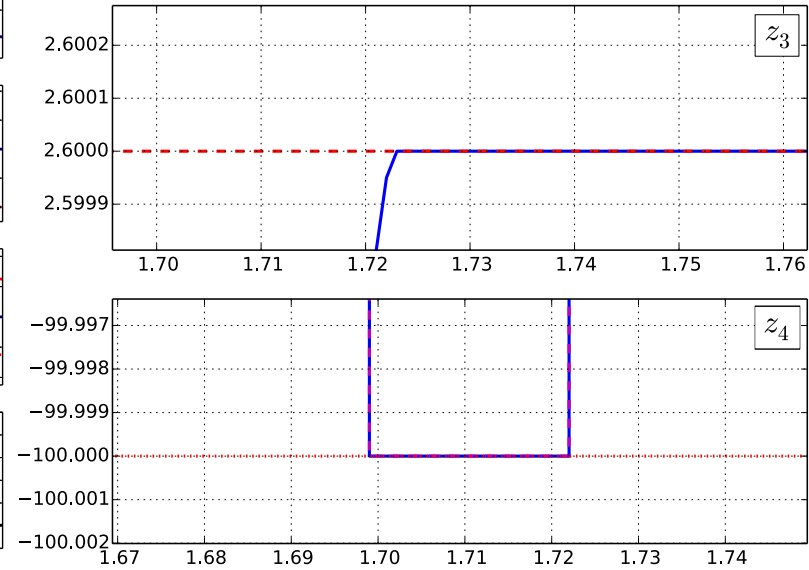


Figure 2.15: Zooms corresponding to Fig. 2.14.

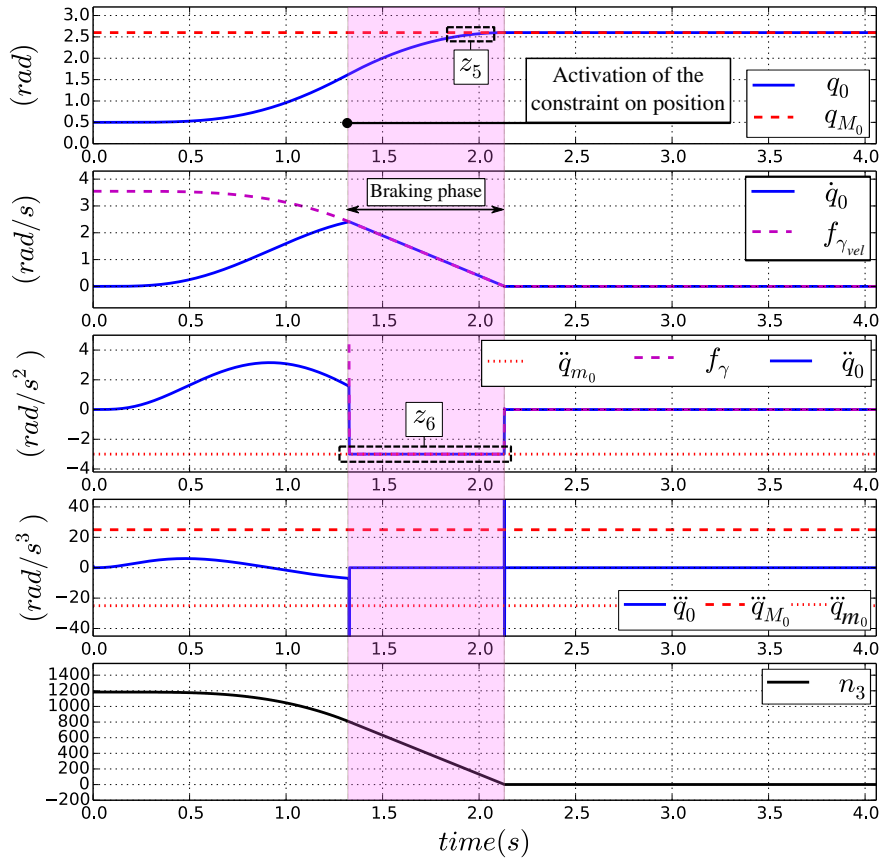


Figure 2.16: Extended state  $S$  of joint 0 during the braking phase implicitly induced by the joint's actuator to cope with the upper position limit  $q_{M_0}$ . The new formulation of the constraint on articular position (2.27) that takes into account the deceleration capability  $\ddot{q}_{m_0} = -3 \text{ rad.s}^{-2}$  is used. Top to bottom: position, velocity, acceleration, jerk and  $n_3$ . See  $z_5$  and  $z_6$  in Fig. 2.17

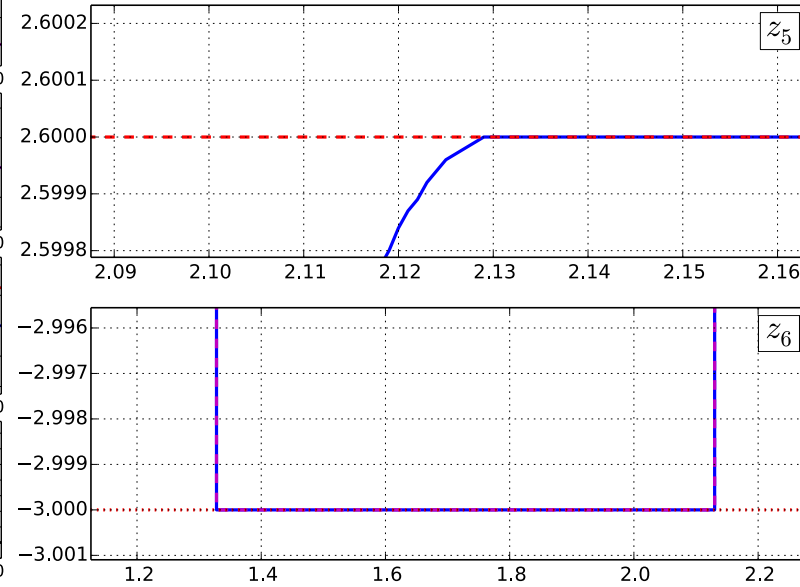


Figure 2.17: Zooms corresponding to Fig. 2.16.

### 2.4.2.2 Joint position constraint incompatibility with jerk limits 1

We consider a braking phase for a joint moving towards its upper position limit  $q_M$ . In this case, only the lower limit  $\ddot{q}_m$  of jerk capabilities is considered. Maximum producible deceleration  $\ddot{q}_m$  is not taken into account. The extended state  $S$  of the system during this implicitly induced braking phase, that makes the joint's actuator stop at its upper position limit  $q_M$ , can be described as:

$$\begin{aligned}
 S_{|k+1} & \begin{cases} q_{|k+1} = q_{|k} + \dot{q}_{|k} \delta t, \\ \dot{q}_{|k+1} = \dot{q}_{|k} + \ddot{q} \delta t, \\ \ddot{q}_{|k+1} = \ddot{q}_{|k} + \ddot{q}_m \delta t; \end{cases} \\
 S_{|k+2} & \begin{cases} q_{|k+2} = q_{|k+1} + \dot{q}_{|k+1} \delta t, \\ \dot{q}_{|k+2} = \dot{q}_{|k+1} + \ddot{q}_{|k+1} \delta t, \\ \ddot{q}_{|k+2} = \ddot{q}_{|k+1} + \ddot{q}_m \delta t; \end{cases} \\
 & \quad \vdots \quad \quad \quad \vdots \\
 S_{|k+n_5} & \begin{cases} q_{|k+n_5} = q_{|k+n_5-1} + \dot{q}_{|k+n_5-1} \delta t, \\ \dot{q}_{|k+n_5} = \dot{q}_{|k+n_5-1} + \ddot{q}_{|k+n_5-1} \delta t, \\ \ddot{q}_{|k+n_5} = \ddot{q}_{|k+n_5-1} + \ddot{q}_m \delta t. \end{cases}
 \end{aligned} \tag{2.31}$$

With:  $q_{|k} \geq 0$ ,  $\dot{q}_{|k} \geq 0$  and  $\ddot{q}_m \leq 0$ . The joint position evolution within  $n_5$  iterations is equal to the general form<sup>15</sup> of the numerical sequence (2.31):

$$q_{|k+n_5} = q_{|k} + n_5 \dot{q}_{|k} \delta t + \frac{(n_5^2 - n_5)}{2} \ddot{q}_{|k} \delta t^2 + \left( \frac{n_5^3}{6} - \frac{n_5^2}{2} + \frac{n_5}{3} \right) \ddot{q}_m \delta t^3. \tag{2.32}$$

The condition  $q_{|k+n_5} \leq q_M$  for all integer  $n_5$  leads to:

$$\ddot{q}_{|k}^c \leq \frac{2(q_M - q_{|k})}{(n_5^2 - n_5) \delta t^2} - \frac{2\dot{q}_{|k}}{(n_5 - 1) \delta t} - \frac{\left( \frac{n_5^2}{3} - n_5 + \frac{2}{3} \right)}{(n_5 - 1)} \ddot{q}_m \delta t, \tag{2.33}$$

with  $n_5$ , the integer minimizing the right-hand side of (2.33). It can be computed analytically. By differentiating this expression w.r.t  $n_5$ :

$$\begin{cases} n_5 \geq 3, \\ - \left( \frac{1}{3} \ddot{q}_m \delta t^3 \right) n_5^4 + \left( \frac{2}{3} \ddot{q}_{|m} \delta t^3 \right) n_5^3 \\ + \left( 2\dot{q}_{|k} \delta t - \frac{1}{3} \ddot{q}_m \delta t^3 \right) n_5^2 \\ + 4(q_M - q_{|k}) n_5 - 2(q_M - q_{|k}) = 0. \end{cases} \tag{2.34}$$

<sup>15</sup>Computed using Maple [Maplesoft Inc. 2016].

$n_5$  is the rounded-up real root of (2.34). It must be  $\geq 3$  so that  $\ddot{q}_m$  is always part of (2.33). Following the same reasoning for the lower position limit, the condition  $q_{|k+n_6} \geq q_m$  for all integer  $n_6 \geq 3$  can be reflected on the acceleration control variable as:

$$\ddot{q}_{|k}^c \geq \frac{2(q_m - q_{|k})}{(n_6^2 - n_6) \delta t^2} - \frac{2\dot{q}_{|k}}{(n_6 - 1)\delta t} - \frac{\left(\frac{n_6^2}{3} - n_6 + \frac{2}{3}\right) \ddot{q}_M \delta t}{(n_6 - 1)} \quad (2.35)$$

With:

$$\left\{ \begin{array}{l} n_6 \geq 3, \\ -\left(\frac{1}{3} \ddot{q}_M \delta t^3\right) n_6^4 + \left(\frac{2}{3} \ddot{q}_{|M} \delta t^3\right) n_6^3 \\ + \left(2\dot{q}_{|k} \delta t - \frac{1}{3} \ddot{q}_M \delta t^3\right) n_6^2 \\ + 4(q_m - q_{|k})n_6 - 2(q_m - q_{|k}) = 0. \end{array} \right. \quad (2.36)$$

$n_6$  is the rounded up real root of (2.36) that maximizes the right-hand side of (2.35). We recall that:  $n_5$  is the number of control time-steps corresponding to the duration of a braking phase implicitly induced by an actuator moving towards its upper position limit  $q_M$ , and during which both its acceleration and velocity are brought to zero by jerking with maximum negative producible jerk  $\ddot{q}_m$  and;  $n_6$  is the number of control time-steps corresponding to the duration of a braking phase implicitly induced by an actuator moving towards its lower position limit  $q_m$ , and during which both its acceleration and velocity are brought to zero by jerking with maximum positive producible jerk  $\ddot{q}_M$ .

$f_\theta$  and  $f_\eta$  are respectively equivalent to the right-hand sides of (2.35) and (2.33).

---

**Algorithm 1** Compute  $n_5, n_6, f_\theta$  and  $f_\eta$

---

**Require:**  $q_M, q_m, q_{|k}, \dot{q}_{|k}, \ddot{q}_M, \ddot{q}_m, \delta t$

```

1:  $f_{\theta_{max}} \leftarrow \ddot{q}_m$  ▷ temporary lower bound of  $\ddot{q}_{|k}^c$ 
2:  $f_{\eta_{min}} \leftarrow \ddot{q}_M$  ▷ temporary upper bound of  $\ddot{q}_{|k}^c$ 
3: for ( $i = 3 \rightarrow N$ ) do
4:    $n_5^* \leftarrow i$     $n_6^* \leftarrow i$ 
5:    $f_\theta^* \leftarrow f_\theta(q_m, \ddot{q}_M, q_{|k}, \dot{q}_{|k}, \ddot{q}_M, n_6^*)$ 
6:    $f_\eta^* \leftarrow f_\eta(q_M, \ddot{q}_m, q_{|k}, \dot{q}_{|k}, \ddot{q}_m, n_5^*)$ 
7:   if ( $f_\theta^* \geq f_{\theta_{max}}$ ) then
8:      $f_{\theta_{max}} \leftarrow f_\theta^*$     $n_6 \leftarrow n_6^*$ 
9:   end if
10:  if ( $f_\eta^* \leq f_{\eta_{min}}$ ) then
11:     $f_{\eta_{min}} \leftarrow f_\eta^*$     $n_5 \leftarrow n_5^*$ 
12:  end if
13: end for
14:  $f_\theta \leftarrow f_{\theta_{max}}$     $f_\eta \leftarrow f_{\eta_{min}}$ 
15: return  $n_5, n_6, f_\theta, f_\eta$ 

```

---

Finally,  $n_5, n_6, f_\theta$  and  $f_\eta$  can be computed numerically as shown by Algorithm 1.  $N$  in Algorithm 1 can be fixed heuristically, it must be however  $\geq$  to the solutions of (2.34) and (2.36). More details on how to compute this parameter in Section 2.6.



### 2.4.2.2.1 Illustration 3

For this simulation, using the test case scenario, the controller is implemented with the new formulation of the constraint on articular position (2.33), that takes into account the available jerk capability ( $\ddot{q}_{m_0} = -25 \text{ rad.s}^{-3}$ ). The right-hand side of the constraint on articular jerk (2.11d) ( $\ddot{q}_{0|k} \leq \ddot{q}_{M_0} = 25 \text{ rad.s}^{-3}$ ) is also included in the configuration of the controller. As for the previous simulations, joint 0 moves towards its upper position limit  $q_{M_0}$ .

Fig. 2.18 illustrates how joint 0 starts braking with maximum negative jerk  $\ddot{q}_{m_0}$  until its upper position limit is reached. The amount of “charged” deceleration should then be brought to 0. However, because of the constraint on articular jerk, the amount of deceleration *loaded* in the joint cannot be “de-charged” instantaneously. The joint is then forced to move away from  $q_{M_0}$  and oscillates until completely and progressively “de-charging” the amount of deceleration ( $\ddot{q}_{m_0} \leq 0$ ) it contains.

On the other hand, in case of an actuator with higher jerk capabilities (e.g.,  $\ddot{q}_{m_0} = -2500 \text{ rad.s}^{-3}$ ), the decaying oscillations can be quite reduced (see Fig. 2.20). This makes the proposed new formulation of the constraint on articular position more convenient when using actuators that can generate larger amounts of jerk. The maximum amplitude of the induced oscillations is noticeably diminished: from  $0.8 \text{ rad}$  in Fig. 2.18 to  $0.025 \text{ rad}$  in Fig. 2.20. Note that the main reason for these oscillations is the non-**complete** description of the braking phase in (2.31). Indeed, the *deceleration de-charging with positive jerk* sub-phase (sub-phase ③ in Fig. 2.11) is not considered in this version of the formulation of the articular position constraint. Also, as it appears in Fig. 2.19 and Fig. 2.21, the lower jerk limit is not perfectly satisfied. This is mainly due to the *discrete* description (2.31) of the state of the joint during the braking phase. In such case, including the *hard-coded* constraint (2.11d) on the negative articular jerk ( $\ddot{q}_{0|k} \geq \ddot{q}_{m_0} = -25 \text{ rad.s}^{-3}$ ) in the configuration of the controller will inevitably result into an infeasible control problem.

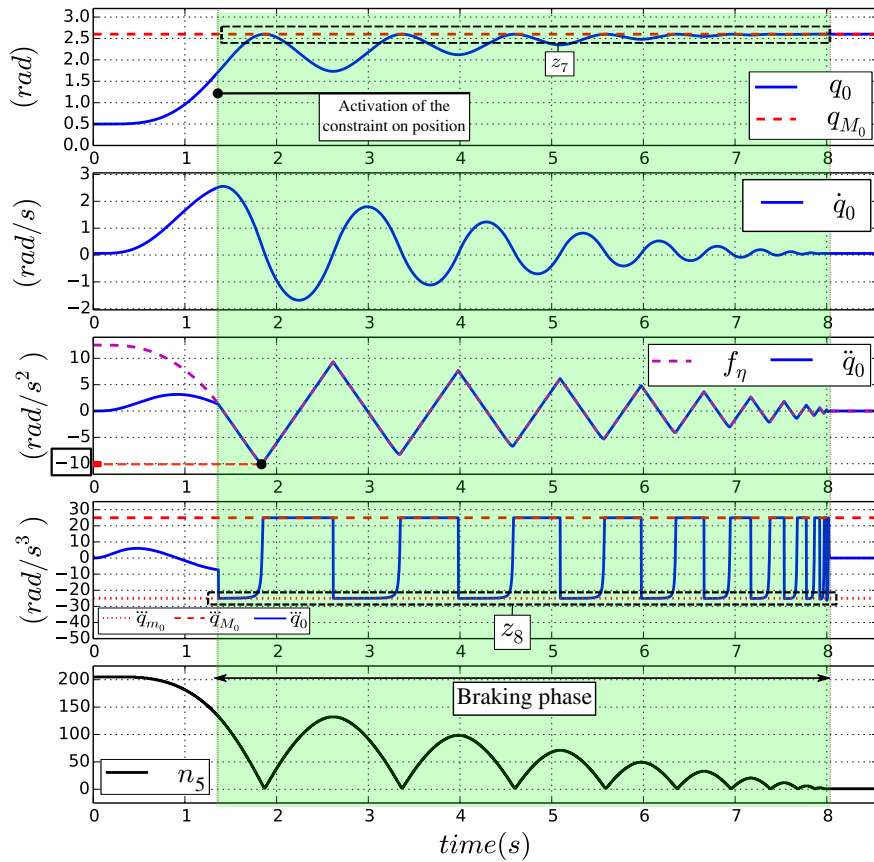


Figure 2.18: Extended state  $S$  of joint 0 during the braking phase implicitly induced to cope with an upper position limit  $q_{M_0}$ . The new version of the formulation of the constraint on articular position (2.33) that takes into account the jerk capability ( $\ddot{q}_{m_0} = -25 \text{ rad.s}^{-3}$ ) is used. Top to bottom: position, velocity, acceleration, jerk and  $n_5$ . See  $z_7$  and  $z_8$  in Fig. 2.15.

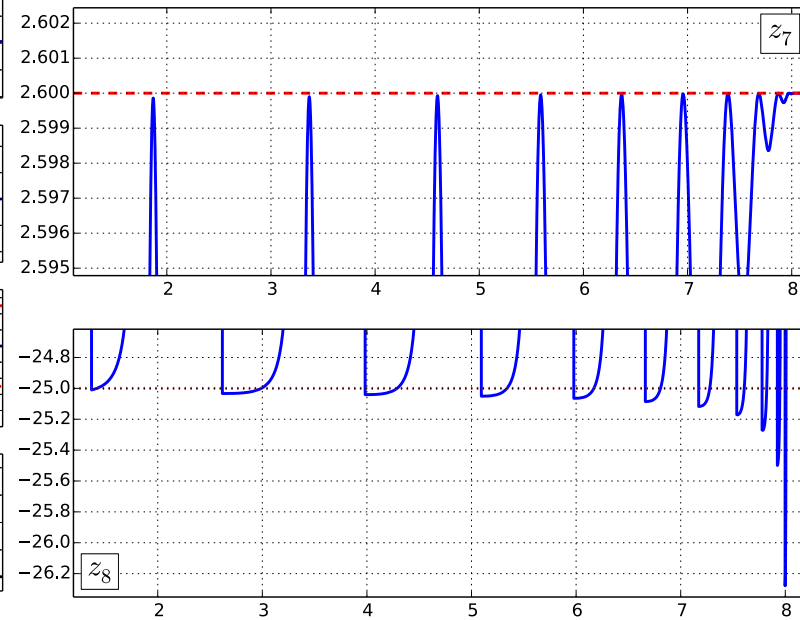


Figure 2.19: Zooms corresponding to Fig. 2.18.

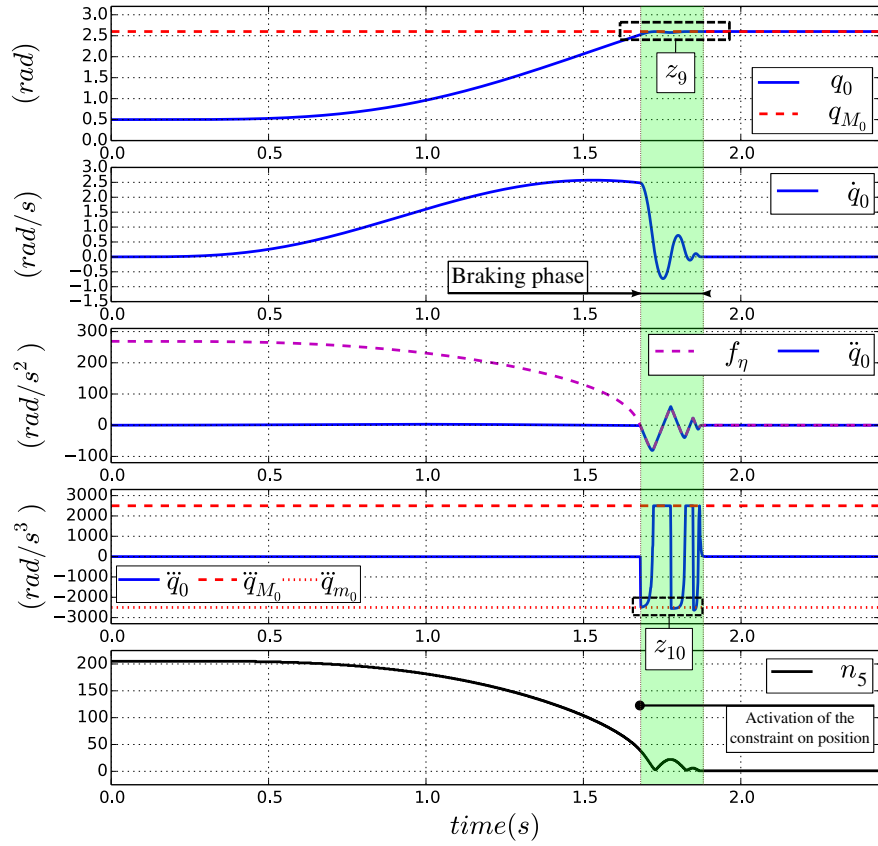


Figure 2.20: Extended state  $S$  of joint 0 during the braking phase implicitly induced to cope with an upper position limit  $q_{M_0}$ . The new version of the formulation of the constraint on articular position (2.33) that takes into account the jerk capability ( $\ddot{q}_{m_0} = -2500 \text{ rad.s}^{-3}$ ) is used. Top to bottom: position, velocity, acceleration, jerk and  $n_5$ . See  $z_9$  and  $z_{10}$  in Fig. 2.21.

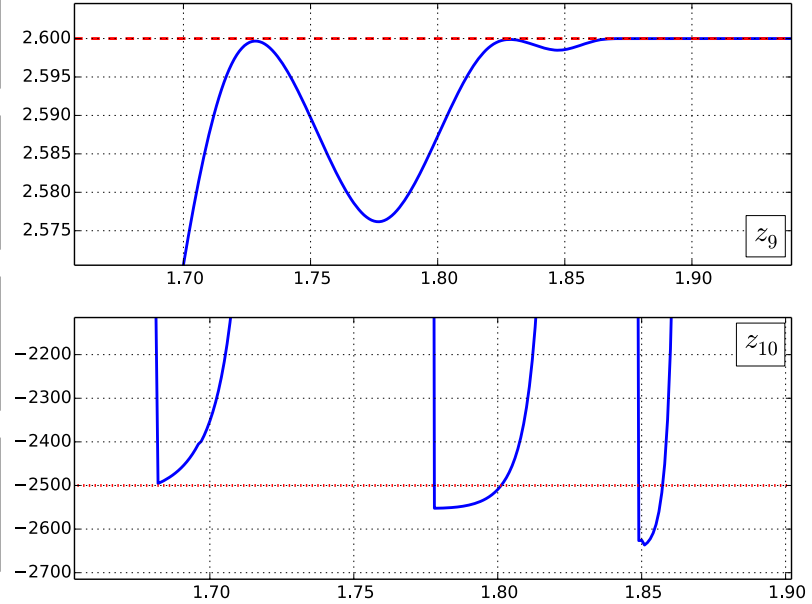


Figure 2.21: Zooms corresponding to Fig. 2.20.



When developed, (2.38) is written:

$$\begin{aligned}
q_{|k+n_7+n_9} &= q_{|k} + (n_7 + n_9) \dot{q}_{|k} \delta t + \left[ (n_7 n_9) + \frac{(n_7^2 - n_7) + (n_9^2 - n_9)}{2} \right] \ddot{q}_{|k} \delta t^2 \\
&+ \left[ \frac{n_7 (n_9^2 - n_9) + n_9 (n_7^2 - n_7)}{2} + \left( \frac{n_7^3}{6} - \frac{n_7^2}{2} + \frac{n_7}{3} \right) \right] \ddot{q}_m \delta t^3 \\
&+ \left( \frac{n_9^3}{6} - \frac{n_9^2}{2} + \frac{n_9}{3} \right) \ddot{q}_M \delta t^3.
\end{aligned} \tag{2.40}$$

Finally, the condition  $q_{|k+n_7+n_9} \leq q_M$  for all integers  $(n_7, n_9)$  leads to:

$$\begin{aligned}
\ddot{q}_{|k}^c &\leq \frac{(q_M - q_{|k})}{\left[ (n_7 n_9) + \frac{(n_7^2 - n_7) + (n_9^2 - n_9)}{2} \right] \delta t^2} + \frac{(n_7 + n_9) \dot{q}_{|k}}{\left[ (n_7 n_9) + \frac{(n_7^2 - n_7) + (n_9^2 - n_9)}{2} \right] \delta t} \\
&+ \frac{\left[ \frac{n_7 (n_9^2 - n_9) + n_9 (n_7^2 - n_7)}{2} + \left( \frac{n_7^3}{6} - \frac{n_7^2}{2} + \frac{n_7}{3} \right) \right] \ddot{q}_m \delta t}{\left[ (n_7 n_9) + \frac{(n_7^2 - n_7) + (n_9^2 - n_9)}{2} \right]} \\
&+ \frac{\left( \frac{n_9^3}{6} - \frac{n_9^2}{2} + \frac{n_9}{3} \right) \ddot{q}_M \delta t}{\left[ (n_7 n_9) + \frac{(n_7^2 - n_7) + (n_9^2 - n_9)}{2} \right]}.
\end{aligned} \tag{2.41}$$

$(n_7, n_9)$  are the two integers minimizing the right-hand side of (2.41). Following the same reasoning for the lower position limit, the condition  $q_{|k+n_8+n_{10}} \geq q_m$  can be reflected on the acceleration control variable as:

$$\begin{aligned}
\ddot{q}_{|k}^c &\geq \frac{(q_m - q_{|k})}{\left[ (n_8 n_{10}) + \frac{(n_8^2 - n_8) + (n_{10}^2 - n_{10})}{2} \right] \delta t^2} + \frac{(n_8 + n_{10}) \dot{q}_{|k}}{\left[ (n_8 n_{10}) + \frac{(n_8^2 - n_8) + (n_{10}^2 - n_{10})}{2} \right] \delta t} \\
&+ \frac{\left[ \frac{n_8 (n_{10}^2 - n_{10}) + n_{10} (n_8^2 - n_8)}{2} + \left( \frac{n_8^3}{6} - \frac{n_8^2}{2} + \frac{n_8}{3} \right) \right] \ddot{q}_M \delta t}{\left[ (n_8 n_{10}) + \frac{(n_8^2 - n_8) + (n_{10}^2 - n_{10})}{2} \right]} \\
&+ \frac{\left( \frac{n_{10}^3}{6} - \frac{n_{10}^2}{2} + \frac{n_{10}}{3} \right) \ddot{q}_m \delta t}{\left[ (n_8 n_{10}) + \frac{(n_8^2 - n_8) + (n_{10}^2 - n_{10})}{2} \right]},
\end{aligned} \tag{2.42}$$

with  $(n_8, n_{10})$ : the two integers maximizing the right-hand side of (2.41).

$f_\mu$  and  $f_\lambda$  are respectively equivalent to the right-hand sides of (2.42) and (2.41).

---

**Algorithm 2** Compute  $n_7, n_8, n_9, n_{10}, f_\mu$  and  $f_\lambda$ 


---

**Require:**  $q_M, q_m, q_{|k}, \dot{q}_{|k}, \ddot{q}_M, \ddot{q}_m, \delta t$ 

```

1:  $f_{\mu_{max}} \leftarrow \ddot{q}_m$  ▷ temporary lower bound of  $\ddot{q}_{|k}^c$ 
2:  $f_{\lambda_{min}} \leftarrow \ddot{q}_M$  ▷ temporary upper bound of  $\ddot{q}_{|k}^c$ 
3: for ( $i = 1 \rightarrow N$ ) do
4:    $n_8^* \leftarrow i$     $n_7^* \leftarrow i$ 
5:   if ( $\dot{q}_{|k} \geq 0$ ) then  $n_9^* \leftarrow n_7^* - \left| \frac{\ddot{q}_{|k}}{\ddot{q}_m \delta t} \right|$     $n_{10}^* \leftarrow n_8^* + \left| \frac{\ddot{q}_{|k}}{\ddot{q}_M \delta t} \right|$ 
6:   if ( $\dot{q}_{|k} < 0$ ) then
7:      $n_9^* \leftarrow n_7^* + \left| \frac{\ddot{q}_{|k}}{\ddot{q}_M \delta t} \right|$     $n_{10}^* \leftarrow n_8^* - \left| \frac{\ddot{q}_{|k}}{\ddot{q}_m \delta t} \right|$ 
8:     if ( $n_7^* \leq 1$ ) then  $n_9^* \leftarrow \left| \frac{\ddot{q}_{|k}}{\ddot{q}_M \delta t} \right|$ 
9:     if ( $n_8^* \leq 1$ ) then  $n_{10}^* \leftarrow \left| \frac{\ddot{q}_{|k}}{\ddot{q}_m \delta t} \right|$ 
10:  end if
11:  if ( $n_7^* \leq 1$ ) then  $n_7^* \leftarrow 1$ 
12:  if ( $n_8^* \leq 1$ ) then  $n_8^* \leftarrow 1$ 
13:  if ( $n_9^* \leq 2$ ) then  $n_9^* \leftarrow 2$ 
14:  if ( $n_{10}^* \leq 2$ ) then  $n_{10}^* \leftarrow 2$ 
15:   $f_\mu^* \leftarrow f_\mu(q_m, \ddot{q}_M, q_{|k}, \dot{q}_{|k}, \ddot{q}_m, n_8^*, n_{10}^*)$ 
16:   $f_\lambda^* \leftarrow f_\lambda(q_M, \ddot{q}_m, q_{|k}, \dot{q}_{|k}, \ddot{q}_M, n_7^*, n_9^*)$ 
17:  if ( $f_\mu^* \geq f_{\mu_{max}}$ ) then
18:     $f_{\mu_{max}} \leftarrow f_\mu^*$     $n_8 \leftarrow n_8^*$     $n_{10}^a \leftarrow n_{10}^*$ 
19:  end if
20:  if ( $f_\lambda^* \leq f_{\lambda_{min}}$ ) then
21:     $f_{\lambda_{min}} \leftarrow f_\lambda^*$     $n_7 \leftarrow n_7^*$     $n_9^a \leftarrow n_9^*$ 
22:  end if
23: end for
24:  $f_\mu \leftarrow f_{\mu_{max}}$     $f_\lambda \leftarrow f_{\lambda_{min}}$ 
25: return  $n_7, n_8, n_9, n_{10}, f_\mu, f_\lambda$ 

```

---

<sup>a</sup> For stability in the computation of  $f_\mu$  and  $f_\lambda$ ,  $(n_9^*, n_{10}^*)$  are used as real numbers (even if previously defined as numbers of iterations). The same for  $n_9$  and  $n_{10}$ .

$n_7, n_8, n_9, n_{10}, f_\mu$  and  $f_\lambda$  can be computed numerically as shown by Algorithm 2.  $N$  in Algorithm 2 is fixed heuristically, it must however be  $\geq$  to the total number of iterations needed to perform the braking movement described in (2.37). More details on how to compute this parameter in Section 2.6.

#### 2.4.2.3.1 Illustration 4

For this simulation, using the test case scenario, the controller is implemented with the new version of the formulation of the constraint on articular position (2.41), that takes into account both the positive and negative jerk capabilities ( $[\ddot{q}_{m_0}, \ddot{q}_{M_0}] = [-25, 25] \text{ rad.s}^{-3}$ ). In this case, joint 0 is also moving towards its upper position limit  $q_{M_0}$ .

Fig. 2.22 depicts how joint 0 starts braking with maximum producible negative (sub-phase ①) then positive (sub-phase ②) jerk until reaching its upper position limit. The amount of “charged” deceleration is then released and brought to 0 as the joint stops its motion at  $q_{M_0}$ . We also highlight that in this case, no oscillations are induced when using this *more complete* description of the braking phase for the formulation of the constraint on articular position. On the other hand, we underline an exceeding of  $0.013 \text{ rad.s}^{-3}$  over the lower jerk limit ( $\ddot{q}_{m_0} = -25 \text{ rad.s}^{-3}$ ) (see Fig. 2.23). Also, as it appears in the articular jerk profile, the positive jerk capability is not used to its full potential for the “de-charging” of the accumulated deceleration sub-phase. Which is mainly caused by the *discrete* description of the state of joint 0 during its braking phase (2.37) and consequently the discrete new formulation of the articular position constraint.

#### 2.4.2.4 Joint position constraint incompatibility with both deceleration and jerk limits 1

In all the previously introduced new formulations, the articular jerk and deceleration capabilities have been independently considered for the different mathematical expressions of the constraint on articular position. As explained in Section 2.4.2 regarding the description of the *complete* braking phase needed to *properly* cope with an articular position limit, these capabilities ( $\ddot{q}_m, \ddot{q}_m, \ddot{q}_M$ ), considering an upper position limit  $q_M$ , should all appear together in the expression of the constraint on articular position. During the needed braking phase, the joint starts jerking negatively with maximum producible jerk  $\ddot{q}_m$  during  $n_{11}$  iterations. If maximum deceleration  $\ddot{q}_m$  is reached, the joint brings its articular jerk to 0 and the “charged” deceleration ( $\ddot{q} = \ddot{q}_m$ ) is applied for  $n_{13}$  iterations. Finally, the amount of deceleration in the joint is “de-charged” and released during several time-steps by jerking positively towards the position limit. In the following example, for **pedagogical** reasons, the braking phase to cope with an upper position limit is considered without the *deceleration de-charging* sub-phase (phase ③ in Fig. 2.11). The *complete* braking phase described hereby will be used for the final and complete formulation of the articular position constraint. The extended state  $S$  of the system during the  $(n_{11} + n_{13})$  iterations corresponding to this **truncated** “*complete* braking phase” can be described as:

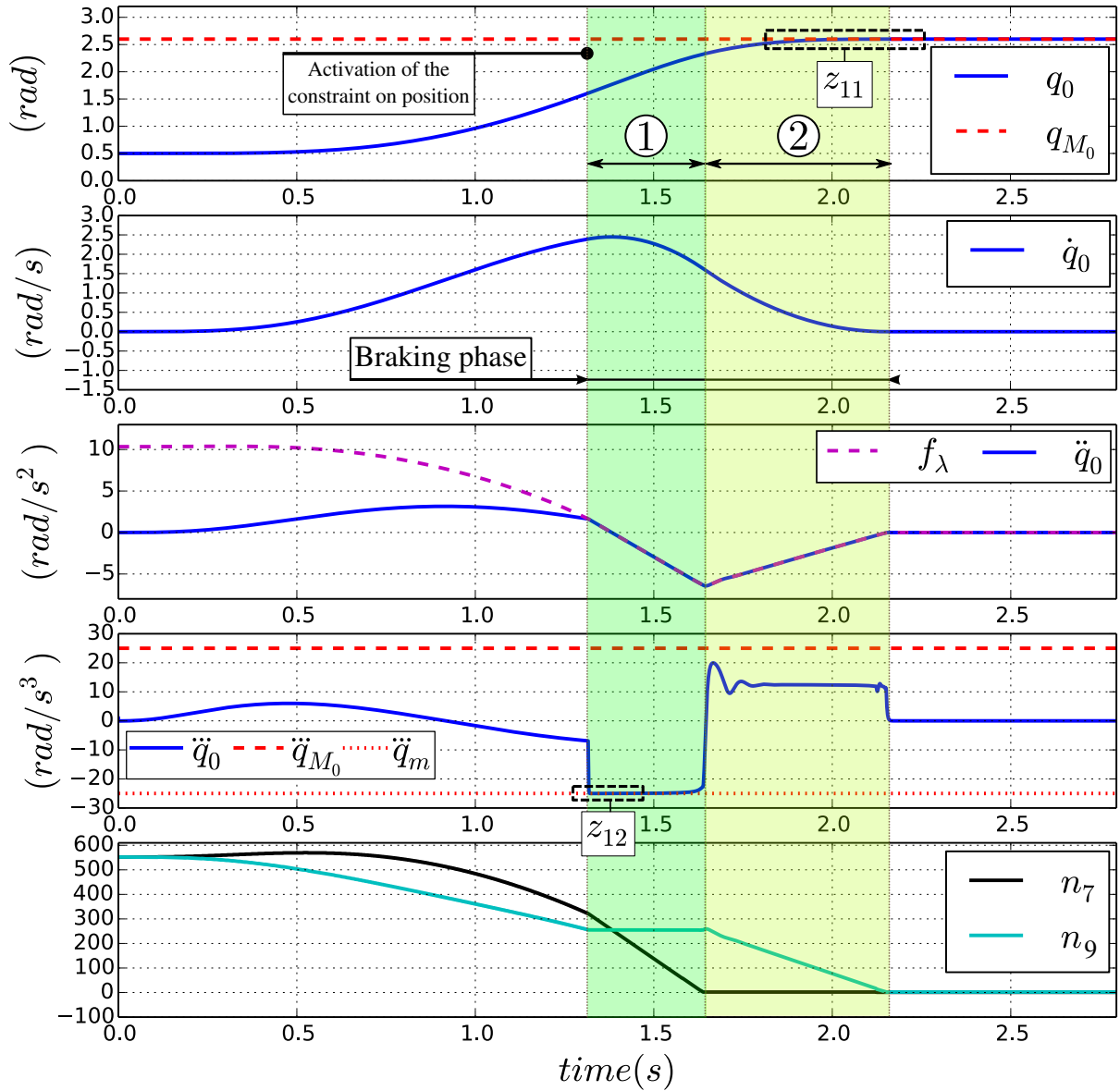


Figure 2.22: Extended state  $S$  of joint 0 during the braking phase implicitly induced to cope with an upper position limit  $q_{M_0}$ . The new formulation of the constraint on articular position (2.41) that takes into account both the positive and negative jerk capabilities  $[\ddot{q}_{m_0}, \ddot{q}_{M_0}] = [-25, 25] \text{ rad.s}^{-3}$  is used. Top to bottom: position, velocity, acceleration, jerk and  $(n_7, n_9)$ . See  $z_{10}$  and  $z_{11}$  in Fig. 2.23.



