Handling uncertainty and variability in robot control

Nirmal Giftsun

To cite this version:

Nirmal Giftsun. Handling uncertainty and variability in robot control. Automatic. INSA de Toulouse, 2017. English. NNT : 2017ISAT0028 . tel-02010919v2

HAL Id: tel-02010919
https://laas.hal.science/tel-02010919v2

Submitted on 20 Feb 2018

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.

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ FÉDÉRALE TOULOUSE MIDI-PYRÉNÉES

Délivré par :
l'Institut National des Sciences Appliquées de Toulouse (INSA de Toulouse)

Présentée et soutenue le *13/12/2017* **par :**

NIRMAL GIFTSUN

Handling Uncertainty and Variability in Robot Control

JURY

VINCENT PADOIS	Maître de Conférences	Rapporteur
PAUL PLOEGER	Professeur d'Université	Rapporteur
FLORENT LAMIRAUX	Directeur de Recherche	Directeur de thèse
ANDREA DEL PRETE	Chercheur Postdoctoral	Codirecteur de thèse

École doctorale et spécialité :
EDSYS : Robotique 4200046
Unité de Recherche :
Laboratoire d'analyse et d'architecture des systèmes(LAAS)
Directeur(s) de Thèse :
Florent LAMIRAUX et *Andrea DEL PRETE*
Rapporteurs :
Vincent PADOIS et *Paul PLOEGER*

Abstract

Amidst a lot of research in motion planning and control in concern with robotic applications, the mankind has never reached a point yet, where the robots are perfectly functional and autonomous in dynamic settings. Though it is controversial to discuss about the necessity of such robots, it is very important to address the issues that stop us from achieving such a level of autonomy. Industrial robots have evolved to be very reliable and highly productive with more than 1.5 million operational robots in a variety of industries. These robots work in static settings and they literally do what they are programmed for specific usecases, though the robots are flexible enough to be programmed for a variety of tasks. This research work makes an attempt to address these issues that separate both these settings in a profound way with special focus on uncertainties. Practical impossibilities of precise sensing abilities lead to a variety of uncertainties in scenarios where the robot is mobile or the environment is dynamic. This work focuses on developing smart strategies to improve the ability to handle uncertainties robustly in humanoid and industrial robots. First, we focus on a dynamical obstacle avoidance framework proposed for industrial robots equipped with skin sensors for reactivity. Path planning and motion control are usually formalized as separate problems in robotics. High dimensional configuration spaces, changing environment and uncertainties do not allow to plan real-time motion ahead of time requiring a controller to execute the planned trajectory. The fundamental inability to unify both these problems has led to handle the planned trajectory amidst perturbations and unforeseen obstacles using various trajectory execution and deformation mechanisms. The proposed framework uses 'Stack of Tasks', a hierarchical controller using proximity information to avoid obstacles. Experiments are performed on a UR5 robot to check the validity of the framework and its potential use for collaborative robot applications.

Second, we focus on a strategy to model inertial parameters uncertainties in a balance controller for legged robots. Model-based control has become more and more popular in the legged robots community in the last ten years. The key idea is to exploit a model of the system to compute precise motor commands that result in the desired motion. This allows to improve the quality of the motion tracking, while using lower feedback gains, leading so to higher compliance. However, the main flaw of this approach is typically its lack of robustness to modeling errors. In this paper we focus on the robustness of inverse-dynamics control to errors in the inertial parameters of the robot. We assume these parameters to be known, but only with a certain accuracy. We then propose a computationally-efficient optimization-based controller that ensures the balance of the robot despite these uncertainties. We used the proposed controller in simulation to perform different reaching tasks with the HRP-2 humanoid robot, in the presence of various modeling errors. Comparisons against a standard inverse-dynamics controller through hundreds of simulations show the superiority of the proposed controller in ensuring the robot balance.

Acknowledgments

In the first place, I thank Florent Lamiraux for giving me a wonderful opportunity to do my Ph.d. in LAAS-CNRS, one of the most renowned robotics institutes in the world. The first contribution of this thesis is specific within the realms of the Factory-in-a-day project focusing on improving the efficiency in deployment of industrial robots. Florent's experience and knowledge combined with his flexibility to let me free to solve problems in my own way, allowed me to become a very confident and an independent researcher & developer. His relaxed nature and his quickness in allocating the necessary resources to carry out my development and experiments in remote locations, helped me to succeed in my Ph.d and present an interesting demonstration of the work in the final project meeting. Secondly, I thank Andrea Del Prete for joining me in the middle of my Ph.d. to help me focus on another interesting problem in humanoid robotics. Andrea's competence in optimization based inverse dynamics control and his maturity in mentoring gave way to an interesting contribution with a good scope for future work. Also Andrea's highly critical support in writing this manuscript is undoubtedly phenomenal.

I also thank the jury members, Paul Ploeger and Vincent Padois for showing sincere interest in my work by giving appropriate feedback and suggestions to orient my future work. The experiments in the first contribution were done in TU-Delft under the support of Mukunda Bharatheesa, Carlos Hernández Corbato and Martijn Wisse. I thank them all for their wonderful support in making this demonstration a success. I sincerely thank Olivier Stasse and Nicolas Mansard for their kind support in handling technical issues during the development of my first contribution. A special thanks to Philippe Souères for his motivation at several occassions during my stay in LAAS-CNRS.

I thank all the former and current team members of the Gepetto team who provided a friendly environment during my Ph.d. I must say that, I pretty much understood what needs to be a great researcher after my precious time in LAAS-CNRS. Finally, I thank the entire laboratory for helping me develop a strong and anti-fragile attitude, with which I'm going to do a lot of wonders in future and everybody mentioned here deserves a genuine credit.

Contents

List of Figures

Notation Table

Table 1: Notations used in the manuscript

Term	Meaning
CoP	Center of Pressure
CoM	Center of Mass
ZMP	Zero Moment Point
CP	Capture Point
DoF	Degrees of Freedom
IK	Inverse Kinematics
ID	Inverse Dynamics
TSID	Task Space Inverse Dynamics
SoT	Stack of Tasks
$SE(3)$	Special Euclidean Group
$SO(3)$	Special Orthogonal Group
HQP	Hierarchical Quadratic Program
QP	Quadratic Program

Chapter 1
Introduction

This chapter introduces the fields of industrial robots and humanoid robots to motivate the work presented in this thesis. Industrial robots operate in structured environments, where they can be highly productive for laborious industrial tasks. However, when collaborating with human beings, these robots have to move conservatively to ensure safety. This chapter discusses this issue in detail, presenting the existing approaches, and their current limitations. Another key issue in robotics is the lack of robustness of current balance controllers for legged robots. Classic balance controllers rely on a precise estimation of the robot center of mass position and velocity. However, accurate models of legged robots are extremely hard to identify in practice, due to the complexity of these machines. We discuss the need for robustness, and the current approaches to tackle this issue, in order to provide the context of our contribution on robust balance control.

1.1 Overview of the Work

Autonomy is undoubtedly the main goal in robotics research. We wish to develop robots that are self-reliant, and we wish to do it through a technology that is as generic as possible. Every incremental research in robotics directly or indirectly strives to automatize processes or mechanisms of interest, efficiently and effectively. Though the need of advanced autonomy in robots is a controversial subject, it is undeniable that robotics researchers try to solve problems using generic and self-reliant algorithms. Autonomy is not only about higher awareness and control in structured environments, but also about the ability to cope with unknown and dynamic environments. Developing algorithms with an ability to change, adapt or be flexible is one of the core problems to achieve autonomy in robots.

In this thesis, we address two specific control problems, aiming to advance autonomy in robots: environment *variability*, and *uncertainty* in the robot model. With *variability*, we refer to situations in which algorithms need to adapt online, but without compromising the final goal. Instead by *uncertainty*, we refer to the incomplete knowledge of qualitative or quantitative parameters of a robot model, requiring the control algorithm to achieve given goals despite this lack of knowledge. Though these two terms look similar, and they may even mean the same thing in some cases, they are fundamentally different in general, and need to be handled differently. We have addressed these two problems using two different robotic platforms, motivated by the current essential need in robotics research and development.

The first part of the thesis (Chapter 2) presents a control framework to handle environment variability in industrial robots. Classic industrial robots work in static settings and repeatedly execute the action programmed by the users. Though these robots are flexible enough to execute a large variety of tasks, their control algorithms lack any adaptation capability. This work attempts to address this issue, focusing on dynamic obstacle avoidance, i.e. when the robot encounters unforeseen obstacles while executing a task. The second part of the thesis (Chapter 3) presents a balance controller for legged robots that can cope with imperfect knowledge of the robot inertial parameters. Legged robots are subject to a variety of uncertainties, both in the robot model and in the environment model. Due to the complexity of their complete dynamic model, reduced/approximated models are often used for balance control. Unfortunately, these model-based control approaches are sensitive to model uncertainties. To tackle this issue, we propose a robust control strategy to improve balance control in the presence of inertial parameter uncertainty.

In a nutshell, this thesis focuses on developing intelligent strategies to handle variability and uncertainties in industrial and humanoid robots, respectively. The two main contributions of this thesis are:

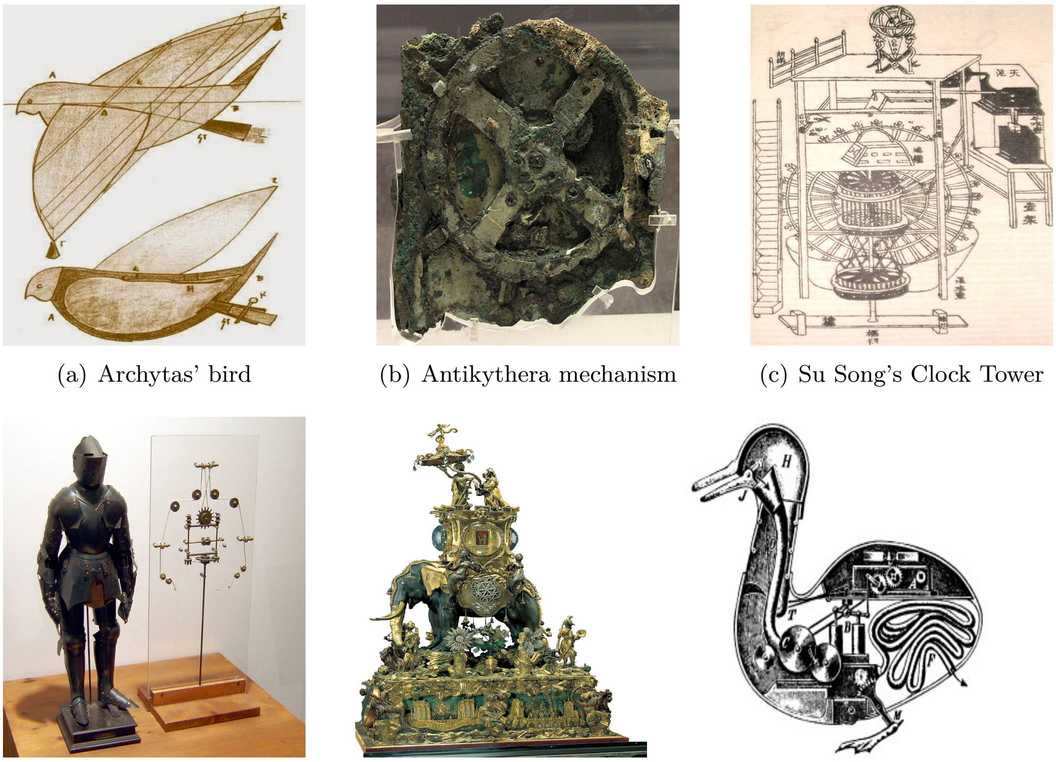
- A dynamic obstacle avoidance control framework for industrial robots equipped with skin proximity sensors.
- A control strategy for balancing humanoid robots that is robust to bounded uncertainties in their inertial parameters.

1.2 Towards Collaborative Robots

This section briefly discusses the evolution of industrial robotics to stress the need for collaborative robots. We also discuss the current state of the industry and its technological limitations, establishing the context of the thesis.

1.2.1 Evolution of Industrial Robotics

The idea of automatic machines has been in existence since many centuries, with documented illustrations in various interesting applications [Berryman 2003]. Fig. 1.1 shows snapshots of some important automatons known to be popular during ancient times. Archytas (4th century B.C.), the founder of mathematical mechanics, invented an autonomous wooden flying pigeon (Fig. 1.1(a)) which can fly uninterrupted for hundreds of meters and is considered to be the first robot ever documented in the history. Fig. 1.1(b) shows an analog device (developed during 1st century B.C.) that can predict astronomical positions for the purpose of maintaining a calendar and astrological reasons, making it one of the most ancient computers ever developed. The first ever human-like robot in the records was designed by Leonardo da Vinci, and is shown in Fig. 1.1(d). It has joints similar to human beings, with an ability to perform motions such as waving its arms, moving its head and jaws, sitting and standing up. Many automatons developed were capable of



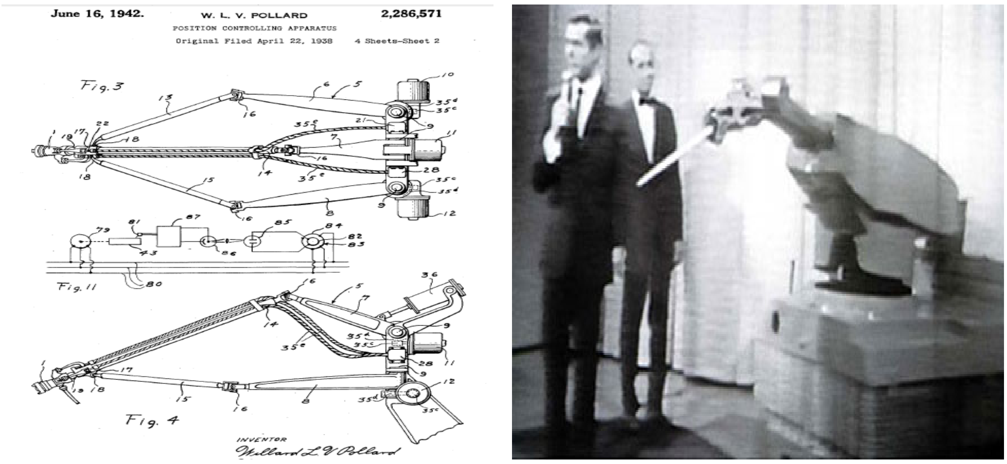
(a) Archytas' bird (b) Antikythera mechanism (c) Su Song's Clock Tower

(d) Da Vinci's robot (e) Elephant Automaton (f) Digesting Duck

Figure 1.1: Automatons in the Past

entertaining, speaking and playing musical instruments. A musical automaton in the form of an elephant is shown in Fig. 1.1(e); it was designed by Hubert Martinet, a French clock-maker, in 1774. Another musical automaton that has mechanically controlled mannequins chiming and ringing bells in an artistic clock tower, developed by Su Song, is shown in Fig. 1.1(c). A funny golden mechanical duck (see Fig. 1.1(f)) is known for imitating the ability to eat and defecate grains, trying to illustrate metabolic capabilities.