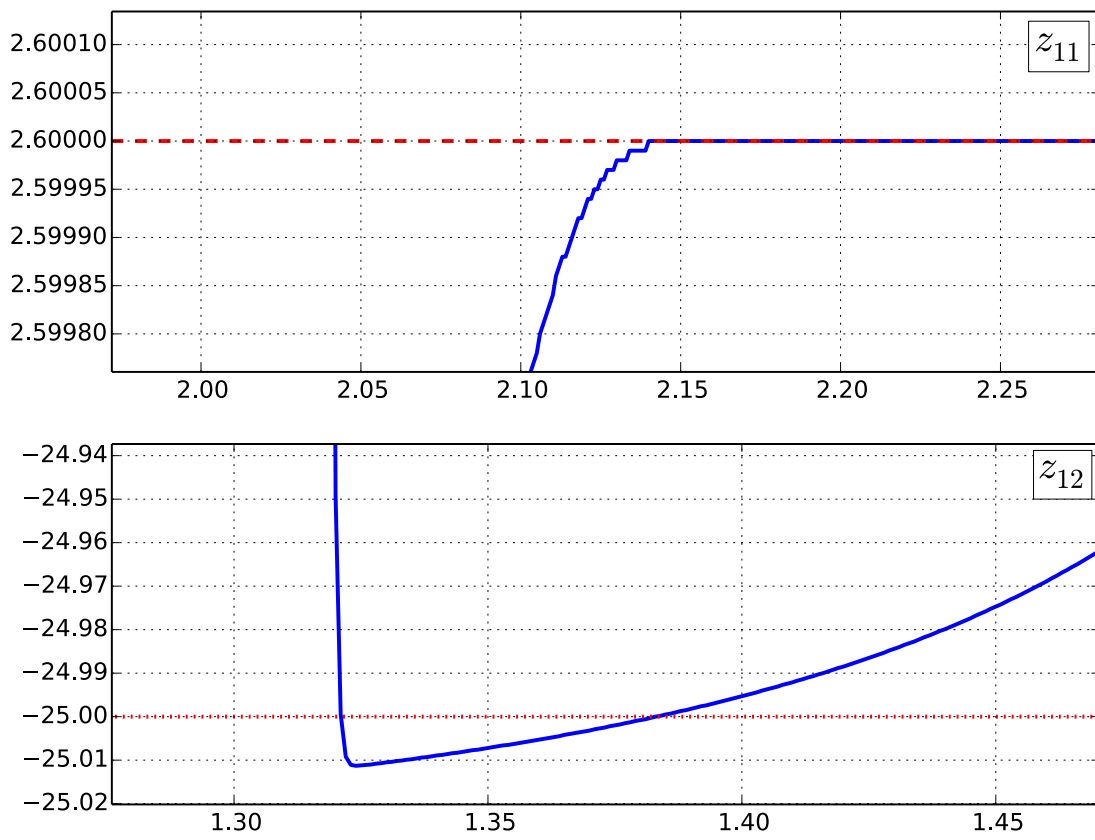


Figure 2.23: Zooms corresponding to Fig. 2.22.

$$\begin{aligned}
& \left. \begin{aligned}
S_{|k+1} \begin{cases} q_{|k+1} & = q_{|k} + \dot{q}_{|k}\delta t, \\ \dot{q}_{|k+1} & = \dot{q}_{|k} + \ddot{q}_{|k}\delta t, \\ \ddot{q}_{|k+1} & = \ddot{q}_{|k} + \ddot{q}_m\delta t; \\ \vdots & \vdots \end{cases} \\
S_{|k+n_{11}} \begin{cases} q_{|k+n_{11}} & = q_{|k+n_{11}-1} + \dot{q}_{|k+n_{11}-1}\delta t, \\ \dot{q}_{|k+n_{11}} & = \dot{q}_{|k+n_{11}-1} + \ddot{q}_{|k+n_{11}-1}\delta t, \\ \ddot{q}_{|k+n_{11}} & = \ddot{q}_{|k+n_{11}-1} + \ddot{q}_m\delta t; \end{cases}
\end{aligned} \right\} \text{Equivalent to (2.31)} \\
& \left. \begin{aligned}
S_{|k+n_{11}+1} \begin{cases} q_{|k+n_{11}+1} & = q_{|k+n_{11}} + \dot{q}_{|k+n_{11}}\delta t, \\ \dot{q}_{|k+n_{11}+1} & = \dot{q}_{|k+n_{11}} + \ddot{q}_m\delta t; \\ \vdots & \vdots \end{cases} \\
S_{|k+n_{11}+n_{13}} \begin{cases} q_{|k+n_{11}+n_{13}} & = q_{|k+n_{11}+n_{13}-1} + \dot{q}_{|k+n_{11}+n_{13}-1}\delta t, \\ \dot{q}_{|k+n_{11}+n_{13}} & = \dot{q}_{|k+n_{11}+n_{13}-1} + \ddot{q}_m\delta t. \end{cases}
\end{aligned} \right\} \text{Equivalent to (2.24)}
\end{aligned} \tag{2.43}$$

The evolution of the joint's articular position in  $(n_{11} + n_{13})$  iterations is equal to the general form<sup>17</sup> of the numerical sequence (2.43):

$$q_{|k+n_{11}+n_{13}} = q_{|k+n_{11}} + n_{13}\dot{q}_{|k+n_{11}}\delta t + \frac{(n_{13}^2 - n_{13})}{2}\ddot{q}_m\delta t^2, \tag{2.44}$$

with  $q_{|k+n_{11}}$  and  $\dot{q}_{|k+n_{11}}$ , respectively equivalent to (2.32) and (2.13). When developed, (2.44) is written:

$$\begin{aligned}
q_{|k+n_{11}+n_{13}} &= q_{|k} + (n_{11} + n_{13})\dot{q}_{|k}\delta t \\
&+ \left( \frac{n_{11}^2 - n_{11}}{2} + n_{11}n_{13} \right) \ddot{q}_{|k}\delta t^2 \\
&+ \left[ \left( \frac{n_{11}^3}{6} - \frac{n_{11}^2}{2} + \frac{n_{11}}{3} \right) + \frac{n_{13}(n_{11}^2 - n_{11})}{2} \right] \ddot{q}_m\delta t^3 \\
&+ \frac{(n_{13}^2 - n_{13})}{2}\ddot{q}_m\delta t^2,
\end{aligned} \tag{2.45}$$

<sup>17</sup>Computed using Maple [Maplesoft Inc. 2016].

with:  $q_{|k} \geq 0$ ,  $\dot{q}_{|k} \geq 0$ ,  $\ddot{q}_m \leq 0$  and  $\ddot{q}_m \leq 0$ . The condition  $q_{|k+n_{11}+n_{13}} \leq q_M$  for all integers  $(n_{11}, n_{13})$  in case of a control at the dynamic-level leads to:

$$\begin{aligned} \ddot{q}_{|k}^c \leq & \frac{(q_M - q_{|k})}{\left(\frac{n_{11}^2 - n_{11}}{2} + n_{11}n_{13}\right) \delta t^2} - \frac{(n_{11} + n_{13})}{\left(\frac{n_{11}^2 - n_{11}}{2} + n_{11}n_{13}\right) \delta t} \dot{q}_{|k} \\ & - \frac{\left[\frac{n_{11}^3}{6} - \frac{n_{11}^2}{2} + \frac{n_{11}}{3} + \frac{n_{13}(n_{11}^2 - n_{11})}{2}\right]}{\left(\frac{n_{11}^2 - n_{11}}{2} + n_{11}n_{13}\right)} \ddot{q}_m \delta t \\ & - \frac{(n_{13}^2 - n_{13})}{(n_{11}^2 - n_{11} + 2n_{11}n_{13})} \ddot{q}_m. \end{aligned} \quad (2.46)$$

In case the control is performed at the kinematic-level, the constraint can be reflected on kinematic-control input  $\dot{q}_{|k}^c$  as:

$$\begin{aligned} \dot{q}_{|k}^c \leq & \frac{(q_M - q_{|k})}{(n_{11} + n_{13}) \delta t} - \frac{\left[\frac{(n_{11}^2 - n_{11})}{2} + n_{11}n_{13}\right]}{(n_{11} + n_{13})} \dot{q}_{|k} \delta t \\ & - \frac{\left[\frac{n_{11}^3}{6} + \frac{n_{11}^2}{2} + \frac{n_{11}}{3} + \frac{n_{13}(n_{11}^2 - n_{11})}{2}\right]}{(n_{11} + n_{13})} \ddot{q}_m \delta t^2 \\ & - \frac{(n_{13}^2 - n_{13})}{2(n_{11} + n_{13})} \ddot{q}_m \delta t, \end{aligned} \quad (2.47)$$

with  $(n_{11}, n_{13})$ , the two integers minimizing the right-hand side of (2.46). Following the same reasoning for the lower position limit, for all integers  $(n_{12}, n_{14})$ , the constraint  $q_{|k+n_{12}+n_{14}} \geq q_m$  can be reflected on the acceleration control variable as:

$$\begin{aligned} \ddot{q}_{|k}^c \geq & \frac{(q_m - q_{|k})}{\left(\frac{n_{12}^2 - n_{12}}{2} + n_{12}n_{14}\right) \delta t^2} \\ & - \frac{\left[\frac{n_{12}^3}{6} - \frac{n_{12}^2}{2} + \frac{n_{12}}{3} + \frac{n_{14}(n_{12}^2 - n_{12})}{2}\right]}{\left(\frac{n_{12}^2 - n_{12}}{2} + n_{12}n_{14}\right)} \ddot{q}_M \delta t \\ & - \frac{(n_{12} + n_{14})}{\left(\frac{n_{12}^2 - n_{12}}{2} + n_{12}n_{14}\right) \delta t} \dot{q}_{|k} - \frac{(n_{14}^2 - n_{14})}{(n_{12}^2 - n_{12} + 2n_{12}n_{14})} \ddot{q}_M, \end{aligned} \quad (2.48)$$

with  $(n_{12}, n_{14})$ , the two integers maximizing the right-hand side of (2.48).  $f_\rho$  and  $f_\pi$  are respectively equivalent to the right-hand sides of (2.46) and (2.48).

---

**Algorithm 3** Compute  $n_{11}, n_{12}, n_{13}, n_{14}, f_\rho$  and  $f_\pi$

---

**Require:**  $q_M, q_m, q_{|k}, \ddot{q}_{|k}, \ddot{q}_M, \ddot{q}_m, \ddot{q}_M, \ddot{q}_m, \delta t$

- 1:  $f_{\rho_{max}} \leftarrow \ddot{q}_m$
- 2:  $f_{\pi_{min}} \leftarrow \ddot{q}_M$
- 3: **for** ( $i_1 = 3 \rightarrow N_1$ ) **do**
- 4:      $n_{11}^* \leftarrow 3$
- 5:      $n_{12}^* \leftarrow 3$
- 6:     **for** ( $i_2 = 1 \rightarrow N_2$ ) **do**
- 7:          $n_{13}^* \leftarrow i_2$
- 8:          $n_{14}^* \leftarrow i_2$
- 9:          $f_\rho^* \leftarrow f_\rho(q_m, q_{|k}, \ddot{q}_{|k}, \ddot{q}_M, \ddot{q}_m, n_{12}^*, n_{14}^*)$
- 10:          $f_\pi^* \leftarrow f_\pi(q_M, q_{|k}, \ddot{q}_{|k}, \ddot{q}_M, \ddot{q}_m, n_{11}^*, n_{13}^*)$
- 11:         **if** ( $f_\rho^* \geq f_{\rho_{max}}$ ) **then**
- 12:              $f_{\rho_{max}} \leftarrow f_\rho^*$
- 13:              $n_{12} \leftarrow n_{12}^*$
- 14:              $n_{14} \leftarrow n_{14}^*$
- 15:         **end if**
- 16:         **if** ( $f_\pi^* \leq f_{\pi_{min}}$ ) **then**
- 17:              $f_{\pi_{min}} \leftarrow f_\pi^*$
- 18:              $n_{11} \leftarrow n_{11}^*$
- 19:              $n_{13} \leftarrow n_{13}^*$
- 20:         **end if**
- 21:     **end for**
- 22: **end for**
- 23:  $f_\rho \leftarrow f_{\rho_{max}}$
- 24:  $f_\pi \leftarrow f_{\pi_{min}}$
- 25: **return**  $n_{11}, n_{12}, n_{13}, n_{14}, f_\rho, f_\pi$

---

$n_{11}, n_{13}, n_{12}, n_{14}, f_\rho$  and  $f_\pi$  can be computed numerically as shown by Algorithm 3.  $N_1$  and  $N_2$  in Algorithm 3 are fixed heuristically, they must however be  $\geq$  to the number of iterations needed to perform the braking movement described in (2.43). More details on how to compute this parameter in Section 2.6.

#### 2.4.2.4.1 Illustration 5

To illustrate the behaviour of the robot using this new theoretical result, the controller is implemented with the new version of the formulation of the constraint on articular position (2.46), that takes into account both the articular jerk ( $\ddot{q}_{m_0} = -25 \text{ rad.s}^{-3}$ ) and the producible articular deceleration  $\ddot{q}_{m_0}$ . In Fig. 2.18, the amplitude of the first oscillation of the joint's articular acceleration reaches  $-10 \text{ rad.s}^{-2}$ . To demonstrate the interest of using the new formulation of the position constraint,  $\ddot{q}_{m_0}$  is fixed at  $-3 \text{ rad.s}^{-2}$ . The same movement of joint 0 moving towards its upper position limit  $q_{M_0}$  is considered; and no constraint on the articular jerk (2.11d) is included in the configuration of the controller.

Fig. 2.24 illustrates how joint 0 starts braking with maximum negative jerk ( $\ddot{q}_0 = \ddot{q}_{m_0}$ ) (sub-phase ①) until maximum deceleration ( $\ddot{q}_0 = \ddot{q}_{m_0}$ ) is reached. This deceleration is then

used to stop the joint at its upper position limit  $q_{M_0}$  (sub-phase ②). As the articular jerk is not constrained, the amount of deceleration *loaded* in the joint is almost instantaneously “*de-charged*”, inducing a high peak of jerk. Otherwise, if the constraint on articular jerk (2.11d) are included in the configuration of the controller, decaying oscillations would appear as in Fig. 2.18. We also underline an exceeding of  $(0.013 \text{ rad.s}^{-3})$  over the lower jerk limit  $\ddot{q}_{m_0}$ . On the other hand, the deceleration limit  $\ddot{q}_{m_0}$  is well satisfied (see Fig. 2.24).

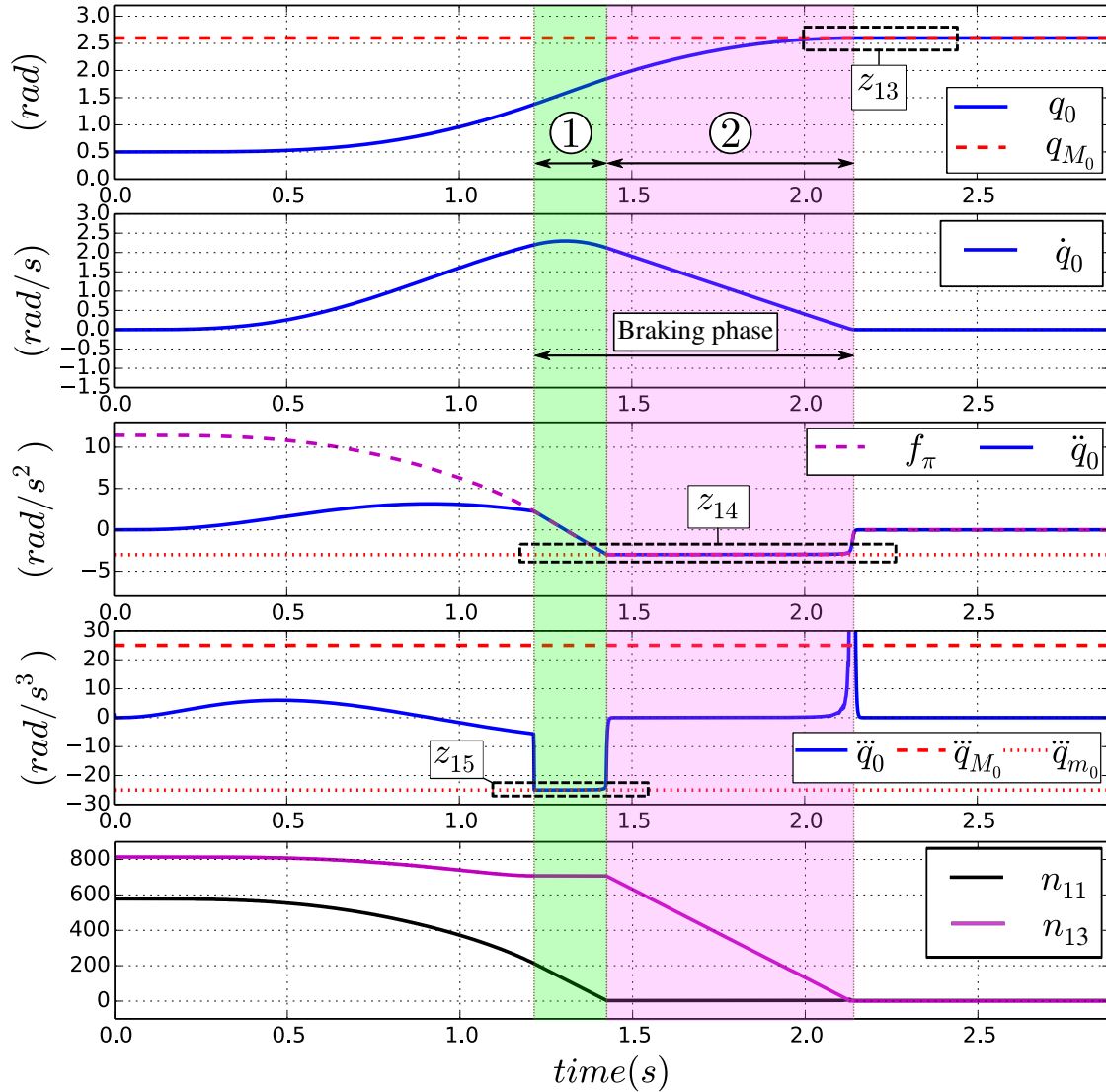


Figure 2.24: Extended state  $S$  of joint 0 during a *non-complete* braking phase implicitly induced to cope with an upper position limit  $q_{M_0}$ . The new formulation of the constraint on articular position (2.46), that takes into account both the articular jerk and deceleration capabilities ( $\ddot{q}_{m_0} = -25 \text{ rad.s}^{-3}$ ), ( $\ddot{q}_{m_0} = -3 \text{ rad.s}^{-2}$ ) is used. Top to bottom: position, velocity, acceleration, jerk and  $(n_{11}, n_{13})$ . See  $z_{13}$  and  $z_{14}$  in Fig. 2.25.

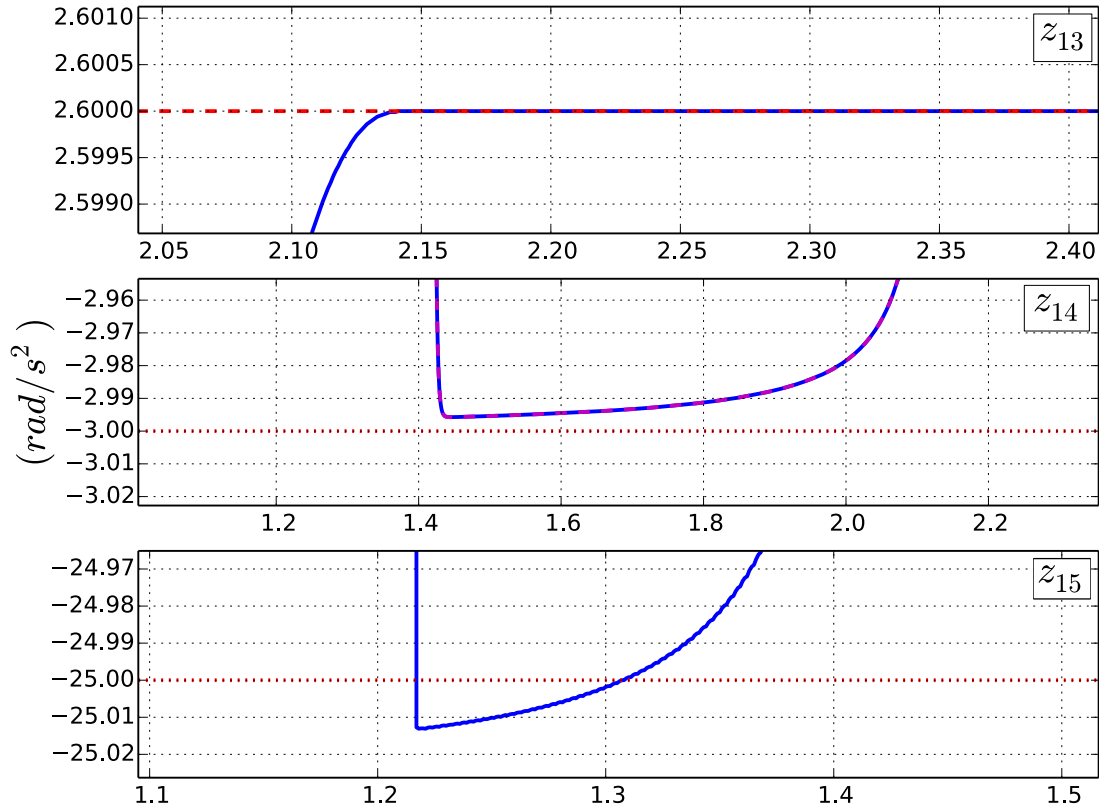


Figure 2.25: Zooms corresponding to Fig. 2.24.

#### 2.4.2.5 Joint position constraint incompatibility with both jerk and deceleration limits 2

Finally, the fully complete description of the braking phase is used to derive the *proper* and complete formulation of the constraint on articular position. When coping with an upper position limit  $q_M$ , such braking phase is as follows: 1) the joint starts jerking with maximum negative jerk  $\ddot{q}_m$  for  $n_{15}$  iterations until maximum producible articular deceleration  $\ddot{q}_m$  is reached. 2) This deceleration is then applied for  $n_{17}$  control time-steps then, 3) “*de-charged*” during  $n_{19}$  iterations by jerking positively with maximum producible jerk  $\ddot{q}_M$  (see Fig. 2.11). The extended state  $S$  of the system during this complete braking phase can be described as:



with:

$$\left\{ \begin{array}{l}
 q_{|k+n_{15}+n_{17}} = q_{|k+n_{15}} + n_{17}\dot{q}_{|k+n_{15}}\delta t + \frac{(n_{17}^2-n_{17})}{2}\ddot{q}_m\delta t^2, \\
 \quad = q_{|k} + (n_{15} + n_{17})\dot{q}_{|k}\delta t + \left[ (n_{15}n_{17}) + \frac{(n_{15}^2-n_{15})}{2} \right] \ddot{q}_{|k}\delta t^2 \\
 \quad \quad + \frac{(n_{17}^2-n_{17})}{2}\ddot{q}_m\delta t^2 \\
 \quad \quad + \left[ \frac{n_{17}(n_{15}^2-n_{15})}{2} + \left( \frac{n_{15}^3}{6} - \frac{n_{15}^2}{2} + \frac{n_{15}}{3} \right) \right] \ddot{q}_m\delta t^3, \\
 \dot{q}_{|k+n_{15}+n_{17}} = \dot{q}_{|k+n_{15}} + n_{17}\ddot{q}_m\delta t, \\
 \quad = \dot{q}_{|k} + n_{15}\ddot{q}_{|k}t + \frac{(n_{15}^2-n_{15})}{2}\ddot{q}_m\delta t^2 + n_{17}\ddot{q}_m\delta t, \\
 \ddot{q}_{|k+n_{15}+n_{17}} = \ddot{q}_{|k} + \ddot{q}_m\delta t.
 \end{array} \right. \quad (2.51)$$

$q_{|k+n_{15}}$ ,  $q_{|k+n_{15}+n_{17}}$  and  $\dot{q}_{|k+n_{15}}$  are respectively equivalent to (2.32), (2.45) and (2.13). When developed, (2.50) becomes:

$$\begin{aligned}
 q_{|k+n_{15}+n_{17}+n_{19}} &= q_{|k} + (n_{15} + n_{17} + n_{19})\dot{q}_{|k}\delta t \\
 &+ \left[ n_{15}(n_{17} + n_{19}) + \frac{(n_{15}^2 - n_{15}) + (n_{19}^2 - n_{19})}{2} \right] \ddot{q}_{|k}\delta t^2 \\
 &+ \left[ \frac{(n_{17} + n_{19}) + (n_{15}^2 - n_{15})}{2} + \left( \frac{n_{15}^3}{6} - \frac{n_{15}^2}{2} + \frac{n_{15}}{3} \right) \right] \ddot{q}_m\delta t^3 \\
 &+ \left[ (n_{17}n_{19}) + \frac{(n_{17}^2 - n_{17}) + (n_{19}^2 - n_{19})}{2} \right] \ddot{q}_m\delta t^2 \\
 &+ \left( \frac{n_{19}^3}{6} - \frac{n_{19}^2}{2} + \frac{n_{19}}{3} \right) \ddot{q}_M\delta t^3.
 \end{aligned} \quad (2.52)$$



With:  $q_{|k} \geq 0$ ,  $\dot{q}_{|k} \geq 0$ ,  $\ddot{q}_m \leq 0$ ,  $\ddot{q}_M \leq 0$  and  $\ddot{q}_M \geq 0$ . The condition  $q_{|k+n_{15}+n_{17}+n_{19}} \leq q_M$  for all integers  $(n_{15}, n_{17}, n_{19})$  leads to:

$$\ddot{q}_{|k}^c \leq \frac{(q_M - q_{|k})}{\left[ n_{15} (n_{17} + n_{19}) + \frac{(n_{15}^2 - n_{15}) + (n_{19}^2 - n_{19})}{2} \right]} \delta t^2 - \frac{(n_{15} + n_{17} + n_{19}) \dot{q}_{|k}}{\left[ n_{15} (n_{17} + n_{19}) + \frac{(n_{15}^2 - n_{15}) + (n_{19}^2 - n_{19})}{2} \right]} \delta t - \frac{\left[ \frac{(n_{17} + n_{19}) + (n_{15}^2 - n_{15})}{2} + \left( \frac{n_{15}^3}{6} - \frac{n_{15}^2}{2} + \frac{n_{15}}{3} \right) \right] \ddot{q}_m \delta t}{\left[ n_{15} (n_{17} + n_{19}) + \frac{(n_{15}^2 - n_{15}) + (n_{19}^2 - n_{19})}{2} \right]} - \frac{\left[ (n_{17} n_{19}) + \frac{(n_{17}^2 - n_{17}) + (n_{19}^2 - n_{19})}{2} \right] \ddot{q}_m}{\left[ n_{15} (n_{17} + n_{19}) + \frac{(n_{15}^2 - n_{15}) + (n_{19}^2 - n_{19})}{2} \right]} - \frac{\left[ \left( \frac{n_{19}^3}{6} - \frac{n_{19}^2}{2} + \frac{n_{19}}{3} \right) \right] \ddot{q}_M \delta t}{\left[ n_{15} (n_{17} + n_{19}) + \frac{(n_{15}^2 - n_{15}) + (n_{19}^2 - n_{19})}{2} \right]}. \quad (2.53)$$

$(n_{15}, n_{17}, n_{19})$  are the integers minimizing the right-hand side of (2.53). Following the same reasoning for the lower position limit, the condition  $q_{|k+n_{16}+n_{18}+n_{20}} \geq q_m$  can be reflected on the acceleration control variable as follows:

$$\ddot{q}_{|k}^c \geq \frac{(q_m - q_{|k})}{\left[ n_{16} (n_{18} + n_{20}) + \frac{(n_{16}^2 - n_{16}) + (n_{20}^2 - n_{20})}{2} \right]} \delta t^2 - \frac{(n_{16} + n_{18} + n_{20}) \dot{q}_{|k}}{\left[ n_{16} (n_{18} + n_{20}) + \frac{(n_{16}^2 - n_{16}) + (n_{20}^2 - n_{20})}{2} \right]} \delta t - \frac{\left[ \frac{(n_{18} + n_{20}) + (n_{16}^2 - n_{16})}{2} + \left( \frac{n_{16}^3}{6} - \frac{n_{16}^2}{2} + \frac{n_{16}}{3} \right) \right] \ddot{q}_M \delta t}{\left[ n_{16} (n_{18} + n_{20}) + \frac{(n_{16}^2 - n_{16}) + (n_{20}^2 - n_{20})}{2} \right]} - \frac{\left[ (n_{18} n_{20}) + \frac{(n_{18}^2 - n_{18}) + (n_{20}^2 - n_{20})}{2} \right] \ddot{q}_M}{\left[ n_{16} (n_{18} + n_{20}) + \frac{(n_{16}^2 - n_{16}) + (n_{20}^2 - n_{20})}{2} \right]} - \frac{\left( \frac{n_{20}^3}{6} - \frac{n_{20}^2}{2} + \frac{n_{20}}{3} \right) \ddot{q}_m \delta t}{\left[ n_{16} (n_{18} + n_{20}) + \frac{(n_{16}^2 - n_{16}) + (n_{20}^2 - n_{20})}{2} \right]}, \quad (2.54)$$

with  $(n_{16}, n_{18}, n_{20})$ : the integers maximizing the right-hand side of (2.54).  $f_\chi$  and  $f_\phi$  are respectively equivalent to the right-hand sides of (2.53) and (2.54).

---

**Algorithm 4** Compute  $n_{15}, n_{16}, n_{17}, n_{18}, n_{19}, n_{20}, f_\chi$  and  $f_\phi$ 


---

**Require:**  $q_M, q_m, q_{|k}, \dot{q}_{|k}, \ddot{q}_M, \ddot{q}_m, \ddot{q}_M, \ddot{q}_m, \delta t$ 

```

1:  $f_{\chi_{max}} \leftarrow \ddot{q}_m$     $f_{\phi_{min}} \leftarrow \ddot{q}_M$ 
2: for ( $i_1 = 1 \rightarrow N_1$ ) do
3:    $n_{16}^* \leftarrow i_1$     $n_{15}^* \leftarrow i_1$ 
4:   for ( $i_2 = 0 \rightarrow N_2$ ) do
5:      $n_{18}^* \leftarrow i_2$     $n_{17}^* \leftarrow i_2$ 
6:     if ( $\ddot{q}_{|k} \geq 0$ ) then
7:        $n_{20_{\mathbb{R}^+}}^* \leftarrow n_{16} - \left| \frac{\ddot{q}_{|k}}{\ddot{q}_m \delta t} \right|$     $n_{19_{\mathbb{R}^+}}^* \leftarrow n_{15} + \left| \frac{\ddot{q}_{|k}}{\ddot{q}_M \delta t} \right|$ 
8:     end if
9:     if ( $\ddot{q}_{|k} < 0$ ) then
10:       $n_{20_{\mathbb{R}^+}}^* \leftarrow n_{16} + \left| \frac{\ddot{q}_{|k}}{\ddot{q}_M \delta t} \right|$     $n_{19_{\mathbb{R}^+}}^* \leftarrow n_{15} - \left| \frac{\ddot{q}_{|k}}{\ddot{q}_m \delta t} \right|$ 
11:      if ( $n_{16}^* \leq 1$ ) then  $n_{20_{\mathbb{R}^+}}^* \leftarrow \left| \frac{\ddot{q}_{|k}}{\ddot{q}_m \delta t} \right|$ 
12:      if ( $n_{15}^* \leq 1$ ) then  $n_{19_{\mathbb{R}^+}}^* \leftarrow \left| \frac{\ddot{q}_{|k}}{\ddot{q}_M \delta t} \right|$ 
13:      end if
14:      if ( $n_{15}^* \leq 1$ ) then  $n_{15}^* \leftarrow 1$ 
15:      if ( $n_{16}^* \leq 1$ ) then  $n_{16}^* \leftarrow 1$ 
16:      if ( $n_{19_{\mathbb{R}^+}}^* \leq 2$ ) then  $n_{19_{\mathbb{R}^+}}^* \leftarrow 2$ 
17:      if ( $n_{20_{\mathbb{R}^+}}^* \leq 2$ ) then  $n_{20_{\mathbb{R}^+}}^* \leftarrow 2$ 
18:       $f_\chi^* \leftarrow f_\chi(q_m, q_{|k}, \dot{q}_{|k}, \ddot{q}_M, \ddot{q}_m, \ddot{q}_M, \ddot{q}_m, n_{15}^*, n_{17}^*, n_{19}^*)$ 
19:       $f_\phi^* \leftarrow f_\phi(q_M, q_{|k}, \dot{q}_{|k}, \ddot{q}_m, \ddot{q}_M, \ddot{q}_m, \ddot{q}_M, n_{16}^*, n_{18}^*, n_{20}^*)$ 
20:      if ( $f_\chi^* \geq f_{\chi_{max}}$ ) then
21:         $f_{\chi_{max}} \leftarrow f_\chi^*$     $n_{16} \leftarrow n_{16}^*$     $n_{18} \leftarrow n_{18}^*$     $n_{20}^a \leftarrow n_{20_{\mathbb{R}^+}}^*$ 
22:      end if
23:      if ( $f_\phi^* \leq f_{\phi_{min}}$ ) then
24:         $f_{\phi_{min}} \leftarrow f_\phi^*$     $n_{15} \leftarrow n_{15}^*$     $n_{17} \leftarrow n_{17}^*$     $n_{19}^a \leftarrow n_{19_{\mathbb{R}^+}}^*$ 
25:      end if
26:    end for
27:  end for
28:  $f_\chi \leftarrow f_{\chi_{max}}$     $f_\phi \leftarrow f_{\phi_{min}}$ 
29: return  $n_{15}, n_{16}, n_{17}, n_{18}, n_{19}, n_{20}, f_\chi, f_\phi$ 

```

---

<sup>a</sup> For stability in the computation of  $f_\chi$  and  $f_\phi$ ,  $(n_{19}^*, n_{20}^*)$  are used as real numbers (even if previously defined as numbers of iterations). The same for  $n_{19}$  and  $n_{20}$ .

$n_{15}, n_{16}, n_{17}, n_{18}, n_{19}, n_{20}, f_\chi$  and  $f_\phi$  can be computed numerically as shown in Algorithm 4.  $N_1$  and  $N_2$  in Algorithm 4 are fixed heuristically, they must however be  $\geq$  to the number of iterations needed to perform the braking movement described in (2.49). More details on how to compute this parameter in Section 2.6.

### 2.4.2.5.1 Illustration 6

For this simulation, the controller is implemented with the last **complete** formulation of the constraint on articular position (2.53), that takes into account both the positive and negative jerk capabilities,  $[\ddot{q}_{m_0}, \ddot{q}_{M_0}] = [-25, 25] \text{ rad.s}^{-3}$  in addition to the deceleration capability ( $\ddot{q}_{m_0} = -3 \text{ rad.s}^{-2}$ ). As for the previous simulations, joint 0 moves towards its upper position limit  $q_{M_0}$ . The *hard-coded* constraints on its articular acceleration (2.11c) and articular jerk (2.11d) are not considered in the configuration of the controller.

Fig. 2.26, illustrates the braking phase induced by the actuator of joint 0 as it copes with its position limit  $q_{M_0}$ . the joint starts jerking negatively with maximum jerk ( $\ddot{q} = \ddot{q}_{m_0}$ ) until maximum deceleration  $\ddot{q}_{m_0}$  is reached (sub-phase ①). This deceleration is then used for several control time-steps (sub-phase ②) then “*de-charged*” and brought to 0 by jerking positively (sub-phase ③). By the end, joint 0 is stopped exactly at its upper position limit  $q_{M_0}$ . The jerk profile in Fig. 2.27 shows an exceeding of ( $0.013 \text{ rad.s}^{-3}$ ) over the lower jerk limit  $\ddot{q}_{m_0}$  and an other one of ( $0.035 \text{ rad.s}^{-2}$ ) over the lower deceleration limit  $\ddot{q}_{m_0}$ . As before, this is mainly due to the *discrete* description of the braking phase that results into a *discrete* formulation of the constraint on articular position.

Fig. 2.28 illustrates the induced braking phase in case the lower deceleration limit  $\ddot{q}_{m_0}$  is fixed at  $-100 \text{ rad.s}^{-2}$ . Consequently, the second braking *sub-phase*<sup>19</sup> disappears resulting into an equivalent behaviour as for (2.41) in Fig. 2.22.

## 2.5 Final bounds on the acceleration control variable

In this section, the final upper and lower bounds  $[f_\psi, f_\omega]$  on the acceleration control variable  $\ddot{q}_{|k}^c$  that allow a joint to cope with all the constraints corresponding to its position, velocity, acceleration and jerk limitations at the same time are computed. Using the following algorithm:

---

**Algorithm 5** Compute  $f_\psi$  and  $f_\omega$

---

**Require:**  $f_\alpha, f_\beta, f_\lambda, f_\mu, \ddot{q}_M, \ddot{q}_m, \ddot{q}_{M_0}, \ddot{q}_{m_0}, \delta t$

```

1:  $f_\psi \leftarrow \ddot{q}_M$ 
2:  $f_\omega \leftarrow \ddot{q}_m$ 
3:  $f_\psi^* \leftarrow \min(f_\alpha(\dot{q}_M, \ddot{q}_m), f_\lambda(q_M, \ddot{q}_m, \ddot{q}_M, \ddot{q}_m), \ddot{q}_M, (\ddot{q}_{|k} + \ddot{q}_M \delta t))$ 
4:  $f_\omega^* \leftarrow \max(f_\beta(\dot{q}_m, \ddot{q}_M), f_\mu(q_m, \ddot{q}_M, \ddot{q}_M, \ddot{q}_m), \ddot{q}_m, (\ddot{q}_{|k} + \ddot{q}_m \delta t))$ 
5: if  $(|f_\psi^* - \ddot{q}_{|k}^c| \leq \epsilon^a)$  then
6:   |  $f_\psi \leftarrow f_\psi^*$ 
7: end if
8: if  $(|f_\omega^* - \ddot{q}_{|k}^c| \leq \epsilon)$  then
9:   |  $f_\omega \leftarrow f_\omega^*$ 
10: end if
11: return  $f_\psi, f_\omega$ 

```

---

<sup>a</sup> e.g.,  $\epsilon = 1$

The final constraint on the acceleration control variable  $\ddot{q}_{|k}^c$  is written:

$$f_\omega \leq \ddot{q}_{|k}^c \leq f_\psi, \quad (2.55)$$

with the upper and lower bounds  $f_\omega$  and  $f_\psi$  computed numerically as shown in Algorithm 5.

<sup>19</sup>Considering an upper position limit, 3 sub-phases are distinguished during the **complete** braking phase: constant negative jerk, constant deceleration then constant positive jerk.

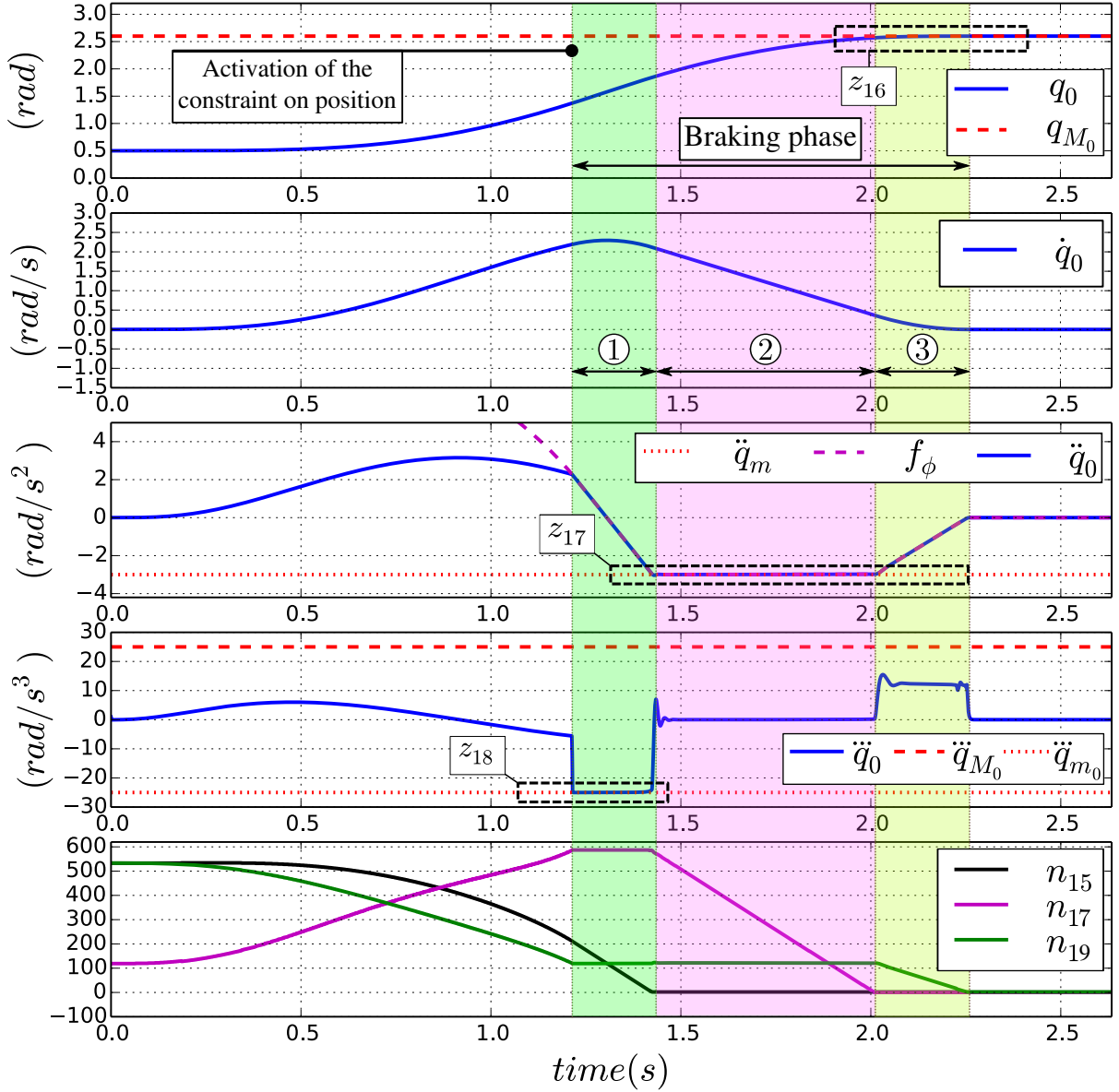


Figure 2.26: Extended state  $S$  of joint 0 during the *complete* braking phase to cope with an upper position limit  $q_{M_0}$ . The new formulation of the constraint on articular position (2.53), that takes into account the deceleration capability  $\ddot{q}_{m_0} = -3 \text{ rad.s}^{-2}$  in addition to the positive and negative jerk limits  $[\ddot{q}_{m_0}, \ddot{q}_{M_0}] = [-25, 25] \text{ rad.s}^{-3}$  is used. Top to bottom: position, velocity, acceleration, jerk and  $(n_{15}, n_{17}, n_{19})$ . See  $z_{16}$ ,  $z_{17}$  and  $z_{18}$  in Fig. 2.27

### 2.5.1 Illustration 7

For this final simulation, the controller is implemented with the constraint (2.55), that takes into account all the articular position, velocity, deceleration and jerk limitations of the actuators of the robot. Physical capabilities of joint 0 are fixed as:  $[q_{m_0}, q_{M_0}] = [-2.6, 2.6] \text{ rad}$ ,  $[\dot{q}_{m_0}, \dot{q}_{M_0}] = [-2, 2] \text{ rad.s}^{-1}$ ,  $[\ddot{q}_{m_0}, \ddot{q}_{M_0}] = [-3, 100] \text{ rad.s}^{-2}$  and  $[\ddot{q}_{m_0}, \ddot{q}_{M_0}] = [-25, 25] \text{ rad.s}^{-3}$ . As for the previous simulations, joint 0 moves towards its upper position limit  $q_{M_0}$ . Note: to ensure compatibility between the *hard-coded* constraint on articular acceleration (2.8c) and the new formulation of the constraint on articular position (2.53),  $\ddot{q}_{m_0}$  is diminished to  $2.95 \text{ rad.s}^{-2}$

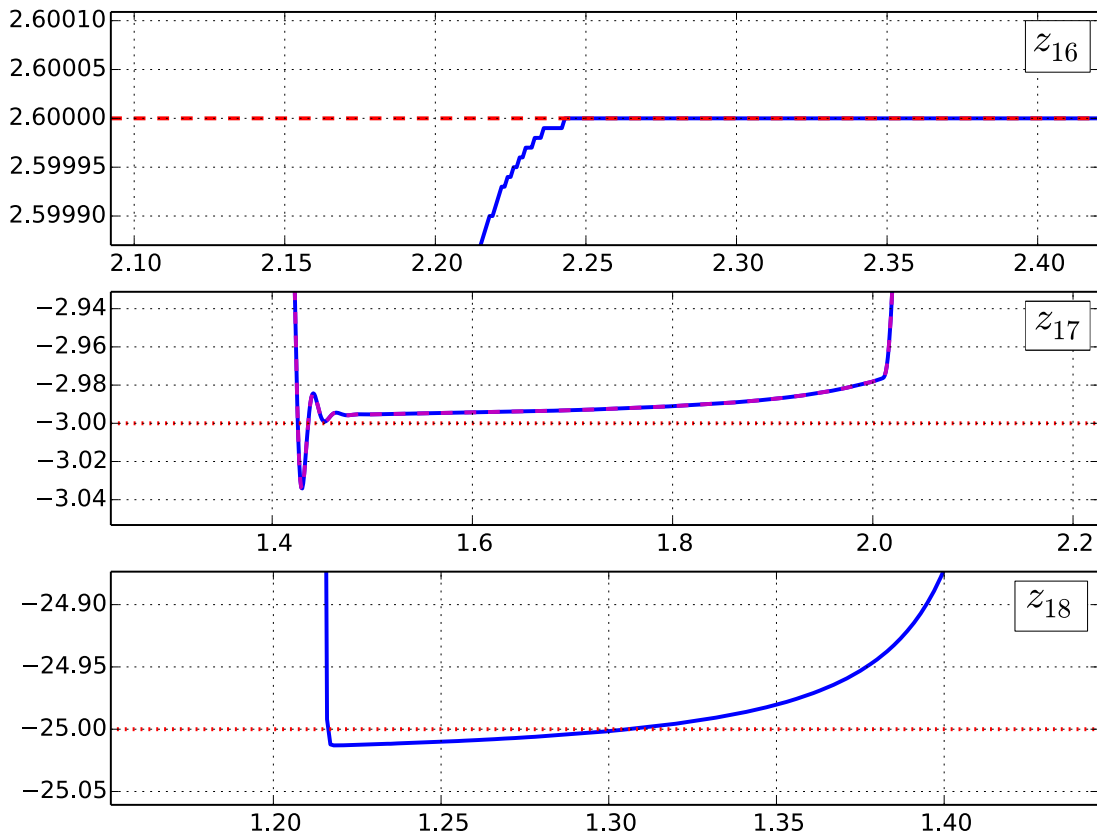


Figure 2.27: Zooms corresponding to Fig. 2.26.

when computing  $f_{\psi}$  using Algorithm 5. This allows to anticipate any exceeding over the lower acceleration limit ( $\ddot{q}_{m_0} = 3 \text{ rad.s}^{-2}$ ).

Fig. 2.29 and Fig. 2.30 show how joint 0 is capable of coping at the same time with all its limitations. Constraints on the articular position, velocity, acceleration and jerk are all satisfied during the movement of the joint and the *viability* for the state of the robot is guaranteed at every time-step over an infinite horizon of time. Two braking phases are distinguished:

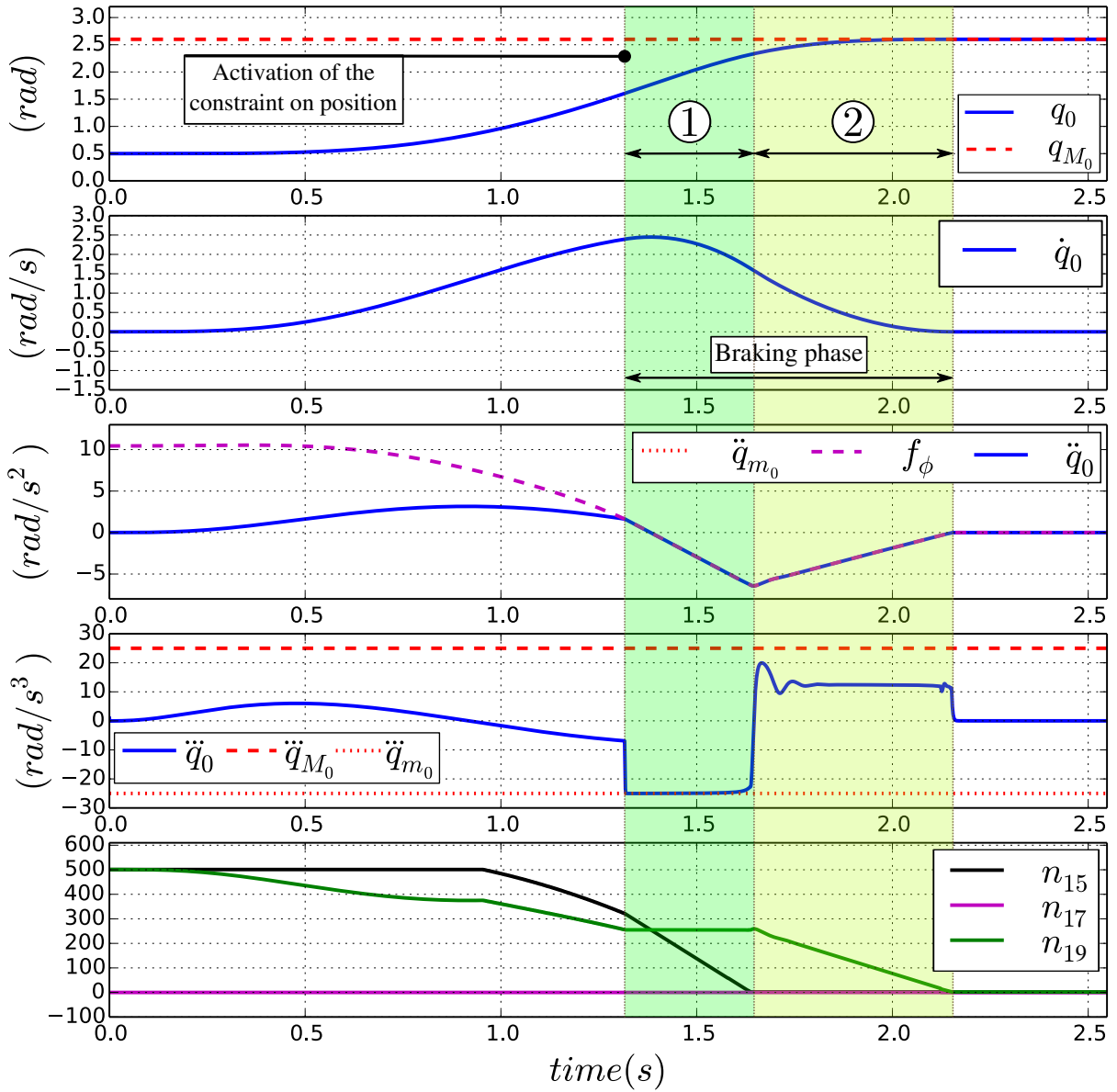


Figure 2.28: State  $S$  of joint 0 during the braking phase to cope with an upper position limit  $q_{M_0}$ ; The new constraint formulation (2.53) that takes into account the deceleration capability  $\ddot{q}_{m_0} = -100 \text{ rad}\cdot\text{s}^{-2}$  in addition to the positive and negative jerk limits  $[\ddot{q}_{m_0}, \ddot{q}_{M_0}] = [-25, 25] \text{ rad}\cdot\text{s}^{-3}$  is used. Top to bottom: position, velocity, acceleration, jerk and  $(n_{15}, n_{17}, n_{19})$ .

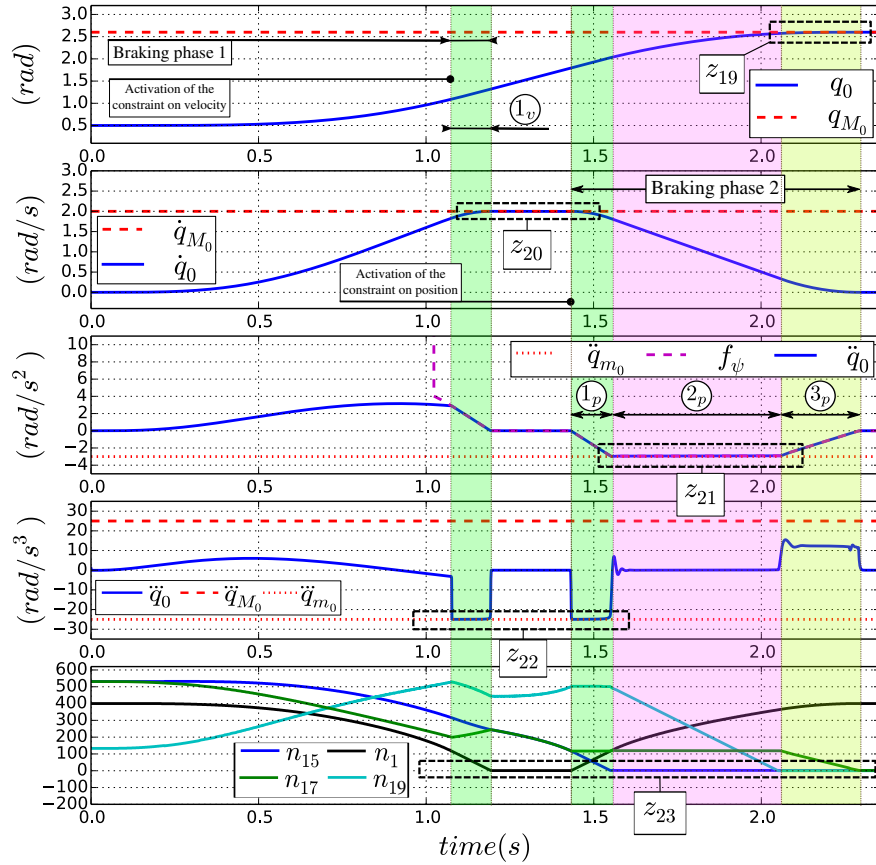


Figure 2.29: Extended state  $S$  of joint 0 during the braking phases implicitly induced to simultaneously cope with its position, velocity, acceleration and jerk limitations. The final constraint on the acceleration control variable (2.55), that takes into account all these limitations is used. Top to bottom: position, velocity, acceleration, jerk and  $(n_1, n_{15}, n_{17}, n_{19})$ . See  $z_{19}$ ,  $z_{20}$ ,  $z_{21}$ ,  $z_{22}$  and  $z_{23}$  in Fig. 2.30.

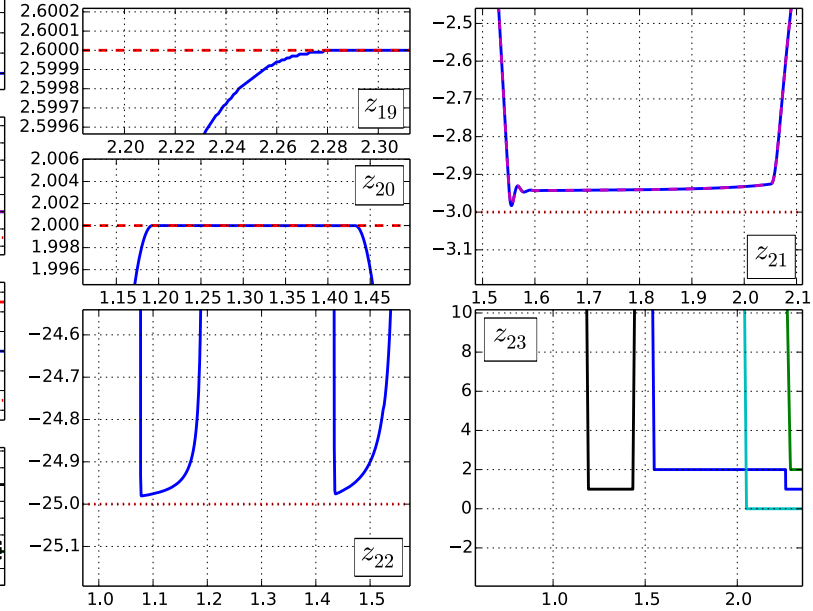


Figure 2.30: Zooms corresponding to Fig. 2.29.

## 2.6 Conclusion

In this chapter, the problem of ensuring *viability* for the state of a robotic manipulator that is reactively controlled at the dynamic-level and whose joints have limited positions, velocities, accelerations and jerks is discussed. As it is impossible to cope with all the constraints related to these limitations using their classic known formulations, new mathematical expressions for the constraints that include the reaction capabilities (i.e., producible articular deceleration/torque and jerk) of the actuators of the robot have been proposed.

For the constraint on articular velocity, the new formulation takes into account the actuators jerk capabilities. For the articular position constraint, both producible articular deceleration and jerk are included in the new formulation. Finally, mathematical expressions of the upper and lower bounds of the robot's dynamic control input, that consider all the physical limitations of its actuators and that ensure every time-step the existence of a solution to the control problem are proposed. Performed simulations on a 7-degree-of-freedom KUKA LWR4 robotic arm clearly show the interest of using the proposed new formulations in comparison to the classic naive ones, to guarantee the satisfaction of the different joint physical limitations without altering the *viability* of the system.

A main drawback with the new introduced formulations of the constraints on articular velocity and position is the computational load that may increase because of the heuristic method used to compute the *number of time-steps needed for the induced braking phases to cope with the considered position and velocity limits*. Fixing  $N, N_1$  and  $N_2$  in Algorithms 2, 3 and 4 to high<sup>20</sup> values (e.g., 10000) and running all these algorithms at the same time for all the joints of the robot can rapidly become computationally heavy. In a more practical way, maximum values for  $N, N_1$  and  $N_2$  can be computed as solutions to the following problems:

- what is the number of time-steps needed for a joint's actuator into an initial state of *maximum* acceleration to bring this acceleration to zero considering the amount of jerk it can produce.
- what is the number of time-steps needed for a joint's actuator moving at *maximum* velocity with *maximum* acceleration to completely stop considering the amount of deceleration and jerk it can generate.

The maximum solution  $N^*$  between these two problems is the biggest number of time-steps regarding the worst case braking movement that can be induced by the actuator.

An other important aspect that must be considered when using the introduced new formulations is their sensitivity regarding the accuracy of the dynamics model of the robot. Indeed, as all the constraints are reflected on the control variable  $\ddot{\mathbf{q}}_k^c$ , the dynamic equation (2.7) that links this parameter to the control torque input  $\boldsymbol{\tau}_k^c$  must be sufficiently accurate to ensure a correspondence between  $\ddot{\mathbf{q}}_k^c$  and the real acceleration performed by the joints of the robot [Del Prete 2016].

Although it has been possible to properly reformulate the constraints on articular velocity and position to successfully guarantee the *viability* of a reactively controlled robotic manipulator; these constraints that are expressed in the system's joint space and characterized by static limits (type 1 in table 2.1); in the upcoming chapters, to ensure the safety of a human-operator physically interacting with a robotic arm, new constraints expressed in operational space and characterized by dynamic limits are introduced and implemented in the controller of the robot. Issues that may occur when coping with such constraints of type 6 (see table 2.1) will be furthermore discussed.

<sup>20</sup>To guarantee finding the right values for the  $n_i$  parameters.





## Chapter 3

# Energy based control for safe human-robot physical interaction

In this chapter, we propose physically meaningful energy related safety indicators for robots sharing their workspace with humans. Based on these indicators, safety criteria that bound the dynamics of the robot during interactions with its environment are introduced as constraints in the control scheme. A first constraint that depends on the real-time distance between the robot and a nearby obstacle is used to limit the amount of kinetic energy dissipated in case of a collision. After the establishment of physical-contact, a second constraint is used to modulate the contact forces by limiting the amount of equivalent energy that accumulates in the controller of the robot. The controller is formulated as an optimization problem and computes every time-step the needed actuation torques for a KUKA LWR4 robotic manipulator given some task to be performed in Cartesian space, the introduced safety related constraints and the physical limitations of the system to cope with. The overall framework tested in simulation allows the robot to safely interact with its environment.

### 3.1 Introduction

Service and intervention robotics, as well as more traditional industrial robotics applications, are evolving in a direction where the workspace of the robot is very likely to be shared with humans. This may induce deliberate<sup>1</sup> and non-intentional physical interactions. Safety in this context becomes a critical issue to be dealt with [Alami 2006].

To ensure safe human-robot interactions, several approaches have been explored in the robotics literature. At the hardware level, the mechanical design can be optimized to reduce the apparent inertia of the robot [Zinn 2004] and compliant components can be introduced to allow smoother contacts and less severe impacts [Haddadin 2012b]. Torque sensing at the joint level also provides a way to actively control the impedance of the robot. The Kuka-DLR lightweight robot [Bischoff 2010], [Loughlin 2007], [Hirzinger 2001] has been specifically designed to meet these challenges.

At the control level, a safe human-robot interaction requires: i) switching between different control modes (before/after physical contact) without causing potentially harmful discontinuities in the movements of the robot; ii) the formulation of safety indicators to reflect the amount of danger towards a considered nearby human-operator. These safety indicators must be related to the impact and contact forces generated during physical interaction; iii) the formulation of

---

<sup>1</sup>For example if the robot can be both used in an autonomous mode or in a co-manipulation mode.

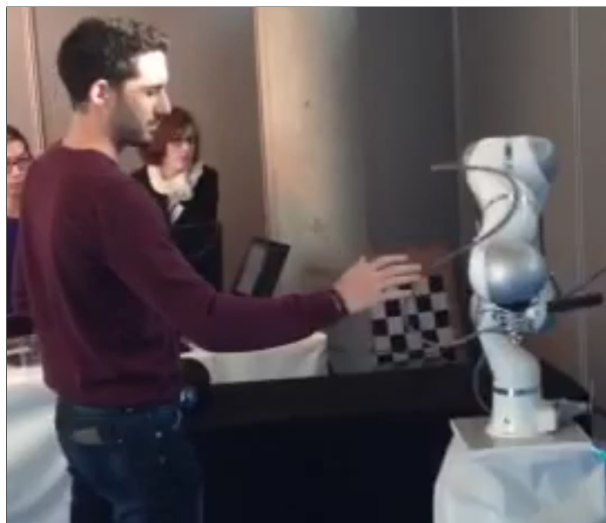


Figure 3.1: View of a user sharing the same workspace with a KUKA LWR4 robotic manipulator. The kinetic energy of the robot is modulated as function of the real-time distance between the human and the robot in order to best perform the task of the robot while ensuring safety.

safety criteria representing the bounds on the dynamic behaviour of the robot and finally, to be easily accounted for, the possibility to express these safety indicators and criteria as constraints related to the control input of the robot.

The most generic way to take into account the different *injury related parameters* (e.g., the impact and contact forces) for synthesising the needed safety indicators is to use an energy-based formulation of the safety problem. Indeed, energy is universal and can describe all the dangerous phenomena occurring during human-robot interaction. The impact force for example is directly caused by the amount of kinetic energy dissipated in case of a collision. This kinetic energy includes the velocity and equivalent mass data of both the robot and the human-operator. Contact forces on the other hand, derive from the amount of equivalent energy that accumulates in the controller of the robot during physical contact. And when contact is released, the risk of injury that can be caused by a fast transformation of the equivalent energy accumulated in the controller into kinetic one can also be monitored and reduced by modulating the energy of the robot.

Kinetic energy displayed by a robot during its movement has already been discussed in [Haddadin 2008] and [Haddadin 2012a] as a good representation of the risk of injury in case of a collision with a nearby human. Also, maximum values for this property considered to be safe in case of an impact are available in the latest ISO/TS 15066 [ISO 2016] for different regions of the human body. Unlike the various safety indicators from the literature, energy is significant during all the different interaction phases of a robot with its environment: before/after collisions, during physical contact, when the robot moves and also when it is motionless. Therefore, energy is used for the work presented in this chapter to synthesize *physically meaningful* safety indicators to assess and *control* the degree of danger a robotic manipulator represents when it physically interacts with its environment. These safety indicators include both the kinetic energy of the robot during its free movements and the *equivalent energy* accumulated in its controller during physical contact.

During the *free movements* of the KUKA LWR4 robotic arm, a safety criterion based on the maximum allowed kinetic energy for the robot is introduced and used to constrain its dynamical

behaviour in the direction of a considered obstacle<sup>2</sup> entering its workspace. This imposed constraint accounts for the dynamic capabilities (i.e., producible deceleration in operational space) of the robot and is modulated as function of the distance between the robot and the approaching obstacle. During physical contact, an other criterion based on the maximum value allowed for the *equivalent energy that accumulates in the controller of the robot*<sup>3</sup>, is used to modulate the contact forces between the robot and its environment.

In order to properly account for the introduced safety constraints, the control problem is expressed as a Linearly Constrained Quadratic Program (LQP) [Boyd 2004]. The computation of the adequate actuation torque needed to perform a trajectory tracking in operational space is subject to several linear inequality constraints accounting for the physical limitations of the robot (joint limits, joint velocity, torque and jerk saturations) as well as for the limit values of the energy-based safety indicators.

Finally, thanks to the constraint on the kinetic energy of the robot during its free movements, the proposed control framework is expected to decrease the impact forces at collisions. Contact forces on the other hand, are saturated through the constraint on the *equivalent energy* that accumulates in the controller of the robot during physical interaction. Using the same framework, impact and contact forces can be modulated and the dynamics of the robot controlled by a straightforward modification of the values of the safety criteria imposed to the system. Fig. 3.1 illustrates a typical workspace-sharing scenario for the proposed controller.

This chapter is organised as follows: In Section 3.2, the proposed safety indicators are formulated for both the pre-collision and physical-contact phases. In Section 3.3, the associated safety criteria, namely the maximum allowed values for the introduced safety indicators are introduced. The controller is described in Section 3.6: task related objectives are formulated and the energy related constraints needed for a safe human-robot interaction are expressed as function of the dynamic control input of the robot. In Section 3.7, a test case scenario is introduced using a KUKA LWR4 serial robot in a simulated world, based on which, the possibilities offered by the proposed controller are illustrated and discussed. Finally, Section 3.8 summarizes and discusses the contributions and provides an overview of the needed developments.

## 3.2 Interaction forces and safety indicators

In this section, safety indicators quantifying the degree of danger<sup>4</sup> represented by the robot towards a nearby human-operator are introduced. These indicators must be *physically meaningful* so they can satisfy the safety limits recommended by the ISO/TS 15066 standard [ISO 2016], related to the control input of the robot and computable in real-time.

### 3.2.1 Kinetic energy

The generated impact force in case of a collision between the robot and its environment can be written as function of the dissipated kinetic energies and the shock absorption distance:

$$\begin{aligned} \int_u F_{\text{impact}} du &= E_{\text{dissipated}} \\ &= E_c^{\text{hum}} + E_c^{\text{rob}}, \end{aligned} \tag{3.1}$$

<sup>2</sup>All along the manuscript, “obstacle” is used as a generic term for any external element of the environment, e.g., a “human-operator”. these two can be used interchangeably.

<sup>3</sup>All along the manuscript, this type of energy is by *abuse of language* referred to as “potential energy”.

<sup>4</sup>Risk induced by a collision or physical contact.

$F_{impact}$  is the generated impact force at collision,  $u$  the shock absorption distance and  $E_{dissipated}$  the dissipated energy which is equal to the sum of kinetic energies  $E_c$  of both the human-operator and the robot. On the one hand, the left side parameters of the shock absorption equation (3.1) are not directly related to the actuation torque of the robot. Moreover, it is impossible to have an accurate model of the human body-robot impedance<sup>5</sup>. As a matter of fact, the use of the impact force or of the shock absorption distance as a safety indicator is neither desirable nor possible. On the other hand, the dissipated energy is closely related to the impact force and can be directly related to the actuation torque of the robot and thus, controlled in order to reduce the impact of a collision.

At a given time, very few assumptions can be made on the energy-state of the nearby human-operator and on its future evolution. As a consequence, the retained safety indicator  $S_c$  is robot-centred.  $E_c^{rob}$  is directly related to the impact force and can be expressed using the control torque input. It is therefore considered for the formulation of the first safety indicator  $S_c$ :

$$S_c^{i,j} = E_c^{i,j} = \frac{1}{2} m(\mathbf{q}_{|k})_{i,j}^{eq} v_C^{i,j2}, \quad (3.2)$$

with:  $1/m(\mathbf{q}_{|k})_{i,j}^{eq} = J(\mathbf{q}_{|k})_C^{i,j} M(\mathbf{q}_{|k})^{-1} J(\mathbf{q}_{|k})_C^{i,jT}$ ;  $m(\mathbf{q}_{|k})_{i,j}^{eq}$  being the equivalent mass of the robot segment  $i$  in the direction of obstacle  $j$  expressed in the Cartesian space [Khatib 1995].  $M(\mathbf{q}_{|k})$  is the joint space inertia matrix of the robot and  $\mathbf{q}_{|k}$  its joint space configuration at time-step  $k$ ;  $v_C^{i,j} = J(\mathbf{q}_{|k})_C^{i,j} \dot{\mathbf{q}}_{|k}$  is the relative linear velocity of the closest point  $C$  belonging to the robot segment  $i$  in the direction of obstacle  $j$ ;  $J(\mathbf{q}_{|k})_C^{i,j}$  is the Jacobian of the robot segment  $i$  expressed at point  $C$  and projected along the distance vector towards obstacle  $j$ .

To ensure safety for both the robot and any nearby obstacle, the introduced safety indicator must be considered for each (robot segment  $i$ , obstacle  $j$ ) pair, i.e., for  $N_O$  obstacles and a robot composed of  $N_b$  mobile bodies,  $N_O \times N_b$  safety indicators are needed. An other solution is to consider a safety indicator related to the real-time closest points  $C$  and  $O$  respectively on the robot and the human-operator. The main drawback with such approach is however the inevitable discontinuities that can be caused by the geometrical shapes of both the robot and the human, and that will be directly reflected on the expression of the safety indicator.

Within the framework of this chapter, and without loss of generality, a single obstacle  $O$  is considered and the only mobile part of the robot considered for the formulation of the *pre-collision* safety indicator is the end-effector (EE). Indeed, generally speaking, it is usually the last segment of the fixed-base serial robot KUKA LWR4 that holds the practical load and that consequently deploys the maximum amount of kinetic energy in contrast with the other segments of the robot. The kinetic energy based safety indicator is then written:

$$S_c = E_{c_{|k}}^{EE,O} = \frac{1}{2} m(\mathbf{q}_{|k})^{eq} v^2, \quad (3.3)$$

with:  $m(\mathbf{q}_{|k})^{eq} = m(\mathbf{q}_{|k})_{EE,O}^{eq}$  and  $v = v_{EE}^{EE,O} = \dot{\mathbf{X}}_{EE|k} \mathbf{n}_{O|k}$ . This indicator represents the energy that will be dissipated by the end-effector of the robot in case of a collision with a nearby obstacle (see Fig. 3.2).

<sup>5</sup>This model would have to be individual and body-part specific.

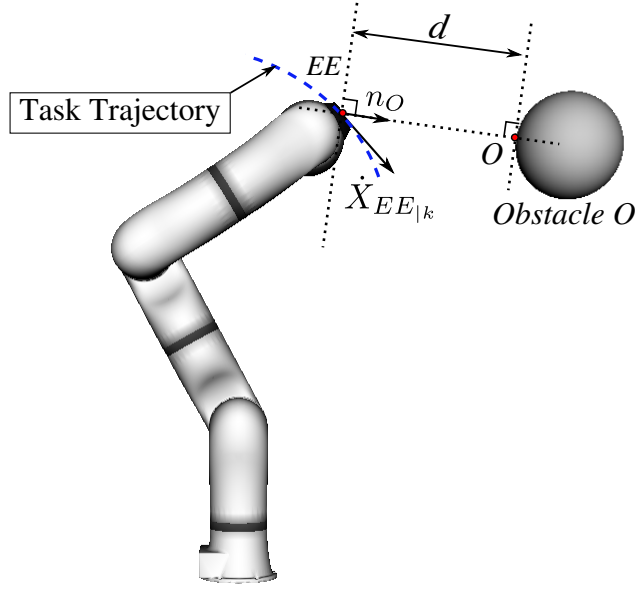


Figure 3.2: The closest distance  $d$  between the end-effector of the robot  $EE$  and an obstacle  $O$  entering its workspace.

### 3.2.2 Potential energy

When physical contact between the robotic manipulator and an obstacle in its environment is established at a contact point  $C$ , the resulting contact force is created as a consequence of the amount of energy that accumulates in the controller of the robot. The instantaneous force  $F_{C|k}$  that pulls the contact point  $C$  in the direction of its desired position<sup>6</sup> in operational space is derived from the controller's energy  $E_{p|k}$ :

$$F_{C|k} = -\nabla E_{p|k}. \quad (3.4)$$

Therefore:

$$E_{p|k} = - \int_{\mathbf{X}_{C|k}^*}^{\mathbf{X}_{C|k}} F_{C|k} d\mathbf{x} = \int_{\mathbf{X}_{C|k}}^{\mathbf{X}_{C|k}^*} F_{C|k} d\mathbf{x} = F_{C|k} \left( \mathbf{X}_{C|k}^* - \mathbf{X}_{C|k} \right) \mathbf{n}_{C|k}, \quad (3.5)$$

with:

$$F_{C|k} = m(\mathbf{q}_{|k})_{C,C^*}^{eq} \ddot{\mathbf{X}}_{C|k}, \quad (3.6)$$

$\mathbf{n}_{C|k}$  represents the instantaneous directing vector between the contact point  $C$  (on the considered segment  $i$ ) and its desired position  $C^*$  (see Fig. 3.3).

$\ddot{\mathbf{X}}_{C|k} = \left( \dot{\mathbf{J}}(\mathbf{q}_{|k})_C \dot{\mathbf{q}}_{|k} + \mathbf{J}(\mathbf{q}_{|k})_C \ddot{\mathbf{q}}_{|k}^c \right) \mathbf{n}_{C|k}$  is the Cartesian acceleration of the contact point  $C$  along the  $C, C^*$  instantaneous distance vector  $\mathbf{n}_{C|k}$  and  $\ddot{\mathbf{q}}_{|k}^c$  is the dynamic acceleration control variable that corresponds to the control torque input  $\boldsymbol{\tau}_{|k}^c$  (considering a dynamic controller).

Depending on the type of controller used to track the desired trajectory for the robot in operational space, the contact force applied by the robot to its environment when its movement is restrained during physical contact can be either *conservative* or *non-conservative*. For example,

<sup>6</sup>Considering a trajectory tracking task.

in case of a purely proportional controller ( $K_p \Delta X$ ), the amount of energy that accumulates in the controller of the robot depends on the instantaneous real and desired positions ( $\mathbf{X}_{C|k}$  and  $\mathbf{X}_{C|k}^*$ ) of the contact point  $C$ . The nature of the controller's energy in this case is similar to a *potential energy* stored into a compressed physical mechanical spring. However, when different control schemes are used, for example, a proportional derivative controller ( $K_p \Delta X + K_d \Delta \dot{X}$ ) or a complete PID with a feed-forward term, *non-conservative* forces that participate in the traction<sup>7</sup> of the contact point  $C$  during physical interaction are added. The energy that accumulates in the controller of the robot in such case is partially *path dependent* and does not *exclusively* depend on the instantaneous real and desired positions of the robot's contact point  $C$ .

On the other hand, the equivalent force *virtually pulling the robot* that results from any type of controller that is adapted for trajectory tracking, will<sup>8</sup> always make the real position of the robot  $\mathbf{X}_{C|k}$  ultimately converge towards its desired position  $\mathbf{X}_{C|k}^*$ . The work carried out even by a non or only partially conservative force will always be between the two positions:  $\mathbf{X}_{C|k}$  and  $\mathbf{X}_{C|k}^*$ . Which are therefore used for the formulation of the amount of energy that accumulates in the controller of the robot during physical contact (3.5). For convenience, and even if the equivalent energy of the controller during physical contact is not always linked to a fully conservative force, in the upcoming sections, this energy can be *by abuse of language* referred to as *potential energy*.

$E_{p|k}$  is directly related to the contact force generated during physical contact and can be expressed using the dynamic actuation variables (articular acceleration/torque  $[\ddot{\mathbf{q}}_{|k}^c, \boldsymbol{\tau}_{|k}^c]$ ). It is therefore used for the formulation of the safety indicator that reflects the degree of danger of the robot during physical interaction phases. Finally, the retained safety indicator  $S_{p_{contact}}$  is robot centred and can be written:

$$S_{p_{contact}} = E_{p|k} = F_{C|k} \left( \mathbf{X}_{C|k}^* - \mathbf{X}_{C|k} \right) \mathbf{n}_{C|k}. \quad (3.7)$$

In case of a physical contact that occurs at the level of the end-effector of the robot (EE) (see Fig. 3.3),  $S_{p_{contact}}$  can be written:

$$\begin{aligned} S_{p_{contact}} &= E_{p|k}^{EE,EE^*} = F_{EE|k} \left( \mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k} \right) \mathbf{n}_{EE|k}, \\ &= m(\mathbf{q})_{EE,EE^*}^{eq} \ddot{X}_{EE|k} \left( \mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k} \right) \mathbf{n}_{EE|k}. \end{aligned} \quad (3.8)$$

<sup>7</sup>Note that this traction does not result in any movement of the robot's contacted area, static contact is considered.

<sup>8</sup>Tracking errors aside.

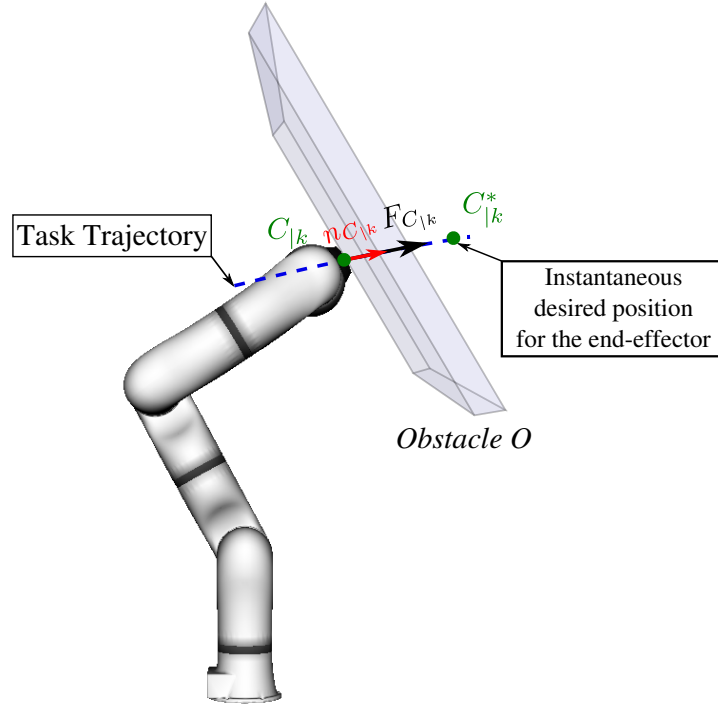


Figure 3.3: Instantaneous force  $F_{EE|_k}$  applied by a robotic manipulator to an obstacle in its environment. The contact point  $C$  in this case is the point  $EE$  of the end-effector of the robot.

Finally,  $S_{p_{contact}}$  represents the amount of potential energy that accumulates in the controller of the robot during a physical contact with its environment. This same potential energy, in case physical contact is released is to be transformed into kinetic one as the robot catches its desired position.

### 3.2.3 Relation between kinetic and potential energies

During *free movements*<sup>9</sup> of the robot, the equivalent force:

$$F_{C|_k} = m(\mathbf{q}|_k)_{C|_k, C|_{k+1}}^{eq} \ddot{X}_{C|_k}, \quad (3.9)$$

derived from the potential energy  $E_{p|_k}$ , pulls the potential collision point  $C$  along its desired trajectory from its current position  $\mathbf{X}_{C|_k}$  to its future position  $\mathbf{X}_{C|_{k+1}}$ .  $E_{p|_k}$  results from feeding the desired Cartesian position, velocity and feed-forward acceleration<sup>10</sup> to the controller of the robot. During these movements, Newton's third law of motion is satisfied. It can be written:

$$2\ddot{X}_{C|_k} \left( \mathbf{X}_{C|_{k+1}} - \mathbf{X}_{C|_k} \right) \mathbf{n}_{C|_k} = v_{C|_{k+1}}^2 - v_{C|_k}^2, \quad (3.10)$$

$v_{C|_{k+1}}$  and  $v_{C|_k}$  are respectively the current and future Cartesian velocities of the robot expressed at the potential collision point  $C$ . Multiplying both sides of (3.10) by  $\frac{1}{2}m(\mathbf{q})_{C|_k, C|_{k+1}}^{eq}$  results in:

$$m(\mathbf{q}|_k)_{C|_k, C|_{k+1}}^{eq} \ddot{X}_{C|_k} \left( \mathbf{X}_{C|_{k+1}} - \mathbf{X}_{C|_k} \right) \mathbf{n}_{C|_k} = \frac{1}{2}m(\mathbf{q}|_k)_{C|_k, C|_{k+1}}^{eq} \left( v_{C|_{k+1}}^2 - v_{C|_k}^2 \right), \quad (3.11)$$

<sup>9</sup>Movements without any physical interaction with the environment.

<sup>10</sup>We consider a trajectory tracking task in Cartesian space. The desired position, velocity and feed-forward acceleration are generated using a trajectory generator.



which is equivalent to:

$$E_{p|k}^{C|k, C|k+1} = E_{c|k+1}^{C|k, C|k+1} - E_{c|k}^{C|k, C|k+1} \Leftrightarrow E_{c|k+1}^{C|k, C|k+1} = E_{c|k}^{C|k, C|k+1} + E_{p|k}^{C|k, C|k+1}. \quad (3.12)$$

Meaning that, the kinetic energy  $E_{c|k+1}^{C|k, C|k+1}$  of the robot expressed at the potential collision point  $C$  at time-step  $k+1$  is equal to its kinetic energy at time-step  $k$  plus the *injected* potential energy  $E_{p|k}^{C|k, C|k+1}$ ; that represents the equivalent energy instantaneously injected in the controller of the robot along the  $\mathbf{n}_{C|k}$  vector (see Fig. 3.4). This energy makes the potential collision point  $C$  move from its initial position  $X_{C|k}$  to its next position  $X_{C|k+1}$ .  $\ddot{X}_{C|k} = \left( \dot{J}(\mathbf{q}|_k)_C \dot{\mathbf{q}}|_k + J(\mathbf{q}|_k)_C \ddot{\mathbf{q}}|_k^c \right) \mathbf{n}_{C|k}$  is the acceleration of the potential collision point in operational space.

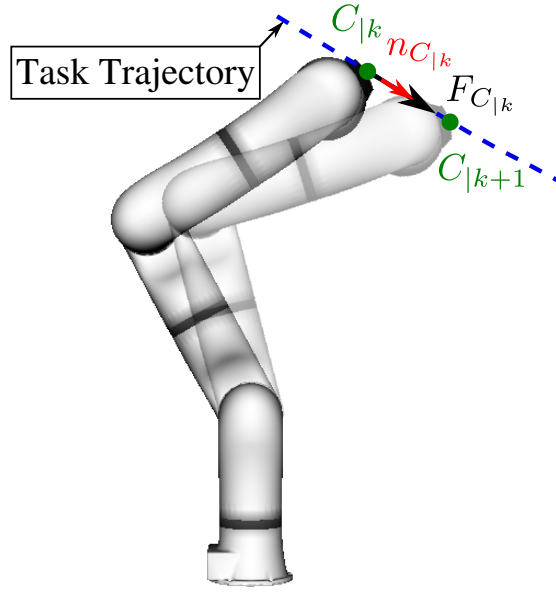


Figure 3.4: Equivalent force  $\mathbf{F}_{C|k}$  expressed at the potential collision point  $C$  pulling a robotic arm towards its desired trajectory from its current position  $\mathbf{X}|_k$  to its future position  $\mathbf{X}|_{k+1}$ . The potential collision point  $C$  in this case is the end-effector point  $EE$ .

According to (3.12), the *injected* potential energy  $E_{p|k}^{C|k, C|k+1}$ , when released, modifies the kinetic energy of the robot in the right direction to accomplish the trajectory tracking task. Therefore, the kinetic energy of the system  $E_{c|n}$  at a given time-step  $k = n$  can be expressed as the sum of all the previously injected potential energies  $E_{p|n}$ :

$$E_{c|k} = \sum_{n=1}^{k-1} E_{p|n}. \quad (3.13)$$

As the desired trajectory for the end-effector is discretized, every time step, only a small amount of potential energy  $E_{p|k}$  is *injected* then promptly transformed into kinetic one as the robot moves. The modulation of this *injected* potential energy can then directly influence the resulting kinetic energy of the robot and therefore the impact force  $\mathbf{F}_{impact}$  in case of a collision. Thus, the instantaneously injected potential energy in the controller of the robot before collision can also be used as a safety indicator during its *free* movements. When expressed at the level of the end-effector ( $EE$ ), the injected potential energy in the direction of a considered obstacle  $O$  is written:

$$S_{p_{free}} = E_{p|k}^{EE, O} = m(\mathbf{q}|_k)_{EE, O}^{eq} \ddot{X}_{EE|k}^{EE, O} \left( \mathbf{X}_{EE|k+1} - \mathbf{X}_{EE|k} \right) \mathbf{n}_{O|k}. \quad (3.14)$$

$\mathbf{n}_{O|k}$  is the unitary vector that corresponds to the distance between the end-effector of the robot (EE) and the closest point from a considered nearby obstacle  $O$  (see Fig. 3.2). Notice that the expression of this *potential energy based safety indicator* during the *free* movements of the robot is different from the expression of the safety indicator  $S_{p_{contact}}$  previously proposed for the physical contact phases (3.8). Indeed, the two points in Cartesian space between which the two *potential energy base safety indicators*  $S_{p_{contact}}$  and  $S_{p_{free}}$  are considered, are different. For  $S_{p_{contact}}$ , the potential energy that accumulates in the controller of the robot during physical contact is considered between the instantaneous real position of its end-effector  $\mathbf{X}_{EE|k}$  and the desired position  $\mathbf{X}_{EE|k}^*$ . Indeed, the contact force that derives from  $S_{p_{contact}}$  during such phase depends directly on the distance  $(\mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k})$ . On the other hand, for  $S_{p_{free}}$ , during *free* movements of the robot, the potential energy instantaneously *injected* in the controller of the robot modifies the kinetic energy expressed at its end-effector between its current position  $\mathbf{X}_{EE|k}$  and its position  $\mathbf{X}_{EE|k+1}$  at the next time-step. Because of tracking errors that increase particularly when coping with constraints, the end-effector of the robot *practically* never reaches exactly its imposed desired position  $\mathbf{X}_{EE|k}^*$  at time-step  $k + 1$ .  $\mathbf{X}_{EE|k+1}$  is therefore *practically* never equal to  $\mathbf{X}_{EE|k}^*$ . Instead of  $(\mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k})$ , the distance between  $\mathbf{X}_{EE|k}$  and  $\mathbf{X}_{EE|k+1}$  is therefore used for the expression of the second formulation (3.14) of the pre-collision safety indicator  $S_c$  during the *free* movements of the robot.

### 3.3 Safety limit values

In this section, safety criteria that bound the energy-based safety indicators previously presented are formulated. The introduced safety indicators together with the safety criteria will then be used to formulate the proper constraints on the energy, the robot is allowed to display during interaction with its environment.

#### 3.3.1 Safety limit value for the pre-collision safety indicator

For  $S_c$ , the safety criterion represents the maximum amount  $E_{c_{limit}}$  of kinetic energy allowed to be dissipated in case of a collision between the robot and a human within its workspace. The first safety constraint can be written:  $S_c \leq E_{c_{limit}}$  and must be satisfied at every time-step during the movements of the robot. Given the nature of  $S_c$ , such constraint when imposed at the control level, may have two consequences: a limitation of the end-effector's velocity in direction of the considered obstacle and a modification of its apparent mass along the same direction. However, when no human is at a close distance from the robot, it becomes unnecessary to saturate its kinetic energy of the robot as it accomplishes its task.

Therefore, to prevent over-limiting the dynamics of the system without needed cause, the reaction of the robot towards a nearby obstacle can be as follows: when the human is far from the robot, the system can be as dynamic as possible to accomplish its main task (maximum kinetic energy  $E_{c_{max}}$  allowed). As the operator starts approaching the robot, a limit  $E_{c_{limit}}$  that depends on the distance between the end-effector of the robot and the person is used to saturate its kinetic energy. The robot is forced into a safe dynamical state. At this time, if any physical contact between the robot and the human-operator within its workspace is engaged, the resulting impact will be harmless.