Though most of the automatons were created for entertainment or artistic satisfaction in the beginning, the characterization of robots to be human look-alike machines that can serve human beings came up in the 20th century. The term *robot* itself was born in 1921 from R.U.R. (Rossum's Universal Robots), a Czech play written by Karel Čapek [Hockstein 2007]. The play depicts robots as good and benevolent workers serving the human beings in the beginning, but later gaining super human strength to revolt against humans, leading to the end of life. This negative idea changed after a Russian writer, Isaac Asimov, made a contrasting characterization of robots as just mechanical creatures with no emotions. Asimov's laws of robotics in 1942 gave way to a new perspective of robots to be seen as a product that could be developed by engineers to improve productivity in manufacturing industries. During the same time period, a mechanism to do spray painting was



(a) Pollard's Paint Sprayer (b) Unimate Robot's Television Appearance

Figure 1.2: Robotic Mechanisms in the Beginning

designed by Pollard and Roselund (see Fig. 1.2(a) showing the first spray painter design), pioneering the first programmable device in history [Wallén 2008].

The first truly programmable robot was *Unimate*, which consisted of an arm and a drum memory box with pre-programmed tasks. *Unimation*, the first company to make robots that were used to transport and weld die castings on automobile bodies, sparing humans dangerous working conditions (Fig. 1.2(b) shows the first television appearance of an *Unimate* robot). Though the numerically controlled turning and milling machines and the hydraulic assembly machines were programmable, the industrial robots differed in the sophistication of re-programmability and the versatility to be used for different tasks. This is purely because of the invention of digital computers and integrated circuit technology, which allowed to develop the brains of industrial robots. Also these robots have more than 3 DoF providing more flexibility in the workspace. Ford's interest to install *Unimation* robots triggered American manufacturing industries to pay attention to the robotics industry a bit more seriously. Installations in General Motors in Ohio (US) in the beginning of the 1960s marks the real beginning of industrial robotics. After an intense research and development during the next 15 years, the introduction of micro-processors provided the basis for low cost control systems. A Norwegian company, *Trallfa*, designed and developed a cost-effective alternative to *Unimate* robots for spray painting applications. Several companies such as *Electrolux, ESAB, Atlas Copco,* and *ASEA* followed the same path designing in-house robots for their own purposes, which suited the requirements of other customers resulting in a product of its own. This phenomenon gave birth to more than 70 robot manufacturers by the mid 1970s. In the beginning, industrial robots were hydraulic or pneumatic, though they are very suitable for heavier loads. Vicarm turned out to be the first electric robot to suit the lighter loads of assembly lines and arc welding **??**. The 6-degree-of-freedom robot was designed with simplified analytic solutions by Victor Scheinman, allowing the

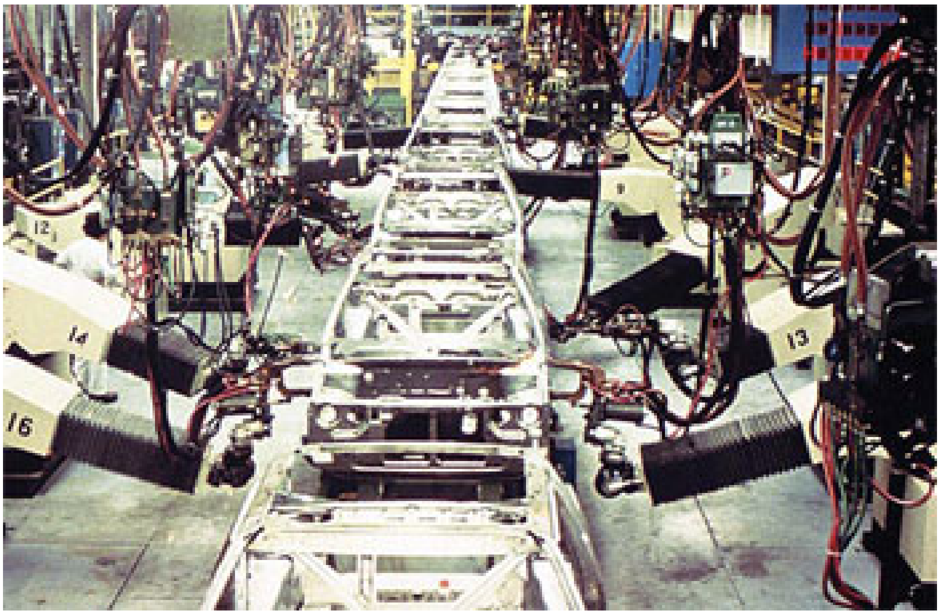
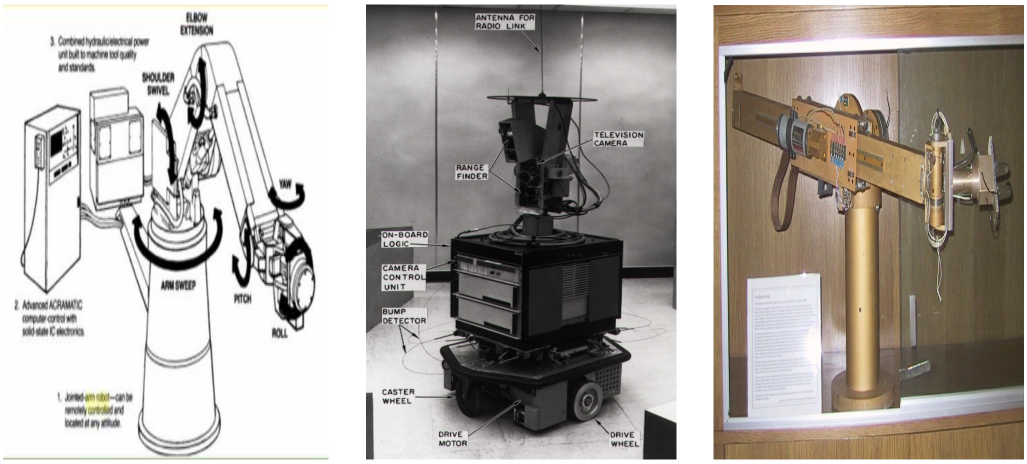


Figure 1.3: Manufacturing unit in General Motors with 'Unimate' robots in 1969

robot to track arbitrary paths in its work space. Cincinnati Milacron, the largest machine tool constructor during the 1970s, developed "The Tomorrow Tool", the first microcomputer-based robot shown in Fig. 1.4.

In the beginning robots were used for simple tasks, such as pick and place with no external sensing. External sensing along with the ability of robots to perform advanced motion behaviors gave rise to complex applications like welding, grinding and deburring. Robots were deployed in three major areas: assembly lines, process operations and material handling. The main motivation of industrial robotics is to apply productive, cost-effective and safe automation solutions, without compromising the quality of the products. The capabilities of robots were purely driven by the manufacturing industries, with different industries focusing on different requirements. Material handling required robots of increased loading capacity, while arc welding and motion dependent applications required the robot to have better electrical motors and path control. In the beginning of 1980s the assembly lines were mostly focused on achieving shorter cycle time, which required robots to be highly dynamic and repeatable. Metal industries required robots to be very stubborn to operate in hot and unsafe working environments. Though robots had complex applications, simple applications such as pick-and-place, or material transportation were economically advantageous for automatization. These customer demands provided way to the industrial robotic revolution in the 80's [Wallén 2008].

Robotics was unanimously accepted as a key focus area to increase industrial development and achieve competitive edge. Advanced sensors such as force sensors, vision cameras and laser scanners were introduced in the late 1980s to allow physical interaction with the environment and to improve the system "intelligence". In the


ELBOW EXTENSION
3. Combined hydraulic/electrical power unit built to machine tool quality and standards.
SHOULDER SWIVEL
YAW
ARM SWEEP
PITCH
ROLL
2. Advanced ACRAMATIC computer-control with solid-state IC electronics.
1. Jointed-arm robot—can be remotely controlled and located at any attitude.
ANTENNA FOR RADIO LINK
TELEVISION CAMERA
RANGE FINDER
ON-BOARD LOGIC
CAMERA CONTROL UNIT
BUMP DETECTOR
CASTER WHEEL
DRIVE MOTOR
DRIVE WHEEL

(a) The Tomorrow Tool-T3 (b) Shakey (c) Stanford Arm

Figure 1.4: Robots in the 1970s

1980s, the ambitious vision of robotics was to completely automatize factories with robots. However, it was soon realized that certain human work mechanisms were difficult to be replicated. Heterogenous integration of complex systems involves a lot of problems and robotic workcells were more expensive than the workers, though they were economically beneficial for simpler tasks. Productivity is a complex concept, and the idea of robots solely responsible for improving productivity changed in the beginning of 1990s. The dependence on robotics for more ambitious tasks started to decrease, though it was and is still an inevitable part of the mechatronic technology to automize and improve productivity. Now robotics is used for medical applications, service, entertainment and disaster recovery. Though industrial robots have been in use for quite a long time, many challenges have not been completely addressed yet. This is where the 'Factory in a Day' project comes in to the picture. Let us have a look at the project and its goals, to position the first part of this thesis, before continuing the discussion on the development of *Collaborative Robots* creating safe, shared and inexpensive work spaces.

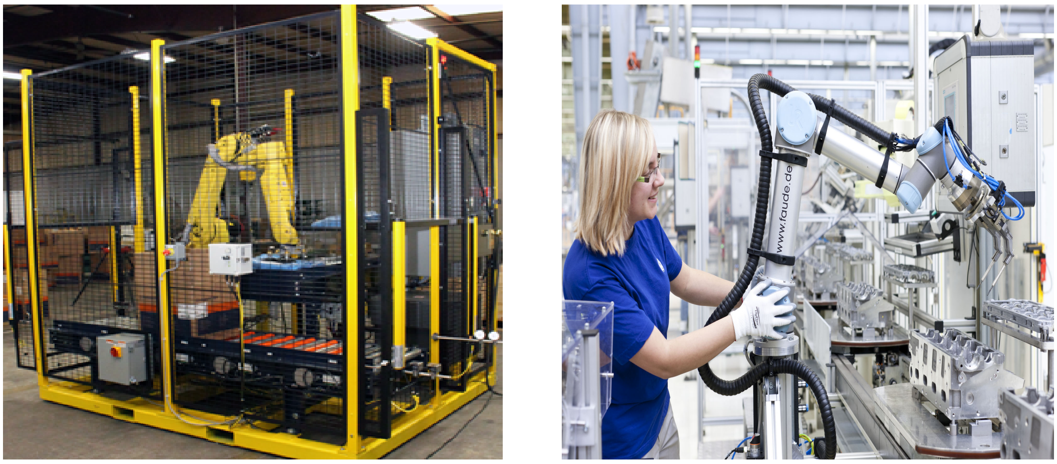
1.2.2 The 'Factory in a Day' EU project

We are aware that robot automation has been into existence since the 1960s and has seen a lot of technological advancements, but it is still challenged by the time and cost needed to set up robots specific to the functional needs of the factories. There is a lot of risk involved in such investments, making them economically less attractive to smaller companies. Moreover, the factory setups are fixed environments with hard coded settings of millimeter precision, so that they are perfectly under control at all times. In contrast to fixed robot arms, mobile robots provide kinematic flexibility to handle multiple tasks, thanks to the proper use of extra base degrees of freedom. Though there exist state-of-the-art controllers that can handle redundancy, mobile manipulation is difficult because of the odometry errors and uncertainties introduced by the wheels in the base. Another critical concern is safety for human beings working in robot environments. Though recent technologies are focused towards collaborative robot control, where human beings and robots can work together, these algorithms are either not mature enough to be used in factories, or not yet known in the industrial community.

The Factory-in-a-Day project puts forward the idea of reducing the installation time (and the associated costs) from several months to just a single day. The project focuses on the following aspects of robotics, which are the steps taken to install a robotic setup in a day [Wisse 2013a].

- Standardized procedures to design 3D printed custom parts, which are usually attached to an existing robot arm and grippers using novel templates to minimize the time taken.
- The flexibility to be placed in factories without any alteration, exploiting an adaptive framework that helps to connect with the existing machineries.
- Use rapid teaching to program production tasks in the setup from a rich set of learnable skills for applications like mould finishing, welding and assembly.
- Visually intuitive tools for the factory workers to assess the robot behavior. Augmented reality can be used to visualize the robot planned path to be aware of its activity allowing improved collaboration.
- A dynamic obstacle avoidance framework based on proximity skin sensors to allow human-robot collaboration in an unfenced workspace.

These aspects, along with the proposed certification procedures and a complete focus on manufacturing industry, can improve the automation sector. The first part of this thesis focuses on the last aspect in the list above, which is human-robot collaboration. Standard industrial robots cannot share the workspace with human beings, as they were primarily designed for productivity, not for safety. The robots are programmed to operate at higher speed wihtout any awareness of an obstacle or human presence which could be dangerous when a human worker accidentally gets into the robot workspace. The Fig. 1.5 (a) shows the reality of many manufacturing units with fences to avoid dangerous accidents with human beings. In case a human has to get inside the fence to prepare the workspace or repair something, strict protocols need to be followed to ensure safety. Moreover, it is definitely not efficient in case of frequent human interruption of the workspace, and it is expensive to build cages and integrate additional safety devices. A need for safety, shared workspace, and inexpensive installations gave birth to a new breed of robots called *Collaborative Robots*. Current *Collaborative Robots* need to be conservative to ensure safety in the vicinity of human beings. This work proposes a collision-avoidance framework that focuses on providing adaptability to the existing technology. A mature human-robot interaction requires advanced



(a) Fenced Robot for Safety

(b) New Gen Universal Robot

Figure 1.5: Motivation towards Collaborative Robots

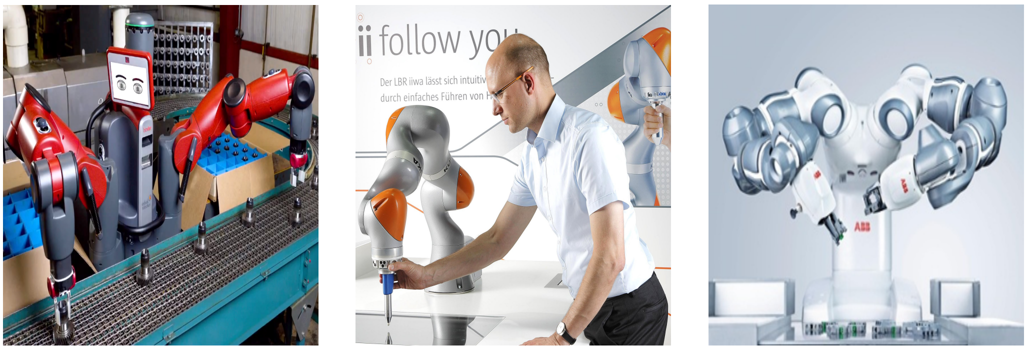
collision handling strategies, motivating the development and integration of reactive motion generation strategies. Let us discuss the current scenario of *Collaborative Robots* to position our framework in the state of the art.

1.2.3 Collaborative Robots

The world *Collaboration* literally means the action of working with somebody to create or produce something. Though collaborative robots share the same goal of regular industrial robots, the functionalities and workings necessary for a robot to be called *collaborative* depend on the kind of safety features. According to the international standards ISO10218, there are four kinds of features that qualify a robot as collaborative in nature [Fryman 2012].