Based on this strategy,  $E_{c_{limit}}$  should therefore depend on the amount of kinetic energy that is considered to be safe just before the occurrence of a contact/collision but also be function of the real-time distance  $d$  between the end-effector of the robot and the approaching human-

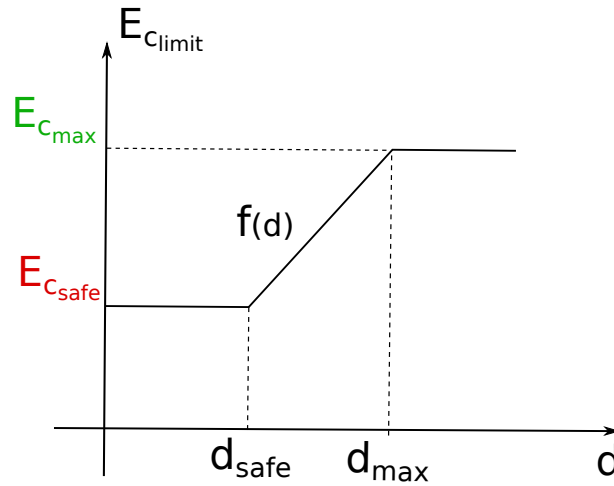


Figure 3.5: Evolution of the constraint on the kinetic energy of the robot in function of the distance  $d$  between its end-effector and a considered obstacle.

operator<sup>11</sup> (see Fig. 3.5). The constraint on the kinetic energy deployed by the robot during such interaction can be written:

$$S_c = E_c^{EE,O} = \frac{1}{2}m(\mathbf{q}_{|k})^{eq}v^2 \leq E_{climit} = E_{csafe} + f(d), \quad (3.15)$$

with  $E_{csafe}$ , the maximum amount of kinetic energy expressed at the level of the end-effector of the robot considered to be safe in case of any collision with a human. Such value depends on the body's impact zone, the shape of the tool/load carried by the end-effector, its apparent mass and its maximum allowed velocity [Haddadin 2012a]. It also depends on the nature of the interaction authorized between the robot and its environment. For example, if any contact between the robot and the nearby human is forbidden, fixing  $E_{csafe} = 0 \rightarrow v = 0$  forces the end-effector of the robot to stop at a distance  $d = d_{safe}$  from the considered obstacle. When physical contact is permitted:  $E_{csafe} > 0$  is the maximum amount of kinetic energy allowed for the robot just before collision. Standardized values for  $E_{csafe}$  are available in the latest ISO/TS 15066 [ISO 2016] for different regions of the human body.

$f(d)$  is an energy function that depends on the real-time distance  $d$  between the end-effector of the robot and any considered approaching obstacle (see Fig. 3.5). Based on the previous statements, three working zones, illustrated in Figure 3.6, are distinguished for the dynamic behaviour of the robot when sharing its workspace with humans:

1. a safe zone for  $d < d_{safe}$  in which the kinetic energy must be at most  $E_{csafe}$ ;
2. a working zone for  $d_{safe} < d < d_{max}$  where the kinetic energy of the system is constrained as the human moves towards the robot;
3. a third zone for  $d > d_{max}$  in which maximum dynamic performances are allowed for the robot.

When the relative distance between the approaching operator and the robot is decreasing, the system must be able to generate sufficient deceleration and jerk capabilities at the level of its end-effector<sup>12</sup> so it can properly react and cope with the imposed constraint on its kinetic energy.

<sup>11</sup>We recall that the collision here is considered only at the level of the end-effector of the robot.

<sup>12</sup>Or at the level of any potential collision point.

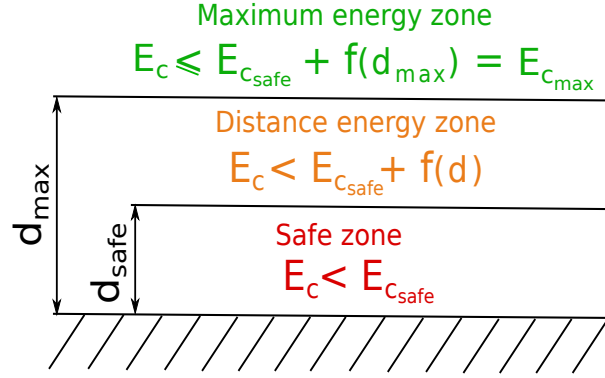


Figure 3.6: Energy zones for the dynamic behaviour of the robot.

Two phases are distinguished when coping with such constraint in operational space: first, at the activation of the constraint, the amount of torque and jerk producible by the actuators of the robot and their corresponding dynamic capabilities in operational space must imperatively be considered to ensure that the system is capable of modulating its kinetic energy to make it cope with  $E_{c_{limit}}$ . The second phase is when the kinetic energy of the robot is equal to the maximum allowed value  $E_{c_{limit}}$  (the constraint is activated), the variation of the function  $f(d)$  must imperatively account for the reaction capabilities (i.e., producible deceleration and jerk in operational space) of the robot and this at every time-step.

When reactively coping with the constraint on kinetic energy, from the Work-Energy theorem, the amount of work exerted on the robot is equal to the variation of its kinetic energy. Moreover, this work can be expressed as a product between the *equivalent braking force*  $F_{eq}$  applied on the end-effector and the braking distance:

$$\begin{aligned}
 W &= \Delta E_c \\
 &= F_{eq}(d - d_{safe}) \\
 &= E_{c_{limit}}(d) - E_{c_{limit}}(d_{safe}) \\
 &= f(d) - f(d_{safe}).
 \end{aligned} \tag{3.16}$$

The term  $f$  represents the maximum energy that can be dissipated during the braking phase as the robot decreases its kinetic energy. By choosing this function to be linear inside the distance energy working zone (see Figure 3.5), it can be written:

$$f(d) = K(d - d_{safe}). \tag{3.17}$$

The smaller the braking distance  $(d - d_{safe})$ , the higher must be the slope coefficient of the function  $f(d)$ .  $K$  represents the equivalent braking force applied at the level of the end-effector of the robot in the opposite direction to the obstacle. It depends on the available braking torque  $\tau_{braking}$ , jerk capabilities and the instantaneous Jacobian  $J(\mathbf{q}_{|k})_{EE}$  of the end-effector along the direction of the considered obstacle:

$$\tau_{braking} = J(\mathbf{q}_{|k})_{EE}^{EE, O^T} K. \tag{3.18}$$

For every time-step, the instantaneous equivalent braking force in Cartesian space at the level of the end-effector can be computed:

$$\arg \max_K \|K\|^2, \tag{3.19}$$

s.t:

$$\begin{cases} \tau_{braking} = J(\mathbf{q}_{|k})_{EE}^{EE,O^T} K, & (3.20a) \\ \tau_m \leq \tau_{braking} \leq \tau_M, & (3.20b) \end{cases}$$

with  $\tau_{braking}$  and  $K$  the optimization variables.  $\tau_M, \tau_m$  are the maximum and minimum torques producible by the actuators of the robot.

For (3.17),  $K$  must be guaranteed over all the braking distance  $d$ . However,  $\tau_{braking}$  and  $J(\mathbf{q}_{|k})_{EE}^{EE,O}$  can only be considered constant locally; the computation of  $K$  depends then on the future configurations of the robot along the braking distance  $d$ . Given the non linear nature of robotic manipulators, predicting the evolution of  $K$  is a complex problem. In the worst case, its value is very close to  $0^{13}$  and to ensure safety,  $E_{c_{limit}}$  should always be equal to  $E_{c_{safe}}$ , strongly limiting the dynamic performances of the robot. On the other hand, using actuators capable of deploying high braking torques/jerks allows to significantly decrease the braking distance  $d$ . In such case, a local estimation of  $\tau_{braking}$  and  $J(\mathbf{q}_{|k})_{EE}^{EE,O}$  can be a good approximation. Given the global objectives of this work, an average value of  $K$  ( $> 0$ ) is considered all over the workspace of the robot.

### 3.3.1.1 Pre-collision safety criterion extension

The safety criterion previously introduced considers the squared relative velocity between the end-effector of the robot and a nearby obstacle. Therefore, there is no differentiation between the case where the robot is going towards the obstacle and when it is moving away from it. In a forbidden contact situation ( $E_{c_{safe}} = 0$ ),  $v = 0$  is imposed which forbids the robot from going towards the obstacle but also from moving away from it. To avoid constraining the motion of the robot in the opposite direction to the obstacle, the safety indicator  $S_c$  during the *free* movements of the robot can be signed:

$$S_c = \frac{1}{2} \text{sign}(v) m(\mathbf{q}_{|k})^{eq} v^2, \quad (3.21)$$

with  $\text{sign}(v) = 1$  when the end-effector is getting closer to the considered obstacle.

The safety criterion can finally be written:

$$S_c \leq E_{c_{safe}} + K(d - d_{safe}), \quad (3.22)$$

with  $S_c$  defined by (3.21).

As previously explained in Subsection 3.2.3, an other approach to constrain the kinetic energy of the robot is to limit the amount of potential energy instantaneously *injected* in its controller during *free* movements. Between two consecutive control time-steps, this potential energy changes the kinetic energy of the robot from  $E_{c_{|k}}^{EE,O}$  to  $E_{c_{|k+1}}^{EE,O}$ :

$$E_{c_{|k+1}}^{EE,O} - E_{c_{|k}}^{EE,O} = E_{p_{|k}}^{EE,O} \Rightarrow E_{c_{|k}}^{EE,O} + E_{p_{|k}}^{EE,O} = E_{c_{|k+1}}^{EE,O} \leq E_{c_{limit}} \quad (3.23)$$

Based on (3.12) and (3.23), the constraint on kinetic energy can be written as a constraint on the potential energy *injected* in the controller of the robot at each time-step:

$$S_c = E_{c_{|k+1}}^{EE,O} = \underbrace{E_{c_{|k}}^{EE,O}}_{\text{measured}} + S_{p_{free}} \leq E_{c_{limit}} \Leftrightarrow S_{p_{free}} \leq \underbrace{E_{c_{safe}} + K(d - d_{safe})}_{E_{c_{limit}}} - \underbrace{E_{c_{|k}}^{EE,O}}_{\text{measured}}, \quad (3.24)$$

<sup>13</sup>Especially near singularity configurations.

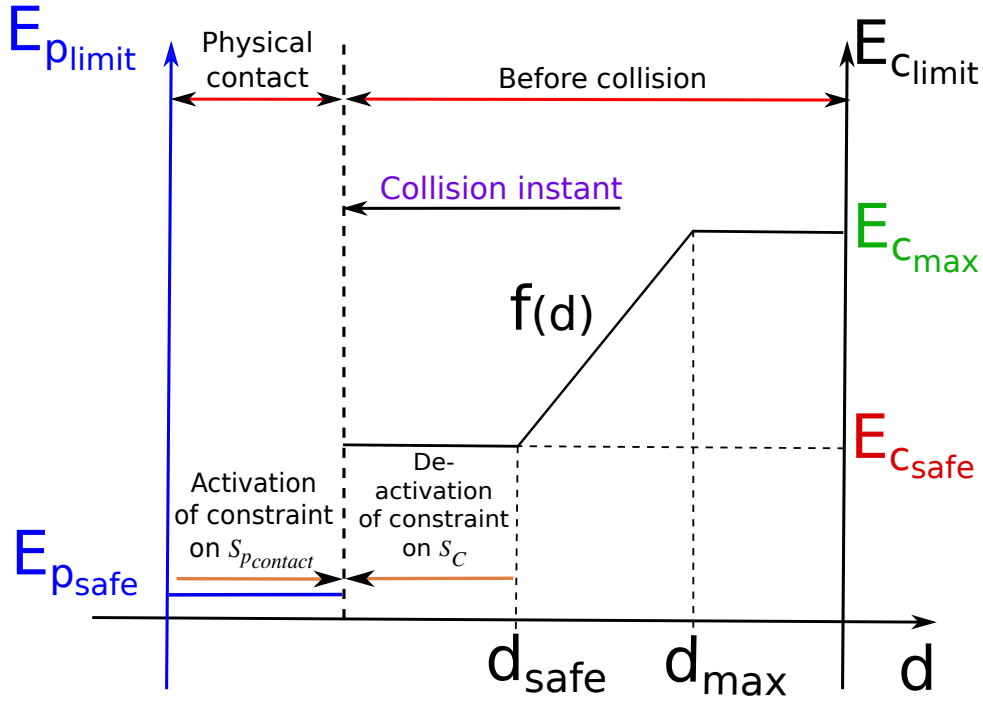


Figure 3.7: Evolution of the constraints on the kinetic and potential energies of the robot in function of the distance  $d$  between its end-effector and the considered obstacle.

with  $S_{p_{free}}$  as in (3.14) the amount of potential energy instantaneously injected in the controller of the robot considering its movement towards a considered obstacle  $O$ . Using (3.24), the kinetic energy of the robot is tracked then saturated when it reaches its maximum allowed limit by acting on the amount of potential energy  $S_{p_{free}}$  injected in the controller of the robot.

### 3.3.2 Safety limit value for the physical-contact safety indicator

For  $S_{p_{contact}}$ , the safety criterion represents the maximum amount  $E_{p_{limit}}$  of potential energy allowed to be accumulated in the controller of the robot during physical contact with its environment. The value of  $E_{p_{limit}}$  depends on several aspects: The desired degree of compliance of the robot during physical contact, the maximum allowed contact force and more importantly, the amount of kinetic energy this potential energy transforms into in case physical contact is released. Indeed, in such situation, the potential energy  $E_{p_{limit}}$  accumulated in the controller of the robot is to be rapidly transformed into kinetic one and if any collision with a nearby operator occurs, the resulting impact force  $F_{impact}$  may be harmful. Therefore, the maximum value acceptable for  $E_{p_{limit}} = E_{p_{safe}}$  should never be superior to  $E_{c_{safe}}$ . The constraint on the potential energy generated between the robot and its environment during physical contact can be written:

$$S_{p_{contact}} \leq E_{p_{limit}} = E_{p_{safe}} \quad (3.25)$$

With:  $0 \leq E_{p_{safe}} \leq E_{c_{safe}}$

### 3.4 Energy constraints implementation

Constraints on kinetic (3.15) and potential (3.25) energies can be implemented into an optimization control scheme and used to modulate the energy of a robotic manipulator during different interaction phases with its environment. Before physical contact and during the approach of the human operator, only the constraint on kinetic energy (3.15) is needed. At collision, this constraint ensures that a safe amount of kinetic energy is dissipated creating a non hazardous impact force. At this stage, collision between the robot and its environment can be detected (using for example the proprioceptive articular torque sensors), and as it is no longer needed, the constraint on  $S_c$  can be removed and replaced by the constraint on potential energy (3.25) (see Fig. 3.7). At the establishment of physical contact, potential energy within the human-robot system starts increasing to reach its maximum allowed value  $E_{p_{limit}} = E_{p_{safe}}$ . Depending on  $E_{p_{safe}}$ , the robot is more or less compliant and can be moved easily. When physical contact is released, the constraint on potential energy can be removed and the one on kinetic energy (3.15) reintroduced in the configuration of the controller to prevent hazardous movements of the robot.

### 3.5 Task energy profile

In case of a trajectory tracking task, the amount of potential energy  $E_{p|k}$  (3.13) instantaneously *injected* in the controller of the robot can be measured at every time-step. As soon as  $E_{p|k}$  is injected, the pulling force derived from it is created and drives the robot along its discretized trajectory. Because of this discretization, every time-step, only a small amount of potential energy  $E_{p|k}$  is injected and promptly transformed into kinetic one as the robot moves. For a repetitive trajectory tracking task, the energy profile  $E_{p_{profile}}(t)$  (with  $E_{p_{profile}}(t = k\delta t) = E_{p|k}$ ) can be initially registered during the repetitive *free* movements of the robot. Finally this profile can be used to limit the instantaneous amount of potential energy  $E_{p|k}$  the controller of the robot is allowed to contain at every time-step. Therefore, in case of a deliberate or accidental collision/contact, this constraint prevents the generation of large amounts of potential energy<sup>14</sup> and thus of hazardous contact forces between the robot and its environment.  $E_{p_{profile}}(t)$  is then used as  $E_{p_{safe}}$  during physical contact phases. To resume:

- Every repetitive trajectory tracking task has its own energy profile. Namely, the minimum amount of energy needed to accomplish the assigned task.
- This energy profile defined in operational space<sup>15</sup> is altered in case of any contact with the environment, more energy accumulates in the controller of the robot.
- A constraint on the instantaneously allowed amount of potential energy in the controller prevents the robot from crushing any obstacle at the establishment of a physical contact.

The constraint on the *task energy profile* of a trajectory tracking task can be expressed at the level of any potential collision point  $C$ . It is written at the end-effector of the robot:

$$S_{p_{profile}} = m(\mathbf{q}|k)_{EE,EE^*}^{eq} \ddot{\mathbf{X}}_{EE|k}^{EE,EE^*} \left( \mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k} \right) \mathbf{n}_{EE} \leq E_{p_{limit}}. \quad (3.26)$$

Note that the expression of  $S_{p_{profile}}$  is the same as the expression of  $S_{p_{contact}}$  that has been defined as the robot's safety indicator exclusively for the physical contact phase (3.8). The safety

<sup>14</sup>The potential energy during physical contact is the extension of the potential energy *injected* in the controller of the robot during its free movements.

<sup>15</sup>Can also be defined in joint space.

indicator related to the controller's task energy profile  $S_{p_{profile}}$  on the other hand, is defined for both the *free movements* and the *physical contact* phases. Constraining the task energy profile of the controller using (3.26), prevents the robot from deploying any additional energy than what is needed at time-step  $k$  for an optimal completion of its trajectory tracking task. As it is activated only in case of collisions/physical contact between the robot and its environment, this constraint can always be included in the controller and used as a low-level security layer without altering the tracking performances related to the assigned task. Even if collisions with the environment are not detected<sup>16</sup>, the robot will automatically comply to any external force preventing therefore the generation of harmful contact forces.

When the constraint on the task energy profile (3.26) is used to limit the potential energy injected at each time-step in the system, (3.24) or (3.3) can simultaneously be implemented in the controller and used to limit the deployed kinetic energy. Both constraints ((3.26) with (3.24)) or ((3.26) with (3.3)) are compatible and can be implemented *at the same time* in the optimization control scheme.

On the other hand, in case  $E_{p_{limit}}$  is fixed  $< E_{p_{profile}}$  in (3.26), the performance of the trajectory tracking task will inevitably be deteriorated. Indeed,  $E_{p_{profile}}$  is the minimum amount of potential energy the system needs to transform into kinetic one to accomplish its task in the most optimal way. At every time-step  $k$ ,  $E_{p_{profile}}$  can be written:

$$E_{p_{profile}}(t = k\delta t) = m(\mathbf{q}_{|k})_{EE,EE}^{eq} \ddot{X}_{EE|k} \left( \mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k} \right) \mathbf{n}_{EE}. \quad (3.27)$$

Or when decoupled along the  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  axis in operational space (the desired trajectory in Cartesian space being fed to the controller independently along these 3 axis):

$$\begin{cases} E_{p_{profile}}^x = m(\mathbf{q}_{|k})_x^{eq} \ddot{X}_{EE|k}^x \left( \mathbf{X}_{x|k}^* - \mathbf{X}_{x|k} \right) \mathbf{x}, & (3.28a) \\ E_{p_{profile}}^y = m(\mathbf{q}_{|k})_y^{eq} \ddot{X}_{EE|k}^y \left( \mathbf{X}_{y|k}^* - \mathbf{X}_{y|k} \right) \mathbf{y}, & (3.28b) \\ E_{p_{profile}}^z = m(\mathbf{q}_{|k})_z^{eq} \ddot{X}_{EE|k}^z \left( \mathbf{X}_{z|k}^* - \mathbf{X}_{z|k} \right) \mathbf{z}. & (3.28c) \end{cases}$$

Which when saturated allows enabling compliance to physical contact independently along the desired axis.

## 3.6 Safe dynamic controller

In this section a dynamic controller that can ensure safety for both the robot and its environment is proposed. The objective is to compute at every time-step the control torque  $\boldsymbol{\tau}_{|k}^c$  in order to perform a trajectory tracking task while coping with a number of constraints:

- satisfying the introduced safety criteria to prevent harmful collision and contact forces;
- coping with the limits corresponding to the articular physical capabilities of the robotic manipulator.

### 3.6.1 Task formulation

The objective function of the controller is defined as an error to be minimized. It could be for example an acceleration task if the robot has to track a trajectory, or a wrench task if

<sup>16</sup>For example in case of a fail of an exteroceptive sensor used to detect the collision.



the interaction wrench with the environment needs to be controlled. In the presented work, a trajectory tracking performance is considered. A Cartesian acceleration task is defined as an error between the desired acceleration  $\ddot{\mathbf{X}}^*$  and the expected acceleration  $\ddot{\mathbf{X}}^c$  of the end-effector of the robot. Considering  $\ddot{\mathbf{X}}^c = J(\mathbf{q}_{|k})_{EE}\ddot{\mathbf{q}}_{|k}^c + \dot{J}(\mathbf{q}_{|k})_{EE}\dot{\mathbf{q}}_{|k}$ , with  $J(\mathbf{q}_{|k})_{EE}$  the instantaneous Jacobian related to the end-effector, using the equation of motion of the system,  $\ddot{\mathbf{X}}^c$  can also be written as function of the control input  $\tau_{|k}^c$ :

$$\ddot{\mathbf{X}}^c = J(\mathbf{q}_{|k})M(\mathbf{q}_{|k})^{-1} \left( \tau_{|k}^c - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}) \right) + \dot{J}(\mathbf{q}_{|k})\dot{\mathbf{q}}_{|k}, \quad (3.29)$$

$\mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k})$  are the non linear terms of the equation of motion, namely gravity, Coriolis and centrifugal induced generalized forces.  $\ddot{\mathbf{X}}^*$  can be computed with a PD controller and a feed-forward acceleration term  $\ddot{\mathbf{X}}_{ff}^*$  in order to track a desired position  $\mathbf{X}^*$  and velocity  $\dot{\mathbf{X}}^*$  in operational space:

$$\ddot{\mathbf{X}}^* = \ddot{\mathbf{X}}_{ff}^* + K_p(\mathbf{X}^* - \mathbf{X}) + K_d(\dot{\mathbf{X}}^* - \dot{\mathbf{X}}), \quad (3.30)$$

where  $K_p, K_d \in \mathbb{R}^+$  are the proportional and derivative gains.

The acceleration task function to be minimized is finally written:

$$\mathbf{g} \left( \tau_{|k}^c, \ddot{\mathbf{X}}^c \right) = \ddot{\mathbf{X}}^* - \left( J(\mathbf{q}_{|k})M(\mathbf{q}_{|k})^{-1} \left( \tau_{|k}^c - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}) \right) + \dot{J}(\mathbf{q}_{|k})\dot{\mathbf{q}}_{|k} \right). \quad (3.31)$$

## 3.6.2 Constraints formulation

### 3.6.2.1 Articular constraints

First, the limitations corresponding to the articular physical capabilities of the robot must be accounted for when solving the control problem. The computed control input  $\tau_{|k}^c$  at time-step  $k$  must be such that the considered limits are never violated and the *viability* of the state of the robot preserved. The actuators limitations are considered at the following levels:  $\mathbf{q}$ ,  $\dot{\mathbf{q}}$ ,  $\ddot{\mathbf{q}}$  and  $\ddot{\ddot{\mathbf{q}}}$ . Usually, the constraints corresponding to the actuation limitations are expressed as inequalities and activated only one time-step before reaching their respective Max/mi boundaries. As explained in Chapter 2, this can result into *incompatibility problems* that may render the control problem impossible solve. For e.g., no sufficient articular torque/jerk may be producible to cope with an articular position or velocity limit within only one control time-step = 1ms. Therefore, it is the new formulations of the constraints introduced in Chapter 2 that are hereby used to bound the physical capabilities of the actuators of the robot. The inequality constraints corresponding to the articular physical limitations of the robot are written:

$$\left\{ \begin{array}{l} \mathbf{q}_{|k+n_{15}+n_{17}+n_{19}} \leq \mathbf{q}_M, \\ \mathbf{q}_{|k+n_{16}+n_{18}+n_{20}} \geq \mathbf{q}_m, \\ \dot{\mathbf{q}}_{|k+n_1} \leq \dot{\mathbf{q}}_M, \\ \dot{\mathbf{q}}_{|k+n_2} \geq \dot{\mathbf{q}}_m, \\ \tau_m \leq \tau_{|k}^c \leq \tau_M, \\ \ddot{\mathbf{q}}_m \leq \ddot{\mathbf{q}}_{|k+1} \leq \ddot{\mathbf{q}}_M, \end{array} \right. \quad \begin{array}{l} (3.32a) \\ (3.32b) \\ (3.32c) \\ (3.32d) \\ (3.32e) \\ (3.32f) \end{array}$$

with  $\mathbf{q}_{|k+n_{15}+n_{17}+n_{19}}$ ,  $\mathbf{q}_{|k+n_{16}+n_{18}+n_{20}}$  as in (2.50) and  $\dot{\mathbf{q}}_{|k+n_1}$ ,  $\dot{\mathbf{q}}_{|k+n_2}$  equivalent to (2.13) from Chapter 2. To be easily accounted for, these constraints must be expressed as a function of the

control variable  $\ddot{\mathbf{q}}_{|k}^c$ . Given the state of the system at instant  $k$  and a discrete linear approximation of the behaviour of the robot in joint space:

$$\left\{ \begin{array}{l} f_{\chi}(\mathbf{q}_m, \ddot{\mathbf{q}}_M, \ddot{\mathbf{q}}_m, n_{16}, n_{18}, n_{20}) \leq \ddot{\mathbf{q}}_{|k}^c \leq f_{\phi}(\mathbf{q}_M, \ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M, n_{15}, n_{17}, n_{19}), \\ f_{\beta}(\dot{\mathbf{q}}_{|k}, \dot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M, n_2) \leq \ddot{\mathbf{q}}_{|k}^c \leq f_{\alpha}(\dot{\mathbf{q}}_{|k}, \dot{\mathbf{q}}_M, \ddot{\mathbf{q}}_m, n_1), \\ \tau_m \leq \tau_{|k}^c \leq \tau_M, \\ \ddot{\mathbf{q}}_m \delta t + \dot{\mathbf{q}}_{|k} \leq \ddot{\mathbf{q}}_{|k}^c \leq \ddot{\mathbf{q}}_M \delta t + \dot{\mathbf{q}}_{|k}, \end{array} \right. \quad (3.33a)$$

$$f_{\beta}(\dot{\mathbf{q}}_{|k}, \dot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M, n_2) \leq \ddot{\mathbf{q}}_{|k}^c \leq f_{\alpha}(\dot{\mathbf{q}}_{|k}, \dot{\mathbf{q}}_M, \ddot{\mathbf{q}}_m, n_1), \quad (3.33b)$$

$$\tau_m \leq \tau_{|k}^c \leq \tau_M, \quad (3.33c)$$

$$\ddot{\mathbf{q}}_m \delta t + \dot{\mathbf{q}}_{|k} \leq \ddot{\mathbf{q}}_{|k}^c \leq \ddot{\mathbf{q}}_M \delta t + \dot{\mathbf{q}}_{|k}, \quad (3.33d)$$

with  $f_{\chi}$ ,  $f_{\phi}$ ,  $f_{\beta}$  and  $f_{\alpha}$  respectively equivalent to (2.53), (2.54), (2.14) and (2.16) from Chapter 2. Finally, in addition to the articular inequality constraints, the system is also subject to linear equality constraints corresponding to its dynamic model:

$$M(\mathbf{q}_{|k})\ddot{\mathbf{q}}_{|k}^c + \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}) = \tau_{|k}^c \quad (3.34)$$

### 3.6.2.2 Energy related constraints

In an equivalent way, the introduced safety indicators  $S_c$ ,  $S_{p_{free}}$ ,  $S_{p_{contact}}$  and  $S_{p_{profile}}$  can be expressed as functions of the dynamic control variable  $\ddot{\mathbf{q}}_{|k}^c$ .

First, based on (3.15), during the *free* movements of the robot,  $S_c$  can be written:

$$S_c = E_{c_{|k+1}}^{EE,O} = \frac{1}{2} \text{sign} \left( v_{EE|k+1}^{EE,O} \right) m(\mathbf{q}_{|k})_{EE,O}^{eq} v_{EE|k+1}^{EE,O^2}, \quad (3.35)$$

With:

$$\left\{ \begin{array}{l} v_{EE|k+1}^{EE,O} = J(\mathbf{q}_{|k})_{EE}^{EE,O} \dot{\mathbf{q}}_{|k+1}, \\ \dot{\mathbf{q}}_{|k+1} = \dot{\mathbf{q}}_{|k} + \ddot{\mathbf{q}}_{|k}^c \delta t. \end{array} \right. \quad (3.36a)$$

$$\dot{\mathbf{q}}_{|k+1} = \dot{\mathbf{q}}_{|k} + \ddot{\mathbf{q}}_{|k}^c \delta t. \quad (3.36b)$$

Considering the *second formulation* (3.24) of the same kinetic energy based safety indicator  $S_c$ , it can also be written:

$$S_c = E_{c_{|k+1}}^{EE,O} = \underbrace{E_{c_{|k}}^{EE,O}}_{\text{measured}} + S_{p_{free}}^{17} = E_{c_{|k}}^{EE,O} + m(\mathbf{q}_{|k})_{EE,O}^{eq} \ddot{X}_{EE|k}^{EE,O} \left( \mathbf{X}_{EE|k+1} - \mathbf{X}_{EE|k} \right) \mathbf{n}_O, \quad (3.37)$$

$$\text{with: } \ddot{X}_{EE|k}^{EE,O} = \dot{J}(\mathbf{q}_{|k})_{EE}^{EE,O} \dot{\mathbf{q}}_{|k} + J(\mathbf{q}_{|k})_{EE}^{EE,O} \ddot{\mathbf{q}}_{|k}^c. \quad (3.38)$$

$X_{EE|k+1}$  is the future position of the end-effector that can be estimated as:

$$\mathbf{X}_{EE|k+1} = \underbrace{\mathbf{X}_{EE|k}}_{\text{measured}} + \underbrace{\dot{\mathbf{X}}_{EE|k}}_{\text{measured}} \delta t + \frac{1}{2} \underbrace{\ddot{\mathbf{X}}_{EE|k}}_{\text{measured}} \delta t^2. \quad (3.39)$$

On the other hand, during physical contact,  $S_{p_{contact}}$  is expressed:

$$S_{p_{contact}} = m(\mathbf{q}_{|k})_{EE,*}^{eq} \ddot{X}_{EE|k}^{EE,*} \left( \mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k} \right) \mathbf{n}_{EE}, \quad (3.40)$$

$$\text{with: } \ddot{X}_{EE|k}^{EE,*} = \dot{J}(\mathbf{q}_{|k})_{EE}^{EE,*} \dot{\mathbf{q}}_{|k} + J(\mathbf{q}_{|k})_{EE}^{EE,*} \ddot{\mathbf{q}}_{|k}^c. \quad (3.41)$$

<sup>17</sup>Here,  $S_{p_{free}}$  is the injected *potential energy* expressed at the level of the end-effector of the robot in the direction of the considered obstacle  $O$  during the *free* movements of the robot.

Or independently along the  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  axis in Cartesian space,  $S_{p_{contact}}$  can be decomposed:

$$\begin{cases} S_{p_{contact}}^x = m(\mathbf{q}_{|k})_{xx}^{eq} \ddot{X}_{EE|k}^{\mathbf{x}} \left( \mathbf{X}_{x|k}^* - \mathbf{X}_{x|k} \right) \mathbf{x}, & (3.42a) \\ S_{p_{contact}}^y = m(\mathbf{q}_{|k})_{yy}^{eq} \ddot{X}_{EE|k}^{\mathbf{y}} \left( \mathbf{X}_{y|k}^* - \mathbf{X}_{y|k} \right) \mathbf{y}, & (3.42b) \\ S_{p_{contact}}^z = m(\mathbf{q}_{|k})_{zz}^{eq} \ddot{X}_{EE|k}^{\mathbf{z}} \left( \mathbf{X}_{z|k}^* - \mathbf{X}_{z|k} \right) \mathbf{z}. & (3.42c) \end{cases}$$

Finally, the constraints related to  $S_c$  and  $S_p$  that will be tested in simulation then on the real robot using the proposed controller are as follows:

- During the *pre-collision* phase:

1. The constraint on the kinetic energy of the robot (first formulation):

$$S_c = \frac{1}{2} \text{sign} \left( v_{EE|k+1}^{EE,O} \right) m(\mathbf{q}_{|k})_{EE,O}^{eq} v_{EE|k+1}^{EE,O^2} \leq E_{climit} = E_{c_{safe}} + K(d - d_{safe}), \quad (3.43)$$

with  $\text{sign} \left( v_{EE|k+1}^{EE,O} \right)$  estimated by:

$$\text{sign} \left( v_{EE|k+1}^{EE,O} \right) = \text{sign} \left( v_{EE|k}^{EE,O} + \underbrace{\ddot{X}_{EE|k}^{EE,O}}_{measured} \delta t \right). \quad (3.44)$$

We recall that  $\text{sign} \left( v_{EE|k+1}^{EE,O} \right)$  is positive for the movements of the robot towards the considered obstacle and negative otherwise.

2. The constraint on the kinetic energy of the robot (second formulation):

$$S_c = E_{c_{|k}}^{EE,O} + S_{p_{free}} \leq E_{climit} = E_{c_{safe}} + K(d - d_{safe}). \quad (3.45)$$

Which transforms into:

$$\underbrace{m(\mathbf{q}_{|k})_{EE,O}^{eq} \ddot{X}_{EE|k}^{EE,O} \left( \mathbf{X}_{EE|k+1} - \mathbf{X}_{EE|k} \right) \mathbf{n}_O}_{S_{p_{free}}} \leq E_{climit} - \underbrace{E_{c_{|k}}^{EE,O}}_{measured}, \quad (3.46)$$

with  $\mathbf{X}_{EE|k+1}$  and  $\ddot{X}_{EE|k}^{EE,O}$  respectively as in (3.39) and (3.38).

- During the *physical-contact* phase:

$$S_{p_{contact}} = m(\mathbf{q}_{|k})_{EE,EE^*}^{eq} \ddot{X}_{EE|k}^{EE,EE^*} \left( \mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k} \right) \mathbf{n}_{EE} \leq E_{plimit} = E_{p_{safe}}. \quad (3.47)$$

When decoupled along the  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  axis in Cartesian space:

$$\begin{cases} S_{p_{contact}}^x = m(\mathbf{q}_{|k})_{xx}^{eq} \ddot{X}_{EE|k}^{\mathbf{x}} \left( \mathbf{X}_{x|k}^* - \mathbf{X}_{x|k} \right) \mathbf{x} \leq E_{p_{safe}}^x, & (3.48a) \\ S_{p_{contact}}^y = m(\mathbf{q}_{|k})_{yy}^{eq} \ddot{X}_{EE|k}^{\mathbf{y}} \left( \mathbf{X}_{y|k}^* - \mathbf{X}_{y|k} \right) \mathbf{y} \leq E_{p_{safe}}^y, & (3.48b) \\ S_{p_{contact}}^z = m(\mathbf{q}_{|k})_{zz}^{eq} \ddot{X}_{EE|k}^{\mathbf{z}} \left( \mathbf{X}_{z|k}^* - \mathbf{X}_{z|k} \right) \mathbf{z} \leq E_{p_{safe}}^z & (3.48c) \end{cases}$$

Note: as formulated, constraints in (3.47) and (3.48) prevent the robot from generating harmful contact forces but only along the positive direction of the considered axis. With such formulations, contact forces are saturated during physical contact but as nothing prevents the accumulation of *negative potential energy* in the controller of the robot during physical contact, the robot cannot be pushed in the opposite direction. Which makes it dangerous regarding the expected *safe* human-robot interaction. To enable the robot to be pushed along all the directions in operational space, it is the absolute value of its *potential energy* that should be constrained. It is then saturated in both the positive and negative directions of the considered axis. (3.47) and (3.48) are thus rewritten:

$$|S_{p_{contact}}| \leq E_{p_{safe}} \Leftrightarrow \begin{cases} S_{p_{contact}} \leq E_{p_{safe}}, \\ S_{p_{contact}} \geq -E_{p_{safe}}. \end{cases} \quad (3.49a)$$

$$(3.49b)$$

And when decoupled:

$$\begin{cases} S_{p_{contact}}^x \leq E_{p_{safe}}^x, \\ S_{p_{contact}}^x \geq -E_{p_{safe}}^x, \\ S_{p_{contact}}^y \leq E_{p_{safe}}^y, \\ S_{p_{contact}}^y \geq -E_{p_{safe}}^y, \\ S_{p_{contact}}^z \leq E_{p_{safe}}^z, \\ S_{p_{contact}}^z \geq -E_{p_{safe}}^z. \end{cases} \quad (3.50a)$$

$$(3.50b)$$

$$(3.50c)$$

$$(3.50d)$$

$$(3.50e)$$

$$(3.50f)$$

- **During both the *pre-collision* and the *physical-contact* phases:**

The constraint on the *task energy profile* of the robot can be written:

$$|S_{p_{profile}}| = \left| m(\mathbf{q})_{EE,*}^{eq} \ddot{\mathbf{X}}_{EE|k}^{EE,*} \left( \mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k} \right) \mathbf{n}_{EE} \right| \leq E_{p_{profile}}^{18} + \epsilon_{E_p}. \quad (3.51)$$

Which can also be decoupled along the  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  axis in Cartesian space:

$$\begin{cases} \left| S_{p_{profile}}^x \right| = \left| m(\mathbf{q}|_k)_x^{eq} \ddot{\mathbf{X}}_{EE|k}^x \left( \mathbf{X}_{x|k}^* - \mathbf{X}_{x|k} \right) \mathbf{x} \right| \leq E_{p_{profile}}^x + \epsilon_{E_p}^x, \\ \left| S_{p_{profile}}^y \right| = \left| m(\mathbf{q}|_k)_y^{eq} \ddot{\mathbf{X}}_{EE|k}^y \left( \mathbf{X}_{y|k}^* - \mathbf{X}_{y|k} \right) \mathbf{y} \right| \leq E_{p_{profile}}^y + \epsilon_{E_p}^y, \\ \left| S_{p_{profile}}^z \right| = \left| m(\mathbf{q}|_k)_z^{eq} \ddot{\mathbf{X}}_{EE|k}^z \left( \mathbf{X}_{z|k}^* - \mathbf{X}_{z|k} \right) \mathbf{z} \right| \leq E_{p_{profile}}^z + \epsilon_{E_p}^z. \end{cases} \quad (3.52a)$$

$$(3.52b)$$

$$(3.52c)$$

The *task energy profile* related constraints (3.51) or (3.52) can be implemented simultaneously with the constraints on the kinetic energy of the robot: (3.43) or (3.45). A *task energy profile* related constraint is defined for both the *pre-collision* and *physical-contact* interaction phases. It can therefore replace the constraint on the robot's *potential energy* (3.50) to render the robot more compliant when physical contact is established.  $\epsilon_{E_p}$ ,  $\epsilon_{E_p}^x$ ,  $\epsilon_{E_p}^y$  and  $\epsilon_{E_p}^z$  are constants used to compensate the repeatability related variations when registering the energy profiles:  $E_{p_{profile}}(t)$ ,  $E_{p_{profile}}^x(t)$ ,  $E_{p_{profile}}^y(t)$  and  $E_{p_{profile}}^z(t)$ .

<sup>18</sup>Measured during the *free* unconstrained movements of the robot.

### 3.6.3 Controller formulation

The proposed controller computes the command torque by minimizing the norm of the Cartesian acceleration task function expressed in the following quadratic form:

$$\arg \min_{\tau_{|k}^c, \ddot{\mathbf{q}}_{|k}^c} \left\| \mathbf{g} \left( \tau_{|k}^c, \ddot{\mathbf{X}}^c \right) \right\|_{Q_t}^2 + \epsilon \left\| \tau_{|k}^c \right\|_{Q_r}^2, \quad (3.53)$$

subject to the constraints regarding the physical articular capabilities of the robot (3.33) and to the equality constraints related to its dynamic model (3.34). The controller will also be tested with: 1) the first (3.43) and second (3.45) formulations of the constraint on the kinetic energy of the robot, 2) the constraint on the *potential energy* that accumulates in its controller during physical contact (3.47) and, 3) the constraint related to the *task energy profile* (3.51).

Note: because of possible incompatibility issues that may occur between the *energy based constraints* and the constraint on articular jerk (3.33d), the latter will be removed from the controller whenever an energy related constraint is to be activated. Indeed, the activation of such constraint can require high values of articular jerk the actuators of the robot may not be capable of producing. Consequently, the *viability* of the state of the robot can be altered and the LQP may become impossible to solve.

$Q_t$  and  $Q_r$  are positive semi-definite weighting matrices and  $\|\mathbf{a}\|_Q$  is the  $Q$ -weighted euclidean norm of  $\mathbf{a}$ . In the upcoming simulations,  $Q_t$  and  $Q_r$  are fixed as identity matrices.  $\epsilon \|\tau_{|k}^c\|_{Q_r}^2$  with  $\epsilon = 10^{-6}$  serves as a regularization task in order to ensure the uniqueness of the control solution and to minimize the norm of the computed control torque.

It can be shown that the quadratic forms composing the tasks expression can be written as a function of positive semidefinite matrix. Constraints expressed in (3.33), (3.43), (3.45), (3.51) and (3.47) are linear and can be written on the form:  $\mathbf{G}\mathbf{r} \leq \mathbf{h}$  with:  $\mathbf{r} = [\tau_{|k}^c, \ddot{\mathbf{q}}_{|k}^c]^T$ . This LQP optimization problem is then convex and admits a unique global minimum.

## 3.7 Simulation

The controller described in Section 3.6 is implemented as a C++ Orocos component [Soetens 2017] on a virtual model of the KUKA LWR4 serial robot using XDE, a robotics-oriented physics simulation engine [Merlhiot 2012].

In this section, as it interacts with the environment, different behaviours of the robotic manipulator that can be induced by a straightforward modification of its controller's parameters ( $d_{safe}$ ,  $E_{c_{safe}}$ ,  $E_{p_{safe}}$  and  $E_{p_{limit}}$ ) are presented and discussed. First, a test case scenario used as a basis for the various configurations<sup>19</sup> of the controller is presented. An obstacle is then introduced in the workspace of the robot and different interaction modes are simulated. To evaluate the advantage of using the proposed control scheme and the introduced energy related constraints, non-physical interactions and collision tests between the robot and its environment are performed *with* and *without* the introduced constraints.

### 3.7.1 Test case scenario

As a main activity, the robot performs a repetitive pick and place movement where it tracks a desired position and orientation in Cartesian space. As in Fig. 3.8, the trajectory for the

<sup>19</sup>For each test and interaction phase, only the needed energy related constraints are added to the *default* configuration of the controller, i.e., the task objective function (3.53) along with the two constraints: (3.33) and (3.34).

pick and place movement is segmented into 3 main segments: segment ①-②, ②-③ and ③-④. The generation of trajectory for the next segment is not performed until the end-effector of the robot reaches the end of the segment it is currently going over. While performing its trajectory tracking task, the robot in this scenario carries a load of 5 kg by its end-effector. First, the controller described in (3.53) is implemented without any energy related constraint. Only the linear constraints (3.33) that correspond to the system's articular limitations in addition to the equality constraints (3.34) related to its dynamic model are considered. The LQP is solved at every time-step using Gurobi, a commercial optimization software [Gurobi Optimization Inc. 2015] to compute in real-time the needed control torque input. In the test case scenario,

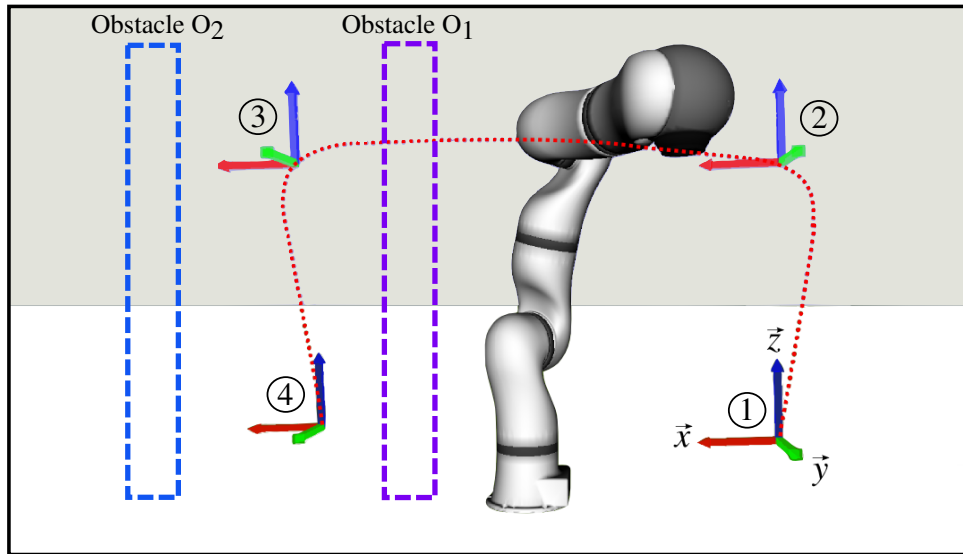


Figure 3.8: The KUKA LWR4 robotic manipulator within the XDE simulated world near its considered obstacle. Case  $O_1$  is when the obstacle intersects with the trajectory of the robot. Case  $O_2$  is when the obstacle is nearby the robot but does not intersect with its trajectory.

the movement of the robot is not obstructed and the pick and place task is performed with the needed kinetic energy to properly track the desired trajectory  $X^*(t)$  in the most optimal way. Trajectory tracking performances can be observed in Fig. 3.9 and Fig. 3.10. Maximum tracking errors in Cartesian space are:  $1 \times 10^{-3} m$  for position and  $2.3 \times 10^{-2} rad$  for orientation.

One of the main advantages of using a QP to compute the actuation torque is the possibility to take into account the physical limitations of the system. Fig. 3.11 and Fig. 3.12 show how the limits on articular position, velocity, torque and also jerk are satisfied whenever a joint reaches its considered bounds. Along the movement of the robot, articular positions are within their max/min boundaries. And thanks to the new formulation of the constraint on articular velocity ((3.32c) & (3.32d)), the compatibility with the constraint on articular jerk (3.32f) is ensured. Indeed, when coping with a joint velocity limit (see joint 0 in Fig. 3.11.b and Fig. 3.12.b), maximum producible jerk is used to bring the articular acceleration to zero. More details about this new formulation of the joint velocity constraint can be found in Chapter 2, Subsection 2.4.1.

The trajectory of the robot in this scenario does not intersect with any obstacle. During this *free movement*, a number of parameters related to the dynamic capabilities of the robot in operational space and to the accomplished task are registered. Fig. 3.10.a illustrates the instantaneous producible equivalent braking force in Cartesian space expressed at the level of the end-effector in the opposite direction to the nearby obstacle (case  $O_2$  in Fig. 3.8). This force depends on

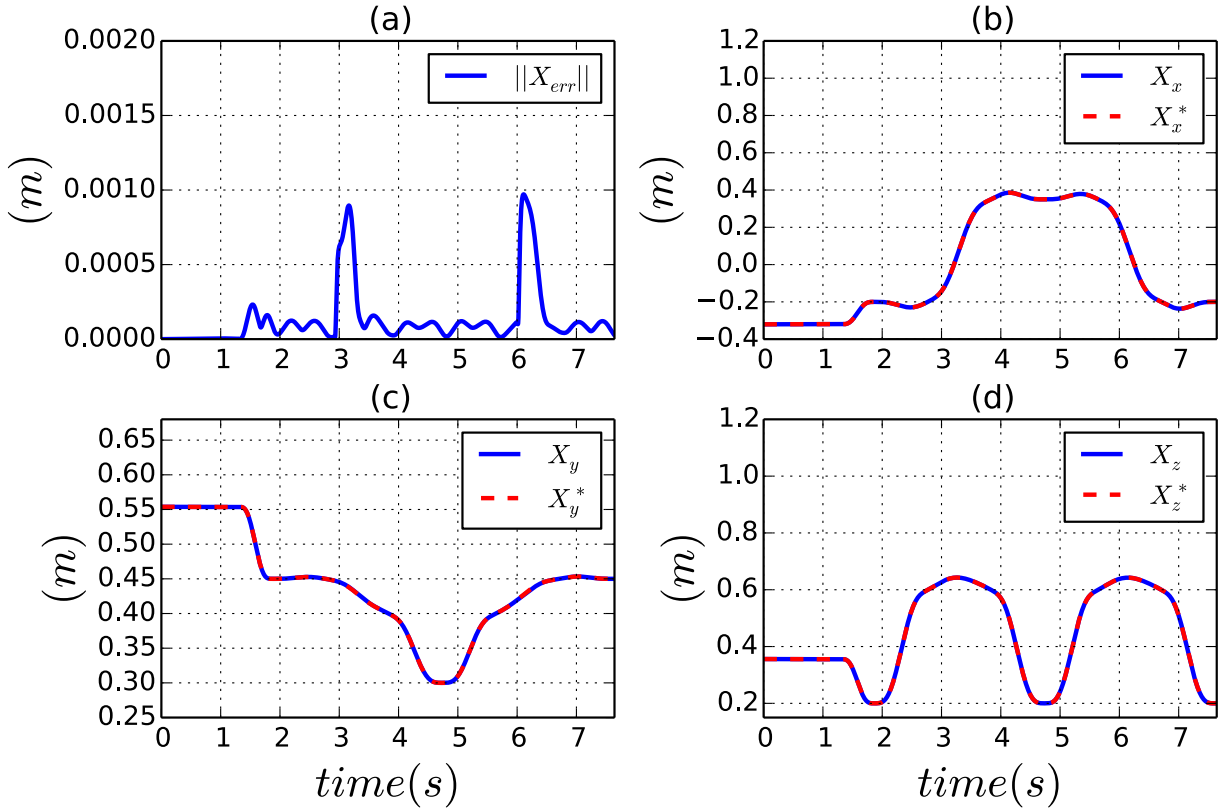


Figure 3.9: (a) Norm of the position tracking error for the end-effector of the robot in Cartesian space. (b), (c) and (d) are the real and desired position for the end-effector along the  $x$ ,  $y$  and  $z$  axis in Cartesian space.

the instantaneous articular configuration  $\mathbf{q}_{|k}$  and on the amount of torque producible by the actuators of the robot.  $K$  is computed as described in (3.19) and its minimum guaranteed value along this movement will be used to synthesise the kinetic energy based safety criteria introduced in (3.17). Fig. 3.10.b, c and d depict how the desired velocity  $\dot{\mathbf{X}}^*(t)$  of the end-effector of the robot is properly tracked in real-time. Kinetic energy of the KUKA LWR4, expressed at its end-effector and generated in the direction of the nearby considered obstacle (case  $O_2$  in Fig. 3.8) is shown in Fig. 3.13.a. Its maximum value is  $2.15 J$ ; which is equivalent to a  $5 kg$  mass, moving at  $0.92 m/s$ . At every time-step, the plotted kinetic energy is computed using two different formulations:

$$S_c = E_{c_{|k}}^{EE,O} = \frac{1}{2} m(\mathbf{q}_{|k})_{EE,O}^{eq} v_{EE_{|k}}^{EE,O^2}, \quad (3.54)$$

and:

$$S_c = E_{c_{|k}}^{EE,O} = \sum_{n=1}^{k-1} S_{p_{free|n}} = \sum_{n=1}^{k-1} m(\mathbf{q}_{|n})_{EE,O}^{eq} \ddot{X}_{EE_{|n}}^{EE,O} (\mathbf{X}_{EE_{|n+1}} - \mathbf{X}_{EE_{|n}}) \mathbf{n}_O. \quad (3.55)$$

Fig. 3.13.a illustrates how the kinetic energy of the robot at time-step  $k$  is indeed equal (errors aside) to the sum of all the previously *injected* potential energies.

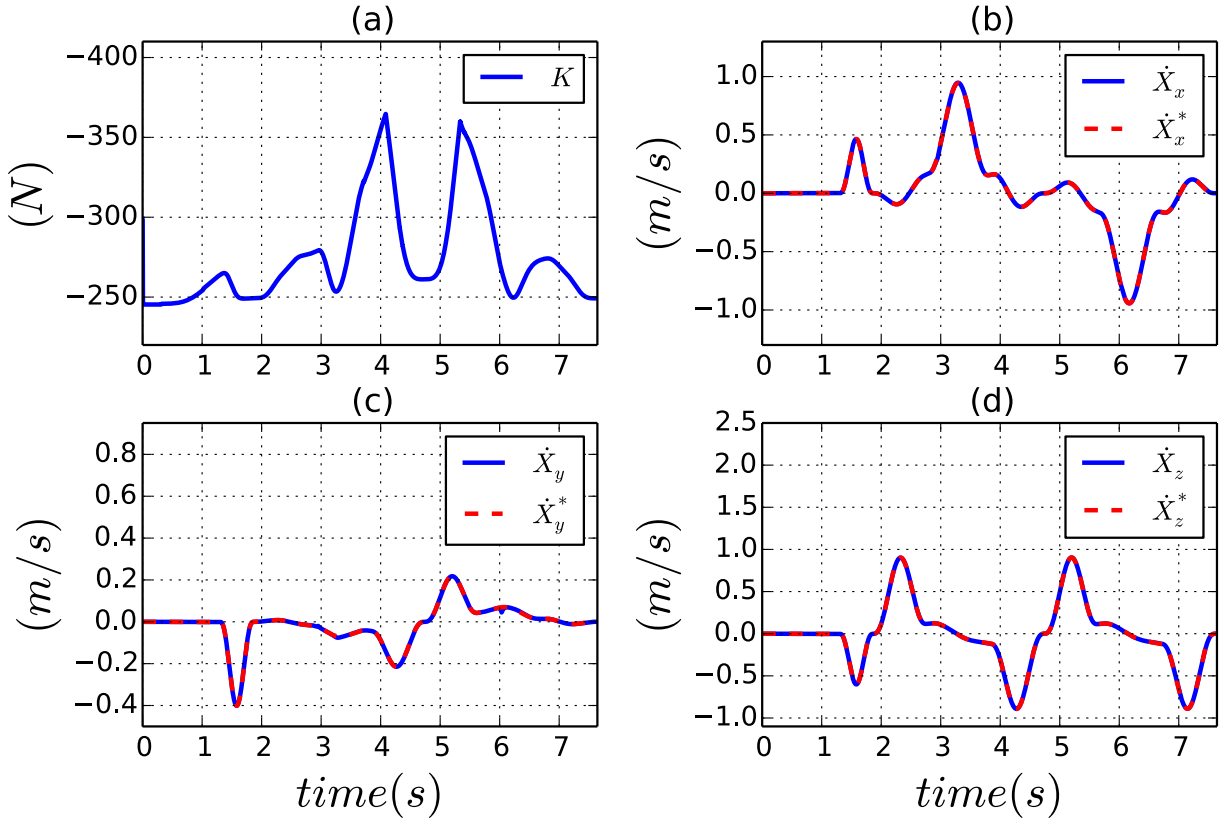


Figure 3.10: (a) Producibile braking force of the robot expressed at the level of its end-effector in the opposite direction to the considered obstacle (case  $O_2$  in Fig. 3.8). (b), (c) and (d) are the instantaneous real and desired velocities along the  $x$ ,  $y$  and  $z$  axis in Cartesian space.

### 3.7.2 Scenario 1: obstacle intersecting with the trajectory of the robot and no constraints on its energy

In this scenario, the obstacle intersects with the ②-③ segment of the pick and place movement (case  $O_1$  in Fig. 3.8). When collision occurs between the robot and the rigid object, most of its kinetic energy is rapidly dissipated. Fig. 3.14 shows how 1.66  $J$  of this kinetic energy are instantaneously dissipated at impact. According to (3.1) and as can be seen in Fig. 3.15.b, this fast dissipation induces a large impact force of 551  $N$ . Which could clearly damage the collided obstacle or the robot. In this simulation and also in all the upcoming ones, a force sensor is linked to the base of the collided object (i.e., the wall). The main impact force is along the  $x$  axis (see Fig. 3.8), which is the main direction of movement of the end-effector of the robot before collision. After the peak of force corresponding to the first collision, physical contact between the robot and the wall is established. As this obstacle obstructs the movement of the KUKA LWR4, the divergence between the desired and real positions for its end-effector causes the augmentation of the amount of *potential energy* that accumulates in the controller of the robot (see Fig. 3.15.a). Consequently, the resulting contact force along the  $x$  axis also intensifies. The contact force along the  $x$  axis is mainly caused by the torques generated by joint 0 and joint 2. Fig. 3.15.c shows how  $\tau_0$  and  $\tau_2$  increase to respectively reach  $\approx -115 N.m$  and  $\approx -95 N.m$ <sup>20</sup>. As depicted in Fig. 3.15.b, the contact force that correspond to these generated torques settles

<sup>20</sup>In the configuration of the robot as seen in Fig. 3.8, negative torque on joint 0 and on joint 2 causes the end-effector of the robot to move towards the wall. Positive torque makes it move in the opposite direction.



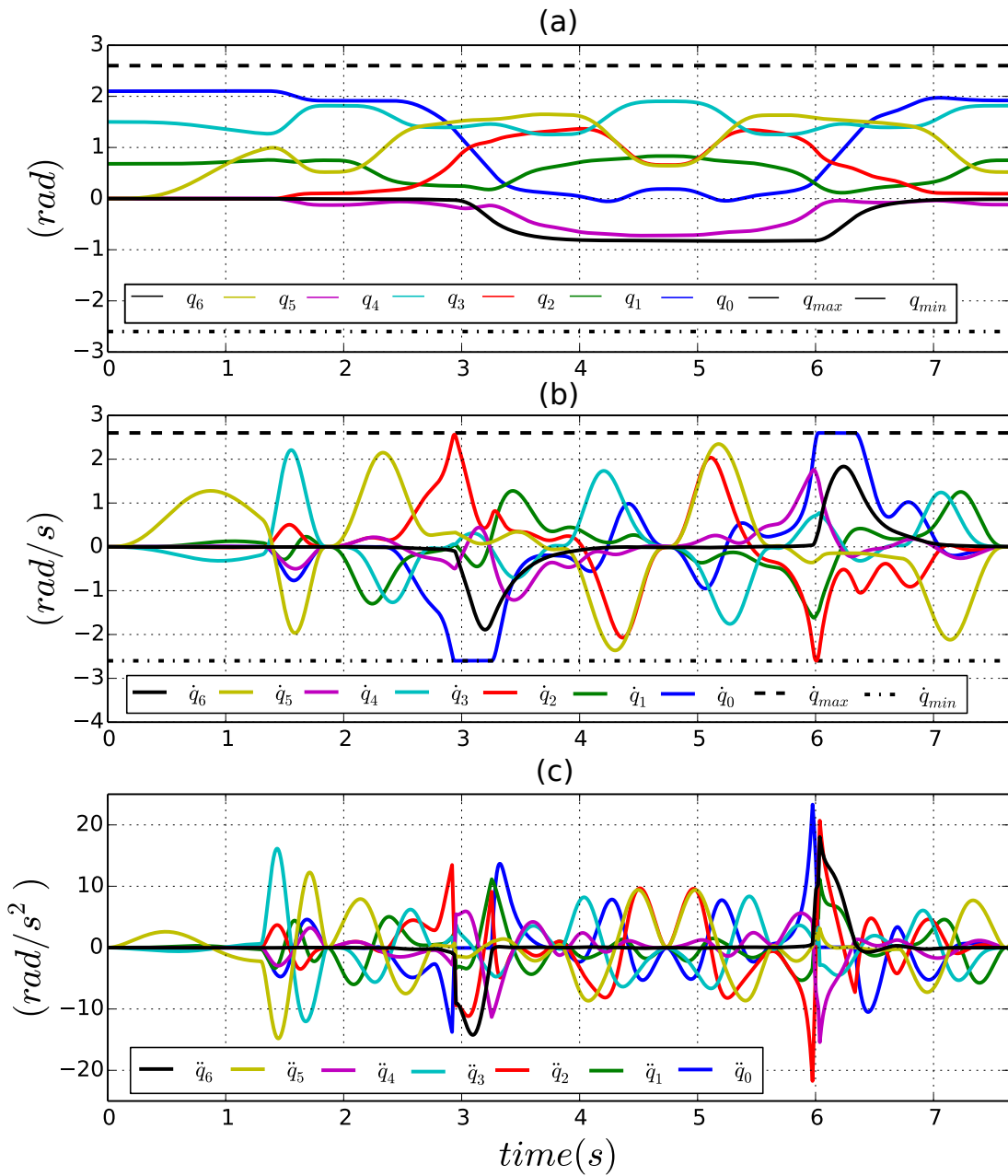


Figure 3.11: Articular position, velocity and acceleration that correspond to the pick and place movement when the energy of the robot is not constrained.

around 250 N. The *potential energy* that accumulates in the controller of the robot and the contact force derived from it stop increasing when the desired position for the end-effector stops diverging once at point ③ from the ②-③ segment (see Fig. 3.8).

Clearly, considering the big amounts of generated impact and contact forces, the controller with its current configuration is not appropriate for enabling *safe* physical interaction between the robot and its environment. Therefore, the introduced energy related constraints should be included for the upcoming configurations.

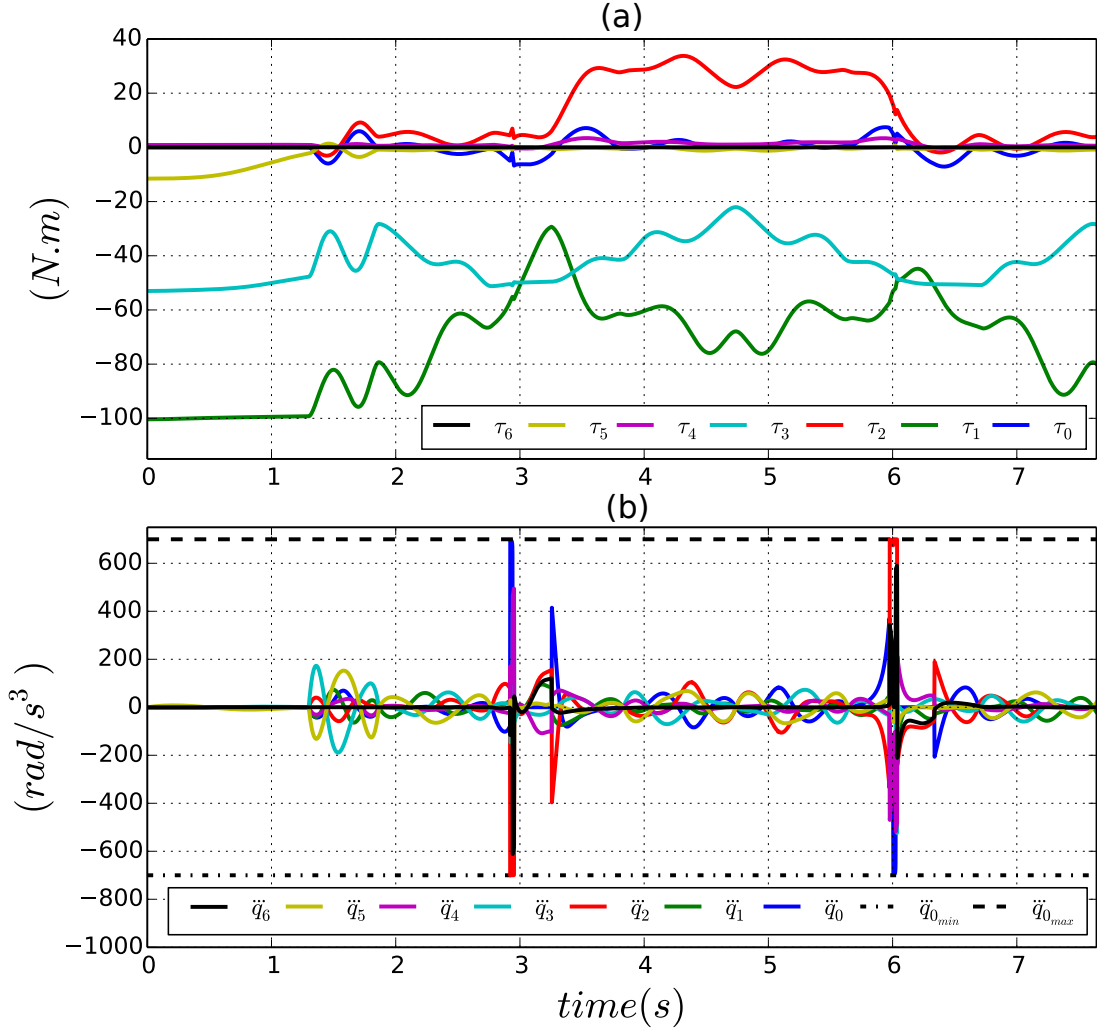


Figure 3.12: Articular torque and jerk that correspond to the pick and place movement when the energy of the robot is not constrained.

### 3.7.3 Scenario 2: nearby obstacle not intersecting with the trajectory of the robot and constraint on its kinetic energy

In this case, the first formulation of the kinetic energy related constraint (3.43) is included in the configuration of the controller to safely account for the presence of a nearby obstacle (i.e., the wall, case  $O_2$  in Fig. 3.8). This constraint limits the actuation torques and, according to (3.21), has a direct impact on the velocity of the end-effector. Depending on how the controller parameters  $d_{safe}$ ,  $E_{c_{safe}}$  and  $K$  are fixed, physical contact can be allowed or prevented. For example, fixing  $d_{safe} = 2$  and  $E_{c_{safe}} = 0$  forces the end-effector of the robot to stop at 2 m from the obstacle. In this scenario, as shown in Fig. 3.16, the obstacle does not intersect with the trajectory of the pick and place movement. In addition to the constraint on the amount of kinetic energy generated in the direction of obstacle  $O_2$ , the controller is implemented with the inequality constraints that correspond to the articular physical limitations of the actuators of the robot (3.33). Because of possible *incompatibility problems*, the constraint (3.33d) on the producible articular jerk is omitted from the configuration of the controller whenever the constraint (3.43) on the kinetic energy of the robot is activated. Finally, the equality constraints (3.34)

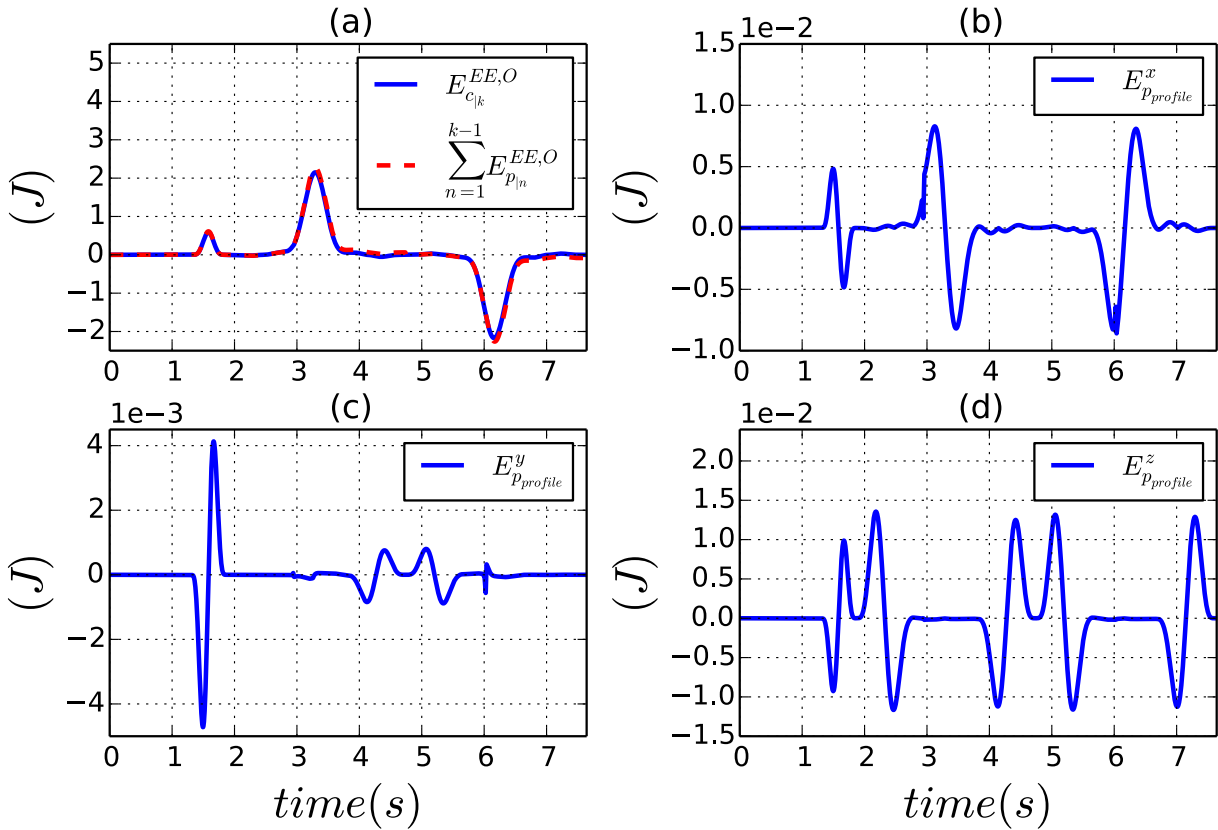


Figure 3.13: (a) Kinetic energy of the robot in the direction of the considered obstacle (case  $O_2$  in Fig. 3.8), computed using (3.54) and (3.55) during the pick and place movement. (b), (c) and (d) are the energy profiles that correspond to the pick and place task along the  $x$ ,  $y$  and  $z$  axis in Cartesian space.

that corresponds to the dynamic model of the robot are also included in the control scheme. Finally, the control parameters are fixed as:  $E_{c_{safe}} = 0.1 J$ ,  $K = 5 N$ ,  $d_{safe} = 0.25 m$  and  $d_{max} = 0.8 m$ .

With this particular configuration of its controller, the robot succeeds in achieving the pick and place task but with diminished dynamic performances compared to the test case scenario in which the kinetic energy of the system is not constrained. Indeed, as shown in Fig. 3.19, constraining the kinetic energy of the robot has a direct influence on its velocity and also on its apparent inertia. Fig. 3.18 shows how the kinetic energy related constraint is satisfied at every time-step; accordingly, a drop in the velocity of the end-effector of the robot can be observed in Fig. 3.19.d along the  $x$  axis component in Cartesian space.  $x$  here is the directing vector of the closest distance between the end-effector and obstacle  $O_2$ . At the activation of the constraint on kinetic energy, we highlight the resulting de-synchronisation between the real and desired position/velocity of the end-effector. Caused by this same constraint, discontinuities in the actuation torque (mainly on  $\tau_0$  and  $\tau_2$ ) can also be seen in Fig. 3.20.a.

Because the kinetic energy related constraint is activated only one time-step before reaching the considered limit  $E_{c_{limit}}$ , high jerk performances ( $10788 rad/s^3$  for joint 0 as shown in Fig. 3.20.b) are needed to comply with this bound in such a small window of time. At the activation of such constraint, the braking movement needed to decrease the kinetic energy of the robot is induced by its actuators. As the braking capabilities (i.e., producible deceleration and jerk) of the system

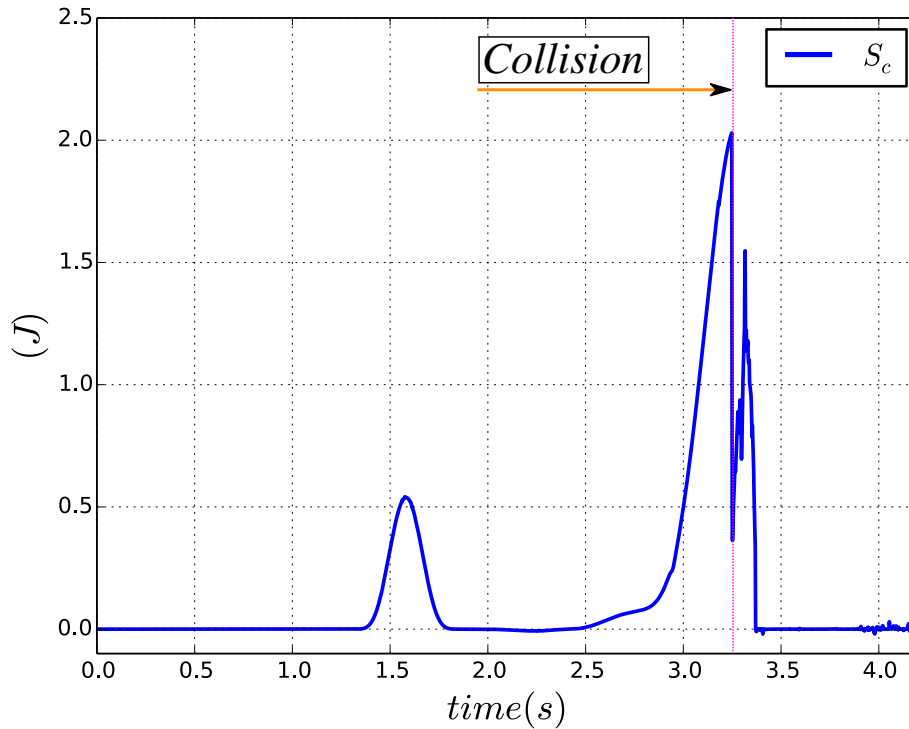


Figure 3.14: Kinetic energy of the KUKA LWR4 expressed at the level of its end-effector in the direction of the considered obstacle during, at and after collision.

are not considered in the mathematical formulation of the constraint on kinetic energy (3.43), incompatibility issues between this constraint and the hard-coded constraint on articular jerk (3.33d) may occur. In such case, coping with the maximum allowed amount of kinetic energy  $E_{c_{limit}}$  may be impossible during only one control time-step, as it is impossible to satisfy both constraints at the same time. The *viability* of the state of the robot can then be altered and the the LQP problem may become impossible to solve. On the other hand, chattering phenomena in both the articular torque and jerk, observed in Fig. 3.20, are caused by the discontinuities in the measured distance between the robot and the considered obstacle. This smallest distance between the end-effector of the robot and obstacle  $O_2$  is computed in real-time and directly fed to the controller without being filtered. This actually mimics the quality of data that can be acquired using a real-life acquisition system like the Microsoft Kinect sensor in a real-life situation. These discontinuities are reflected on the maximum allowed kinetic energy  $E_{c_{limit}}$  and thus on the control input  $\tau_{|k}^c$ .

Finally, as the kinetic energy of the robot is successfully constrained, the controller with its current configuration seems more appropriate for establishing safe physical interactions between the robot and its environment. Physical contact is allowed in the following simulation.

### 3.7.4 Scenario 3: obstacle intersecting with the trajectory of the robot and constraint on its kinetic energy

In this scenario, obstacle  $O_1$  intersects with the ②-③ segment of the pick and place movement. As the robot moves towards the wall, the first formulation of the kinetic energy related constraint (3.43) is included in the configuration of the controller and used to reduce the kinetic energy the

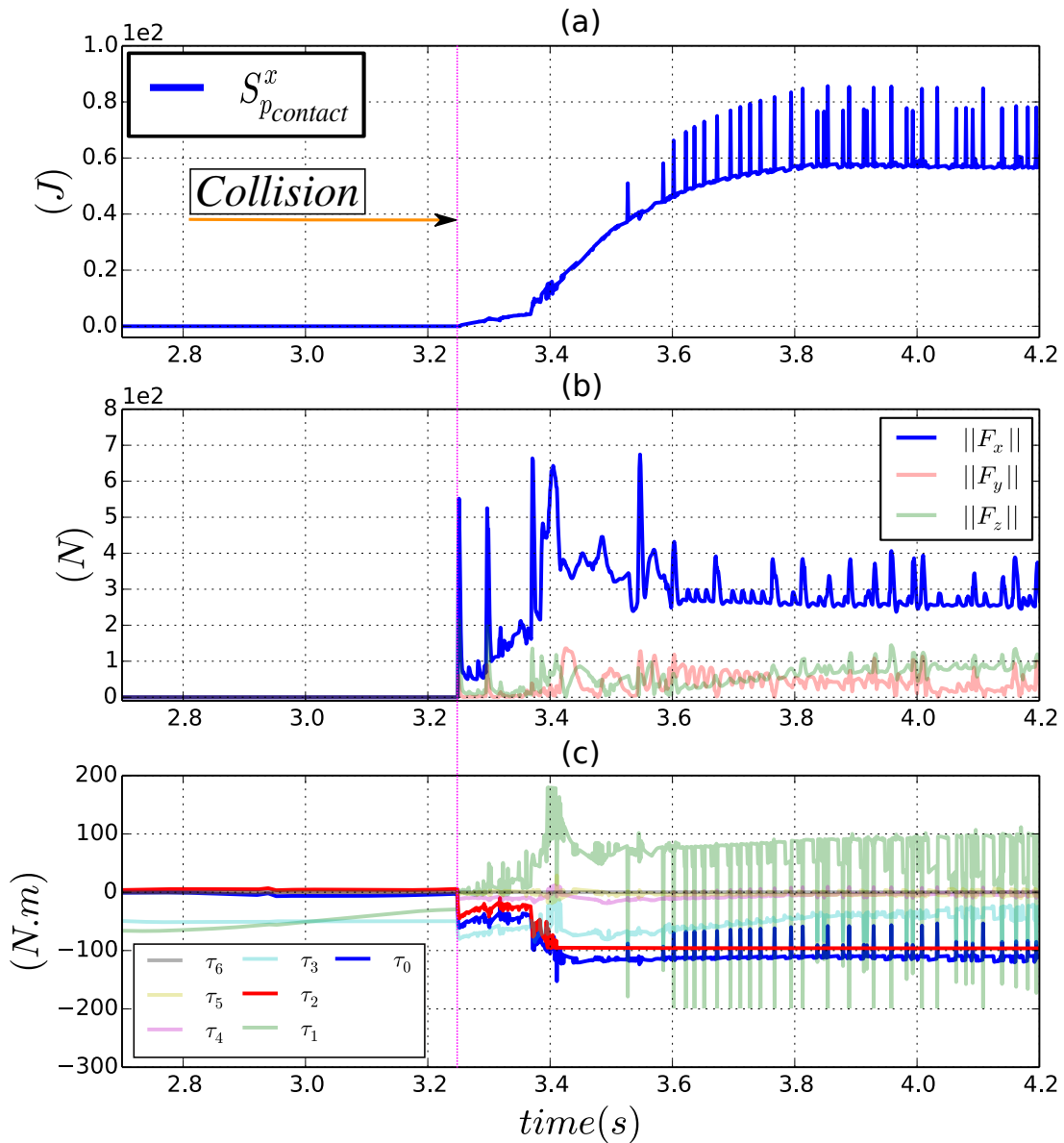


Figure 3.15: (a) *Potential energy* accumulated in the controller of the robot during physical contact along the  $x$  axis in Cartesian space. (b) Contact forces between the robot and the wall after the establishment of physical contact along the  $x$ ,  $y$  and  $z$  axis in Cartesian space. (c) Actuation torques during physical contact with the considered obstacle.

robotic manipulator displays before collision (see Fig. 3.17). Articular inequality<sup>21</sup> and equality constraints described in (3.33) and (3.34) are also considered. The parameters of the controller are fixed as:  $E_{c_{safe}} = 0.05$  J,  $K = 50$  N,  $d_{safe} = 0.1$  m and  $d_{max} = 0.3$  m.

Dissipated kinetic energy at collision is shown in Fig. 3.21. Compared to the collision case where the kinetic energy of the robot is not constrained (as in Fig. 3.14), we underline the benefit of using the kinetic energy based safety criterion introduced in (3.3). Indeed, at collision, the amount of dissipated kinetic energy (0.035 J) when this latter is initially constrained is far less

<sup>21</sup>Similarly to the previous simulation, the constraint on articular jerk is removed from the configuration of the controller whenever the constraint on kinetic energy (3.43) is activated.

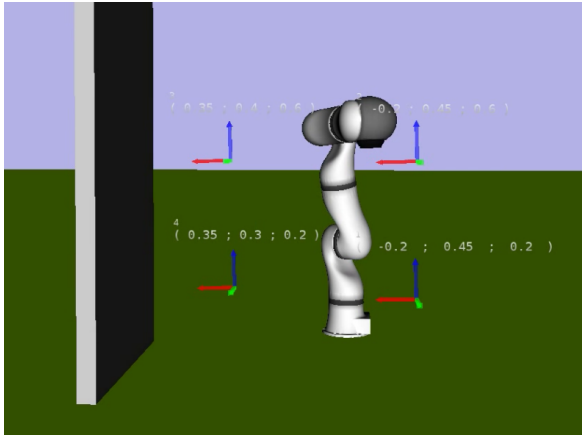


Figure 3.16: Screen-shot of the KUKA LWR4 robot within the XDE simulated world with the nearby wall not intersecting its trajectory.

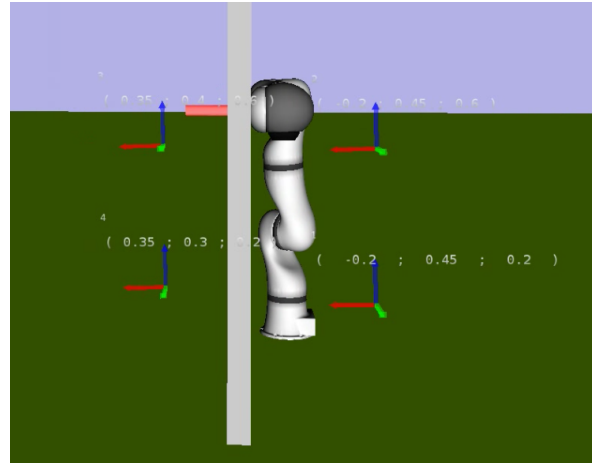


Figure 3.17: Screen-shot of the KUKA LWR4 robot within the XDE simulator during a collision with the wall intersecting its trajectory.

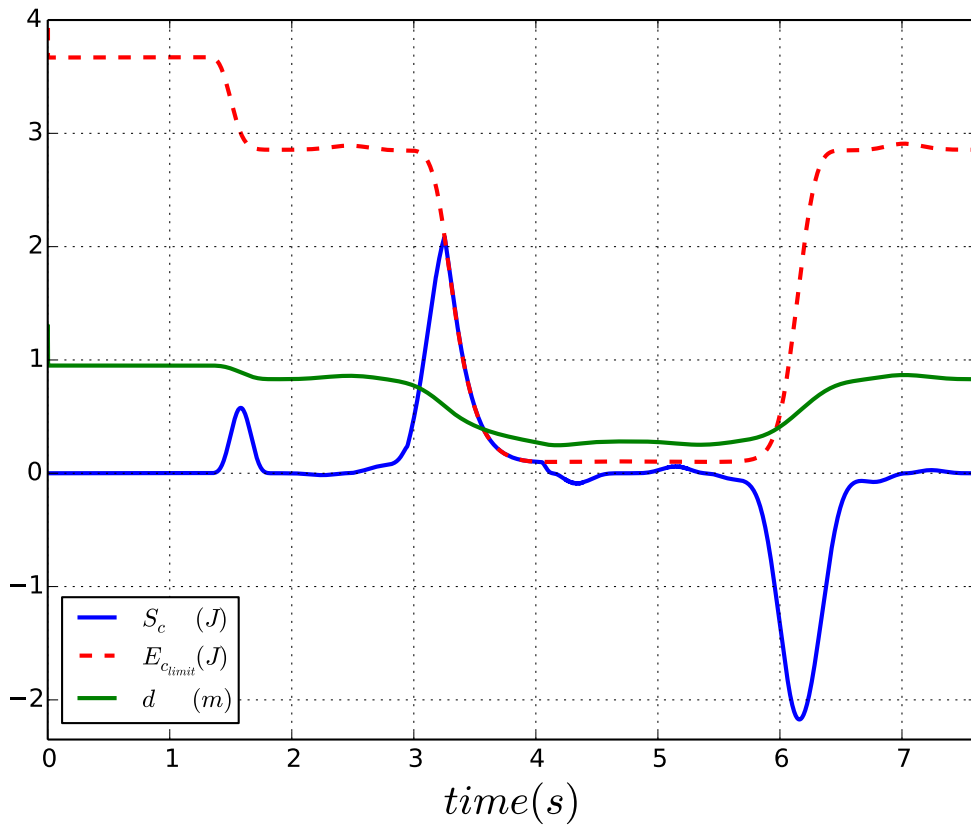


Figure 3.18: Constrained kinetic energy of the robot expressed at its end-effector in the direction of a nearby considered obstacle (case  $O_2$  in Fig. 3.8).

than its amount if the introduced constraint on kinetic energy is not included in the configuration of the controller ( $1.66 J$ ). Accordingly, the resulting impact force shown in Fig. 3.22.a is largely reduced:  $86 N$  compared to the collision force when the kinetic energy of the robot is not pre-constrained ( $551 N$  as shown in Fig. 3.15.b). Using the kinetic energy related constraint (3.43),

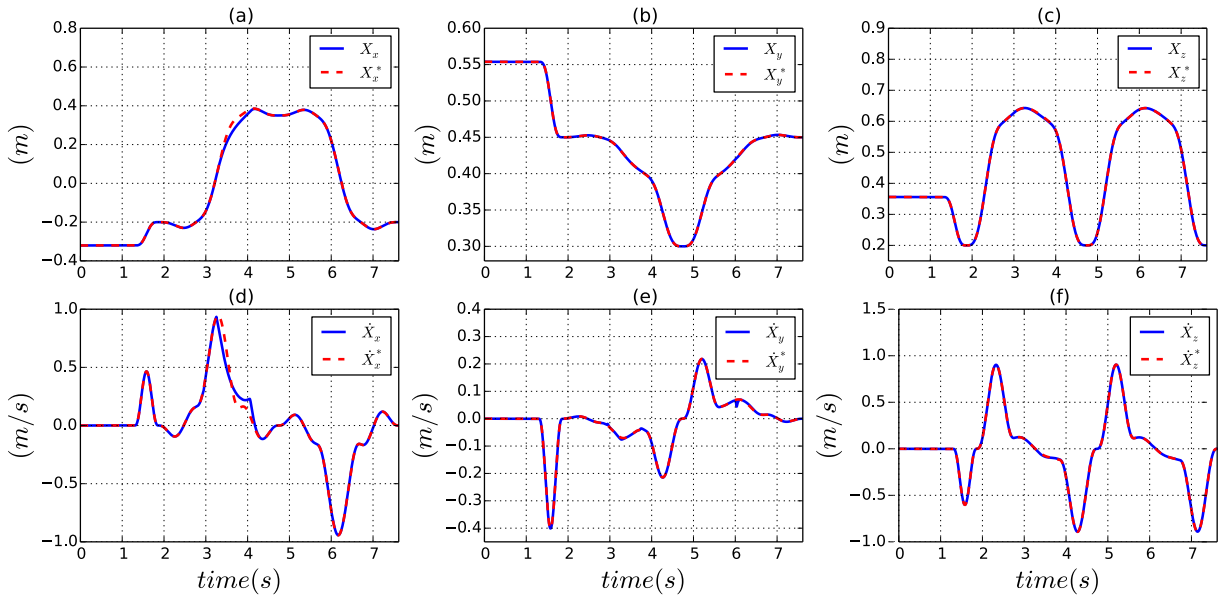


Figure 3.19: (a), (b) and (c) depict the real and desired position for the end-effector of the robot in Cartesian space as its kinetic energy is constrained. (d), (e) and (f) are the real and desired velocity for the end-effector as the system is submitted to the same constraint (case  $O_2$  in Fig. 3.8).

the ability to saturate the kinetic energy of the robot at a safe limit before collision enables a safer physical-contact establishment/impact between the robot and its environment.