1. **Safety Monitored Stop**: The robot needs to stop moving when a human worker gets into the restricted zone of safety. The proximity of the robot is monitored using external sensors to ensure safety of the workers without the need of a cage. The robot is stopped by switching on the brakes, instead of shutting it down interrupting its behavior, which allows the worker to perform his task in the shared workspace. It is reasonably efficient if the frequency of human interruption is minimal. This safety feature is the easier to implement in regular industrial robots.

2. **Hand Guiding**: This collaborative feature of the robot is an additional control mode that can be added in regular robots with an obvious necessity to detect applied forces on the arm. As the name says, a human can guide the robot by moving the tool around to learn scenario specific behaviors. A force torque sensor is used to measure these forces applied on the robot tool. The limiting aspect is that this type of collaboration works only in hand


ii follow yo
Der LBR iiwa lässt sich intuitiv
durch einfaches Führen von H
ABB

(a) Baxter Robot (b) KUKA iiwa (c) ABB Yumi

Figure 1.6: Collaborative Robots

guiding control mode, requiring the robot to safeguarded in the other control modes. This feature is very useful in developing applications without explicit programming.

3. **Speed and Separation Monitoring**: An improvement of the "safety monitored stop" feature is to adjust the speed of the robot when a worker is nearby. These regular robots use a vision system to detect the workers presence. In this case they operate at the stipulated speed, which is usually slow, and stop completely if a worker gets too close. The zones close to the robot are gradated with different robot behaviors depending upon the proximity. The new ANSI/RIA standards dictate a maximum end-effector speed of 250 mm/s when the distance between the robot and human beings is less than a minimum separation distance, which is determined by risk assessment of the setup and the application [Michalos 2015].

4. **Force Limiting Robots**: These are non-regular robots specially designed to be safe, so they do not need any other extra device to ensure safety. These robots react to any abnormal force that they detect with safe and adaptive motion behaviors. They are designed to dissipate impact forces thanks to their rounded shapes and the absence of sharp edges. These robots work within regulated force and energy limits, which can be applied without any harm. Before their deployment in the industry, they require a risk assessment verifying i) the severity of injury, ii) the possibility of occurrence and avoidance, and iii) the frequency of exposure. A direct connection with the worker and its ability to be compliant makes it look more collaborative in nature when compared to other robots with limited features. In terms of functionalities, these robots can measure forces in the joints, which allows the detection of unexpected contacts.

Universal Robots launched a series of cobots such as UR5(2008) and UR10(2012) followed by UR3(2015), the smallest cobot in the market. These robots are known for increasing productivity without causing injuries, making *Universal Robots* the largest collaborative robot manufacturer. Rethink Robotics launched Baxter(2012), two 7-degree-of-freedom arms with hand guided training to intuitively program the robot behavior without explicit programming (See Fig. 1.6 (a)). Later in 2015, they launched Sawyer, an extension of Baxter having larger workspace and better repeatability. Though the robot has good safety features, its shaky and irregular motions make it more a lab platform than an industrial product [Bélanger-Barrette 2015]. The LBR iiwa by KUKA, promoted as the first sensitive robot produced in series, can perform adaptive assembly tasks such as plugging hoses on the connector and human collaborated assembly exploiting torque sensing. As shown in this video, the workspace is separated into two areas: with and without permissible contact. In the contact permissible area, reduced velocities and force detection are implemented to ensure safety.

1.2.4 Motivation and Objectives

Though reduced velocities and compliant control are a simple way to ensure safety on collaborative robots, they do it in a conservative way: the fact that human beings do not engage with the robot all the time slows down the production process. The motivation of our work is to remove the barriers that separate permissible and non-permissible contact area. To do so, we want to implement adaptive behaviors by applying advanced perception and intelligent control. Force limited robots can feel contact forces, but they cannot detect an oncoming object if they don't have a proximity sensor. Also manipulating unstructured environments requires a good awareness of the surroundings. Visual sensors can be used to detect obstacles, but the continuous computation of the robot-obstacle distance after fitting an approximate model is computationally expensive.

A fast obstacle avoidance using on-board sensors is essential to remove physical cages and multiple control modes, without compromising safety, nor performance. We propose a framework that uses proximity sensors for dynamic obstacle avoidance. My contribution was the development of a reactive controller as a ROS-based component.

1.3 Towards Robust Balance Control

To motivate the necessity of robust balance control, this section discusses the evolution of humanoid robots and the current challenges preventing the deployment of humanoid robots in real working environments.

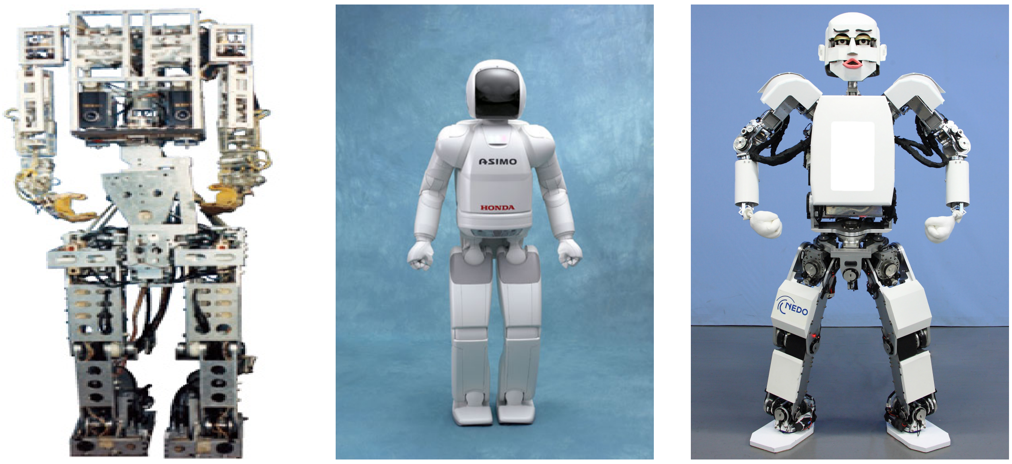
1.3.1 Evolution of Humanoid Robots

Humanoid robots are robotic mechanisms whose kinematics structure resembles the one of human beings, with a head and a torso bridging two arms and two legs. Many variants exist, considering only certain parts of the human body, but we focus on legged robots with arms in this thesis. The anthropomorphic shape of these robots helps their social acceptance, making human-robot interaction feasible [Fink 2012]. Also, the potential abilities of humanoids could make them suitable for rescue or disaster recovery scenarios. All these aspects make investment on humanoid robots important in robotics research.

In the beginning, humanoid robots were only controlled using traditional joint position control methodologies similar to industrial robots focusing only on precision tasks. WABOT-1(1973), from the University of Waseda, was the first humanoid robot to walk, communicate, visually recognize objects and manipulate them [Kato 1973]. The same laboratory lead a series of developments leading to WABIAN-2, which can walk with stretched knees [Ogura 2006]. P2(1996), from Honda, was the first robot to perform stable walking after 10 years of work on dynamic biped walking and stability control [Hirai 1998]. The next version P3 was much lighter and led to the launch of the famous ASIMO robot in 2000, having a more friendly appearance and improved mobility [Hirose 2007]. ASIMO's impressive capabilities attracted the attention of robotics researchers and created a perspective of humanoid robots to be exploited for service robotics [Kaneko 2009]. Many humanoids were developed in Japan during the last decade for both entertainment and demonstrating physical capabilities, including the *Humanoid Robotics Project* series: HRP-2P(2002) [Kaneko 2002], HRP-2(2004) [Kaneko 2004], HRP-3(2008) [Kaneko 2008], HRP-4C(2009) [Kaneko 2009] and HRP4 (2010) [Kaneko 2011], developed by Kawada Industries in collaboration with the Japanese National Institute of Advanced Industrial Science and Technology (AIST). Other Japanese popular platforms include QRIO(2004) [Geppert 2004], HOAP series(Latest HOAP-3 in 2005), H7 [Nishiwaki 2007], Kenta [Inaba 2003], Kojiro(2007) [Mizuuchi 2007], and Kenshiro(2012) [Nakanishi 2012].

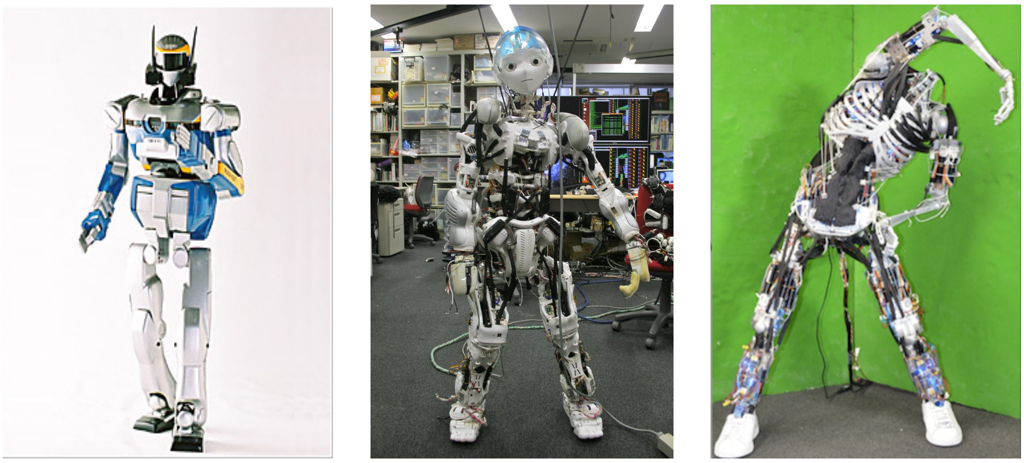
The SARCOS Research Corporation and the ATR (Advanced Telecommunications Research Institute International) from Japan, built the robots Erato DB (Dynamic Brain) based on hydraulic actuation in 2000 [Atkeson 2000], and CBi in 2006 [Cheng 2007], which explores the background neural processes in the human brain. The Korea Advanced Institute of Science and Technology (KAIST) developed humanoid robots (KHR) since 2002, with the latest HUBO 2 (2011) [Kim 2002, Park 2005a, Park 2005b, Grey 2013] winning the DARPA robotics challenge in 2015. Other Korean robots include MAHRU, and the AHRA series [You 2005, Kwon 2007, Kim 2011].

The Technical University of Munich (TUM) has built the humanoid robot Johnnie [Gienger 1999], followed by an improved version called Lola [Lohmeier 2009], known for its fast and robust motions. The Italian Institute of Technology (IIT) has built the small-size torque-controlled robots iCub [Metta 2010] and COMAN(2012) [Tsagarakis 2013]. The Spanish University Carlos III has built Rh-1(2007) [Arbulu 2007] and its successor TEO(2011) [Monje 2011]. Pal Robotics, a robotics company based in Barcelona, has built the humanoid robots REEM-B [Tellez 2008], and REEM-C, a human-friendly robot in 2014 [Robotics 2014]. In 2007, Aldebaran Robotics, a French robotics company, launched Nao, one



(a) WABOT-1 (b) ASIMO (c) Kobian

Figure 1.7: Humanoids of the first generation

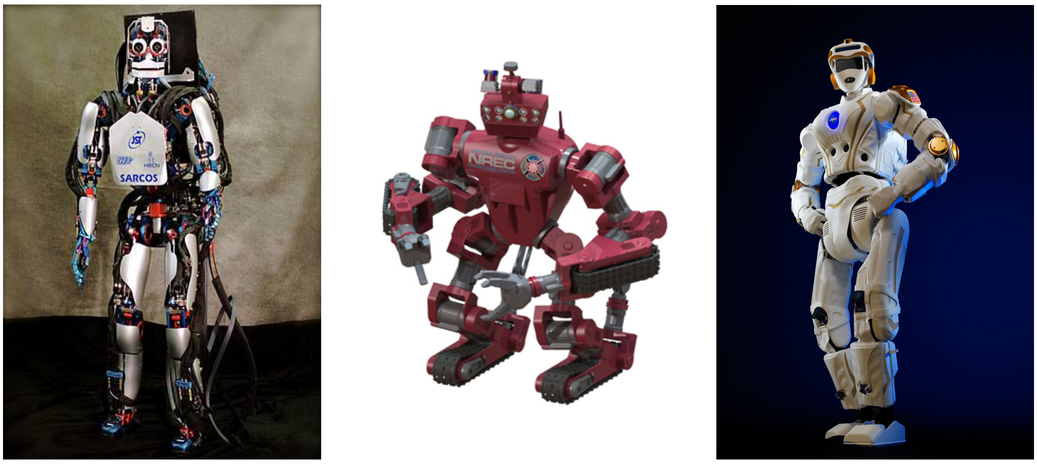


(a) HRP2 (b) Kojiro (c) Kenshiro

Figure 1.8: Japanese Humanoids

of the most popular small humanoid robots in the world, allowing several laboratories to work on humanoids with a smaller budget [Gouaillier 2009]. The same company presented a torque-controlled child-size robot, called Romeo. The DARPA Robotics Challenge (DRC), a competition funded by the US Defense Advanced Research Projects Agency (DARPA), was held between 2012 and 2015. This has motivated the development of new humanoid robots, such as CHARLI(2010) [Knabe 2013], THOR(2014) [Yi 2015], CHIMP(2013) [Stentz 2015], and Valkyrie(2013) [Radford 2015].

Most of the robots mentioned above are rigid, and fully actuated (i.e. each joint can be independently controlled) by electric motors. Compliances are sometimes used in the feet, to attenuate impacts with the ground while walk-



(a) SARCOS (b) CHIMP (c) Valkyrie

Figure 1.9: Advanced Humanoid Robots

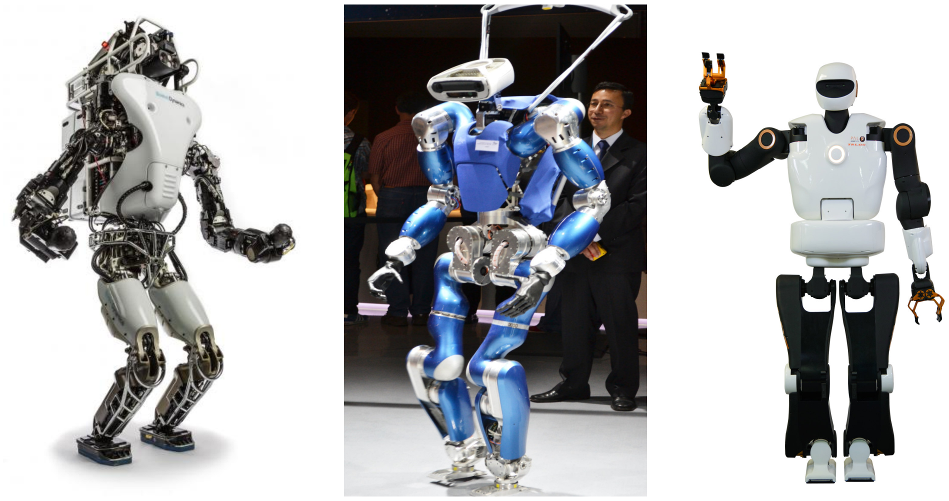
ing. These robots mostly use traditional high-gain position-control methodologies requiring a precise robot dynamic model. For walking and balance control, the zero moment point (ZMP) concept [Vukobratović 1972] has applied to many bipeds [Hirai 1998, Kaneko 2004, Grey 2013]. Advanced robots are expected to perform dexterous interactions in unstructured environments. High-gain position control severely limits the capabilities of these systems, as it does not allow them to be compliant, and thus adapt to the environment in case of modeling errors. For instance, in a jumping task the feet need to be compliant. The same can be observed in drilling tasks, where the robot hands need to apply force on the wall with a driller, motivating a need for compliant control in humanoid robots. Torque control has gained much attention in recent years for its ability to make robust and compliant interactions with the environment and human beings resulting in greater safety. Human safety is a crucial issue that does not allow service robots, either in mobile or humanoid form, to be commercialized as domestic robots. Higher compliance brings automatic adaptation to un-modeled and uncertain environments, making interactions safer than for traditional position-controlled robots. The interest in torque control has led to a new generation of humanoid robots with torque sensing. Even though joint torques cannot be directly measured in older platforms such as HRP-2, iCub, HRP-4 and Asimo, torque control has been implemented in [Del Prete 2016b, Nori 2015, Khatib 2008].