On the other hand, as in scenario 1, after the establishment of physical-contact, *potential energy* starts increasing in the controller of the robot (see Fig. 3.22.a). Consequently, as illustrated in Fig. 3.22.b, the resulting contact force also increases to reach an average value of 100 N, which may be potentially harmful for any human-operator within the workspace of the robot. Actuation torques are shown in Fig. 3.22.

In the current scenario, even after the establishment of physical contact, the constraint on kinetic energy is never removed from the configuration of the controller. However, as it depends on the real-time distance  $d$  between the end-effector of the robot and the considered obstacle  $O_1$ , this constraint may be released as case  $d$  is equal to zero.

To summarize, in this scenario, the configuration of the controller that accounts for the constraint on the kinetic energy of the robot is considered more suitable for physical contact establishment/collision, compared to the case where the kinetic energy related constraint is not included in the control scheme. However, to ensure safety even during the physical contact phase, the *potential energy* that accumulates in the controller of the robot from which the hazardous contact force is derived must also be constrained. This constraint is considered in the scenario of the following simulation.

### 3.7.5 Scenario 4: obstacle intersecting the trajectory of the robot and constraints on its kinetic and *potential* energies

Similarly to the previous scenario, the wall intersects with the ②-③ segment of the pick and place movement. As the robot moves, the second formulation of the constraint on kinetic energy (3.45) is included in the configuration of the controller and used to limit the amount of kinetic energy the robotic manipulator deploys in the direction of obstacle  $O_1$ . This constraint also depends

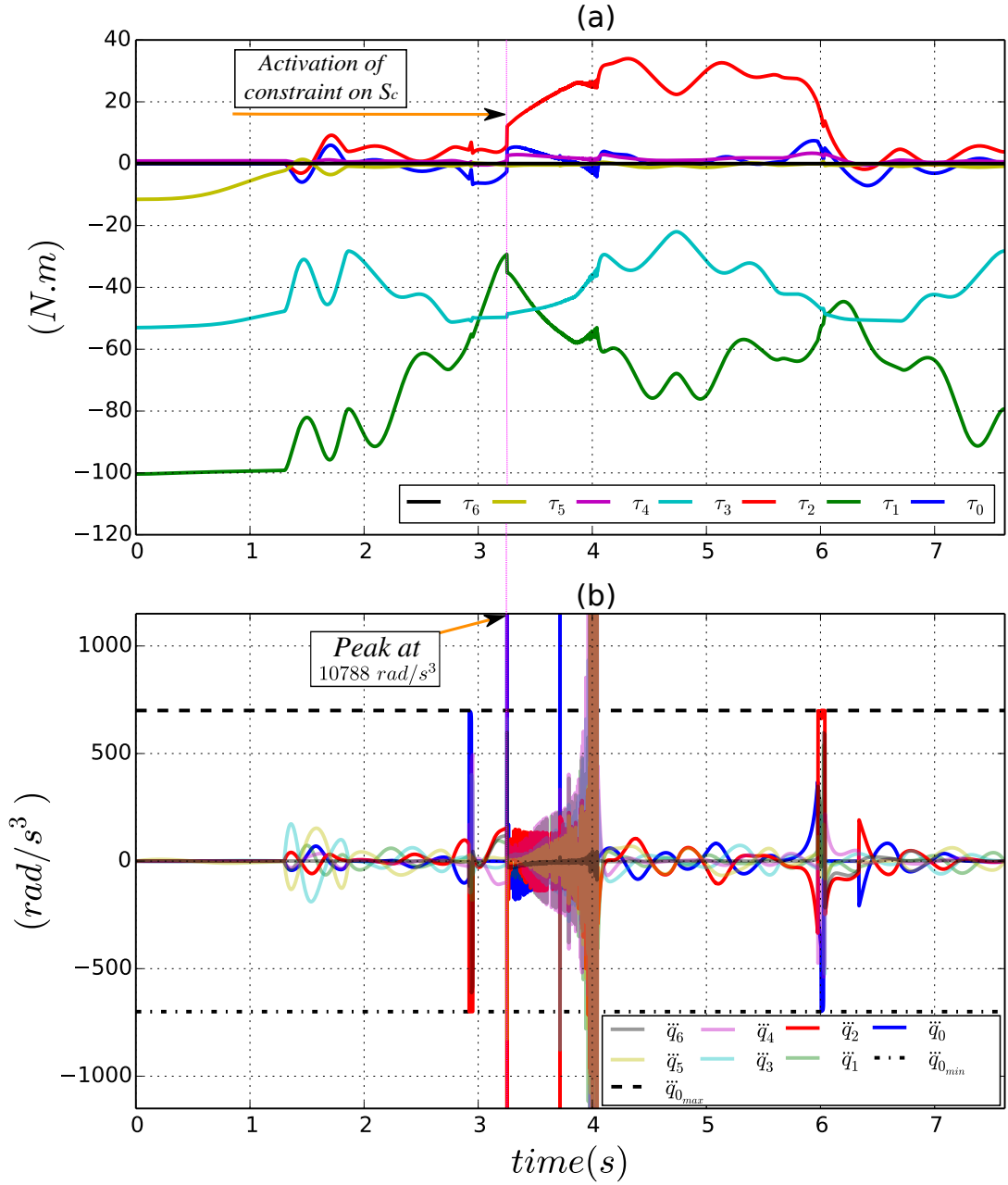


Figure 3.20: Generated articular torque and jerk as the constraint on the kinetic energy of the robot is activated (case  $O_2$  in Fig. 3.8).

on the real-time closest distance between the wall and the end-effector of the robot. After the establishment of physical-contact, without removing the constraint on kinetic energy(3.45), the constraint (3.52a) is added to the controller and used to saturate the amount of *potential energy* that accumulates in the controller of the robot. Articular inequality<sup>22</sup> and equality constraints described in (3.33) and (3.34) are also considered in the control scheme. The parameters of the controller are fixed as:  $E_{c_{safe}} = 0.05 J$ ,  $K = 50 N$ ,  $d_{safe} = 0.1 m$ ,  $d_{max} = 0.3 m$  and  $E_{p_{safe}}^x = 0.1 J$ . During the physical-contact phase, contact forces applied by the robot to the

<sup>22</sup>Similarly to the previous case, the constraint on articular jerk is removed from the controller whenever an energy related constraint (kinetic (3.45) or potential (3.48a)) is activated.



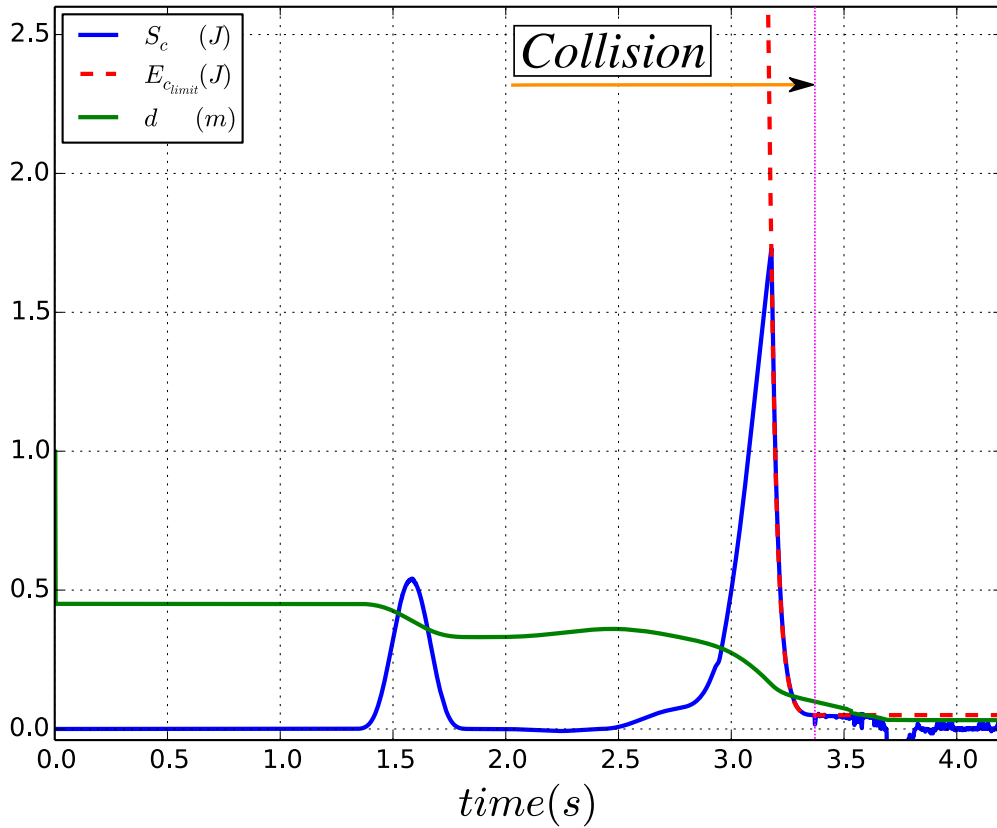


Figure 3.21: Constrained kinetic energy of the robot expressed at the level of its end-effector in the direction of the collided obstacle (case  $O_1$  in Fig. 3.8).

wall are *mainly* along the  $\boldsymbol{x}$  axis. Consequently, only the *potential energy* along this vector is constrained.

Dissipated kinetic energy at collision is shown in Fig. 3.24. As it is pre-constrained, only  $0.035 J$  of kinetic energy are dissipated at collision, resulting into an impact force of  $85 N$  (see Fig. 3.25.b). As shown in Fig. 3.24, the second formulation of the constraint on kinetic energy (3.45) is satisfied at every time-step, exactly the same as when the first formulation (3.43) is used (see subsection 3.7.4). On the other hand, Fig. 3.25.a shows how the constraint on the *potential energy* that accumulates in the controller of the robot during physical-contact is also satisfied at every time-step. Because this *potential energy* is saturated at a constant value  $0.1 J$ , the contact force derived from it decreases over time to reach  $0 N$  (see Fig. 3.25.b) as the Cartesian tracking error between the real and desired positions for the end-effector of the robot increases. This finally causes the release of physical contact between the robot and the wall intersecting with its trajectory. When using the current configuration of the controller, if any physical-contact is established between the robotic manipulator and a human-operator in its workspace, the robot can easily and safely be moved or pushed away. *The decrease* in the resulting contact force while the error between the desired and real positions for the end-effector of the robot *increases* is actually exactly the inverse of what happens when the *potential energy* related constraint is not included in the controller of the robot. In such case, tracking errors increase, *potential energy* accumulates in the controller and accordingly hazardous contact forces (as shown in scenario 1). Fig. 3.25.c shows the actuation torques of joint 0 and joint 2 as a peak is created when the braking phase implicitly induced to cope with the kinetic energy related constraint is engaged.

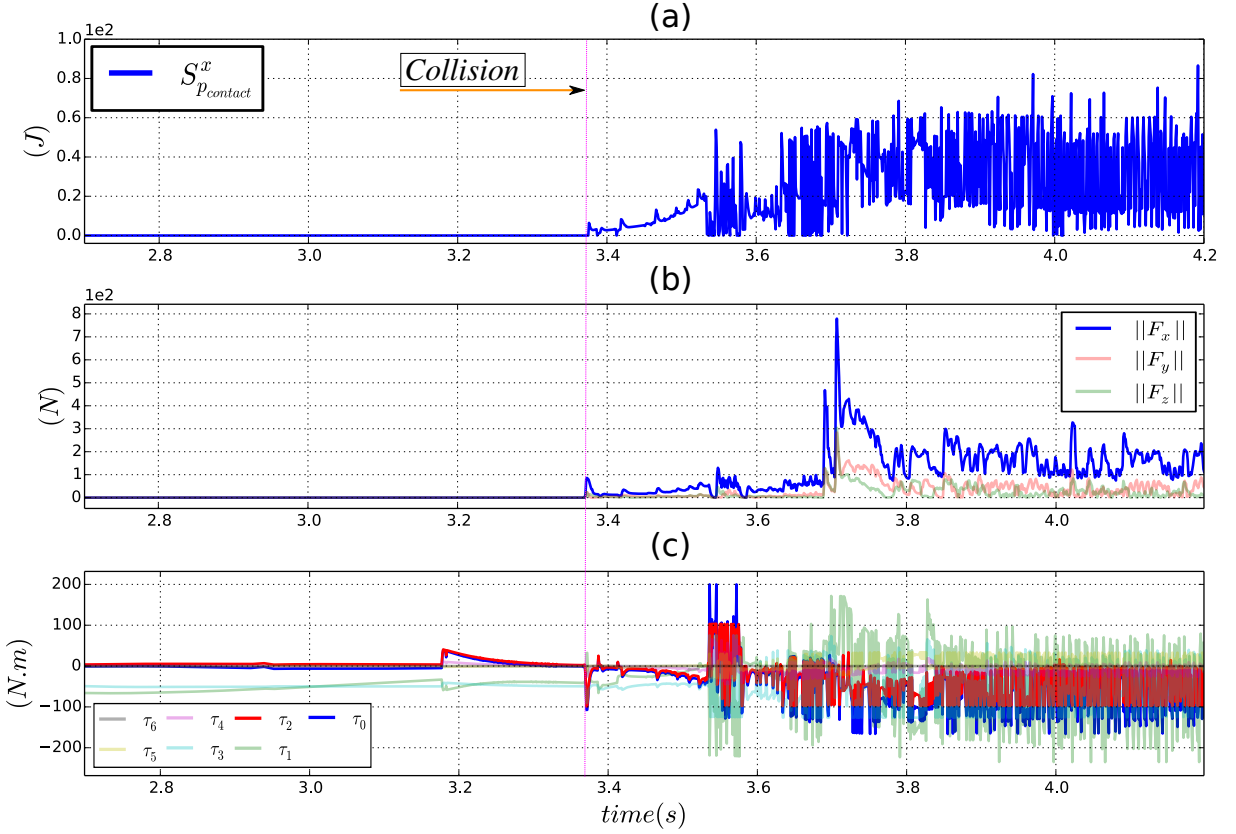


Figure 3.22: (a) *Potential energy* stored in the controller of the robot during physical contact along the  $x$  axis in Cartesian space. (b) Corresponding contact forces along the  $x$ ,  $y$  and  $z$  axis in Cartesian space. (c) Actuation torques during physical contact with the considered obstacle (case  $O_1$  in Fig. 3.8).

### 3.7.6 Scenario 5: obstacle intersecting with the trajectory of the robot and constraint on its task energy profile

In this scenario, obstacle  $O_1$  also intersects with the ②-③ segment of the pick and place movement. During the movement of the robot, its energy profile (described in Section 3.5) related to the accomplished pick and place task is constrained using (3.52). The kinetic energy of the robot is not constrained and the articular inequality and equality constraints respectively described in (3.33) and (3.34) are also included in the configuration of the controller. As shown in Fig. 3.13, based on the measured energy profiles along the  $x$ ,  $y$  and  $z$  axis, the parameters of the controller are fixed as:  $E_{p_{profile}}^x + \epsilon_{E_p}^x = 0.01 J$ ,  $E_{p_{profile}}^y + \epsilon_{E_p}^y = 0.0045 J$  and  $E_{p_{profile}}^z + \epsilon_{E_p}^z = 0.015 J$ . With  $\epsilon_{E_p}^x$ ,  $\epsilon_{E_p}^y$  and  $\epsilon_{E_p}^z$  used to compensate the repeatability related variations on the controller's *potential energy* needed to accomplish the pick and place task.

As can be seen in Fig. 3.26, during the pre-collision phase, the tracking performances of the controller are not altered by the *task energy profile* related constraint. Indeed, during the *free* movement of the robot, this constraint is never activated. The desired position, orientation and velocity for the end-effector are properly tracked, exactly the same as if this constraint is not included in the control scheme.

At collision, the impact of the robot with the wall is not detected and the desired position (also the velocity) for the end-effector starts diverging from its real value. Consequently, the *potential*

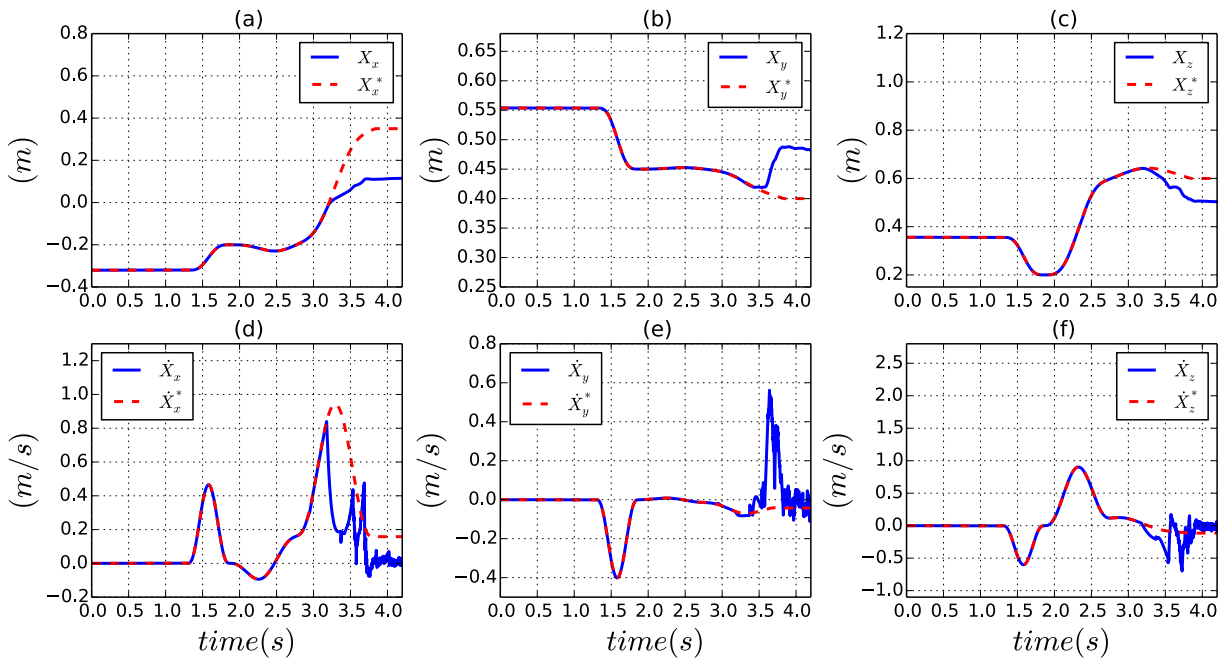


Figure 3.23: (a), (b) and (c) depict the real and desired position for the end-effector of the robot in Cartesian space as its kinetic energy is constrained. (d), (e) and (f) are the real and desired velocity for the end-effector as the robot is submitted to the same constraint (case  $O_1$  in Fig. 3.8).

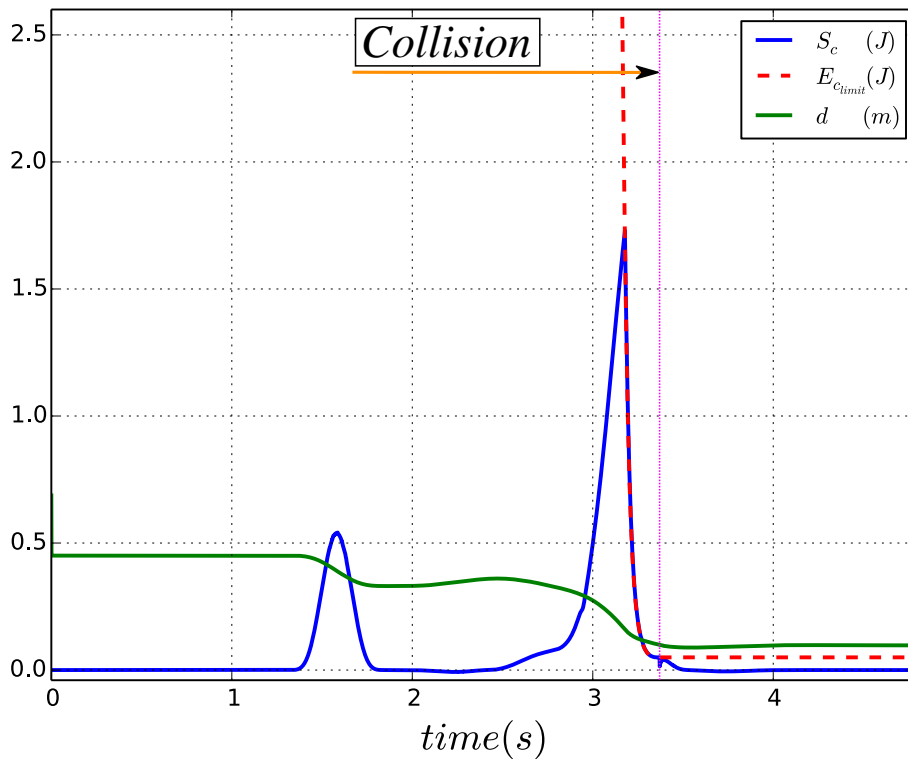


Figure 3.24: Constrained kinetic energy of the robot expressed at the level of its end-effector in the direction of the collided obstacle (case  $O_1$  in Fig. 3.8).

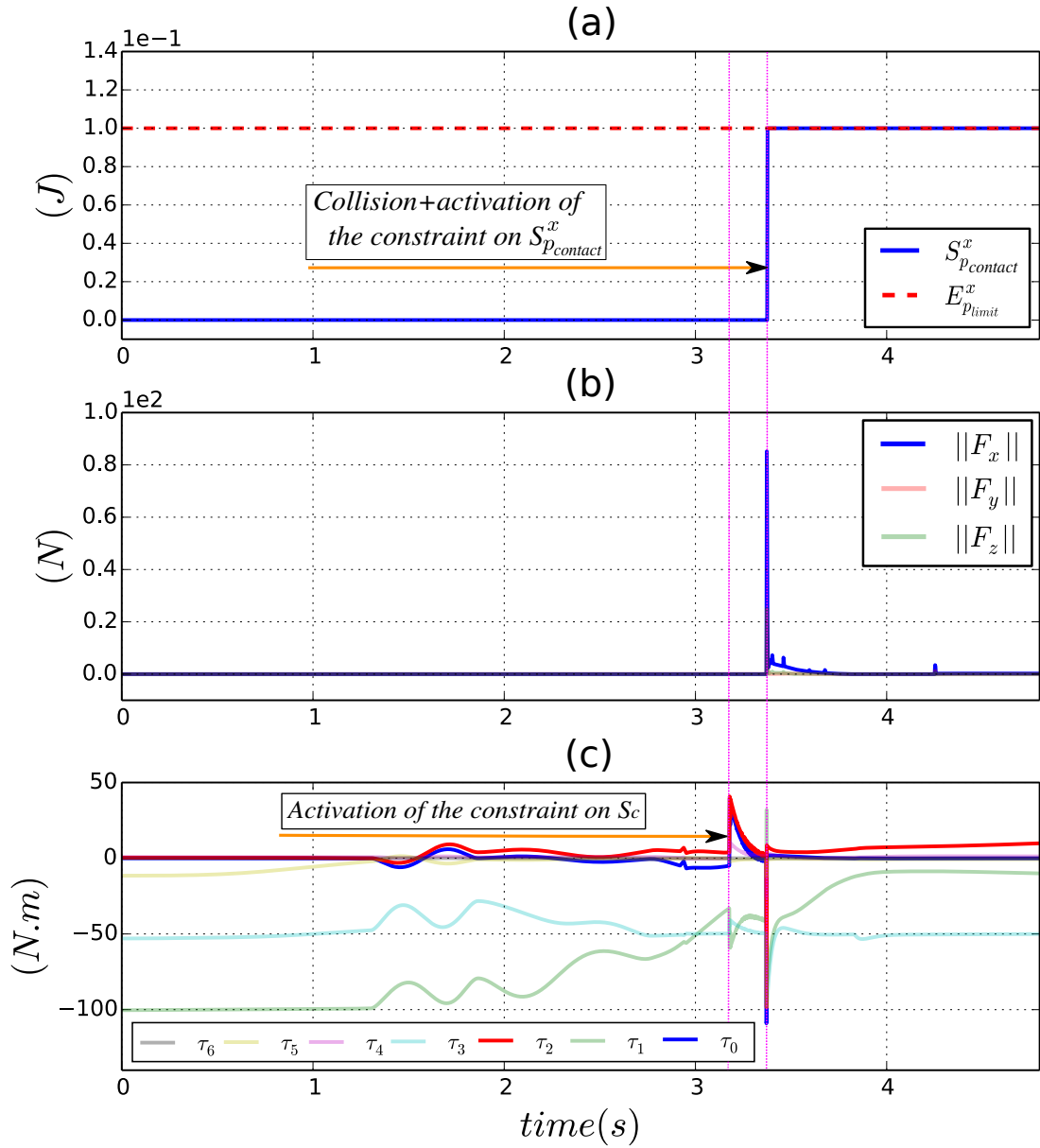


Figure 3.25: (a) Constrained *potential energy* stored in the controller of the robot during physical contact along the  $x$  axis in Cartesian space. (b) Corresponding contact forces along the  $x$ ,  $y$  and  $z$  axis in Cartesian space. (c) Actuation torques during the physical contact with the considered obstacle (case  $O_1$  in Fig. 3.8).

*energy* stored in the controller starts to increase and triggers the activation of the constraint on the *task energy profile*. During physical contact, this constraint behaves exactly the same as (3.50). According to Fig. 3.27, the *potential energy* related constraint is successfully satisfied along the  $x$ ,  $y$  and  $z$  axis. The peak of *potential energy* along the  $x$  axis at collision is however caused by the induced impact force.

Fig. 3.28.a shows how 1.66  $J$  of kinetic energy are dissipated at impact. Fig. 3.28.b depicts the resulting collision and contact force: 551  $N$  are induced at the first impact. The other collision peaks are caused by collisions between other parts of the robot and the wall. During physical contact, the constraint on the controller's accumulated *potential energy* when expressed at the

level of the end-effector, does not prevent the movement of other *free* parts of the body of the robot.

Thanks to the constraint on the *task energy profile* that has been included in the configuration of the controller, the resulting contact force is also reduced and it decreases over time as the desired and real positions for the end-effector diverge. The robot at this stage is compliant and can easily and safely be moved by a human-operator. The actuation torque generated during this simulation is shown in Fig. 3.28.

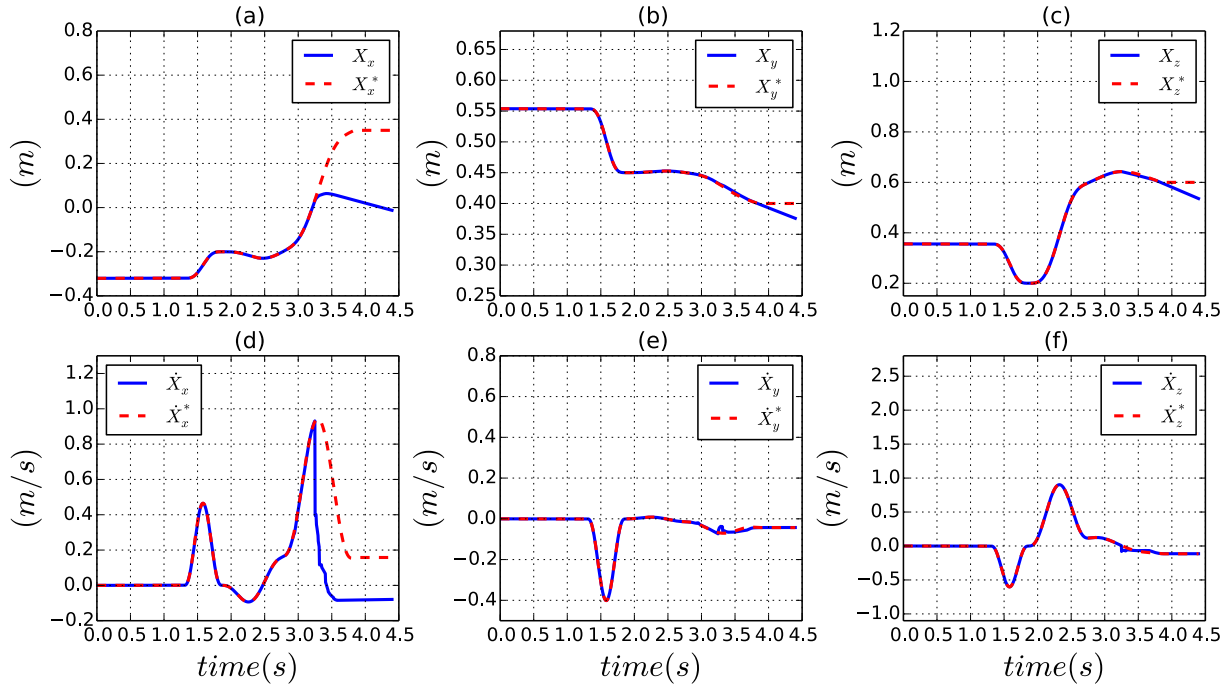


Figure 3.26: (a), (b) and (c) depict the real and desired position for the end-effector of the robot in Cartesian space as its task energy profile is constrained. (d), (e) and (f) are the real and desired velocity for the end-effector as the robot is submitted to the same constraint (case  $O_1$  in Fig. 3.8).

### 3.8 Conclusion

Energy-based safety indicators proposed and validated in this chapter hold a great potential for human-robot collaboration tasks. Indeed, energy is a universal component that can describe all the physical phenomena occurring during an interaction between the robot and its environment. Velocity, inertia, impact and also contact forces can all be expressed and modulated with this same property. Using the presented control framework and the introduced energy-based criteria, it has been proven possible to control the energy of the robot. During its movement and at every time-step, the kinetic energy of the robot can be tracked then saturated to meet the imposed safety criteria needed for a safe establishment of physical contact. Dissipated kinetic energy at impact is reduced and the induced collision force is harmless.

After the establishment of physical contact, contact forces can also be saturated by constraining the *potential energy* that accumulates in the controller of the robot. In case of a human-robot interaction, the robot can easily and safely be pushed and moved as needed. In addition to the kinetic and *potential energy* related constraints, the constraint on kinetic energy is reformulated

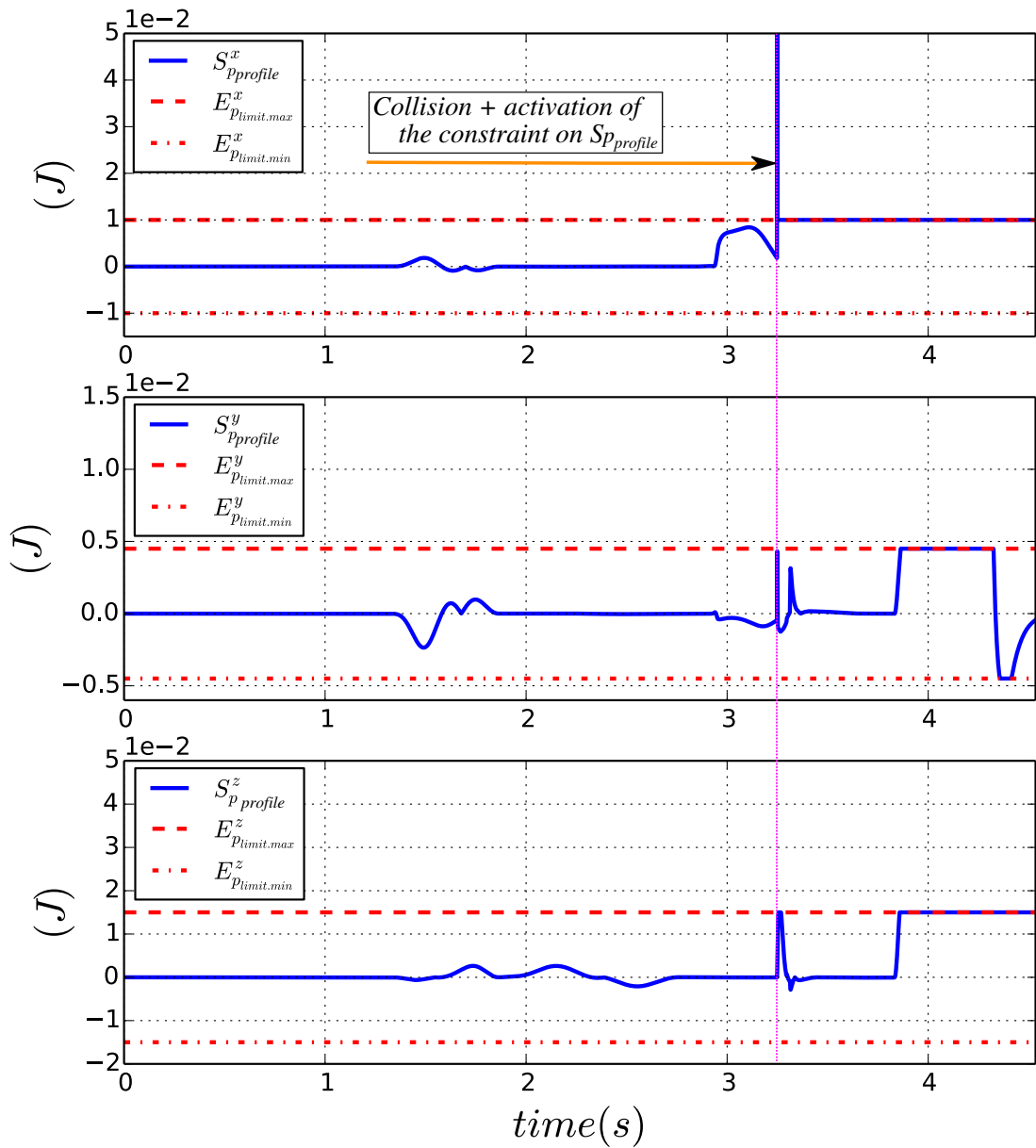


Figure 3.27: Constrained energy profile for the pick and place task along the  $x$ ,  $y$  and  $z$  axis in Cartesian space.

as a constraint on the *potential energy* instantaneously held in the controller of the robot. Consequently, the amount of *potential energy injected* in the controller of the robot at every time-step can be used as a safety indicator for both the pre and post collision phases.

Moreover, the concept of *task energy profile* has been introduced and used to synthesise a low-level security layer to deal with non-deliberate or undetected collisions between the robot and its environment. The proposed constraint on the *task energy profile* prevents the robot from accumulating any additional amount of energy than what it needs to perform its main task. This has the effect of preventing any harmful contact forces as the controller of the robot is restrained from “loading” any additional amount of *potential energy* in case a physical-contact with the environment is established.

Constraints on kinetic and potential energies are expressed in Cartesian space and are non-

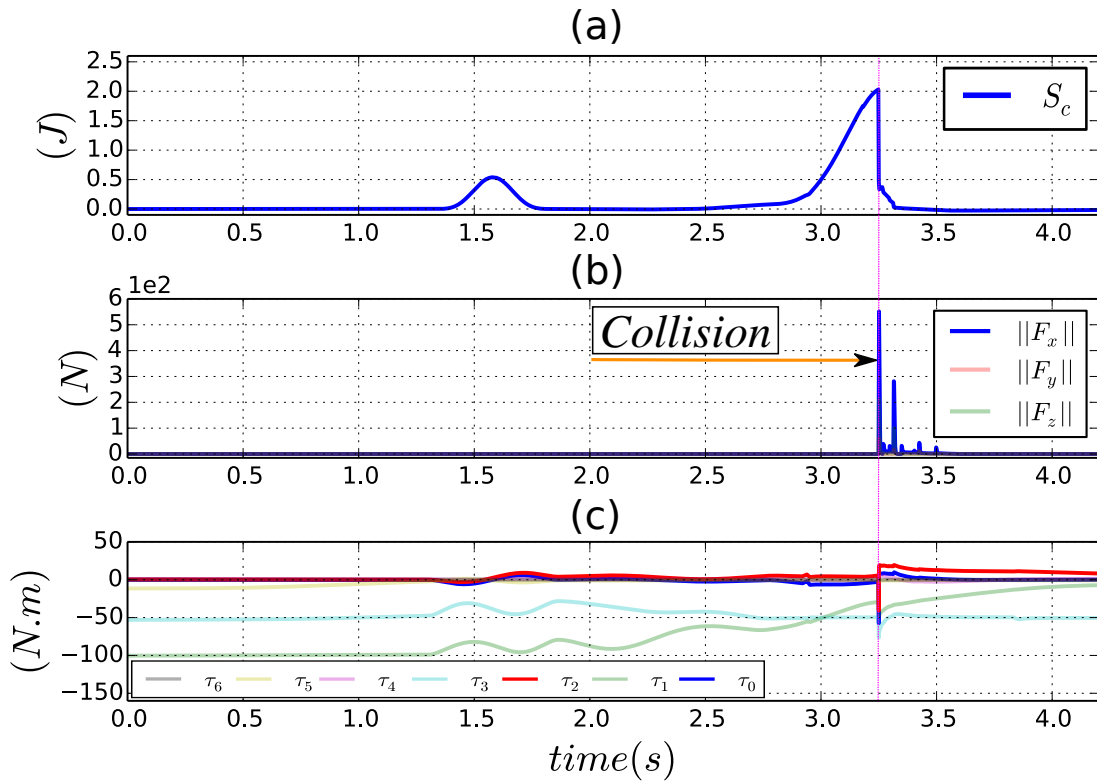


Figure 3.28: (a) The kinetic energy of the robot expressed at the level of its end-effector as it physically interacts with the nearby obstacle (case  $O_1$  in Fig. 3.8). (b) Corresponding contact forces along the  $x$ ,  $y$  and  $z$  axis in Cartesian space. (c) Actuation torques during physical contact.

linearly related to the instantaneous state  $S_{|k} = \{\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}, \ddot{\mathbf{q}}_{|k}\}$  of the robot in articular space. Such constraints are of type 6 (see table 2.1 in Chapter 2) and due to how these are currently formulated, the energy related constraints are activated only one time-step before reaching their respective limits, which may render the control problem impossible to solve. Indeed, for example, the robotic manipulator may not have sufficient *braking* capabilities to react and cope with an imposed kinetic energy limit  $E_{c_{limit}}$  within only one control time-step. Its braking capabilities, namely: producible articular deceleration  $[\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M]$  and jerk  $[\dddot{\mathbf{q}}_m, \dddot{\mathbf{q}}_M]$  should therefore imperatively be part of the formulation of these constraints. With such new formulations, the energy related constraints can be activated at the right moment in order to guarantee the preservation of the *viability* of the state of the robot. Otherwise, *constraints incompatibility* issues as tackled Chapter 2 may appear and it may be impossible to implement these constraints on a real robot which actuators have limited dynamic capabilities. In the following chapter, the introduced energy related constraints are tested on a real KUKA LWR4 robotic manipulator as it physically interacts with a human-operator.

## Chapter 4

# Safe human-robot interaction: experimental results

In this chapter, the controller and the energy related constraints introduced in the previous chapter are implemented and tested on a real KUKA LWR4 robot for physical and non-physical interactions with a nearby human-operator. As the person approaches the robot, the distance between the two is computed in real-time with an algorithm that uses point-clouds acquired from the scene by the mean of a set of depth sensors (Kinects). This distance is used to generate the instantaneous maximum amount of kinetic energy the robot is allowed to exhibit. At collision, thanks to the kinetic energy related constraint, dissipated energy and the resulting impact force are reduced. When physical contact between the KUKA LWR4 and the human is established, a second constraint is used to limit the amount of *potential energy* that accumulates in the controller of the robot. Consequently, contact forces are saturated and the robot can safely be pushed and moved if needed. The overall control framework allows a human-operator to enter the workspace of the robotic manipulator, physically interact with it then safely leave its workspace.

### 4.1 Introduction

In the following sections, the experimental setup for the KUKA LWR4 robotic arm and the vision system used to detect the human-operator approaching and entering its workspace are described. A test case scenario is presented and the various reactions of the robot that can be induced using the safe controller presented in Chapter 3 along with the energy related constraints are discussed.