In 2012, Boston Dynamics has built the hydraulics robot Petman [Nelson 2012], and then a series of Atlas robots starting from 2013. The demonstration video of Atlas in 2016 illustrated the powerful capability of this hydraulic robot by performing tasks which were impossible for previous robots. Hydraulic actuation provides high bandwidth control and high power density, with the price of huge power requirements and noisy hydraulic pumps. High un-modeled friction/stiction makes it harder to control and also the high strength of actuators does not allow safe human-robot interaction. Series elastic actuators are also used by some humanoid platforms to implement torque control: StarlETH from ETH [Hutter 2012], CO-MAN [Tsagarakis 2013] and WALKMAN [Negrello 2016] from IIT, Hume from Human Centered Robotics lab [Slovich 2012], and M2V2 from IHMC [Pratt 2008]. The deflection in the spring is measured to compute the torque, and regulated to control the robot. Series elastic actuators are mechanically robust, shock absorbing and energy efficient [Kormushev 2011].

For certain applications, electrical drive units with torque sensors are an even better choice, as they allow for both stiff position control and controlled compliance, along with less acoustic noise and low maintenance. DLR's advanced Light Weight Robot (LWR) technology [Hirzinger 2002] has resulted in the development of the Rollin' Justin, a humanoid upper body, and the DLR biped robot [Ott 2010]. LWR drives were exploited in such a way that it does not require any customization to develop a complete humanoid robot specifically for a purpose such as walking or running. TORO, a torque-controlled humanoid robot with DLR-Biped legs can perform multi-contact interaction and dynamic whole-body motions [Englsberger 2014]. Each joint of the robot is equipped with both torque and position sensors, which makes it appropriate for torque control. DURUS from SRI [Hereid 2016] is another humanoid robot equipped with torque sensors and an efficient energy transmission system with high mechanical compliance at the ankles. Compliance is an important aspect in the design of humanoid robots. A typical task scenario of humanoids requires a mixture of precision and compliance in control. High stiffness is required for precision tasks, such as manipulation and foot positioning, which can be achieved by high gain position control. Low stiffness is required for compliant interaction with unstructured environment or human beings, which can be achieved by low gain force control. Ideally, we need a robot that can handle both the control with ease, which leads to the idea of developing controlled (active) compliances rather than just mechanical (passive) ones.

1.3.2 Humanoids in Real Environments

Even though there exist a large number of humanoid robots, their applications are rather limited. The DRC competition challenged the limits of humanoid robots by putting them in complex scenarios. It has been a great milestone in terms of the amount of attention received and the participation of teams from all around the world. After a first pre-selection in the DRC simulation, 16 teams managed to contest in the trails, consisting of a rich set of tasks: driving a utility vehicle, locomoting across rubbles, removing debris, manipulating various tools such as valves, fire hoses and more. TEAM Kaist has won the contest with their robot DRC-HUBO. Hubo's transformer design providing the ability to switch between bipedal walking and rolling on wheels made them perform faster [Guizzo 2015]. Though the DRC has encouraged humanoid research labs to deploy the current technologies on their robots to showcase their capabilities in complex and unstructured envi-



(a) ATLAS (b) TORO (c) TALOS

Figure 1.10: Latest Humanoid Robots

Figure 1.11: DARPA Robotic Challenge Illustration

ronments, we observed a variety of problems including failed task attempts, robot inactivity for longer period, and human operator mistakes [Atkeson 2016]. The contest has shown us how difficult it is to deploy humanoids in reality, in spite of all the technologies working in controlled environments or in simulation.

There are several interesting problems to overcome to make the technology mature enough to be used in reality. Different aspects need to be considered when discussing these failures.

- *Robot Design*: The mechanical design of the robot has influence on the kind of failures it encounters with the environment. Walking on a flat surface is different from rough surfaces or ground full of rubbles. An appropriate mechanical design that can adapt to unstructured environments for locomotion is necessary to eliminate failures. The transformer design of Hubo has been a clear example of this concept. The wheels allowed the robot to navigate in moderately rough terrains, eliminating the risks of fall associated to walking. A systematic way to combine wheels, limbs and tracks can breed a new kind of humanoid robots for disaster handling. Another important mechanical detail is the ratio between feet size and height of center of mass which affects the stability of walking robots.

- *Behavior Design*: The generated motion behaviors should be robust to hardware failures. The kind of behaviors that we choose have a crucial impact on the failures as well. This means that robots should be able to handle variations in the task, making the behavior robust. The generated motion behaviors do not fully exploit the environment to locomote. Humanoids have arms that could be used to rest or hold static or rigid objects in the scene to support locomotion. For instance, it is natural to hold the railings to climb up the stairs, which significantly reduces the amount of actuation needed in the joints. Most robots have shown no ability to locomote using external contacts.

- *Planning and Control*: Humanoid robots are redundant systems with more than 30 degrees of freedom, which makes their whole-body control complex. Also, the pose of the robot can be only controlled indirectly by appropriate joint motions and its interaction with the environment, making these robots under-actuated. Making and breaking contacts with the environment makes the motion generation problem even harder because it introduces discontinuity in the derivative of the robot dynamics function. Switching control behaviors due to physical contact, joint limits, or kinematic singularities, challenges the limits of optimization solvers, resulting in discontinuous control trajectories. All these problems make it easy for the robot to lose its balance.

- *Error Detection*: Robot falls are mostly caused by errors generated by the system itself rather than perturbations from the environment. Detecting errors earlier can give enough time to take appropriate control actions to maintain balance. Though there are strategies available to handle external perturbations (so-called ’Push-Recovery’ strategies), robot generated errors are given less attention. Aborting the current behavior after detecting errors can work, but there is a necessity to handle them systematically.

These aspects show the challenges of humanoid robots, and the need for various components to work appropriately to be robust to failures. For legged robots, balancing is an essential safety constraint, and has been our motivation to focus on robust balancing under model uncertainties.

1.3.3 Motivations and Objectives

Balancing a humanoid without any failures is difficult as it would require a perfect robot model. Humanoid robot models are typically less accurate due to a variety of reasons:

- The high number of degrees of freedom and the impossibility to take the robot apart to measure the parameters of each link makes the parameter identification problem harder. Moreover small errors at each link can accumulate to large errors at the end-effectors.

- The rigid body assumption is violated due to the link and joint flexibilities, which are typically higher than in industrial manipulators mostly due to loading because of gravity when standing.

- The extended use of the robot, or the replacement of its components, introduces small parameter variations, which are hard to model.

- It is challenging to model the interactions between foot and contacts which depends upon the nature of the ground surface.

The Center of Mass (CoM) of the complete robot plays a crucial role for balance control. Most balance control strategies are indeed revolving around it. This requires an accurate computation of the robot CoM, but due to modeling errors at each link, a model-based estimation of the CoM position and velocity may not be accurate enough. These errors affect the robot balancing, and can be the main reason of failure of the control algorithm. This scenario defines the context of the second part of this thesis, which presents a computationally-efficient optimization-based controller that account for these uncertainties in the balance of the robot.

1.4 State of the Art

In this section we discuss the current state of the art of motion generation in robotics, providing the basis for the inverse-kinematic and inverse-dynamic control techniques used in the next chapters.

Control methodologies generally define a control law that generates feasible motions, with respect to robot and environmental constraints, to achieve one or more given tasks. There are a variety of methods to generate motion depending upon the robot, the environment and the task complexity. The control of humanoid robots is particularly challenging because of their kinematic redundancy and their dynamic complexity.

- They have a non-trivial kinematic tree structure, with an essential need to stabilize the position of their center of mass (CoM) with respect to the contact locations, while executing other tasks. In simple words, a humanoid robot cannot walk or run without knowing how to balance itself while in motion.

- The dimension of their task space typically does not equal the dimension of their actuator space. Typical tasks consist in controlling the position and orientation of an end-effector (i.e. 6 dimensions), while a humanoid robot has more than 30 degrees of freedom.

- Humanoid robots are under-actuated systems, which means that a subset of their state (i.e. their 6d pose with respect to the inertial frame) cannot be controlled directly, but only as a consequence of their joint actuators.

- The controller must handle the local interactions with the environment, while reaching the global goal. The combined workings of both a motion planner and controller results in a complete working application.

In the following we discuss the three main control methodologies used in industrial and legged robots: kinematic control, dynamic control and optimal control.

1.4.1 Kinematic Control

Kinematic control is one of the most used control techniques for precise robot control. In this section, we will briefly discuss the basics of kinematic control, the shift to task based control which forms the basic foundation for our collision avoidance framework proposed in chapter 2.

Basics of Robot Kinematics Kinematics is the study of movement of kinematic chains. It focuses on the geometry, ignoring the dynamic properties of the system, such as mass, inertia and the forces/torques that generate the motion. Kinematic control relies on the relationship between position, velocity and acceleration of the joints and the links of the system. These relationships of course depend on the kinematic structure of the robot. The joint space of a robot with n degrees of freedom is an n-dimensional manifold $\mathcal{Q}$. The variation in an operational point x can be represented by a twist ξ, comprising linear and angular velocities [Featherstone 2014, Murray 1994]. An operational point is an arbitrary point on the robot that is required to perform a desired motion or achieve a manipulation task or exert a desired force in the environment. The control model can be defined in four different ways:

- *Forward Kinematics*: Given a joint configuration $q \in \mathcal{Q}$, find the pose of an operational point $x \in SE(3)$ such that $x = f(q)$, where $f : \mathcal{Q} \rightarrow SE(3)$ is the forward-kinematics function.

- *Forward differential kinematics*: Given the joint velocities $v \in T_q(\mathcal{Q})$, find the twist of an operational point $\xi \in se(3)$ such that by $\xi = J_o(q)\dot{q}$, where $J_o : T_q(\mathcal{Q}) \rightarrow se(3)$ is the tangent or geometric Jacobian matrix [Spong 2006, Khatib 1987], and $T_q(\mathcal{Q})$ is the space tangential to $\mathcal{Q}$ at q.

- *Inverse Kinematics*: Given a pose $x \in SE(3)$ for an operational point, find the joint configuration $q \in \mathcal{Q}$ such that $x = f(q)$, where $f : \mathcal{Q} \rightarrow$

$SE(3)$ is the kinematic function. The *Inverse Kinematics* problem can be solved analytically for certain kinematic structures, using either algebraic approaches [pau 1981, Raghavan 1993] or geometric approaches [Paden 1985, Peiper 1968].

- *Inverse differential kinematics*: Given a twist $\xi \in se(3)$ for an operational point, find the joint velocities $v \in T_q(\mathcal{Q})$ such that $\xi = J_o(q)\dot{q}$. This problem boils down to solving a (typically underdetermined) system of linear equations, and can thus be easily solved numerically, e.g. by computing the pseudo inverse of J_o, that is $J_o^+ : se(3) \rightarrow T_q(\mathcal{Q})$. In certain cases it can be solved analytically [Chiaverini 1994, Chiaverini 1997, Siciliano 1999].

Kinematic Redundancy The kinematic control problem consists in reaching a reference point, or tracking a trajectory, in $SE(3)$ for one or more operational points, by searching the appropriate instantaneous joint configurations $q(t)$. Such goals are sometimes called tasks. When the dimension of the robot n exceeds the dimension of the task n_t, the robot is said to be kinematically redundant with $n - n_t$ being the degree of redundancy with respect to the task [Nakamura 1990]. Though kinematic redundancy provides flexibility in the joint space to manage the constraints effectively, it can be complex to handle in a multi-task scenario [Siciliano 1991]. Closed-form solutions for redundant robots are not always available or difficult to analytically compute. The approaches used in [Ali 2010, Nunez 2012] treat the complete robot as a set of many kinematic chains making it complex and less generic. Given the complexity of these approaches, instantaneous inverse kinematic solvers are usually preferred. Also a controller that solves both the primary task and the secondary tasks (at its best) is essential for handling multiple tasks in parallel to fully exploit the redundancy of the robot. For instance, a robotic arm may have to manipulate an object (secondary task) while reactively avoiding collisions with obstacles (primary task). Numerical approaches are used to solve these multi-task control problems on redundant robots.

Numerical methods formulate *Inverse Kinematics* as a constrained optimization problem either with global or local constraints. Global methods search for an optimal path for the entire trajectory, which is computational expensive [Baillieul 1990]. Local methods solve them differentially, computing locally optimal joint variations dq corresponding to small end-effector variations dx to generate the joint space trajectory $q(t)$. *Resolved Motion Rate Control* [Whitney 1969] finds the joint velocities v by solving the system: $\dot{x} = J(q)v$. Damped pseudo inverses [Nakamura 1986] are used to avoid singularities and numerical issues in redundant robots, which may appear when the Jacobian matrix has not full rank.

A more generic solution solves each task by projecting it into the null space of the Jacobians J of the tasks with higher priority [Liegeois 1977]. There are two ways to carry out the projection systematically.

- *Task Weighting/Soft priorities* uses weighting to modulate the task space by constraining the joint space [Sciavicco 1987]. Task conflicts are managed by assigning weights to each task, finding a trade off between the different tasks. Proper tuning specific to the scenario is essential for this methodology.

- *Task Prioritization/Hard Priorities* strictly prioritizes the tasks, ensuring that the lower priority tasks do not affect the tracking error of the higher priority tasks [Nakamura 1987]. A systematic framework for a multi-task scenarios is proposed in [Siciliano 1991], which numerically computes v to minimize the task errors in a prioritized way.

Both these approaches are quite popular in the control of redundant systems. *Task Weighting* methods [Tevatia 2000, Salini 2009] suffer from task conflicts, resulting in unsuccessful task execution, whereas *Task Prioritization* has a strict priority on resolving multiple tasks leading to locally optimal results. The main advantage of attaching strict priorities to the tasks, is the simplicity it provides to program multiple tasks in an hierarchical fashion. Several heirarchical approaches have been developed for multiple equality constraints at the kinematic level [Yoshida 2006, Mansard 2007, Gienger 2005]. For inequality tasks, a cascade of quadratic programs is used to solve the stack of tasks [Kanoun 2009]. A much more efficient implementation using complete orthogonal decomposition has been proposed by [Escande 2014a]. [Jarquín 2013] proposed a solution for smooth transitions in control when the priorities are interchanged but does not deal with inequalities.