### 4.2 Experimental setup

The distance between the robot and the human-operator is computed using data from a set of 3 Kinect-like depth sensors strategically placed around the workspace of the robot to avoid occlusions (see Fig. 4.1). RGB and depth images from each sensor are calibrated and the pose of each device in the base frame of the robot is computed. The 3D model of the robot and the background are removed from the depth images [KaewTraKulPong 2002] then the pointclouds from the 3 depth sensors are down sampled and combined together. Finally the cluster of the human-operator is extracted from the resulting pointcloud [Rusu 2011] and the minimum distance between the robot end-effector and the human-operator is computed and published



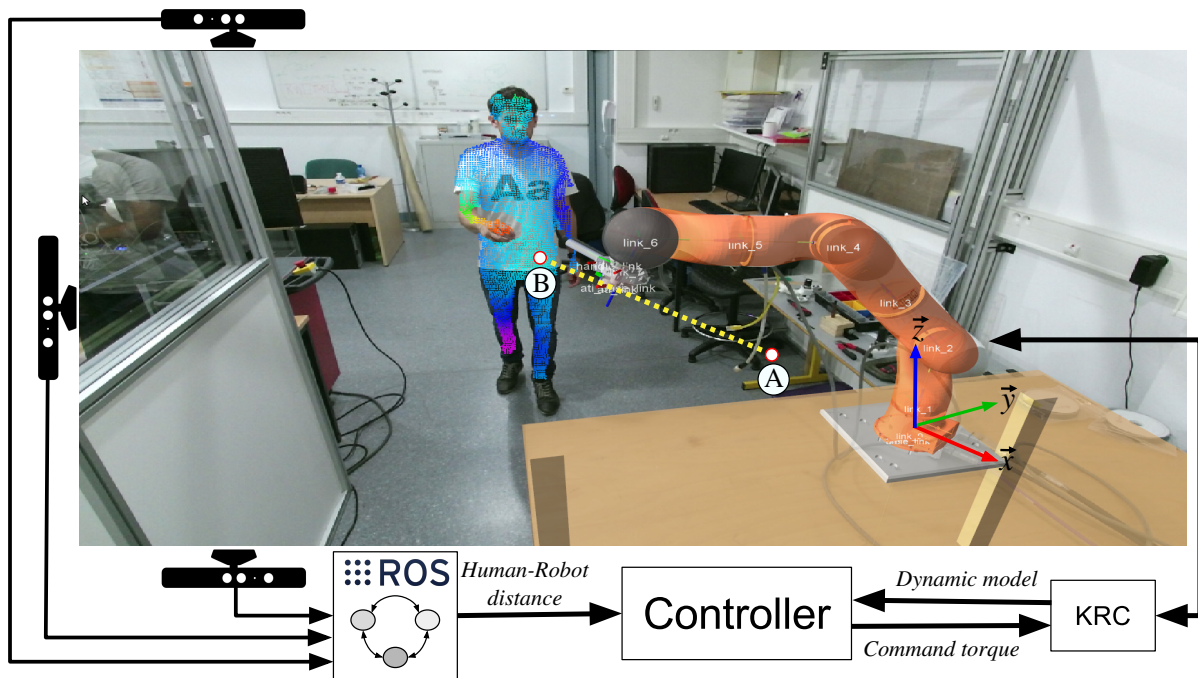


Figure 4.1: View of a user sharing its workspace with a KUKA LWR4 robotic manipulator; with a schematic representation of the experimental setup used to measure the real-time distance between the robot and the human-operator and to control the robot.

via a ROS topic. The controller described in Section 3.6.3 of Chapter 3 is implemented as a C++ OROCOS [Soetens 2017] component inside a generic software architecture developed at ISIR for robot manipulators [Hoarau 2017]. The remote control PC runs a Xenomai kernel [Chantepredrix 2017] with RTnet [Marx 2017] to ensure minimum jitter in the real-time Ethernet communication. Finally, communication with the KUKA LWR4 Robot's Controller (KRC) is performed via the Fast Research Interface (FRI) [Schreiber 2010].

### 4.3 Test case scenario

As a main activity, the robot performs a repetitive movement where it tracks a desired position on a straight line between two points  $\textcircled{A}$  and  $\textcircled{B}$ <sup>1</sup> in Cartesian space (see Fig. 4.1). The controller described in (3.53) is fed with a desired Cartesian position, velocity and a feed-forward acceleration term, and is implemented with the articular constraints corresponding to the physical limitations of the robot (2.11)<sup>2</sup> along with the equality constraints described in (3.34) that correspond to its dynamic model. The LQP problem is solved for each period of time  $\delta t = 15 \text{ ms}$  to compute the needed control torque  $\tau_{|k}^c$ .  $\delta t$  is at the same time the *sample-time* imposed to the controller and the *time-window* used for the discrete formulation of the energy related constraints the robot must cope with. Fixed by trial and error,  $\delta t = 15 \text{ ms}$  is the least amount of time needed by the robot to be able to cope with the introduced (*naively expressed*) energy related constraints: (3.43) and (3.47). In case  $\delta t$  is fixed at  $1 \text{ ms}$ , indeed, better tracking

<sup>1</sup>Only when the end-effector of the robot reaches one point (e.g.,  $\textcircled{A}$ ) that the trajectory to the other point  $\textcircled{B}$  is generated.

<sup>2</sup>As the real jerk capabilities of the actuators of the KUKA LWR4 are not known, the constraint on articular jerk (3.33d) is not considered in the version of the controller used for experiment.

performances for the trajectory tracking task can be realized. However, with limited dynamic capabilities (i.e., producible deceleration and jerk in operational space), the system is incapable to react in time and cope (for example) with a constraint on its kinetic energy within just a 1 *ms* time-window. When using such a small control sample-time, the energy related constraints must be reformulated to include the reaction capabilities of the robot. Exactly as have been realized for the constraints on articular position and velocity in Chapter 2 (see Section 2.4). Finally, the LQP is solved in real-time using Gurobi, a commercial optimization software [Gurobi Optimization Inc. 2015]. The maximum tracking error for the position of the end-effector of the

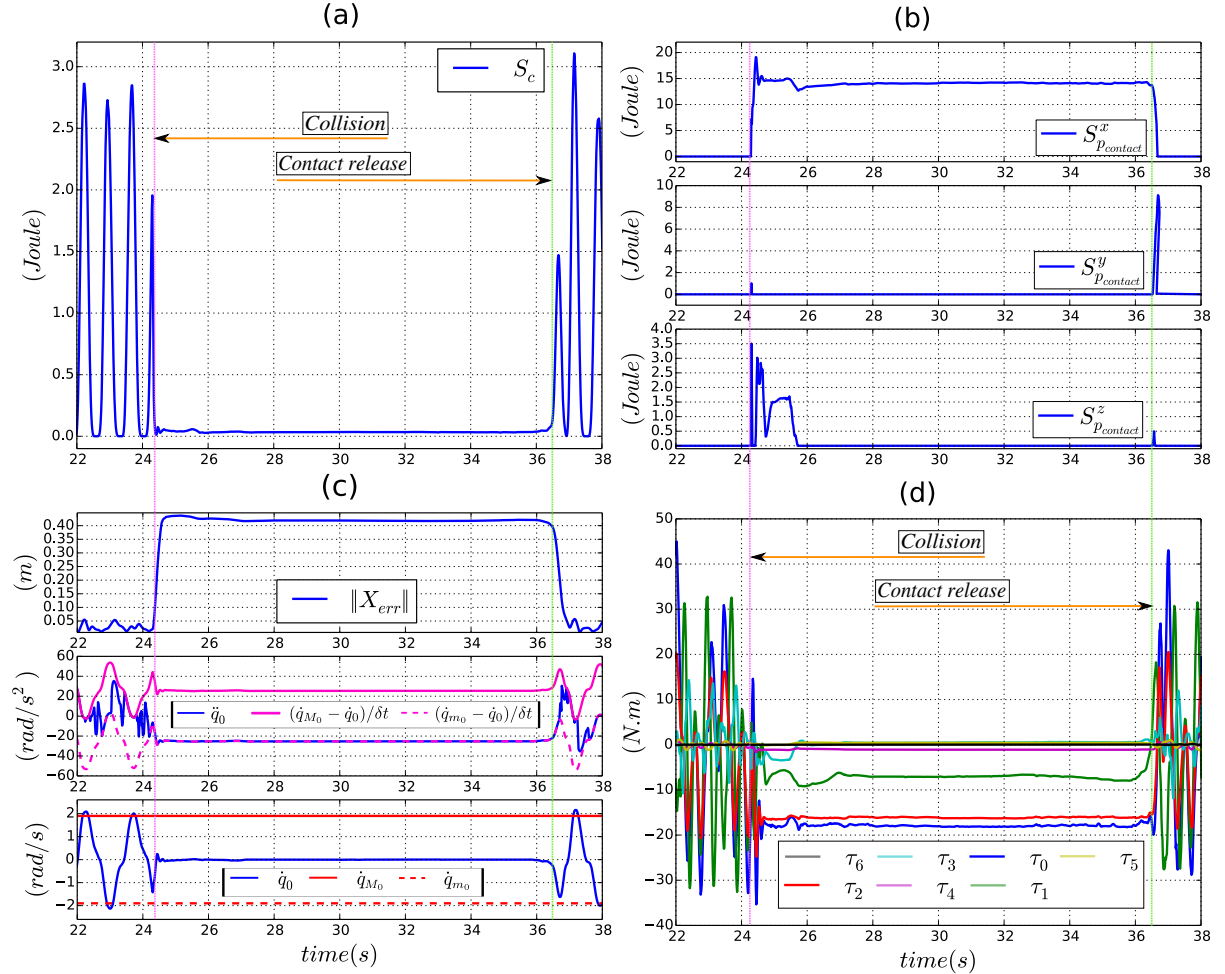


Figure 4.2: (a) Kinetic energy of the robot expressed at its end-effector in the direction of the human-operator. (b) *Potential energy* stored in the controller of the robot during physical-contact. (c) Top: position tracking error. Middle: Reflected articular velocity constraint on the dynamic control variable (articular acceleration) of joint 0. Bottom: Constrained articular velocity of joint 0. d) Articular torques.

robot in Cartesian space is around 0.051 *m* (see Fig. 4.2.c Top). Such a large value is due to the activation of the constraint on articular velocity (2.11b) that is reactively coped with as the velocity of joint 0 reaches its max/min limits during the movements of the robot (see Fig. 4.2.c Middle). During the pre-collision phase, the maximum reached velocity of the end-effector is about 2 *m/s*. When this movement is less dynamic, for example, with a maximum Cartesian velocity of 1 *m/s* instead 2 *m/s*, the position tracking error for the robot end-effector can be

reduced to  $0.005\text{ m}$ . Note that the upper and lower limits for the velocity of this joint are fixed at lower values than what is recommended by KUKA for the KUKA LWR4 robotic arm:  $[\dot{q}_{m_0}, \dot{q}_{M_0}] = [-1.9, 1.9]\text{ rad/s}$  instead of  $[-2.18, 2.18]\text{ rad/s}$ . To explain this choice, note that the experiments presented in this chapter have been performed prior to our work on the *constraints incompatibility* problem presented in Chapter 2, it is therefore the classical formulations of the constraints related to the *articular physical limitations* of the robot (2.11) that are hereby used in the configuration of the controller instead of the new formulations (3.33). Because of the uncertainties related to the dynamic model of the robot and the naive<sup>3</sup> expressions of the constraints linked to its articular physical limitations,  $\delta t$  for the discrete formulation of the joint velocity constraint (2.11b) is fixed at  $30\text{ ms}$  to give sufficient time to the system to react and saturate its articular velocity.

The maximum kinetic energy displayed by the robot in the direction of the nearby human-operator<sup>4</sup> is  $2.8\text{ J}$ .  $K_p$  is fixed at 200 and  $K_d = 28$ .

#### 4.4 Scenario 1: human-operator intersecting with the trajectory of the robot and no constraints on its kinetic nor *potential* energies

In this case, the human-operator enters the workspace of the robot and intersects with its trajectory, a physical-contact is engaged and the movement of the robot is obstructed as it tries to reach its desired position towards point  $\textcircled{B}$ <sup>5</sup>. the controller is implemented as described in the test case scenario. No energy related constraints are considered. At collision (see video in [Meguenani 2016b]),  $2\text{ J}$  of kinetic energy are instantaneously dissipated to create the resulting impact force (see Fig. 4.2.a).

As shown in Fig. 4.2.b, after the establishment of physical-contact, *potential energy* within the controller of the robot increases to reach a maximum value of  $14\text{ J}$ . We recall that this quantity is computed as:

$$S_{p_{\text{contact}}}^x = m(\mathbf{q}_{|k})_{x}^{eq} \ddot{X}_{EE|k}^{\mathbf{x}} \left( \mathbf{X}_{x|k}^* - \mathbf{X}_{x|k} \right) \mathbf{x}, \quad (4.1)$$

with:

$$\begin{aligned} \ddot{X}_{EE|k}^{\mathbf{x}} &= \left( \dot{J}(\mathbf{q}_{|k})_{EE} \dot{\mathbf{q}}_{|k} + J(\mathbf{q}_{|k})_{EE} \ddot{\mathbf{q}}_{|k}^c \right) \mathbf{x} \\ \ddot{\mathbf{q}}_{|k}^c &= M(\mathbf{q}_{|k})^{-1} \left( \boldsymbol{\tau}_{|k}^c - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}) \right). \end{aligned} \quad (4.2)$$

Consequently, a contact force in the direction of the desired position is created. Generated articular torques can be seen in Fig. 4.2.d. We highlight the amount of torques in joint 0 and joint 2:  $\tau_0 \simeq -18\text{ N.m}$  and  $\tau_2 \simeq -16\text{ N.m}$ <sup>6</sup>. Once physical contact is released, the *potential energy* stored in the system is transformed into kinetic one as fast as possible; which can be dangerous considering the nearby operator. Finally, the robot goes back to its normal behaviour.

<sup>3</sup>Naive because they do not take into account the dynamic capabilities, i.e., the max/min deceleration and jerk producible by the actuators of the robot.

<sup>4</sup>The direction of the human-operator is mainly along the  $-\mathbf{x}$  axis (see Fig. 4.1).

<sup>5</sup>The trajectory generator is blind to any collision with the environment. It continues generating desired position, velocity and feed forward acceleration for the end-effector of the robot.

<sup>6</sup>The contact force between the robot and the human-operator is mainly caused by the generated torques in joint 0 and joint 2. The contact force is not directly measured.

## 4.5 Scenario 2: human-operator intersecting with the trajectory of the robot and constraint on its kinetic energy

In this scenario, the human-operator gets within the workspace of the robot and intersects with its trajectory. As the person moves towards the robot, the first formulation of the constraint on kinetic energy (3.43) is included<sup>7</sup> in the configuration of the controller and used to reduce the kinetic energy of the robotic manipulator before collision. We recall that this constraint has the following expression:

$$S_c = \frac{1}{2}m(\mathbf{q}_{|k})_{EE,O}^{eq} v_{EE|k+1}^{EE,O^2} \leq E_{c_{limit}} = E_{c_{safe}} + K(d - d_{safe}), \quad (4.3)$$

Inequality and equality constraints used in the test case scenario are also considered in the control scheme. The parameters of the controller are fixed as:  $E_{c_{safe}} = 0.02 J$  (if fixed at 0, the robot stops moving),  $K = 0.4 N$ ,  $d_{safe} = 0.3 m$  and  $d_{max} = 7 m$ . As seen in Chapter 3 (Subsection 3.7.3), choosing a big value for  $K$  (e.g., 50) will result in high articular jerks when the constraint on kinetic energy is activated (see Fig. 3.20.b). For this reason, a smaller value for  $K$  is used when implementing the kinetic energy related constraint on the real robot. Also, a larger distance ( $d_{max} - d_{safe}$ ) = 6.7 m is allowed for the robot to reduce its kinetic energy. Compared to the braking distance used in simulation ( $d_{max} - d_{safe}$ ) = 0.2 m.

Dissipated kinetic energy is shown in Fig. 4.3. At collision, only 0.02 J of energy are dissipated compared to the previous scenario in which the kinetic energy of the robot is not pre-constrained (2 J). As can be seen, the constraint is respected at every time-step and the impact between the robot and the operator is considered to be safe. We recall the safety limit value of dissipated kinetic energy recommended by the ISO/TS 15066 [ISO 2016] standard in case of a collision between the robot and a human-operator at the level of the hands:  $E_{c_{limit}} = 0.49 J$ . After the establishment of physical-contact, the non-constrained *potential energy* starts accumulating in the controller of the robot to reach  $\approx 14 J$  (see Fig. 4.4.a). Actuation torques of joint 0 and joint 2 that are directly proportional to the generated contact force are shown in Fig. 4.4.b:  $\tau_0 \simeq -20 N.m$  and  $\tau_2 \simeq -17 N.m$ . We highlight the chattering phenomenon on the actuation torque during physical-contact. Indeed, because of the inevitable occlusions, during this phase, the distance between the robot and the human operator is more difficult to compute and may become null. In this case, the constraint on kinetic energy can be released and consequently the robot forces its way towards point (B) as its movement is obstructed by the human-operator. After releasing physical-contact, the constraint on kinetic energy prevents the stored *potential energy* from rapidly and dangerously transforming into kinetic one. The distance related constraint on the kinetic energy of the robot makes both the physical-contact establishment and releasing processes safer for physical human-robot interaction.

## 4.6 Scenario 3: human-operator intersecting with the trajectory of the robot and constraint on its *potential energy*

In this scenario, the person also intersects with the trajectory of the robot between points (A) and (B). The kinetic energy of the KUKA LWR4 is not constrained during the approach of the operator. After the establishment of physical-contact, collision is detected using the system's proprioceptive torque sensors then, the constraint (3.50) is added to the controller and used to

<sup>7</sup>The constraint is not removed from the controller even after the establishment of physical-contact.

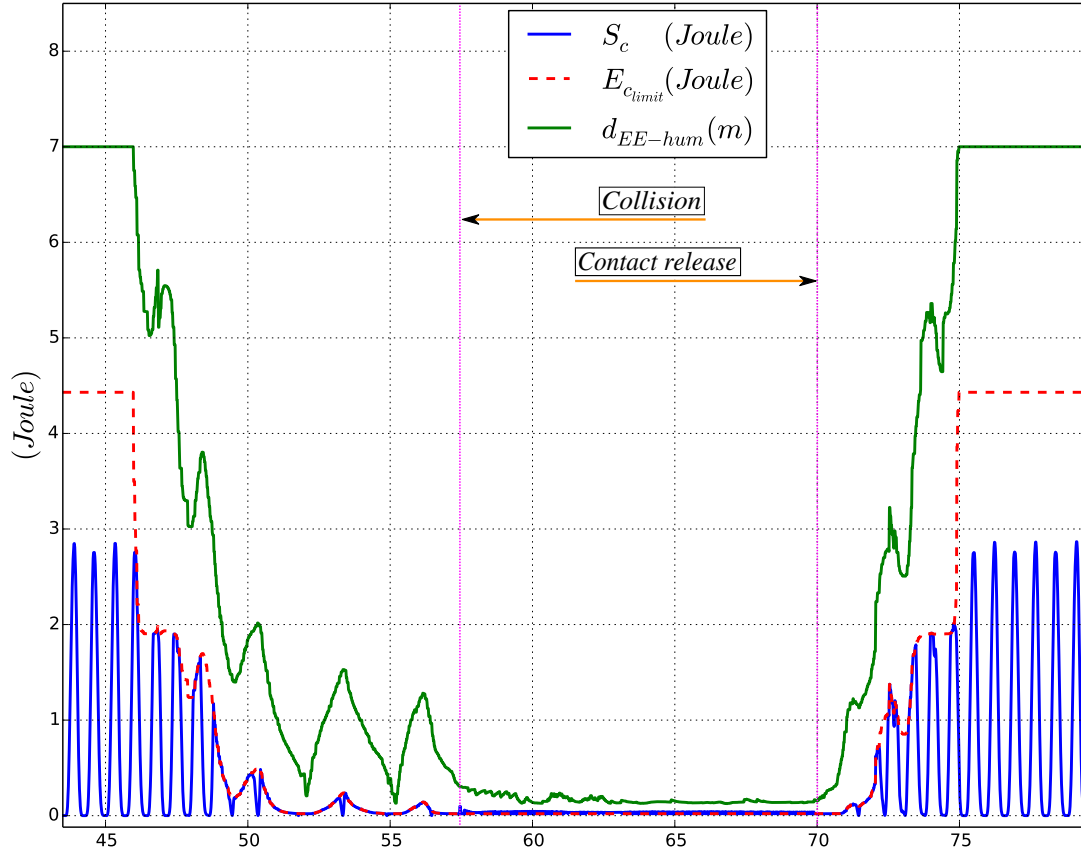


Figure 4.3: Constrained kinetic energy of the robotic manipulator expressed at the level of its end-effector in the direction of the approaching operator.

limit the amount of *potential energy* that accumulates in the controller of the robot. We recall the expression of such constraint:

$$|S_{p_{contact}}^\alpha| = |m(\mathbf{q}_{|k})_{\alpha}^{eq} \ddot{\mathbf{X}}_{EE|k}^\alpha (\mathbf{X}_{|k}^* - \mathbf{X}_{|k}) \alpha| \leq E_{p_{safe}}^\alpha \Leftrightarrow \begin{cases} S_{p_{contact}}^\alpha \leq E_{p_{safe}}^\alpha, \\ S_{p_{contact}}^\alpha \geq -E_{p_{safe}}^\alpha, \end{cases} \quad (4.4a)$$

with  $\alpha$  representing the  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  axis in Cartesian space. Except for the constraint on articular jerk, inequality and equality articular constraints described in (2.11) and (3.34) are also considered in the control scheme. The parameters of the controller are fixed as:  $E_{p_{safe}}^x = 0.001 J$ ,  $E_{p_{safe}}^y = 0 J$ ,  $E_{p_{safe}}^z = 0 J$ .

At impact, as depicted in Fig. 4.5, 1.83 J of kinetic energy are instantaneously dissipated resulting into a more dangerous collision force, compared to the amount of kinetic energy (0.02 J) dissipated in the previous scenario. Fig. 4.6.a shows how the constraint on *potential energy* is satisfied at every time-step all along the physical-contact phase. Consequently, the contact force applied to the human-operator is harmless. Corresponding actuation torque and particularly of joints<sup>8</sup> 0 and 2 are shown in Fig. 4.6.b:  $\tau_0 \simeq -4 N.m$  and  $\tau_2 \simeq -3 N.m$ , compared to the two previous scenarios in which the constraint on *potential energy* is not considered in the configuration of the controller:  $\tau_0 \simeq -20 N.m$  and  $\tau_2 \simeq -17 N.m$  during physical interaction. When using the introduced constraint on *potential energy*, the robot is more compliant during

<sup>8</sup>The generated contact force during physical contact is mainly caused by the articular torques on these two joints.



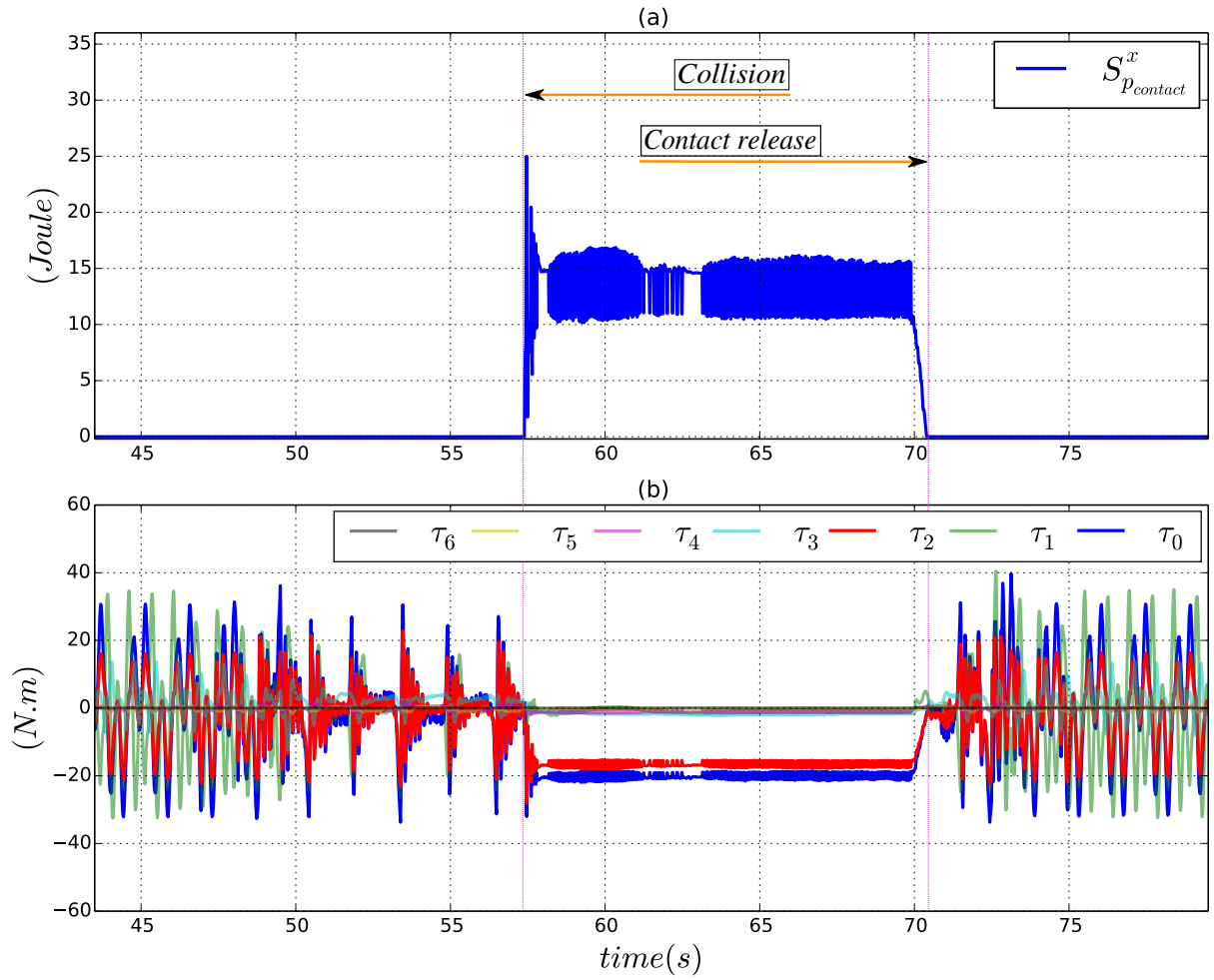


Figure 4.4: (a) *Potential energy* stored in the controller of the robot as it physically interacts with a human-operator within its workspace. (b) Articular torques.

physical-contact and if needed, can safely and easily be moved or pushed away by the human-operator. Also, thanks to this same constraint, no vibrations are induced by the actuators of the robot during physical-contact. Once contact is released, the constraint on *potential energy* is de-activated and removed from the controller. Because of the accumulated position error<sup>9</sup>, at this stage, the controller of the robot holds a large amount of *potential energy*  $S_{p_{profile}}$ <sup>10</sup>. This energy transforms into kinetic one as fast as possible as the robot moves towards point (B). Because the kinetic energy of the robot is not constrained, a harmful second collision is still possible with the nearby human-operator during this *energy transformation* phase. Which makes the contact-releasing phase more dangerous, compared to the previous scenario in which the kinetic energy of the robot is constrained during the *physical-contact release* phase. Finally, the KUKA LWR4 goes back to its normal behaviour.

<sup>9</sup>As the movement of the robot is restrained during-physical contact, the error between the real and desired positions for its end-effector accumulates and increases.

<sup>10</sup> $S_{p_{profile}}$  is the *potential energy injected* in the controller of the robot at each time-step (see 3.5).

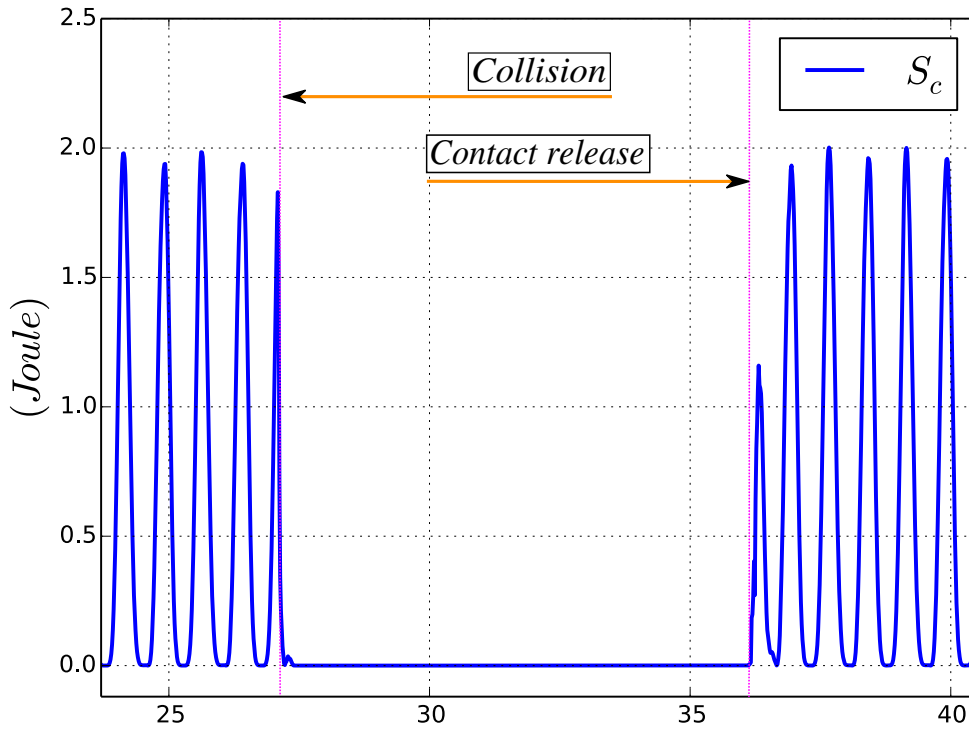


Figure 4.5: Kinetic energy deployed by the robotic manipulator expressed at the level of its end-effector in the direction of the human-operator entering its workspace and physically obstructing its movement.

#### 4.7 Scenario 4: human-operator intersecting with the trajectory of the robot and constraints on both its kinetic and *potential* energies

In this last scenario, the human also gets within the workspace of the KUKA LWR4 and intersects with its trajectory as the end-effector of the robotic arm tracks a desired trajectory between points **(A)** and **(B)** (see Fig. 4.1). Using (4.3), the kinetic energy of the robot is constrained during the approach of the operator. At impact, collision is detected using the system's proprioceptive torque sensors, physical contact is established then without removing the constraint on kinetic energy, the *potential energy* related constraint (4.4) is added to the control scheme. This constraint is used to saturate the amount of *potential energy* that accumulates in the controller of the robot along the  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  axis in Cartesian space (see video in [Meguenani 2016a]). Except for the constraint on articular jerk, inequality and equality articular constraints described in (2.11) and (3.34) are also included in the configuration of the controller. Which parameters are fixed as:  $E_{c_{safe}} = 0.06 J$ ,  $K = 0.4 N$ ,  $d_{safe} = 0.3 m$ ,  $d_{max} = 7 m$ ,  $E_{p_{safe}}^x = 0.001 J$ ,  $E_{p_{safe}}^y = 0 J$  and  $E_{p_{safe}}^z = 0 J$ .

As shown in Fig. 4.7, the kinetic energy related constraint along the direction of the human-operator is respected at every time-step. At collision, only  $0.02 J$  of kinetic energy is dissipated. Which results into a safe physical-contact establishment. During the post-collision phase, the constraint on *potential energy* generated in the controller of the robot is also satisfied at every time-step (see Fig. 4.8.a). This constraint is decoupled along the  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  axis in Cartesian space and results into a reduced and safer contact force between the human and the robot. The KUKA LWR4 at this stage is compliant and can safely and easily be pushed or moved if needed.

Actuation torque of joint 0 and joint 2 that are directly proportional to the contact force are shown in Fig. 4.8.b:  $\tau_0 \simeq -4 \text{ N.m}$  and  $\tau_2 \simeq -3 \text{ N.m}$ , compared to the case when the constraint on *potential energy* is not considered in the control scheme:  $\tau_0 \simeq -20 \text{ N.m}$  and  $\tau_2 \simeq -17 \text{ N.m}$  during physical contact.

Once contact is released, the constraint on *potential energy* (4.4) is removed from the configuration of the controller. At this moment, the robot contains sufficient *potential energy*  $E_p = m(\mathbf{q})_{EE,*}^{eq} \ddot{X}_{EE|k}^{EE,*} (\mathbf{X}_{EE|k}^* - \mathbf{X}_{EE|k}) \mathbf{n}_{EE|k}$  to move towards point  $\textcircled{B}$ . Considering the nearby operator who stills within its workspace, without the constraint on the kinetic energy of the robot,  $E_p$  can rapidly transform into a hazardous amount of kinetic energy. however, thanks to the kinetic energy related constraint (4.3), physical-contact is safely released and the *potential energy* accumulated within the controller of the robot progressively and harmlessly transforms into kinetic one as the end-effector reaches its desired position  $\textcircled{B}$ .

## 4.8 Conclusion

In this chapter, the controller and energy related constraints introduced in Chapter 3 have been successfully implemented on a real KUKA LWR4 robot as it physically interacts with a human-operator. As the person approaches the robot and enters its workspace, using the kinetic energy related constraint, the kinetic energy displayed by the robotic arm in the direction of the human-operator is saturated. The robot still performs its assigned task but with a diminished dynamics. Physical-contact can then safely be established.

During the post-collision phase, the *potential energy* that accumulates in the controller of the robot is also successfully constrained. Depending on the allowed amount of *potential energy*, the resulting contact force can be reduced and if needed, the robot easily and safely moved by the operator.

After releasing physical-contact, the constraint on kinetic energy ensures a smooth and safe transformation of the *potential energy* held in the system's controller into kinetic one. Consequently, the human-operator can safely and progressively leave the workspace of the robotic manipulator. The task dynamical performances of the robot are then fully recovered.

As explained in Chapter 2 and Chapter 3, a solution to the control problem cannot be guaranteed for every time-step if the formulations of the introduced energy related constraints do not take into account the reaction capabilities<sup>11</sup> of the actuators of the robot. This is actually the reason why the control sample-time for the implementation of the introduced energy related constraints on the real robot is increased from 1 *ms* in simulation to 15 *ms* in the real life. The KUKA LWR4, for example, is not able to brake and cope with  $E_{c_{limit}}$  within only 1 *ms* of time-window and 15 *ms* is the smallest duration needed for the robot to cope with the limit of the kinetic energy related constraint. Moreover, increasing the control time-step duration deteriorates the tracking performances of the robot regarding the trajectory tracking task it has to perform.

The only way to guarantee the existence of a solution to the control problem for every time-step is to reformulate the energy related constraints to include the max/min producible articular torque/deceleration and jerk. However, because of the dynamical and non-predictable nature of  $E_{c_{limit}}$  in (3.15), ensuring the *viability* of the state of the robot during human-robot interaction is even more difficult. Indeed,  $E_{c_{limit}}$  depends on the non-predictable movements of the human-operator and on his real-time distance to the robot. The constraint on the kinetic energy displayed by the robot is therefore of type 6 (see table 2.1 in Chapter 2) and can be modified

<sup>11</sup>i.e., producible articular torque/deceleration and jerk.



as follows. Similarly to the joint velocity constraint:  $\dot{\mathbf{q}}_{|k} \leq \dot{\mathbf{q}}_M$ ,  $S_c \leq E_{c_{limit}}$  can be written:

$$v_{|k+1} \leq \sqrt{\frac{2E_{c_{limit}}}{m(\mathbf{q}_{|k})^{eq}}}, \quad (4.5)$$

with  $\tilde{v}_{limit} = \sqrt{\frac{2E_{c_{limit}}}{m(\mathbf{q}_{|k})^{eq}}}$  being a dynamic limit in contrast to  $\dot{\mathbf{q}}_M$  which is a static limit. To ensure at every time-step a solution to the control problem when the system must cope with such a constraint on its kinetic energy, (4.5) can be reformulated as:

$$v_{|k+n} = v_{|k} + n\ddot{X}\delta t + \frac{(n^2 - n)}{2}\ddot{X}_m\delta t^2 \leq \tilde{v}_{limit}^{predicted}, \quad (4.6)$$

with  $\ddot{X} = \dot{J}(\mathbf{q}_{|k})\dot{\mathbf{q}}_{|k} + J(\mathbf{q}_{|k})\ddot{\mathbf{q}}_{|k}^c$ ,  $n$  the maximum number of time-steps needed for the robot in a state  $S_{|k}$  to bring its acceleration expressed in operational space to 0 considering a constant amount of jerk  $\ddot{X}_m$  it can produce.  $\tilde{v}_{limit}^{predicted}$  is the predicted value of  $\tilde{v}_{limit}$ . The main challenge then is to properly compute  $n$  and the prediction horizon  $p$  for  $\tilde{v}_{limit}^{predicted}$ :

$$\tilde{v}_{limit_{|k}}^{predicted} = \tilde{v}_{limit_{|k}} + p\ddot{X}_m, \quad (4.7)$$

with  $\ddot{X}_m$  the maximum variation of  $v_{|k}$  in operational space the robot can handle. Finally, note that when using the formulation of the constraint on the kinetic energy (4.6) that takes into account the dynamic capabilities of the actuators of the robot, the control sample-time in theory may potentially be fixed at 1 *ms* without causing any *constraints incompatibility* problems.

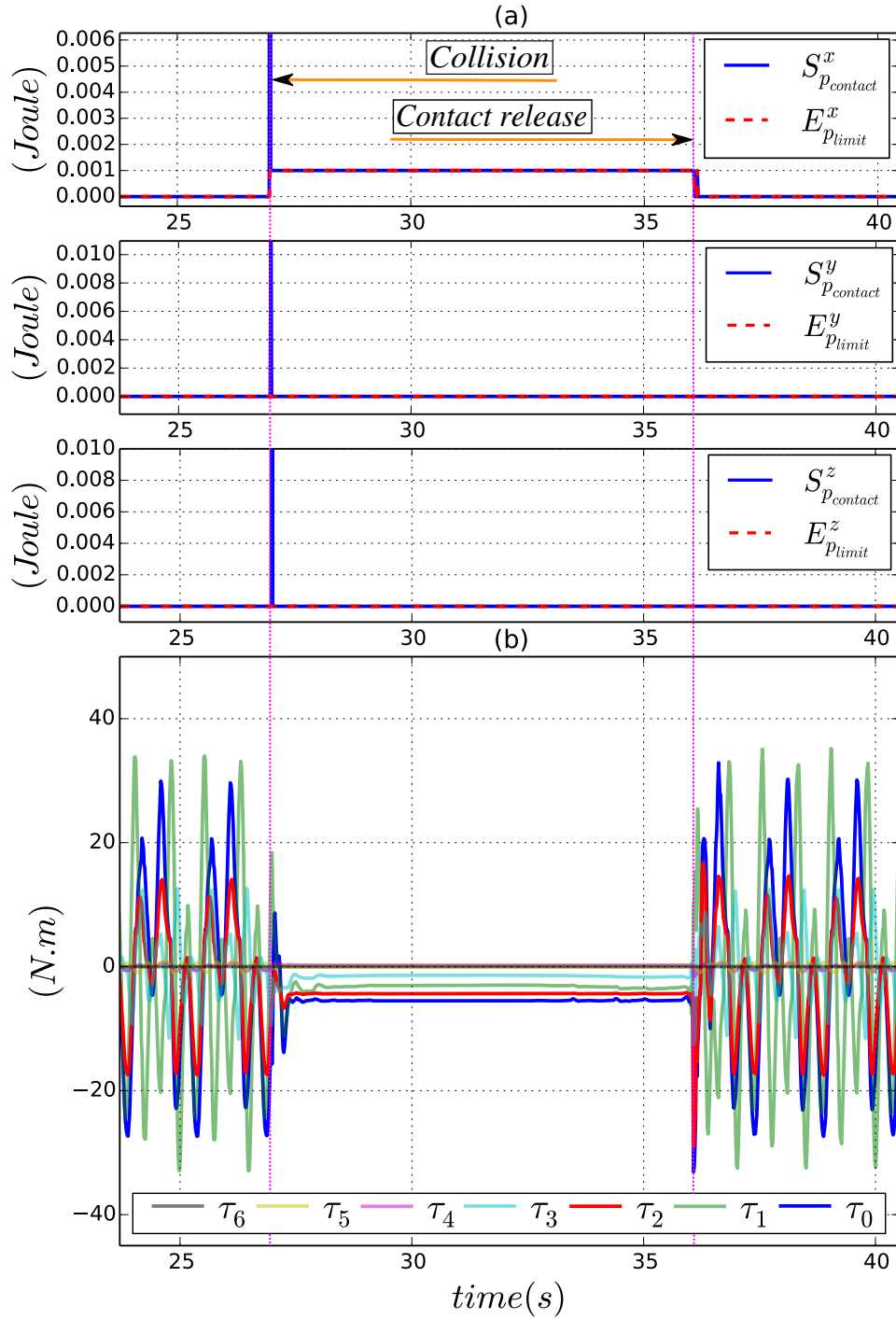


Figure 4.6: (a) *Potential energy* stored in the controller of the robot during physical contact along the  $x$ ,  $y$  and  $z$  axis in Cartesian space. (b) Articular torques.

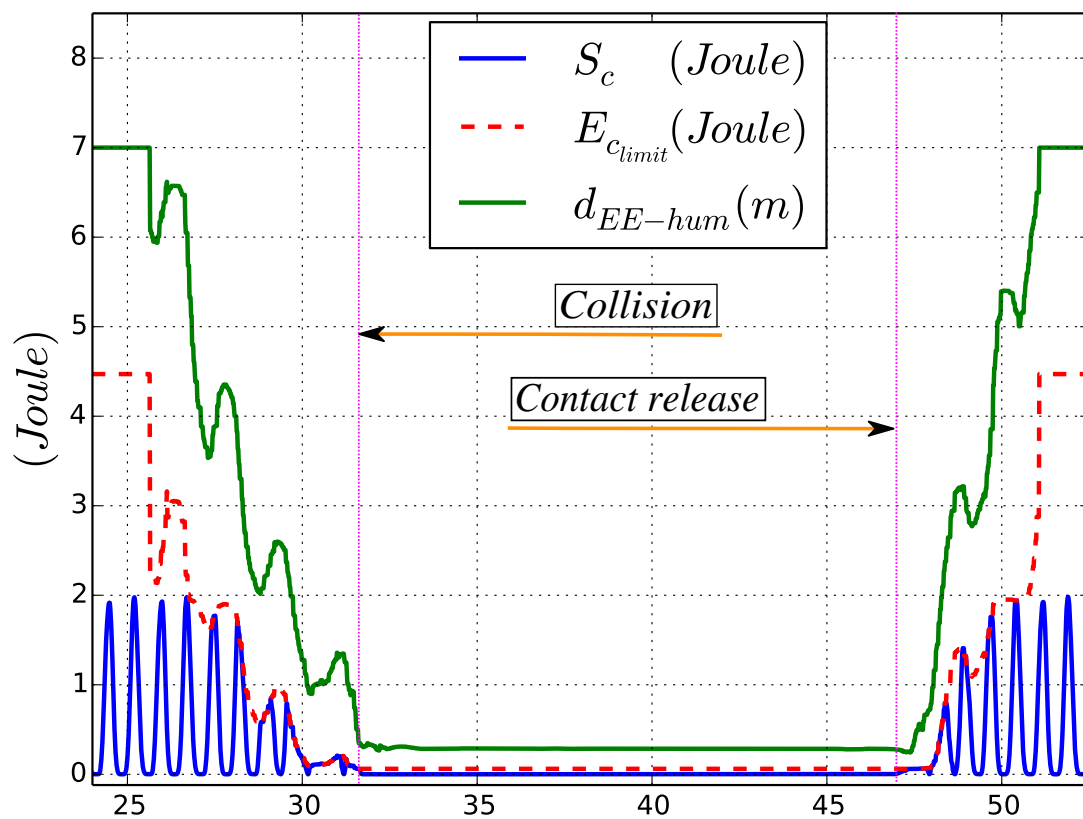


Figure 4.7: Constrained kinetic energy of the robotic manipulator expressed at the level of its end-effector in the direction of the human-operator entering its workspace and physically obstructing its movement.