The controller we used for the reactive control scheme discussed in chapter 2 uses the state of the art hierarchical based control as proposed in [Escande 2014a]. The hierarchical property and the ability to handle inequality constraints provide a way to deform the trajectory in case of unforeseen situation such as a collision with obstacles in the vicinity. Collision avoidance provide safety and is of high priority when compared to the main task such as manipulating an object from the workspace. There are other safety tasks which are basically inequality constraints such as ensuring joint position, joint velocity limits, ensuring balance in legged robots occupying the first level of the hierarchy. This provides a way to achieve the main task only with the available degrees of freedom after the safety tasks has been satisfied. The formulation of these inequality constraints and the hierarchy used to achieve a collision avoidance behaviot while executing the main task will be explained in chapter 2.

1.4.2 Dynamic Control

Applying dynamic control helps in advanced interaction with the environment for dynamic tasks such as balancing, jumping, or running for humanoid robots. In this section, we will briefly discuss the basics of dynamics control that will prepare us for the proposed robust controller strategy discussed in chapter 3.

Basics of Robotic Dynamics Dynamics is the study of the relationship between the motion of the kinematic chain and the generalized forces acting on the system.

The generalized forces include the joint torques for rotational joints, the joint forces for prismatic joints, and also the contact forces. This relationship allows to control the robot at a dynamic level, resulting in a better control of physical interactions. Dynamic parameters such as length, mass, and inertia of each link need to be known in this kind of models. In a robot dynamic model, the motion is defined by joint acceleration $\dot{v}$, and operational point acceleration $\dot{\xi}$ in the task space. The two main problems in robot dynamics are:

- *Forward Dynamics*: find the joint accelerations of the robot given the generalized forces.
- *Inverse Dynamics*: find the generalized forces given the joint accelerations.

The two main approaches to model the robot dynamics are:

- *Lagrange*: this is an energy-based approach resulting in a closed-form expression of the dynamic equations [Uicker 1969, Kahn 1969, Bejczy 1974]. It has a clear separation of each component, but it is very expensive for implementing control schemes. [Hollerbach 1980] presents an efficient formulation, but still not as efficient as the *Newton-Euler* algorithm.
- *Newton-Euler*: this recursive algorithm [Orin 1979] does not clearly separate components by describing the combined translational and rotational dynamics of rigid bodies. The recursivity of the apprach makes it computationally cheaper. [Featherstone 2010] explains the most common algorithms such as the recursive Newton-Euler Algorithm (RNEA).

[Spong 1992] uses an hamiltonian approach for the analysis of the robot dynamics and there exist certain numerical methods to integrate hamiltonian equations efficiently. Alternatively, *centroidal dynamics* [Orin 2013, Orin 2008] models the dynamics of the CoM of the robot capturing the constraints imposed by contact forces on the CoM making the dynamic model very simple. But it does not include joint position and torque limits which makes the model approximate failing to describe the angular momentum of the robot. Humanoids walk with very little angular momentum, which is typically supposed to be zero when using *centroidal dynamics* model. In contrast to the classic joint space formulation, the operational space formulation [Khatib 1987] defines the motion using the task space acceleration, which requires the forces to be formulated as generalized forces in the task space.

Dynamic Control in Humanoid Robotics There are two main categories of torque-based control: impedance control and inverse dynamics control. Impedance control [Part 1985, Albu-Schäffer 2007, Ott 2008, Schäffer 2008] is known for its passivity properties, which guarantee stable interactions with the environment. Inverse dynamics (ID) control [Del Prete 2016b, Buchli 2009, Righetti 2013a] relies on a complete dynamic model of the robot to ensure good trajectory tracking while maintaining high compliance, thanks to the low feedback gains. Recent ID controllers use quadratic programming (QP) solvers, which allow to account for inequality constraints, such as joint position, velocity, and torque bounds, contact force friction cones, and center of pressure (CoP) limits, which are crucial in humanoid robots. The operational-space inverse-dynamics (OSID), a generic framework establishes the whole-body control considering balance, contacts and other constraints. [Khatib 2004] proposes a multi-task formulation with sequential projection on the nullspaces of the tasks at each level. Strict hierarchy allows lower priority tasks not to affect the higher priority tasks. An equivalent approach [Mistry 2010] uses orthogonal decomposition and kinematic projections to simply control when switching contact constraints and avoids the inversion of mass matrix. [Righetti 2011a, Righetti 2011b] proposes an improved version constraining the ground reaction forces with the friction boundaries. This framework does not include inequality constraints, which allow a straightforward implementation of collision avoidance, joint limits and visual servoing. OSID is used within optimization-based methodologies to find optimal solutions. Quadratic Programming (QP) based approaches are more popular for redundant systems, which allow both equality and inequality tasks. Weighting schemes has been used in a QP-based approach that provides robust balance [Collette 2008a]. [Salini 2011] implemented a weighting approach to compute torque commands using a sequence of prioritized dynamic tasks. [Herzog 2013] has used an active-set algorithm to implement an inverse dynamics control of the legs of a hydraulic humanoid robot. OSID in a strictly prioritized QP framework has also been implemented in simulation in [Saab 2013].

The previous approaches focused on the robot dynamics, but did not control the angular momentum, which is an important component of human agile and complex motions [Popovic 2004]. [Kajita 2003c] proposed to control the angular momentum of the robot using the inverse of the inertia matrix. Other approaches focused on controlling the angular momentum by constraining the centroidal momentum of the system. In [Hofmann 2009], this approach is implemented to generate appropriate gait movements. [Moro 2013] used an attractor-based approach to control the angular momentum. Also [Wensing 2013] used conic optimization techniques to control the angular momentum generating complex motions.

1.4.3 Optimal Control

Optimal Control consists in finding control laws for a given dynamical system, such that a given optimality criterium is minimized. It can be seen as a constrained infinite-dimensional optimization problem, which is impossible to solve in general. When the found control law is open loop (i.e. it does not depend on the state, but only on time) it is called *Trajectory Optimization* or *Trajectory Filtering*.

Background Optimal Control is actually an extension of *Calculus of Variations* that uses optimization to find control policies. The first ever optimal control solution was proposed for the brachystochrone problem in Bernouli's work [Sussmann 1997].

Though there were early contributions to the theory of optimal control by Newton, Euler, Leibniz, Jacobi, Hamilton, Bolza and many others, the formalization began to take shape with the introduction of the Linear Quadratic Control (LQC) problem, minimizing a quadratic ojective function [Wiener 1949]. The next milestone was the birth of Dynamic Programming (DP), a recursive approach to solve discrete (time and space) optimal control problems, resulting in a discrete version of the Hamilti-Jacobian Bellman(HJB) equation.

HJB formalizes the optimization problem as a nonlinear partial differential equation. The solution of the HJB equation is the value function, which gives the minimum cost as a function of the initial state. When solved locally, the HJB is a necessary condition, but when solved over the whole state space, the HJB equation is a necessary and sufficient condition for an optimum. The solution is open loop, but it also permits the solution of the closed loop problem. The HJB method can be generalized to stochastic systems as well.

Pontryagin maximum principle [Pontryagin 1987] completely formalizes the problem based on the calculus of variations considering path-wise constraints on the control. In optimal control problems, the final goal is to find a trajectory. The HJB approach is a good way to find a local optimum, but it does not actually constitute a solution, whereas Pontryagin maximum principle gives an actual solution [Bertsekas 1995]. The *Linear Quadratic Regulator* (LQR) and the *Linear Quadratic Gaussian* (LQG) have been formulated to design optimal control policies in [Kalman 1960], which marks another important step in optimal control.

There are two numerical approaches used to handle optimal control problems: direct and indirect methods. In direct methods, the state, control, and the objectives are approximated using a relevant function such as piecewise or polynomial approximation. The control problem is transcribed to a nonlinear optimization problem with the coefficients of the approximated functions as optimization variables. In indirect methods, the first-order optimality conditions are generated using calculus of variations, resulting in a two or a multi-point boundary value problem to be solved.

Model Predictive Control (MPC) also known as Receding Horizon Control, is a popular automatic control methodology in the industries. At each control cycle, it solves a new finite-horizon optimal control problem, using the current state of the system as initial state in the problem. Only the first part of the computed control trajectory is then applied, while the rest is discarded [Richalet 1978]. The high-level task goals are specified as simple cost functions, and the MPC controller generates the behavior details and the control law automatically. The MPC control also avoids the extensive exploration by generating control policies online. The controller predicts the future states of the system to decide the best current control action based on the pre-defined criteria. *Differential Dynamic Programming* (DDP) is an efficient numerical scheme for direct optimal control, generating locally-optimal trajectories [Jacobson 1968]. A hybrid method with constant local controllers was presented in [Atkeson 1994], to speed up global optimization in dynamic programming. An extension was later proposed in [Atkeson 2003], which used second-order

DDP models to make locally-linear controllers.

Optimal Control in Humanoid Robotics For humanoid robot walking, the *Operational Space Inverse Dynamics* or *Inverse Kinematics* cannot properly handle the constraints of the CoM accelerations, thus generating very conservative motions. Optimal control can find trajectories connecting given initial and final postures subject to given constraints. MPC is extensively used in humanoid robotics. [Kajita 2003b, Herdt 2010] used a Linear Inverted Pendulum Model (LIPM) to generate walking motion for a humanoid robot. Though it is a complex machinery and it heavily relies on models, the selling point of optimal control is that it can account for all the system constraints. The key problems limiting the use of optimal control in humanoid robotics are the large number of DoFs and the need for fast reactions, which make computational time a critical resource.

In multi-task scenarios, a weighted average of the task errors can be used to find a compromise between the possibly-conflicting tasks. Choosing the right weights for each task may however be challenging [Dimitrov 2011]. Large weights can also produce numerical errors, making these control schemes hard to use in practice. In [Del Prete 2014], strict priorities have been introduced in the optimal control problem to avoid such issues. *MuJoCo* [Todorov 2012] is a state-of-the-art simulation and control framework based on MPC. It has been used to generate simulated humanoid motions, such as getting up from the ground or rejecting disturbances [Tassa 2012]. The use of contacts in the environment increases the controllable space to successively achieve the goals [Lengagne 2013] generating non-coplanar contact motions. Parkour motions [Dellin 2012], kicking a ball [Miossec 2006] and lifting weights [Arisumi 2008] are examples of complex behaviors generated using optimal control.

Different variants of DDP have been used in humanoid robots. Control-limited DDP [Tassa 2014] allows for box constraints on the control inputs, and it has been applied on a simulated humanoid robot. Square-root DDP [Geoffroy 2014] is an efficient variation of DDP that exploits the least-squares form of the cost function, showing also that IK and OSID are special cases of optimal control without preview horizon. DDP has also been used to generate simulated sample trajectories to train a neural network based policy [Levine 2012]. Behaviors such as walking, running, hopping and swimming have been generated using this approach. Robust walking behaviors have been generated using dynamic programming [Whitman 2013], relying on multi model and learning based dynamic programming variants. Steep climbing motions have been generated in [Noda 2014], which used the Body Retention Load Vector index for modeling severity of physical constraints, i.e. bounds on contact forces, moments and joint torques. Optimal control has also been treated as an offline control problem in [Schultz 2010] using multiple shooting to generate energy efficient running. Walking motions have been generated without using a pattern generator by optimizing joint velocities, torques or ZMP constraints [El Khoury 2013, Koch 2012].

Motion planning is also combined with optimization for collision-free navigation in a cluttered environment [Desai 1999, Kalakrishnan 2011]. Motion planners for high-DoF systems generate trajectory in two stages: planning and optimization. There are three well-known and similar techniques in this domain. CHOMP (Covariant Hamiltonian Optimization for Motion Planning) uses covariant gradient-descent techniques, resulting in a planning algorithm completely relying on trajectory optimization [Zucker 2013]. Starting with a naive trajectory, CHOMP optimizes the trajectory while reacting to the obstacles in the environment. STOMP (Stochastic Trajectory Optimization for Motion Planning) is very similar, except for the use of stochastic perturbations to generate trajectories without computing the Jacobian [Kalakrishnan 2011]. ITOMP (Incremental Trajectory Optimization for Real-Time Replanning in Dynamic Environments) incrementally updates the trajectory online, but it can produce suboptimal solutions because of the time constraints in the solver [Park 2014].

The high dimensionality of humanoid robots is one of the main challenges to get optimal control working in real time. Current optimal control solutions are time consuming and encounter numerical problems, which is still an open issue in robotics.

1.5 Chapter Overview

The thesis describes two main contributions, respectively in Chapter 2 and Chapter 3. Chapter 2 presents a reactive collision-avoidance control scheme using proximity information from the skin sensors, which has been tested on a UR5 robot. Chapter 3 presents a robust balance controller that accounts for inertial parameter uncertainties. Finally, Chapter 4 concludes the thesis and discusses the future work.

1.6 Publications

- Giftsun, N., Prete, A. Del, & Lamiraux, F. (2017). *Robustness to Inertial Parameter Errors for Legged Robots Balancing on Level Ground.* International Conference on Informatics in Control, Automation and Robotics (ICINCO) [**CONFERENCE**].

- Bharatheesha, M., Giftsun, N., Hernandez, C., Bergner, F., Dean-leon, E., & Cheng, G. (2017). *Dynamic Obstacle Avoidance for Collaborative Robot Applications.* IEEE International Conference on Robotics and Automation (ICRA)[**WORKSHOP**].

Chapter 2

Collision Avoidance Framework

Current collaborative robot arms allow for more flexible work cells, where they safely collaborate with human operators augmenting productivity in tasks difficult for traditional automation. However, current robotic solutions for safe interaction simply stops the robot motion when a collision is detected. This reduces the productivity in an operational setup in which unintended, safe collisions can happen often. Active contact evasion by the robot arm is desired so that the production process continues despite regular interferences and path obstructions, without compromising human safety. In the framework of the EU project Factory-in-a-day we have investigated dynamic collision avoidance techniques that exploit proximity-measuring robot skin. We have used reactive motion control to generate collision-free motions in a real-time industrial manipulation setup. These technologies have been integrated into a unique framework for dynamic collision avoidance, which has been successfully tested on a UR5 robot. The primary contribution of this chapter is thus a reactive control mechanism, built upon a state-of-the-art inverse-kinematics Hierarchical Quadratic Programming (HQP) solver. This chapter summarizes the above-mentioned collision avoidance framework, with a special focus on the reactive control components.

2.1 Introduction

A need for robotic solutions, particularly in the small and medium scale enterprises (SMEs) is becoming increasingly prominent. Automation and robotics promise to reduce production costs and increase productivity. However, traditional automation implies an investment prohibitive for SMEs, whose activities mainly involve small batches of production and high variety of products, for example due to a seasonal nature of their operations. Concretely, tasks such as assembly, machine filling or packaging, can be automated with a robot in the work cell. However, economic feasibility requires to reduce the robotization costs. As it was pointed out earlier, the Factory-in-a-day project [Wisse 2013b] focuses on reducing the robotization cost by reducing the cost of system integration and the installation time. The key idea is to develop generic and flexible robotic solutions so that they can be quickly re-installed and configured to another temporary product line. To achieve this flexibility and maintain acceptable levels of productivity, the Factory-in-a-day approach is to *automate the easy 80% of the tasks and leave the hard 20% for human co-workers.* Robot manipulators provide power, repeatability and extended work-space, while the human operators provide flexibility and problem solving capabilities. In addition, fenceless collaborative robots save space and installation cost. However, this approach requires a very high level of safety and agility; the robots should be aware of any obstacle (including dynamic obstacles such as human co-workers) and be able to avoid collisions. Whereas current co-bots guarantee safe contacts, they degrade the performance of the work cell by stopping the production.

This chapter presents the Factory-in-a-day framework, which relies on collision avoidance and skin sensors to make robots able to avoid (dynamic) obstacles while continuing their work. The chapter is structured as follows. Section 2.2 summarizes the state of the art in collision avoidance. Section 2.3 presents the framework, each of its individual components, and their interconnections in a manipulation scenario. Section 2.4 presents the main contribution, which is the reactive motion control part of the framework. Section 2.5 demonstrates collision avoidance using the proposed methodologies on a real robot setup. Section 2.6 discusses the merits and demerits of the proposed methodologies along with conclusive remarks.

2.2 Collision Avoidance

The main motivation behind collision avoidance is to ensure safety of the robots, its connected components and, most importantly, the human beings and the environment. Also, a secondary motivation to avoid collision is to allow the robot to achieve its task efficiently. Collision avoidance is one of the fundamental problems of robotics. It consists in planning a sequence of actions that the robot should take to avoid a detected obstacle in the near future. This means that there is no need to avoid collisions if there are no oncoming collisions, giving rise to the sub-problem of *collision detection*. Clearly, the collision avoidance quality is highly dependent on the collision detection quality. This simple subconscious mechanism that allows human beings to be aware of obstacles and avoid unintended contacts, is actually extremely complicated to automatize in robots. Even if collisions are handled at the planning level, this is not sufficient if un-modeled obstacles appear in the environment, or obstacles start moving. Robots should continuously perform the sense-act cycle based on the instantaneous observations of the world to avoid collisions while executing a task. Moreover, the collision avoidance algorithm has to take into account a variety of aspects, such as the kinematic and the dynamic capability of the robot and its ability to detect collisions at run time. In the following, we briefly discuss the state of the art of collision avoidance, to position our framework.

2.2.1 State of the Art

To start off, we want to highlight that by *collision avoidance* we do not necessarily mean avoiding unintended contacts with the obstacles. While collaborating with robots, it is necessary for human beings to be in contact with the robot to teach or guide them through the work flow. As the main concern of collision avoidance is safety, a collision can be avoided even after contact is made. We can thus identify two kinds of behaviors to avoid collisions:

- *Extrinsic behaviors*: These active behaviors avoid the contact with the obstacles. They require a continuous tracking of the distance between the robot and the obstacles to take an appropriate action for avoiding contact.
- *Intrinsic behaviors*: These approaches are based on the idea to dissipate the contact forces applied on the robot links. They require force/toque sensors to measure the interaction forces and thus be compliant in case of collision to minimize damage. To ensure safety with this kind of behaviors, the speed of the robot must be bounded to give it enough time to react.

The collision avoidance problem has been researched extensively since the 80's, resulting in a variety of methodologies tackling the problem either at the planning or at the control level. Let us have a look at the existing approaches.

2.2.1.1 Extrinsic Approaches

Potential Field Methods Potential field methods are one of the most popular techniques used to date. The first real-time collision avoidance algorithm based on potential fields was introduced in [Khatib 1986], and later extended in [Warren 1989, Ogren 2000]. The control laws are defined based on artificial attractive and repulsive fields, which can pull the robot towards the goal, while pushing it away from the obstacles. This simple approach allows to generate reactive and complex motion behaviors by modifying the trajectory depending on changing environmental conditions. In its extended version [Warren 1989], a collision free path is generated by a path planner by defining the potential functions around obstacles in the configuration space. The linear path connecting initial and final state is exposed to the artificial fields, resulting in incremental deformations. Some local minima are avoided because planning is done at a global level, considering the entire path. However, defining potential functions in configuration space becomes challenging for systems with many degrees of freedom. In [Ogren 2000], coordinated motions between the base and the arm have been generated to reach an end-effector goal, while avoiding the obstacles encountered by the base.

There are different formulations of potential fields with different semantics depending on the kind of robot and the goals of the scenario. This has led to implementations mostly depending on the so-called *separation distance* between robot and obstacles, but also on other factors, such as the human gaze direction, or even the affective state of the human nearby. The affective state describes the psychological experience of emotions. In the safety framework proposed in [Kulić 2007], the controller uses a danger index based on the separation distance, the velocity, and the affective state of the user. The affective state of the human is inferred using skin conductance and heart rate measurements. The controller in [Calinon 2010] used a risk criterion based on the human gaze direction, in addition to the separation distance to the human head to guide the robot motion without compromising safety. The collision avoidance method in [Haddadin 2010] is not only based on the virtual forces caused by the separation distance, but also the real forces due to the physical contact. They used virtual springs and dampers in the workspace to generate collision-free trajectories with an impedance controller. The potential function in this controller depends also on the direction of the approach, smoothing the motions and avoiding unnecessary accelerations. Another interesting framework proposed in [Flacco 2012, De Luca 2012] used two collision avoiding schemes: one for the end-effector, and the other for the surface of the robot. The repulsive vector for the end-effector is considered as a repulsive velocity using a potential field, whereas artificial forces are generated for the other points on the robot, modeled as Cartesian constraints.

In contrast to the repulsive field definition of the obstacles, the idea of a circular field generated by fictitious current on the obstacle surface was introduced in [Singh 1996]. This allows the robot to rotate away from the obstacles avoiding some local minima, which is the main issue with the potential field approaches. A hybrid method presented in [Haddadin 2011b] used an extended circular field along with a potential field to model the interaction between the robot and a complex environment for a 6D operational space real-time motion. A kinetostatic danger field [Lacevic 2010] is a generalization of potential fields that captures the robot kinematics by creating a virtual field on the robot rather than the obstacles. Kinetostatic safety fields [Polverini 2014] , an extension of [Lacevic 2010], includes the geometry of the danger source and the relative motion between the danger source and the field location to model the interaction. The safety field was experimentally verified to avoid self collisions and human-robot collision.

Real time Adaptive Motion Planning The insight to handle dynamic environments by exploiting both global planning and reactive motion control has resulted in a variety of real-time adaptive motion planning methods. These methods perform adaptive online trajectory generation to bridge the motion planning and control components to handle dynamic environments. Usually the planner generates parametrized paths that are modulated at run-time, based on the interactions with the environment. The Elastic band framework [Quinlan 1993] represented paths as curves in configuration space, with properties of an elastic band. The obstacles had a repulsive action resulting in trajectory elongations, as an elastic band. The main difference with respect to [Warren 1989] is that the path is deformed by using proximity information in task space, rather than configuration space. This makes the potential fields easier to define. However, the computational complexity increases with the number of degrees of freedom and the geometrical complexity of the robot, making the framework more suitable for robot with few degrees of freedom. [Baginski 1998] proposed a fast motion planning algorithm called the BB method, which literally means 'Blow up the robot and Bend the trajectory'. The planner starts with an initial path and it reduces the robot size until the path is collision free. The path is then incrementally modified repositioning the colliding bodies to let the complete robot pass. However, this method can get stuck in local minima and it cannot incorporate local motion constraints at run time.

The elastic strip framework [Brock 2002] is a generic and efficient framework that integrates global motion planning and task-based control with reactive obstacle avoidance. The planned motion for redundant robots can be deformed using task-based control to execute robust collision-free motion in dynamic environments. In [Vannoy 2008], spline functions have been used to represent trajectories connecting the waypoints generated by a global planner. The parameters of the spline functions are modified to adapt to the dynamic obstacles in the environment. The approach in [Yang 2006, Yang 2010] uses a roadmap composed of collision free vertices and edges to represent planned trajectories. In [Balan 2006], the robot and the obstacles are modeled as spheres to generate a collision-free trajectory by searching in the space of end-effector in predefined orientations.

Speed and Separation Monitoring These approaches are based on the idea of gradating the space around the robot based on the distance to an obstacle. These complex zones are used to control the speed of the robot to avoid harming the human beings nearby. The SafeMove framework by ABB [Soenke Kock 2006] uses programmable static safety zones to control the robot speed within recommended limits, allowing safe interaction with human beings using extrinsic sensors. In contrast to static zones, [Vogel 2013] implemented varying safety zones depending on the joint positions and velocities, increasing the efficiency of the task with more workspace flexibility to the collaborating human beings. The interesting thing in this work is the projection system used to display the safety zone on floor of the room where the robot is mounted, to give a visual cue to the user about where the robot is approaching. The speed is decreased when a human being or an obstacle enters into this changing virtual space, making it very effective in industries. [Lasota 2014] proposed a framework that eliminates the definition of conservative safety zones by using accurate and instantaneous distance measurements to scale down the speed of the robot using a tunable function incorporating task-related parameters. The approach proposed in [Zanchettin 2016] addressed the loss of productivity of [Lasota 2014] by taking advantage of the robot redundancy to ensure safety, while maintaining the goal end-effector position. The safety region is computed based on the separation distance with uncertainties in measurement, robot velocity, and the braking distance, allowing the robot to maintain a significant distance of separation without compromising the goal.

The main challenge in the above systems is the ability to localize human beings or obstacles accurately. [Flacco 2012] used depth information from 3-d cameras, such as kinect, to estimate the separation distance and the obstacle velocity. In [Rybski 2012], the framework fuses data from stereo and range cameras to generate dynamically changing danger zones based on the robot state and the trajectory. [Avanzini 2014] used multiple infrared distance sensors distributed on the surface of the robot to reduce the possibility of occlusions and sense unstructured environments. The work also proposes an optimization strategy to find out the minimal number of distance sensors and their best placements on the robot for effective collision avoidance. As discussed before, [Lacevic 2010] used a kinetostatic danger field criterion depending on the distance sensors information to control the robot behavior. This framework allows to modulate the task hierarchy depending on whether the danger criterion is above the safety threshold or not. For example, if the danger metric increases because of human interference, the end-effector goal orientation is compromised, but maintaining at least the goal position. In case the metric exceeds the safety threshold, the controller simply abandons the end-effector goal.

Depth-Based Approaches In the last decade, the usage of visual sensors with depth information has become essential to develop a 3d collision avoidance system for robotic arms, even though laser scanners are sufficient for mobile robot bases. Microsoft Kinect is a low-cost depth sensor that can record helpful 3d information to measure distance between objects. The depth information is projected in the robot-centric space and approximate representations of obstacles are built to measure distances. Depth data based collision avoidance implementations are quite a few. In [Bascetta 2010], the authors presented two approaches to avoid collisions in Cartesian space using a laser time of flight sensors—one that preserves the geometrical properties of the trajectory, and another that preserves its time properties. In [Schiavi 2009], both active collision avoidance and passive impedance control in configuration space are discussed to improve the robot safety. [Flacco 2012] used a classic potential field method to generate repulsive commands to avoid collisions with a KUKA LWR IV arm in a dynamic environment. A concept of depth space in proposed to evaluate distances between the robot and the obstacles with estimated velocities from 3d sensors. The virtual force vectors from the distances and velocities measured using the 3d sensors are used to avoid collisions, while executing a generic motion task. [Yang 2010, Yang 2006] presented a solution based on elastic roadmap, claiming to generate robust and task-consistent motions for redundant robots, but not completely verified experimentally. [Pan 2013] proposed a real-time collision detection and distance query algorithm, which is particularly efficient in handling large amounts of point-cloud information. Even though the algorithm has been implemented, there is no record of a complete experimental verification and the software is unavailable.

Optimization-Based Approaches The idea of modeling obstacle avoidance as a linear inequality constraint was originally introduced in [Faverjon 1987]. This approach is a nice alternative to potential fields to generate local collision-free motions for robotic manipulators. In potential-field methods, the controller generates trajectories based on the sum of the repulsive fields coming from the obstacles, and the attractive field coming from the target. The solution work well for few obstacles, but it struggles in complex environment as influences of various nature are added in a single function. Some well-known issues are [Faverjon 1987]:

- Oscillatory repulsions between opposite obstacles. This behavior can be observed when a mobile robot tries to navigate through a narrows passage.

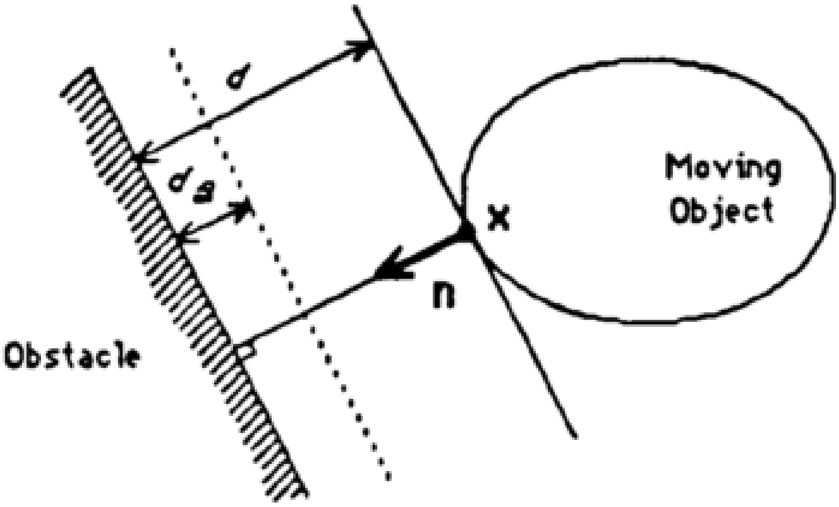


Figure 2.1: Velocity damper constraint [Faverjon 1987]

- Higher repulsions from multiple adjacent obstacles than the repulsion from a single obstacle due to the summation of all forces in a single function.
- Inability to generate trajectories close to obstacles.

In [Faverjon 1987], the main idea is to separate the main task from the collision-avoidance constraints. A main task can be to follow a given trajectory, or to reach a given posture, which can be easily formulated as a minimization of an objective function. The collision avoidance constraints are modeled as geometric constraints in configuration space that are not added in the objective function, allowing to generate improved trajectories. The collision avoidance constraints are defined as linear inequalities:

$$\dot{d} \geq -K\frac{(d-d_s)}{(d_i-d_s)} \qquad \forall d \leq d_i \tag{2.1}$$

This technique is called a *velocity damper*. First of all, this constraint is active only when the separation distance d is less than the distance of influence d_i. In this case, (2.1) constrains d not to decrease too fast, so that d cannot be less than the security distance d_s. The gain K must be tuned depending upon the application. Fig. 2.1 shows the interaction between the moving object and the obstacle describing the *velocity damper* constraint. This control strategy has been implemented on redundant systems to verify the validity of the approach.

This approach [Faverjon 1987] was later extended in [Kanehiro 2008] to handle arbitrary polyhedral objects (i.e. not necessarily convex). The closest points between the objects move discontinuously because of the non-convexity, resulting in discontinuous velocities when a *velocity damper* constraint is applied to avoid collision. The continuity in interaction is achieved by decomposing the interaction between polyhedral objects into a set of interactions between triangular faces of the polyhedra and the edges of triangular faces of the polyhedra. The velocity damping constraints are generated with pairs of points between an edge and a triangle, chosen based on the Voronoi[1] regions in which the edge lies.

An often overlooked aspect of constraint-based methods is the potential incompatibility between the considered constraints. This problem has been thoroughly investigated for the case of joint position, velocity and acceleration bounds [Decré 2009, Rubrecht 2012, Del Prete 2018]. Computationally efficient solutions of this problem exist for the case of constant bounds. However, these approaches do not easily extend to obstacle avoidance because velocity and acceleration bounds are not constant in Cartesian space. Recently in [Meguenani 2016], the authors model energy related safety indicators as constraints in the controller to ensure safety.