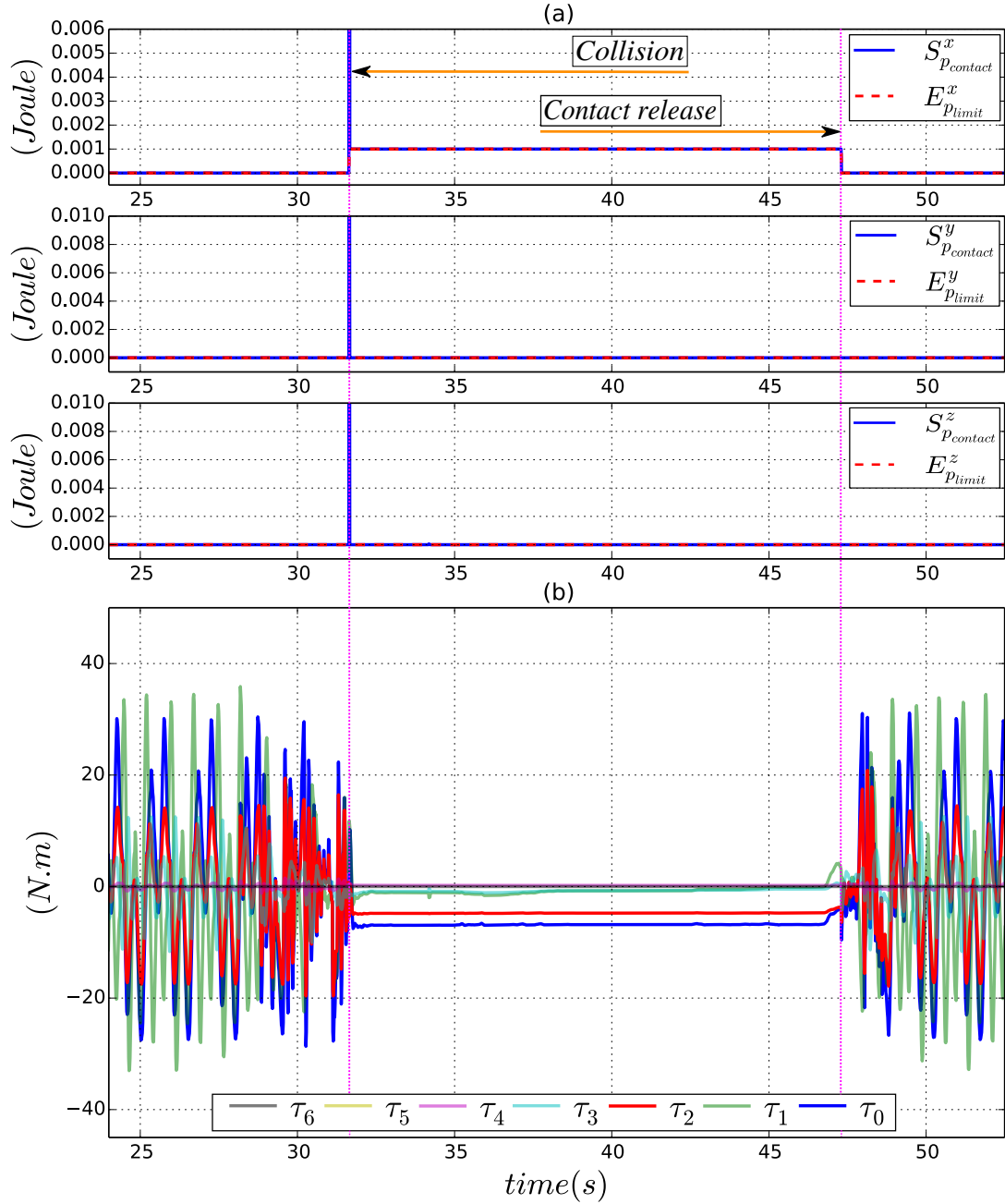


Figure 4.8: (a) *Potential energy* stored in the controller of the robot during physical contact along the  $x$ ,  $y$  and  $z$  axis in Cartesian space. (b) Articular torques.



## Chapter 5

# Conclusions and outlook

In this chapter, the thesis is concluded with a summary of the introduced contributions and an outlook on some lines of research identified as the consequent extension of the presented work.

### 5.1 Contributions

Human-robot interaction is one of the most complex scenarios a robotic system could face. Indeed, in such dynamic and non-predictable environment, several criteria must be satisfied with safety being the most important one. In the work presented in this thesis, the concept of safety is tackled at two different levels. First, when reactive control loops are used to handle the user specified objectives and/or constraints, the *solvability* of the control problem and the effect of the retained solution on the *solvability* of future control problems is rarely studied. Consequently, *constraints incompatibilities* that render the control problem impossible to solve without inevitable constraints violation are frequent in robotics, thus compromising safety. Therefore, using a control law able to solve a feasible problem and maintain its permanent feasibility is the main condition to ensure safety regarding constraints satisfaction. The second part of this work tackles the problem of safety for a human-operator entering the workspace of a robotic arm and physically interacting with it. In a generic way, the physical integrity of both the robot and its environment are considered for synthesizing a safe controller that makes the robot more suitable for human-robot interaction.

In the first chapter, the context of this thesis is presented from the industrial point of view of how to integrate robotic manipulators as collaborative tools which tasks are defined on the fly and constraints discovered in real-time. This objective exhibits general issues regarding: 1) how to properly guarantee safety for humans when interacting with robots and also: 2) in terms of constraint-based control design. For the first field of investigation, a short overview of the state-of-the-art techniques and standards employed to ensure safety during human-robot interaction are presented and used to position our contributions to open problems. These contributions go beyond what is usually proposed in the literature as a generic approach based on the modulation of the robot's energy at the control level is used to continuously guarantee safety for a human-operator during different interaction phases. For the second field of investigation, the issue of *constraints incompatibility* that occurs when using reactive control loops is introduced. The various specifications that must be considered when dealing with such a problem are highlighted. Finally, our contributions to the resolution of the *constraints incompatibility* problem are presented: not only the resolution of the control problem is tackled but its formulation is also addressed. The latter is very critical since it conditions the existence of a solution to this

problem over an infinite horizon of time.

In the second chapter, the problem of constraints incompatibility related to the physical limitations of a serial robot is studied. The relation between the different parameters of the robot's extended state  $S = \{\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \dddot{\mathbf{q}}\}$  along with all the combinations of the set {joint position limits - joint velocity limits - joint acceleration limits - joint jerk limits} is examined to reveal possible incompatibility cases. These incompatibilities are then resolved by reformulating the naive expressions of the joint physical bounds related constraints for a robot controlled at the dynamic-level. The constraint on joint velocity is modified to take into account the amount of jerk producible by the robot actuators and the constraint on articular position is reformulated to include both the robot's maximum articular deceleration and jerk capabilities. The final max/min bounds on the dynamic control input enable the system to successfully and simultaneously cope with all the constraints related to its articular physical limitations as an optimal solution for the control problem is guaranteed at every control time-step.

In the third chapter, a new control strategy that allows the modulation of both the kinetic energy of the robot and the amount of *potential energy* held in its controller is introduced. This approach is successfully used to ensure safety for a human-operator during his interaction with a robotic arm. The impact force at the establishment of a physical contact is reduced by monitoring then constraining the kinetic energy of the robot under some *safe* limit just before collision. The contact force on the other hand is diminished thanks to the constraint on the *potential energy* that accumulates in the controller of the robot during physical-contact. Each constraint is composed of: 1) an introduced energy based safety indicator that reflects the degree of danger the robotic manipulator represents towards the human and, 2) an energy based safety criterion that corresponds to the amount of energy considered to be safe for the robot to exhibit during the interaction.

First used separately to express the energy based constraints, the relation between the robot's kinetic and potential energies is formulated and the robot along with its controller is represented as a system to which *potential energy* is *injected* and transformed into kinetic one. Based on this analysis, the concept of *task energy profile* is introduced. A task energy profile is the instantaneous amount of *potential energy* held in the controller of the robot and that transforms into kinetic one during its free movements. For tasks with repetitive motion cycles, this profile can be registered then used for the formulation of a constraint that saturates the instantaneous amount of *potential energy* loaded in the controller of the robot. With such constraint, this amount, intended to increase in case of a deliberate or non-intentional physical contact with the environment, is saturated; which results into a compliant robot that does not generate harmful contact forces. Illustrated on a KUKA LWR4 robotic arm in simulation, the introduced safe control strategy allows the robot to safely physically interact with its environment.

In the fourth chapter, the controller along with the safety related constraints introduced in Chapter 3 are implemented on a real KUKA LWR4 as it physically interacts with a human-operator. A vision system is used to detect the approach of the human and computes his real-time distance to the robot. Depending on this distance, the robot's kinetic energy is modulated. A large amount is allowed when the person is distant and a safe value is imposed before collision. During physical contact, thanks to the constraint on the *potential energy* that accumulates into the controller of the robot, the resulting contact force is saturated, the robot is compliant and can safely be moved or pushed by the human-operator. The introduced control scheme allows a person to safely enter the workspace of a robotic manipulator, physically interact with it and securely release contact and move away. The *potential energy* held in the controller of the robot during physical contact is prevented from rapidly and dangerously transforming into kinetic one. The system progressively recovers its optimal dynamics.

## 5.2 Perspectives

The scope of the research topics tackled in this thesis leaves many unanswered questions and hence several directions for future research. They are hereby briefly discussed.

### 5.2.1 Extension of the work on the constraints incompatibility problem

Although the new formulations of the articular velocity and position constraints allow to *properly*<sup>1</sup> cope with the articular velocity and position limits when applied to a simulated mechanism, the gap between simulation and real world affects however the quality of the introduced algorithms when used on a real world system. Indeed, unmodeled uncertainties such as sensor noises and delays can result in control strategies that are feasible for the ideal system but not for the real one. Accounting for the robot dynamics model uncertainties in the formulation of the control problem is therefore essential when implementing the introduced algorithms on real world robots. The methods developed in [Del Prete 2016] to account for uncertainties in optimization problems seem to constitute a good approach with which this problem can be tackled. In their work, Del prete et al. model the uncertainties related to the dynamic control input of a humanoid robot using two different techniques: 1) as deterministic variables belonging to a known set, a solution that is feasible for any realization of the uncertainty in the given set is then computed and alternatively, 2) as random variables following known probability distributions so the solution that satisfies the constraints with a large-enough probability is retained. Besides the required work for the implementation of the introduced new constraints formulations on real robots, the proposed solutions for the problem of *constraints incompatibility* resolved for constraints of type 1 (see table 2.1) need to be extended for constraints expressed in operational space, as for example, the constraint on energy introduced in Chapter 3. Because related to human safety, the energy based constraints are of high priority and therefore must imperatively always be satisfied. As explained in Chapter 2, guaranteeing the *viability* of the state of the robot when its controller includes such type of constraints is not straight forward. *Prediction* is however the key word to successfully handle this type of constraints and for which MPC (*Model Predictive Control*) seems to be the most appropriate approach as discussed in Section 2.1.3.

### 5.2.2 Control of collaborative robots for safe human-robot interaction

The energy-based control approach for safe human-robot interaction developed in Chapter 3 and Chapter 4 constitutes a very interesting framework as it:

- defines a unified way to ensure safety for a human-operator during different interaction phases with a robotic manipulator;
- tends to break the classical barrier which consists in making a distinction between the robot motion in free-space and its behaviour in contact with the environment by considering mainly the system's energetic state.

However, several aspects that have been neglected must be investigated in a required future work. The first one is actually of high importance: while monitoring the energy of the robot and constraining it under some safe limit is necessary, it is unlikely that safety can be guaranteed if this value does not account for the energy displayed by the human during physical interaction. Indeed, if not void, the very own kinetic energy of the human for example just before collision could compromise his safety even if the robot's kinetic energy is properly saturated. Its amount

---

<sup>1</sup>Considering the amount of deceleration and jerk producible by the actuators of the robot.



should then also be estimated and considered in the formulation of the energy based safety indicators. A way of effectively accounting for the effective mass of the collided human body region and its velocity is proposed in the recent ISO/TS 15066 standard [ISO 2016].

A second limiting factor is related to the absence of a proper theoretical proof regarding the stability of the introduced energy-based safe controller. We tend to think that our control strategy leads to safe and stable behaviours of the robot during human-robot interaction but no theoretical approach to prove it has yet been proposed. Just like the stability analysis approach used in [Schindlbeck 2015], looking at this problem in terms of *passivity* could be one way to tackle this question. Finally, including safety as a constraint in the formulation of the control problem is the method for guaranteeing human safety proposed in this work. An additional approach for handling safety that can be used simultaneously, is to consider the amount of energy displayed by the robot directly at the level of trajectory generation (considering trajectory tracking tasks). Similarly to fixing the max/min velocity, acceleration and jerk at the generation of a desired trajectory, maximum deployed kinetic energy can also be included as a design parameter in a real-time algorithm of trajectory generation.

## Appendix A

# Computation of the articular deceleration capabilities for a robotic manipulator

For a robotic manipulator, the relation between the control torque input  $\tau_{|k}^c$  and the resulting articular acceleration  $\ddot{\mathbf{q}}_{|k}$  can be expressed using the equations corresponding to its dynamic model:

$$M(\mathbf{q}_{|k})\ddot{\mathbf{q}}_{|k} + \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}) = \tau_{|k}^c. \quad (\text{A.1})$$

Therefore, the maximum producible deceleration/acceleration  $[\ddot{\mathbf{q}}_m, \ddot{\mathbf{q}}_M]$  is function of the minimum and maximum producible torque  $[\tau_m, \tau_M]$  (usually specified by the system's manufacturer) in addition to the instantaneous state  $\{\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k}\}$  of the system. Instantaneous minimum and maximum producible decelerations ( $\ddot{\mathbf{q}}_m^{inst}, \ddot{\mathbf{q}}_M^{inst}$ ) can be computed:

$$\arg \min_{\tau_{|k}} M(\mathbf{q}_{|k})^{-1}(\tau_{|k} - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k})), \quad (\text{A.2})$$

$$\arg \max_{\tau_{|k}} M(\mathbf{q}_{|k})^{-1}(\tau_{|k} - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k})), \quad (\text{A.3})$$

s.t:  $\tau_m \leq \tau_{|k} \leq \tau_M$

Then:

$$\ddot{\mathbf{q}}_M^{inst} = M(\mathbf{q}_{|k})^{-1}(\tau_{max} - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k})), \quad (\text{A.4})$$

$$\ddot{\mathbf{q}}_m^{inst} = M(\mathbf{q}_{|k})^{-1}(\tau_{min} - \mathbf{b}(\mathbf{q}_{|k}, \dot{\mathbf{q}}_{|k})), \quad (\text{A.5})$$

where  $\tau_{min}$  and  $\tau_{max}$  are respectively the instantaneous solutions for (A.2) and (A.3) during the movement of the robot shown in Fig. 2.6. Note that these two parameters differ from  $\tau_m$  and  $\tau_M$ , as  $\tau_m$  and  $\tau_M$  do not obligatory generate the maximum positive and negative producible articular decelerations.

For a KUKA LWR4 serial robot with:  $\tau_M^T = -\tau_m^T = [200, 200, 100, 100, 100, 30, 30]$ , the instantaneous maximum and minimum producible accelerations for joint 0 during the movement shown in Fig. 2.6 are depicted in Fig. A.1.

However, as  $\ddot{\mathbf{q}}_M$  and  $\ddot{\mathbf{q}}_m$  are used as constants in the formulation of the constraints on articular velocity and position, the final values for these parameters can be computed using algorithm 6 :

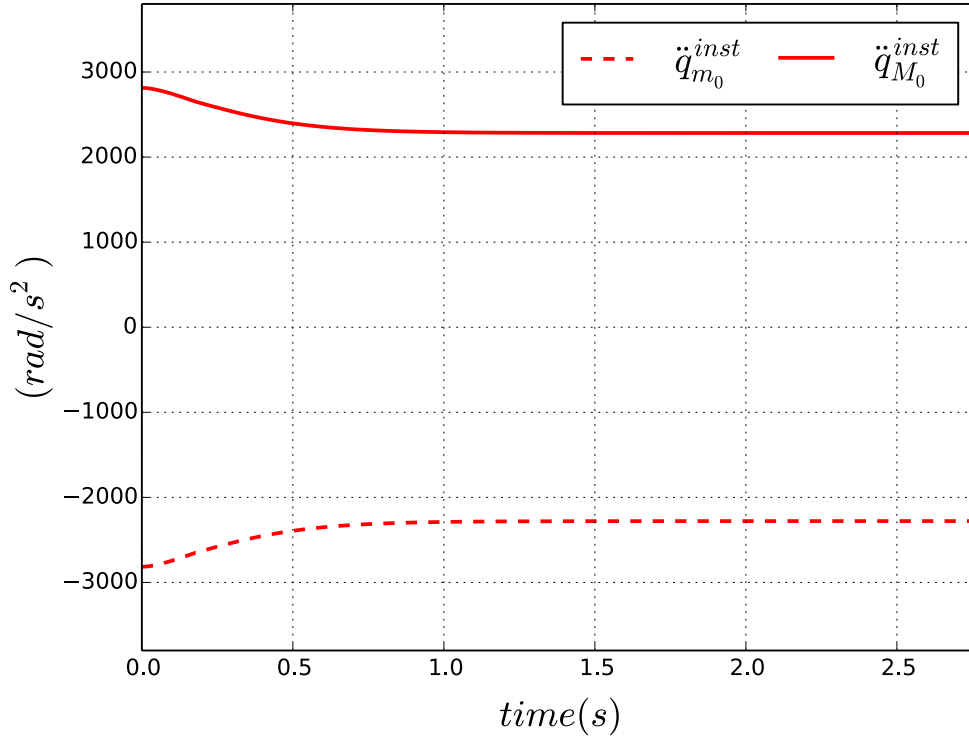


Figure A.1: Instantaneous producible maximum and minimum articular accelerations for joint 0 during the movement of the KUKA LWR4 as shown in Fig. 2.6.

---

**Algorithm 6** Compute  $\ddot{\mathbf{q}}_M$  and  $\ddot{\mathbf{q}}_m$

---

**Require:**  $\mathbf{q}_M, \mathbf{q}_m, \dot{\mathbf{q}}_M, \dot{\mathbf{q}}_m, \boldsymbol{\tau}_m, \boldsymbol{\tau}_M, N$

- 1:  $\ddot{\mathbf{q}}_m \leftarrow -[\infty, \dots, \infty]$
  - 2:  $\ddot{\mathbf{q}}_M \leftarrow [\infty, \dots, \infty]$
  - 3: **for** ( $i = 1 \rightarrow N$ ) **do**
  - 4:      $\mathbf{q} \leftarrow \text{random}(\mathbf{q}_m, \mathbf{q}_M)$
  - 5:      $\dot{\mathbf{q}} \leftarrow \text{random}(\dot{\mathbf{q}}_m, \dot{\mathbf{q}}_M)$
  - 6:     compute  $\boldsymbol{\tau}_{max}, \boldsymbol{\tau}_{min}$  using (A.2) and (A.2)
  - 7:      $\ddot{\mathbf{q}}_M^{inst} \leftarrow M(\mathbf{q})^{-1}(\boldsymbol{\tau}_{max} - \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}))$
  - 8:      $\ddot{\mathbf{q}}_m^{inst} \leftarrow M(\mathbf{q})^{-1}(\boldsymbol{\tau}_{min} - \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}))$
  - 9:     **if** ( $\ddot{\mathbf{q}}_M^{inst} \leq \ddot{\mathbf{q}}_M$ ) **then**
  - 10:          $\ddot{\mathbf{q}}_M \leftarrow \ddot{\mathbf{q}}_M^{inst}$
  - 11:     **end if**
  - 12:     **if** ( $\ddot{\mathbf{q}}_m^{inst} \geq \ddot{\mathbf{q}}_m$ ) **then**
  - 13:          $\ddot{\mathbf{q}}_m \leftarrow \ddot{\mathbf{q}}_m^{inst}$
  - 14:     **end if**
  - 15: **end for**
  - 16: **return**  $\ddot{\mathbf{q}}_M, \ddot{\mathbf{q}}_m$
-

# Bibliography

- [ABB 2017] ABB. *YuMi*. <http://new.abb.com/products/robotics/industrial-robots/yumi>, Accessed: 2017.
- [Alami 2006] R. Alami, A. Albu-Schaeffer, A. Bicchi, R. Bischoff, R. Chatila, A. De Luca, A. De Santis, G. Giralt, J. Guiochet, G. Hirzinger, F. Ingrand, V. Lippiello, R. Mattone, D. Powell, S. Sen, B. Siciliano, G. Tonietti and L. Villani. *Safe and dependable physical human-robot interaction in anthropic domains: State of the art and challenges*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1–16. IEEE, 2006.
- [Albu-Schaffer 2005] A. Albu-Schaffer, A. Bicchi, G. Boccadamo, R. Chatila, A. De Luca, A. De Santis, G. Giralt, G. Hirzinger, V. Lippiello, R. Mattone, R. Schiavi, B. Siciliano, G. Tonietti and L. Villani. *Physical human-robot interaction in anthropic domains: Safety and dependability*. In Proceedings of the 4th IARP/IEEE-RAS/EURON Workshop on Technical Challenges for Dependable Robots in Human Environments, 2005.
- [Albu-Schäffer 2007] Alin Albu-Schäffer, Christian Ott and Gerd Hirzinger. *A unified passivity-based control framework for position, torque and impedance control of flexible joint robots*. The International Journal of Robotics Research, vol. 26, no. 1, pages 23–39, 2007.
- [Aubin 1991] Jean-Pierre Aubin. *Viability theory. Systems & control: Foundations & applications*. Birkhäuser, Boston. doi, vol. 10, no. 1007, pages 978–0, 1991.
- [Aubin 2011] Jean-Pierre Aubin, Alexandre M. Bayen and Patrick Saint-Pierre. *Viability theory: new directions*. Springer Science & Business Media, 2011.
- [Baerlocher 2004] Paolo Baerlocher and Ronan Boulic. *An inverse kinematics architecture enforcing an arbitrary number of strict priority levels*. The visual computer, vol. 20, no. 6, pages 402–417, 2004.
- [Bischoff 2010] Rainer Bischoff, Johannes Kurth, Günter Schreiber, Ralf Koeppe, Alin Albu-Schäffer, Alexander Beyer, Oliver Eiberger, Sami Haddadin, Andreas Stemmer, Gerhard Grunwald and G. Hirzinger. *The KUKA-DLR Lightweight Robot arm—a new reference platform for robotics research and manufacturing*. In Proceedings of the 41st International Symposium on Robotics and 6th German Conference on Robotics, pages 1–8, 2010.
- [Boaventura 2012] Thiago Boaventura, Claudio Semini, Jonas Buchli, Marco Frigerio, Michele Focchi and Darwin G Caldwell. *Dynamic torque control of a hydraulic quadruped robot*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 1889–1894. IEEE, 2012.

- [Boyd 2004] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- [Burdet 2001] Etienne Burdet, Rieko Osu, David W Franklin, Theodore E Milner and Mitsuo Kawato. *The central nervous system stabilizes unstable dynamics by learning optimal impedance*. *Nature*, vol. 414, no. 6862, pages 446–449, 2001.
- [Caldwell 1995] Darwin G Caldwell, Gustavo A Medrano-Cerda and Mike Goodwin. *Control of pneumatic muscle actuators*. *IEEE Control Systems*, vol. 15, no. 1, pages 40–48, 1995.
- [Chanteperdrix 2017] Gerum Chanteperdrix. *Xenomai*. <https://xenomai.org>, Accessed: 2017.
- [Chen 1994] Tsing-Hua Chen, Fan-Tien Cheng, York-Yih Sun and Min-Hsiung Hung. *Torque optimization schemes for kinematically redundant manipulators*. *Journal of Robotic Systems*, vol. 11, no. 4, pages 257–269, 1994.
- [Chen 2013] Lisha Chen, Manolo Garabini, Matteo Laffranchi, Navvab Kashiri, Nikos G Tsagarakis, Antonio Bicchi and Darwin G Caldwell. *Optimal control for maximizing velocity of the CompAct compliant actuator*. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 516–522. IEEE, 2013.
- [De Luca 2006] Alessandro De Luca, Alin Albu-Schäffer, Sami Haddadin and Gerd Hirzinger. *Collision detection and safe reaction with the DLR-III lightweight manipulator arm*. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1623–1630, 2006.
- [De Luca 2008] Alessandro De Luca and Lorenzo Ferrajoli. *Exploiting robot redundancy in collision detection and reaction*. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3299–3305, 2008.
- [De Luca 2012] Alessandro De Luca and Fabrizio Flacco. *Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration*. In *Proceedings of the 4th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 288–295. IEEE, 2012.
- [Decré 2009] Wilm Decré, Ruben Smits, Herman Bruyninckx and Joris De Schutter. *Extending iTaSC to support inequality constraints and non-instantaneous task specification*. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 964–971. IEEE, 2009.
- [Del Prete 2016] Andrea Del Prete and Nicolas Mansard. *Robustness to joint-torque-tracking errors in task-space inverse dynamics*. *IEEE Transactions on Robotics*, vol. 32, no. 5, pages 1091–1105, 2016.
- [Ebert 2002] Dirk M Ebert and Dominik D Henrich. *Safe human-robot-cooperation: Image-based collision detection for industrial robots*. In *Proceedings of the IEEE/RSJ International Conference On Intelligent Robots and Systems*, pages 1826–1831, 2002.
- [EMIKA 2017] Franka EMIKA. *Franka*. <https://www.franka.de/>, Accessed: 2017.
- [Ferraguti 2013] Federica Ferraguti, Cristian Secchi and Cesare Fantuzzi. *A tank-based approach to impedance control with variable stiffness*. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4948–4953. IEEE, 2013.

- [Ficuciello 2015] Fanny Ficuciello, Luigi Villani and Bruno Siciliano. *Variable impedance control of redundant manipulators for intuitive human-robot physical interaction*. IEEE Transactions on Robotics, vol. 31, no. 4, pages 850–863, 2015.
- [Flacco 2012] Fabrizio Flacco, Torsten Kröger, Alessandro De Luca and Oussama Khatib. *A depth space approach to human-robot collision avoidance*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 338–345. IEEE, 2012.
- [Flacco 2015] Fabrizio Flacco, Alessandro De Luca and Oussama Khatib. *Control of redundant robots under hard joint constraints: Saturation in the null space*. IEEE Transactions on Robotics, vol. 31, no. 3, pages 637–654, 2015.
- [Fraichard 2007] Thierry Fraichard. *A short paper about motion safety*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 1140–1145. IEEE, 2007.
- [Fryman 2014] Jeff Fryman. *Updating the Industrial Robot Safety Standard*. In Proceedings of the 41st International Symposium on Robotics, pages 1–4, 2014.
- [Garabini 2011] Manolo Garabini, Andrea Passaglia, Felipe Belo, Paolo Salaris and Antonio Bicchi. *Optimality principles in variable stiffness control: The usa hammer*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3770–3775. IEEE, 2011.
- [Grebenstein 2011] M. Grebenstein, A. Albu-Schäffer, T. Bahls, M. Chalon, O. Eiberger, W. Friedl, R. Gruber, S. Haddadin, U. Hagn, R. Haslinger, H. Hppner, S. Jrg, M. Nickl, A. Nothhelfer, F. Petit, J. Reill, N. Seitz, T. Wimbeck, S. Wolf, T. Wsthoff and G. Hirzinger. *The DLR hand arm system*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 3175–3182. IEEE, 2011.
- [Gurobi Optimization Inc. 2015] Gurobi Optimization Inc. *Gurobi Optimizer Reference Manual*, 2015.
- [Haddadin 2008] Sami Haddadin, Alin Albu-Schäffer, Alessandro De Luca and Gerd Hirzinger. *Collision detection and reaction: A contribution to safe physical human-robot interaction*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3356–3363, 2008.
- [Haddadin 2012a] S. Haddadin, A. Houry, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi and A. Albu-Schäffer. *A truly safely moving robot has to know what injury it may cause*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 5406–5413, 2012.
- [Haddadin 2012b] Sami Haddadin, Nico Mansfeld and A Albu-Schäffer. *Rigid vs. elastic actuation: Requirements & performance*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 5097–5104. IEEE, 2012.
- [Heinzmann 2003] Jochen Heinzmann and Alexander Zelinsky. *Quantitative safety guarantees for physical human-robot interaction*. The International Journal of Robotics Research, vol. 22, no. 7-8, pages 479–504, 2003.
- [Hirzinger 2001] Gerd Hirzinger, A Albu-Schäffer, M Hahnle, Ingo Schaefer and Norbert Sporer. *On a new generation of torque controlled light-weight robots*. In Proceedings of the IEEE International Conference on Robotics and Automation, volume 4, pages 3356–3363, 2001.

- [Hoarau 2017] A. Hoarau, J. DaSilva, A. Ibanez and V. Padois. *rtt-kuka-lwr: Control software architecture for lightweight robots*. [https://www.github.com/kuka-isir/rtt\\_lwr](https://www.github.com/kuka-isir/rtt_lwr), Accessed: 2017.
- [Hogan 1984] Neville Hogan. *Impedance control: An approach to manipulation*. In Proceedings of the American Control Conference, pages 304–313. IEEE, 1984.
- [Hyon 2007] Sang-Ho Hyon, Joshua G Hale and Gordon Cheng. *Full-body compliant human-humanoid interaction: balancing in the presence of unknown external forces*. IEEE Transactions on Robotics, vol. 23, no. 5, pages 884–898, 2007.
- [Ibanez 2015] Aurélien Ibanez. *Emergence of complex behaviors from coordinated predictive control in humanoid robotics*. PhD thesis, Université Pierre et Marie Curie Paris VI, September 2015.
- [Ikuta 2001] K Ikuta, M Nokata and H Ishii. *Safety evaluation method of human-care robot control and special robot simulator. Integration of Assistive Technology in the Information Age*, 2001.
- [Ikuta 2003] Koji Ikuta, Hideki Ishii and Makoto Nokata. *Safety evaluation method of design and control for human-care robots*. The International Journal of Robotics Research, vol. 22, no. 5, pages 281–297, 2003.
- [ISO 2011a] ISO. *Robots and Robotic Devices-Safety Requirements for Industrial Robots-Part1: Robots*. Iso, International Organization for Standardization, Geneva, CH, 2011.
- [ISO 2011b] ISO. *Robots and Robotic Devices-Safety Requirements for Industrial Robots-Part2: Robot systems and integration*. Iso, International Organization for Standardization, Geneva, CH, 2011.
- [ISO 2016] ISO. *robots and robotic devices - collaborative robots*. Iso, International Organization for Standardization, Geneva, CH, 2016.
- [Ivar 2011] Thorson Ivar and D Caldwell. *A Nonlinear Series Elastic Actuator for Highly Dynamic Motions*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 390–394, 2011.
- [KaewTraKulPong 2002] Pakorn KaewTraKulPong and Richard Bowden. *An improved adaptive background mixture model for real-time tracking with shadow detection*. In Video-based surveillance systems, pages 135–144. Springer, 2002.
- [Kashiri 2013] Navvab Kashiri, Nikos G Tsagarakis, Matteo Laffranchi and Darwin G Caldwell. *On the stiffness design of intrinsic compliant manipulators*. In Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pages 1306–1311. IEEE, 2013.
- [Katzschmann 2013] Robert Katzschmann, Torsten Kröger, Tamim Asfour and Oussama Khatib. *Towards online trajectory generation considering robot dynamics and torque limits*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 5644–5651. IEEE, 2013.
- [Khatib 1987] Oussama Khatib. *A unified approach for motion and force control of robot manipulators: The operational space formulation*. IEEE Journal on Robotics and Automation, vol. 3, no. 1, pages 43–53, 1987.

- [Khatib 1995] Oussama Khatib. *Inertial properties in robotic manipulation: An object-level framework*. The International Journal of Robotics Research, vol. 14, no. 1, pages 19–36, 1995.
- [KUKA 2017] KUKA. *LBR iiwa*. <https://www.kuka.com/en-de/products/robot-systems/industrial-robots/lbr-iiwa>, Accessed: 2017.
- [Kulic 2007] Dana Kulic and Elizabeth A Croft. *Affective state estimation for human–robot interaction*. IEEE Transactions on Robotics, vol. 23, no. 5, pages 991–1000, 2007.
- [Laffranchi 2011] Matteo Laffranchi, Nikos Tsagarakis and Darwin G Caldwell. *A compact compliant actuator (CompAct) with variable physical damping*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 4644–4650. IEEE, 2011.
- [Lange 2014] Friedrich Lange and Michael Suppa. *Predictive path-accurate scaling of a sensor-based defined trajectory*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 754–759. IEEE, 2014.
- [Lange 2015] Friedrich Lange and Michael Suppa. *Trajectory generation for immediate path-accurate jerk-limited stopping of industrial robots*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 2021–2026. IEEE, 2015.
- [Loughlin 2007] Clive Loughlin, A Albu-Schäffer, S Haddadin, Ch Ott, A Stemmer, T Wimböck and G Hirzinger. *The DLR lightweight robot: design and control concepts for robots in human environments*. Industrial Robot, vol. 34, no. 5, pages 376–385, 2007.
- [Lumelsky 1993] Vladimir J Lumelsky and Edward Cheung. *Real-time collision avoidance in teleoperated whole-sensitive robot arm manipulators*. Transactions on Systems, Man and Cybernetics, vol. 23, no. 1, pages 194–203, 1993.
- [Ma 2002] Shugen Ma and Mitsuru Watanabe. *Time-optimal control of kinematically redundant manipulators with limit heat characteristics of actuators*. Advanced Robotics, vol. 16, no. 8, pages 735–749, 2002.
- [Maplesoft Inc. 2016] Maplesoft Inc. *Maple 18*, 2016.
- [Marx 2017] et al. Marx. *RTnet: Hard Real-Time Networking for Real-Time Linux*. <http://rtnet.org>, Accessed: 2017.
- [Mason 1981] Matthew T Mason. *Compliance and force control for computer controlled manipulators*. IEEE Transactions on Systems, Man, and Cybernetics, vol. 11, no. 6, pages 418–432, 1981.
- [Mavridis 2015] Nikolaos Mavridis. *A review of verbal and non-verbal human–robot interactive communication*. Robotics and Autonomous Systems, vol. 63, pages 22–35, 2015.
- [Meguenani 2016a] A. Meguenani, V. Padois, J. Da Silva, A. Hoarau and P. Bidaud. *KUKA: HR interaction with constraints on Ec and Ep*. [http://pages.isir.upmc.fr/~padois/website/fichiers/videos/Kuka\\_Human\\_interaction\\_constraints\\_on\\_Ec\\_and\\_Ep.mp4](http://pages.isir.upmc.fr/~padois/website/fichiers/videos/Kuka_Human_interaction_constraints_on_Ec_and_Ep.mp4), 2016.



- [Meguenani 2016b] A. Meguenani, V. Padois, J. Da Silva, A. Hoarau and P. Bidaud. *KUKA: HR interaction with no constraints*. [http://pages.isir.upmc.fr/~padois/website/fichiers/videos/Kuka\\_Human\\_interaction\\_no\\_constraints.mp4](http://pages.isir.upmc.fr/~padois/website/fichiers/videos/Kuka_Human_interaction_no_constraints.mp4), 2016.
- [Merlhiot 2012] X. Merlhiot, J.L. Garrec, G. Saupin and C. Andriot. *The XDE mechanical kernel: efficient and robust simulation of multibody dynamics with intermittent nonsmooth contacts*. In Proceedings of the International Conference on Multibody System Dynamics, Stuttgart, Germany, 2012.
- [Ngo 2005] Khoi B Ngo and Robert Mahony. *Passivity-based control of robot manipulators subject to constraints*. In Proceedings of the Australian Conference on Robotics and Automation, 2005.
- [Ngo 2006] Khoi B Ngo and Robert Mahony. *Bounded torque control for robot manipulators subject to joint velocity constraints*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 7–12. IEEE, 2006.
- [Ott 2008a] Christian Ott. *Cartesian impedance control of redundant and flexible-joint robots*. Springer, 2008.
- [Ott 2008b] Christian Ott, Alin Albu-Schaffer, Andreas Kugi and Gerd Hirzinger. *On the passivity-based impedance control of flexible joint robots*. IEEE Transactions on Robotics, vol. 24, no. 2, pages 416–429, 2008.
- [Ott 2010] Christian Ott, Ranjan Mukherjee and Yoshihiko Nakamura. *Unified impedance and admittance control*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 554–561. IEEE, 2010.
- [Padois 2016] V. Padois. *Control and design of robots with tasks and constraints in mind*. PhD thesis, Universit Pierre et Marie Curie (Paris 6), Paris, France, oct 2016.
- [Park 1998] Ki Cheol Park, Pyung Hun Chang and Seung Ho Kim. *The enhanced compact QP method for redundant manipulators using practical inequality constraints*. In Proceedings of the IEEE International Conference on Robotics and Automation, volume 1, pages 107–114. IEEE, 1998.
- [Pfeifer 2006] Rolf Pfeifer and Josh Bongard. *How the body shapes the way we think: a new view of intelligence*. MIT press, 2006.
- [Pratt 1995] Gill A Pratt and Matthew M Williamson. *Series elastic actuators*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, volume 1, pages 399–406. IEEE, 1995.
- [Raibert 1981] Marc H Raibert and John J Craig. *Hybrid position/force control of manipulators*. Journal of Dynamic Systems, Measurement, and Control, vol. 102, no. 127, pages 126–133, 1981.
- [Robotics 2017a] Fogale Robotics. <http://www.fogale-robotics.com>, Accessed: 2017.
- [Robotics 2017b] Rethink Robotics. *Sawyer*. <http://www.rethinkrobotics.com/sawyer/>, Accessed: 2017.
- [ROBOTS 2017] UNIVERSAL ROBOTS. *UR10*. <https://www.universal-robots.com/products/ur10-robot/>, Accessed: 2017.

- [Rubrecht 2010] Sébastien Rubrecht, Vincent Padois, Philippe Bidaud and Michel De Broissia. *Constraints Compliant Control: constraints compatibility and the displaced configuration approach*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 677–684. IEEE, 2010.
- [Rubrecht 2011] S. Rubrecht. *Contributions to the control of constrained robots*. Theses, Université Pierre et Marie Curie - Paris VI, September 2011.
- [Rusu 2011] RB. Rusu and S. Cousins. *3D is here: Point Cloud Library (PCL)*. In Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, May 9-13 2011.
- [Salisbury 1991] Kenneth Salisbury, Brian Eberman, Michael Levin and William Townsend. *The design and control of an experimental whole-arm manipulator*. In Proceedings of the 5th International Symposium on Robotics Research, pages 233–241. MIT Press, 1991.
- [Schindlbeck 2015] Christopher Schindlbeck and Sami Haddadin. *Unified passivity-based cartesian force/impedance control for rigid and flexible joint robots via task-energy tanks*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 440–447. IEEE, 2015.
- [Schreiber 2010] Günter Schreiber, Andreas Stemmer and Rainer Bischoff. *The fast research interface for the kuka lightweight robot*. In Proceedings of the IEEE International Conference on Robotics and Automation, 2010.
- [Siciliano 2008] B. Siciliano and O. Khatib. [Springer handbook of robotics](#). Springer Science & Business Media, 2008.
- [Siciliano 2016] Bruno Siciliano and Oussama Khatib. Springer handbook of robotics. Springer, 2016.
- [Soetens 2017] P. Soetens. *RTT: Real-Time Toolkit*. <http://www.orocos.org/rtt>, Accessed: 2017.
- [Tassa 2012] Yuval Tassa, Tom Erez and Emanuel Todorov. *Synthesis and stabilization of complex behaviors through online trajectory optimization*. In Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, pages 4906–4913. IEEE, 2012.
- [Townsend 1993] William T Townsend and J Kenneth Salisbury. *Mechanical design for whole-arm manipulation*. Robots and Biological Systems: Towards a New Bionics?, pages 153–164, 1993.
- [Tsagarakis 2009] Nikolaos G Tsagarakis, Matteo Laffranchi, Bram Vanderborght and Darwin G Caldwell. *A compact soft actuator unit for small scale human friendly robots*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 4356–4362. IEEE, 2009.
- [Vanderborght 2012] Bram Vanderborght, Alin Albu-Schäffer, Antonio Bicchi, Etienne Burdet, D Caldwell, Raffaella Carloni, M Catalano, G Ganesh, Manolo Garabini, Markus Grebenstein, G. Grioli, S. Haddadin, A. Jafari, M. Laffranchi, D. Lefeber, F. Petit, S. Stramigioli, N. Tsagarakis, M. Van Damme, R. Van Ham, L. C. Visser and S. Wolf. *Variable impedance actuators: Moving the robots of tomorrow*. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 5454–5455. IEEE, 2012.

- [Vanderborght 2013] B. Vanderborght, A. Albu-Schäffer, A. Bicchi, E. Burdet, D. G. Caldwell, R. Carloni, M. Catalano, O. Eiberger, W. Friedl, G Ganesh, M. Garabini, M. Grebenstein, G. Grioli, S. Haddadin, H. Hoppner, A. Jafari, M. Laffranchi, D. Lefeber, F. Petit, S. Stramigioli, N. Tsagarakis, M. Van Damme, R. Van Ham, L.C. Visser and S. Wolf. *Variable impedance actuators: A review*. Robotics and autonomous systems, vol. 61, no. 12, pages 1601–1614, 2013.
- [VIA 2017] <http://www.viactors.org>, Accessed: 2017.
- [Whitney 1977] Daniel E Whitney. *Force feedback control of manipulator fine motions*. Journal of Dynamic Systems, Measurement, and Control, vol. 99, no. 2, pages 91–97, 1977.
- [Wieber 2006] Pierre-Brice Wieber. *Trajectory free linear model predictive control for stable walking in the presence of strong perturbations*. In Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pages 137–142. IEEE, 2006.
- [Wolf 2011] Sebastian Wolf, Oliver Eiberger and Gerd Hirzinger. *The DLR FSJ: Energy based design of a variable stiffness joint*. In Proceedings of the IEEE International Conference on Robotics and Automation, pages 5082–5089. IEEE, 2011.
- [Yoshikawa 1985] Tsuneo Yoshikawa. *Manipulability of robotic mechanisms*. The international journal of Robotics Research, vol. 4, no. 2, pages 3–9, 1985.
- [Zinn 2004] Michael Zinn, Oussama Khatib, Bernard Roth and J Kenneth Salisbury. *Playing it safe [human-friendly robots]*. Robotics & Automation Magazine, IEEE, vol. 11, no. 2, pages 12–21, 2004.
- [Zollo 2005] Loredana Zollo, Bruno Siciliano, Alessandro De Luca, Eugenio Guglielmelli and Paolo Dario. *Compliance control for an anthropomorphic robot with elastic joints: Theory and experiments*. Journal of dynamic systems, measurement, and control, vol. 127, no. 3, pages 321–328, 2005.