2.2.1.2 Intrinsic Approaches

The idea of strictly avoiding collisions may not be practical in some setups because of a variety of reasons, e.g. robot type, sensing, necessity to guide the robot. As a result, an appealing alternative is to focus on post-collision mechanisms to reduce the impact of collisions and react to it. These methods use quantitative measures to guarantee that a robot motion is not harmful to the human beings after detection of collisions, and they react depending on the task requirement. This is done by constraining the joint velocities, energy, or exertion of forces on the robot body. First, let us have a look at some quantitative measures to ensure safety in a collaborative environment, which is followed by the popular approaches to date.

Safety Measurements The trajectory planner SoftMotion [Broquere 2008, Broquere 2010] can generate trajectories satisfying jerk, acceleration and velocity limits. This approach is based on the 7 segment acceleration profile [Castain 1984], computing connected series of cubic curves for both point to point and continuous motions. The generated trajectory is smoothed at each way point, respecting the error and kinematic bounds in real time. The real-time capability of this trajectory planner makes it appropriate to be used for reacting to unforeseen situations online. Though the proposed planner can be used online for collaborative robot applications [Zhao 2014], it is not complete, because it can react only if the current joint accelerations are zero [Wahl 2010]. Rather than planning trajectories, other approaches focused on real-time tracking and limiting the appropriate quantities to ensure safety. The approach in [Heinzmann 2003] limits the impact forces by constraining the torque generated by a position controller, such that it complies with safety restrictions, reducing the post-collision impact with human beings. To reduce the collision impacts, [Laffranchi 2009] implemented a position controller on a

[1] A Voronoi diagram is a partitioning of a plane into regions, based on a set of points in the plane. Each region is the set of points whose distance to one of the given points is less than or equal to the distance to all of the other points.

single-joint series elastic actuator, limiting the system energy (kinetic, gravitational and elastic potential energy) by modifying the reference commands online.

In [Haddadin 2012], the authors presented a unique approach based on an analysis of the relationship between the robot mass, velocity, impact geometry and resulting injury. Previous approaches relied on the relationships with exerted forces on the robot surfaces. On the contrary, this work carried out an injury analysis on abdominal samples from pigs, resulting in risk curves showing a relationship between impact speed, geometry, mass and the impacted body. These risk curves have been used to bound the robot velocity, ensuring safety in the case of unintended collisions.

Detection, Localization and Reaction Some methodologies proposed not only to detect, but also to localize and react to collisions in the right direction according to the goals of the scenario. The well-known work of [De Luca 2006] detects and localizes collisions using only joint positions, velocities, and commanded torques. This framework incorporates a energy measure—the sum of kinetic and gravitational potential energies—and generalized momentum to localize collisions, allowing for a safe reaction. The approach in [Geravand 2013], similarly to [De Luca 2006], detects collisions without torque sensing. This framework does not require a dynamic model and uses motor current measurements (instead of joint velocities) to distinctly detect unintentional and intentional collisions. A parallel use of high-pass and low-pass filters on the motor current measurements allows to switch between collaborative mode or evasive mode, based on the assumption that unintentional collisions generate a high-frequency signal, whereas intentional impacts generate a low-frequency signal. [Golz 2015] used a nonlinear support vector machine (SVM) to distinguish intentional and unintentional collisions using a physical contact model and data generated from real impacts. [De Luca 2009] developed a momentum-based collision detection system for a variable stiffness actuator without torque sensing, similar to [Geravand 2013]. The system uses nonlinear control to gently move the arm away from the collision, while limiting contact forces by rapidly reducing the stiffness to bring the arm to a halt.

Interactive Control There are special techniques to handle intentional collisions that are made by human beings to collaborate with the robot. The robot should reason about the intent of the human instead of just moving away or switching to another control mode. [De Luca 2012] proposed to switch to a collaborative mode based on user input, either through gestures or through speech. The contact forces are estimated continuously to identify and distinguish between allowed and un-allowed human contact body parts. The controller allows contacts with specific parts (e.g. hands) for collaboration, while it avoids contacts with other parts (e.g. the head). In an approach proposed in [Erdem 2011] for a back-drivable robot, the system determines the intent of the human by estimating the effort based on conservation of momentum, without measuring the joint torques or joint velocities.

Effectiveness of Collision Avoidance Schemes [Haddadin 2008] presented a collision detection and reaction concept with experimental analysis on human dummies to compare the post-impact force time-series profiles of four intrinsic collision reaction schemes.

1. Stopping immediately after a collision is detected.
2. Switching from position control to torque control with gravity compensation resulting in a compliant robot.
3. Switching from position control to torque control with gravity compensation. The difference with respect to 2) is that it uses joint torque feedback and estimated external torques to scale down motor and link inertia resulting in an apparently lighter robot.
4. Switching to admittance control using the estimated external torques. The robot moves away from the collision by setting a reference velocity to the joints opposite to the estimated external torque.

The strategies 2)-4) switch from trajectory following to sensor guided compliant control to actively react to a potential collision. The tests verified that all the approaches succeeded in avoiding danger to human beings at velocities up to 2.7 m/s. The same control schemes were also tested for collisions with a balloon in [De Luca 2006], in which the residual torque time-series during collision were compared.

[Vick 2013] evaluated their post-collision strategy on an industrial robot, which limits torques based on external forces. The effectiveness is tested in two cases: when a human pushes a stationary robot, and when a human interferes with a robot executing a sinusoidal motion. In the former case, the robot moves away from the collision source to limit the contact forces. In the latter case, the robot modifies the sinusoidal path to reduce the impact forces. [Haddadin 2011a] studied soft tissue injuries caused by robots holding sharp tools. They carried out experiments of stabbing and cutting motions on silicone and human beings with two control schemes: i) stop the robot when the collision is detected, or ii) switch from position control to torque control with zero gravity. The experiments proved to be effective at reducing forces and avoiding injury with robot velocities up to 0.75 m/s.

Force based control using skin sensors There are post-collision avoidance schemes using haptic sensors that can directly measure impact forces applied on the robot [De Luca 2006, De Luca 2004] to react to the collision. [Phan 2011] presented an approach using capacitive skin sensors to detect and localize the impact forces on the robot. Passive torsional springs have been used for variable stiffness in the joints, making it independent of the controller bandwidth limitations. The authors carried out an accurate comparison between skin sensors and joint force sensors, considering various parameters such as interface friction, interface stiffness, joint stiffness, and end-effector velocity, as they affect the severity of collisions.

[Shin 2011] presented the concept of instantaneous stiffness during collisions with experimental comparisons of peak impact accelerations at different impact magnitudes, on a robot using pneumatic muscles. [Killpack 2016] used model predictive control with an impact-momentum model in the objective function to regulate joint velocities specifically to reduce the impact forces from unexpected obstacles.

2.2.2 Remarks

In spite of a large variety of methods to avoid collisions with dynamic obstacles, robots are still used conservatively because of the lack of maturity of these approaches. Our control scheme falls in the category of *extrinsic* approaches, where the speed of the robot is adapted depending on the separation distance to the obstacle. Our framework relies on infrared-based proximity sensors in skin cells to measure distance information. The main advantage of these sensors is that they provide a direct distance measurement, rather than computing it after a computationally-expensive and error-prone processing, which is usually the case with other extrinsic sensors. These sensors allow us to perceive unstructured environments without the need for modeling obstacles. Moreover, they are not subject to occlusion issues, which can instead occur with static extrinsic sensors.

Other approaches exist that exploit distance sensors mounted on the robot to avoid collisions. The concept of distributed sensors was introduced in [Lumelsky 2000], focusing on the principles, methodology, and prototypes of sensitive skin-like devices to measure proximity, touch, pressure, and temperature. [Ceriani 2013] mounted distributed proximity sensors on the manipulator links, focusing on the optimal sensor placements according to safety and detection capabilities. The developed prototype with off-the-shelf infrared distance sensors has been used to validate the proposed approach. [Avanzini 2014], an extension of [Ceriani 2013], focused on improving safety by assessing the danger that a robot induces. In our approach, we did not have to focus on the optimal placement of the the skin sensors because they already cover the whole link surface.

The formulation of collision avoidance in our approach is based on [Faverjon 1987], using a velocity-damper function to constrain the velocities of the points of interest (i.e. the skin cell locations). We formulate the collision avoidance and the trajectory tracking as two separate tasks, with higher priority being given to the former, in order to ensure safety. The main difference of our work with respect to similar state-of-the-art approaches is that we used hierarchical control. This makes our optimization problem always feasible because rather than formulating the inequalities as hard constraints, we minimize their violations. This results in exactly the same solution in case the inequalities are feasible. However, if the inequalities are unfeasible, our formulation still allows for a solution, while a non-hierarchical formulation would be unfeasible. We will see in Section 2.5.2 that this property can be extremely useful in certain complex scenarios.

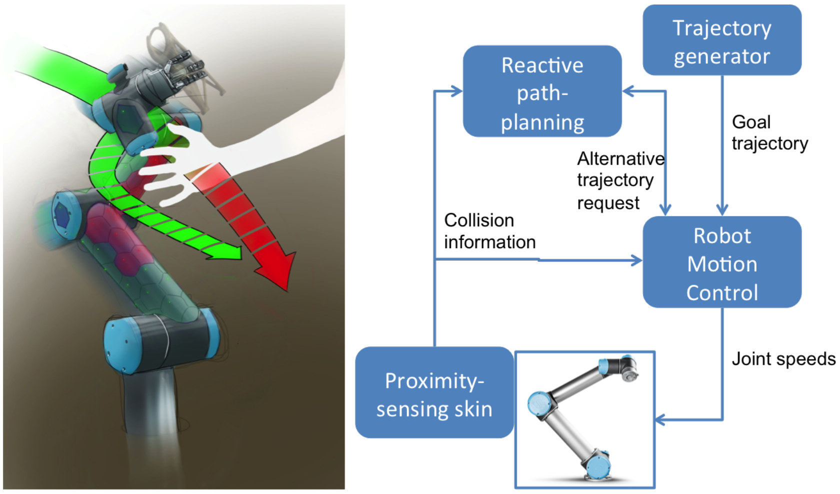


Figure 2.2: An artist's schematization of the Dynamic Obstacle Avoidance (DOA) concept.

2.3 Framework Components

This section provides an overview of the different components of the dynamic obstacle avoidance framework.

2.3.1 Overview

Current collaborative robot solutions guarantee safety, but they stop moving when an obstacle is detected rather than adapting the motion. The objective of the proposed dynamic obstacle avoidance framework is to detect obstacles and move around them while accomplishing the desired tasks. The framework relies on a dynamic motion planner that can fulfill various task specific constraints for typical industrial applications. For example, the work cell 3D model is used to create a consistent model of the work environment, so that collision-free trajectories are flexibly generated for different operations.

An illustration of the proposed dynamic obstacle avoidance solution is shown in Fig. 2.2. The robot motion control component (see Section 2.4) generates the joint velocities set-points for the robot controller. This component uses the proximity-sensing skin that covers the manipulator (see Section 2.3.2) to adapt the motion online to fulfill two objectives: i) following the reference trajectory, and ii) avoiding collisions. If the collision is unavoidable with local deformations of the current trajectory, the module may request a (global) re-planning to the reactive path-planner (see Section 2.3.3). The reactive path planner has been provided by our project partner, Siemens, and is not a contribution of this thesis. Our main focus is the reactive control scheme, which can be used with any path planner.

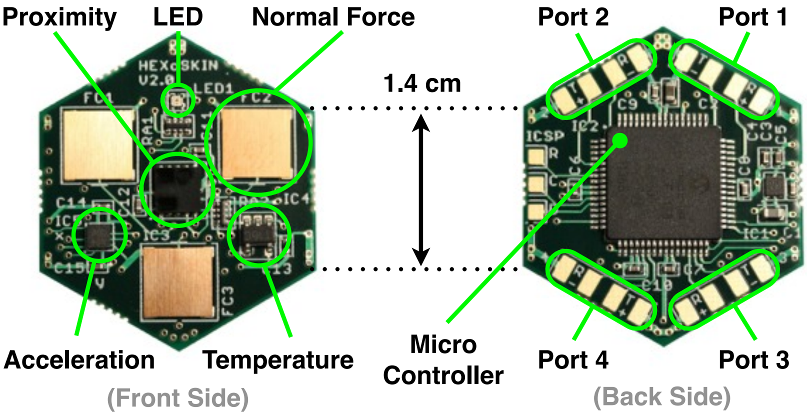


Figure 2.3: Robot skin developed at Institute for Cognitive Systems, TUM.

2.3.2 Artificial Robot Skin

The development of *Artificial Robot Skin*(ARS) is motivated by the necessity to provide robots with a rich and direct feedback of their interactions with the world. This system, called HEX-o-SKIN, assembles multiple intelligent uniform unit cells with cell-2-cell communication, thus allowing automatic cellular network organization [Mittendorfer 2012b]. The robot skin system is modularized and transduces multi-modal tactile stimuli [Mittendorfer 2015]. The robot skin consists of hexagonally shaped Printed Circuit Board (PCB) modules called *skin cells* (see Fig. 2.3). A group of directly connected skin cells is called a *skin patch*. All skin cells are identical and contain the same set of sensors. The sensors sample 9 tactile stimuli of 4 different modalities, namely vibration (3D acceleration sensor), 3 normal forces (capacitive force sensor), 2 temperatures and 1 distance (optical proximity sensor). These sensors are either off-the-shelf standard integrated circuits or, in the case of the force sensors, in-house developments. A micro-controller in the back of each skin cell collects data from its sensors, filters it, and sends data packets containing the most recent values of all sensors. All the skin cells are connected to each other via stretchable flex PCBs, which allows the skin to cover curved surfaces and increases its robustness. The network of skin cells is a meshed bidirectional communication network, which is routed by the micro-controllers of the skin cells. A self-organized algorithm initializes all the skin cells in a skin network and constructs a bidirectional communication path between each skin cell and the network root, the tactile section unit (TSU). The TSU converts skin network packets to standard UDP Ethernet packets and vice versa. This allows for fast connections between the robot skin and the PC (see Fig. 2.4). The skin also supports the auto-calibration of spatial relationships between skin cells of a skin patch [Mittendorfer 2012a], such that the pose of every skin cell with respect to the base frame can be easily determined. The proximity sensors used in the skin cells are infrared-based sensors. The sensor emits infrared light and captures its reflections on obstacles in the range from 0 to 6 cm. The strength of the reflections allows the sensor to estimate the

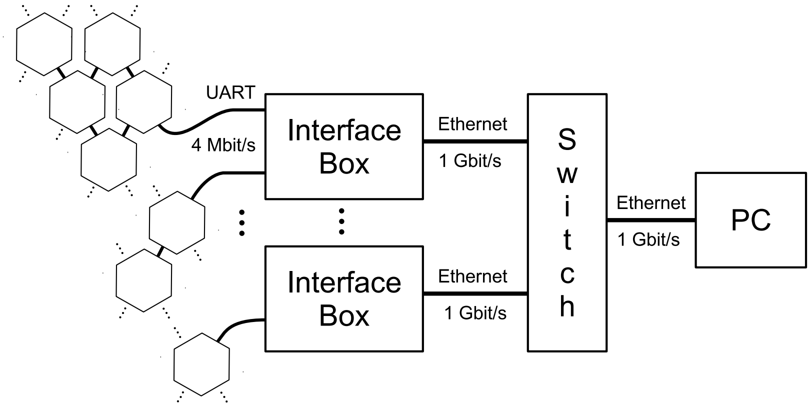


Figure 2.4: The skin cell network architecture and interface to the PC.

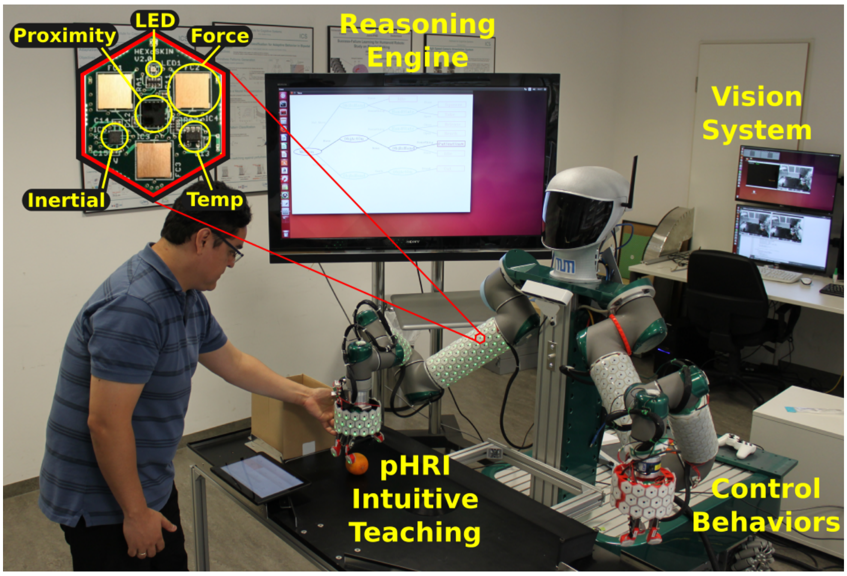


Figure 2.5: Robot TOMM with artificial robot skin.

distance to the detected objects.

Evaluation of Artificial Robot Skin(ARS) The ARS has been successfully deployed on the robot TOMM [Dean-Leon 2017] (see Fig. 2.5). The integration of the ARS signals in the control loop has been demonstrated in [Dean-Leon 2016a], where the controller exploited the ARS to produce compliance in a non-compliant industrial robot. The main advantage of these compliant behaviors is their higher safety in case of physical Human-Robot Interaction. The fusion of the multi-modal signals of the ARS with different sensors (e.g. cameras and joint encoders) in a semantic level has been demonstrated in [Karinne Ramirez-Amaro 2016]. These semantic representations are used to extract general task structures, which together with the obtained knowledge can improve and accelerate the teaching of new tasks [llya Dianov 2016]. Finally, the integration of these technologies has been

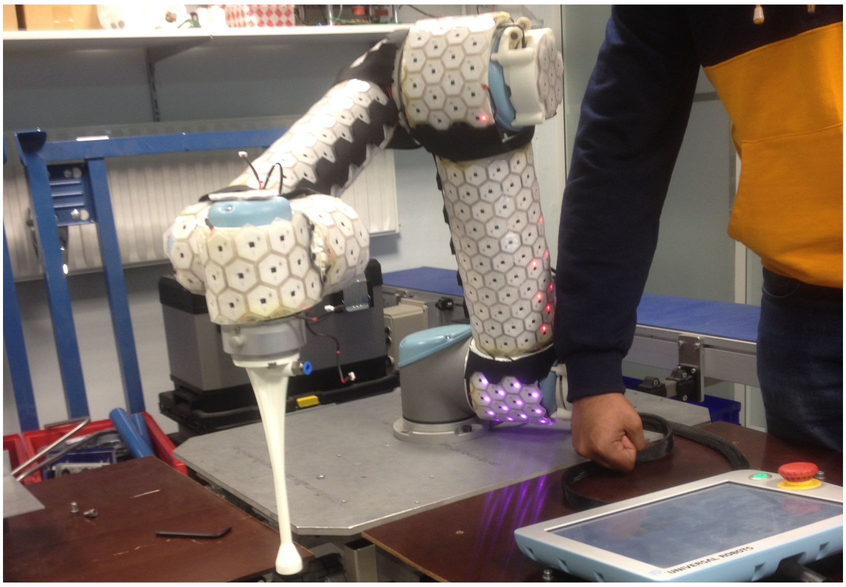


Figure 2.6: UR5 setup with Skin Cells activated (with red LEDs)

evaluated in an industrial scenario, where a human can kinesthetically teach the robot TOMM to sort oranges [Dean-Leon 2016b] (see Fig. 2.5). The ARS has also been successfully deployed on another practical setup with a statically mounted Universal Robots UR5 robot. In this setup, the ARS is being used to provide proximity information related to obstacles in the immediate surroundings of the robot. The proposed reactive control scheme (discussed in the next section) has been experimentally verified in this setup, shown in Fig. 2.6.

2.3.3 Motion Planning

The motion planning component is required in this framework to generate a global plan, represented as a sequence of way points. We integrated a path planner developed by our partner Siemens, which is based on the industry grade KineoWorks™[2] path planning library. KineoWorks library includes innovative algorithms to address path planning, distance computation and collision checking which is basically a result of research activities done in LAAS-CNRS. The algorithms use modern *Probabilistic Roadmaps* to solve the planning problem which are basically graphs with nodes referring to collision free configurations and edges referring to collision free path. A collision checker is used to find collision free configurations at random from the roadmaps which actually captures both the coverage and the topological connectivity of the space. Another advantage of the current state of the planner is its ability to use point-cloud data to compute a collision-free path in an unstructured or un-modeled environment to reach the desired goal configuration. It also allows to add static obstacles present in the workspace, like most planners. The exact algorithm used to achieve this is never disclosed until now for the ob-

[2] See Kineoworks.

vious commercial reasons. The main advantage of 3d sensors is that they provide a global view of the environment, contrary to proximity sensors on the surface of the robot, which provide only local distance measurements. Though skin sensors can be used to implement reactive obstacle avoidance behaviors (as discussed in the next section), a collision-free path generated using point-cloud data can help the controller to recover from *local minima*.

The collision detection in the planner is performed using the Kineo^{TM}Collision Detector (KCD)[3]. KCD performs 3D collision detection and minimal distance analysis between triangular mesh surfaces in assembly environments. KCD has been designed specifically to minimize memory usage and take advantage of parallel processing. The component is synchronized with an OctoMap[4] based 3D occupancy grid map, which is updated at 30Hz with the point clouds acquired by an Xtion or Kinect camera sensor.

Though it is essential to replan when the robot is stuck in *local minima*, due to the practical unavailability of 3d sensors in the experimental setup, we have used the planner only to plan a trajectory at the beginning of the experiment. The planner generates the reference path as a polygonal line connecting a sequence of way points in joint space. These way points can then be connected using a trajectory generation library that takes into account the joint velocity, acceleration and jerk limits (such as Reflexxes [Kröger 2011] or Softmotion [Broquere 2008]). Alternatively, the ways points can simply be connected by straight lines, using an arbitrary time parametrization, which is the approach that we used in our experiments. A trajectory sequencer in the framework takes care of this linear interpolation to generate instantaneous desired posture command for the reactive controller.

2.3.3.1 Software Integration

This framework has been seamlessly integrated into the ROS-ecosystem via a ROS package called `kws_ros_interface`, which provides the planner implementations of KineoWorks as shared objects that are readily usable in ROS-based software via the `kws_ros_planner` ROS node. Robot kinematic models are provided to KineoWorks in the Unified Robot Description Format (URDF), which is a ROS standard. Furthermore, KineoWorks also accepts the standard ROS representation of a `PointCloud`[5] for creating collision models of dynamic obstacles in the environment.

2.4 Reactive Collision Avoidance using SoT

Our reactive controller relies on the *Stack of Tasks* (SoT) [Mansard 2009a], a hierarchical control framework that implements the generalized inverse kinematics

[3] See KCD.

[4] a library that implements a 3D occupancy grid mapping approach, providing data structures and mapping algorithms in C++

[5] See http://wiki.ros.org/pcl

formalism by Hanafusa et al. [Hanafusa 1981, Mansard 2009c]. In this section, we introduce the *Stack of Tasks* controller framework and its integration in ROS based framework. It is followed by the formulation of collision avoidance as a inequality constraint and the task hierarchy for executing a planned path while avoiding collisions.

2.4.1 Stack of Tasks: State of the Art

Redundant systems are more and more popular due to their increased flexibility. However, their control is more complex as in general it is not possible to compute analytic inverse kinematic solutions. Task function based approaches (such as the SoT) uses numerical inversion techniques to resolve redundancy and minimize the task space errors [Samson 1991]. A systematic framework for the control of redundant systems has been proposed in [Siciliano 1991] to allow the execution of multiple tasks with strict priorities. This framework could only handle equality tasks, so various strategies have been proposed to transform inequality constraints to equalities [Nelson 1995, Chan 1995, Mansard 2009b, Raunhardt 2007]. However, these strategies are not generic and lead to priority inversion issues, making them unreliable for practical use.

To overcome these issues, Kanoun et al. proposed to handle inequality constraints by solving a cascade of least-squares program [Kanoun 2011a]. This approach is generic and exact, but computational inefficient, due to the need of solving several optimization problems at each control cycle. Mansard et al. proposed an improved QP solver to manage multiple equality and inequality tasks in a prioritized hierarchy [Escande 2014b]. Kanoun et al. [Kanoun 2011b] and De lasa et al. [de Lasa 2010] used a primal active-set algorithm, which is computationally expensive due to the active-set search involving inappropriate activation and deactivation of constraints at each level along the cascade. This efficiency issue has been addressed by a dedicated Hierarchical Quadratic Programming (HQP) solver, which is based on the Complete Orthogonal Decomposition (rather than the more expensive Singular Value Decomposition) and an improved active-set algorithm [Escande 2014b]. The HQP solver relies on a modified primal active-set algorithm, which is tailored for hierarchical problems. The solver is ten times faster than the classical solvers and can consider inequalities at any levels of the hierarchy [Escande 2014b].

2.4.2 What is a Task?

A task is a control objective defined by a function of the robot state. For equality tasks, the goal of the controller is to minimize the value of the task function (i.e. bring it as close as possible to zero). For inequality tasks, the goal is instead to keep this value negative (or positive). As an example, a task-space reaching task can be defined by a function measuring the distance between the current end-effector position $x(q)$ and the desired end-effector position x^*:

$$e(q) = x^* - x(q),$$

where q contains the current joint angles. Other typical tasks are: reaching a desired joint configuration, avoiding obstacles, or grasping an object.

2.4.3 Redundancy Formalism

Siciliano and Slotine [Siciliano 1991] have been the first ones to propose a systematic control framework to achieve multiple tasks in redundant systems. The key idea is to solve each task in the null space of all higher priority tasks, to ensure the satisfaction of the strict priorities. In other words, the achievement of any task cannot be compromised by the lower-priority tasks.

Let $x_1(q)$ be the task-space position of the first (highest priority) task, and $J_1(q)$ be its Jacobian matrix, defined by:

$$\dot{x}_1 = J_1(q)\dot{q} \tag{2.2}$$

We can then define the task error as $e_1(q) = x_1(q) - x_1^*$, and we can define a desired exponential convergence rate of this error towards zero:

$$\dot{e}_1^* = -K_1 e_1, \tag{2.3}$$

where K_1 is a symmetric positive-definite gain matrix. Now we can compute the desired joint velocities as:

$$\dot{q} = J_1^+ \dot{e}_1^* + P_1 z, \tag{2.4}$$

where J_1^+ is the Moore-Penrose pseudo-inverse of J_1, P_1 is a projector in the null space of J_1, and z is an arbitrary velocity vector, which can be exploited to achieve any secondary objective.

We can easily extend this methodology to an arbitrary number of tasks n. Let $(e_1, J_1)(e_2, J_2)...(e_n, J_n)$ be the errors and Jacobians associated to the n tasks. The desired joint velocities to achieve all these tasks can be computed as:

$$\dot{q}_i = \dot{q}_{i-1} + (J_i P_{i-1}^A)^+ (\dot{e}_i - J_i \dot{q}_{i-1}), \qquad i = 1 \dots n, \tag{2.5}$$

where P_i^A is the projector onto the null space of the augmented Jacobian $J_i^A = (J_1 \dots J_i)$. The algorithm is initialized with $\dot{q}_0 = 0$ and $P_0^A = I$. The joint velocities achieving all the task objectives are $\dot{q} = \dot{q}_n$. The null space projectors can be computed recursively using the following expression:

$$P_i^A = P_{i-1}^A - (J_i P_{i-1}^A)^+ J_i P_{i-1}^A \tag{2.6}$$

This systematic way of prioritizing tasks allows simultaneous execution of potentially conflicting tasks, ensuring the satisfaction of the given priority order.

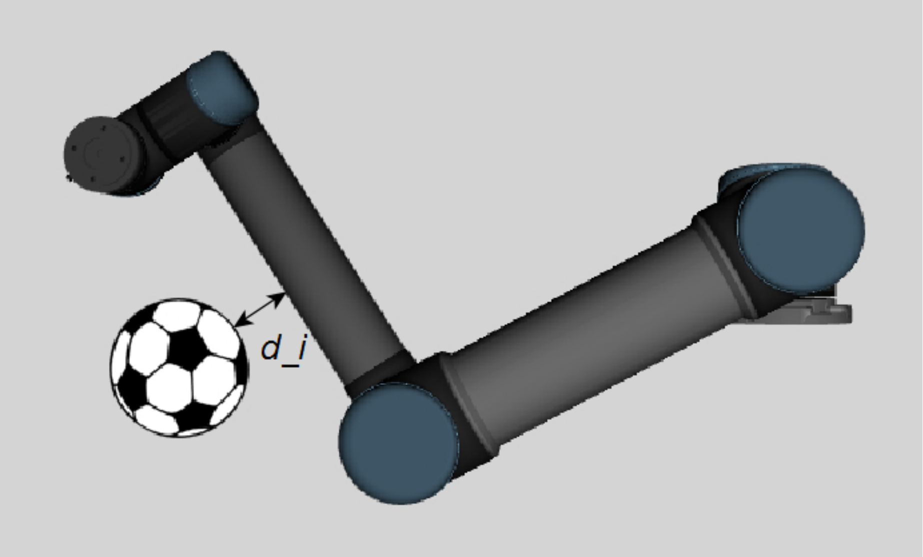


Figure 2.7: Robot-Obstacle Interaction.

2.4.4 Collision Avoidance Task Formulation

Suppose an obstacle in the environment is sufficiently close to the robot to be perceived by its artificial skin sensors. In this case, we would like the robot to exploit the skin data to lower its velocity in the direction of the obstacle, so as to avoid a collision in the near future. The final goal is to maintain a minimum distance d_{min} (defined by the user) between the robot and any obstacle. The i-th skin cell measures its distance d_i (e.g. see Fig. 2.7) to the perceived obstacle (if any). Since we have no way to predict the obstacle motion, our best guess is to assume that it does not move, and thus the distance d_i depends only on the robot configuration q. We can easily express our control objective as an inequality constraint:

$$d_i(q) \geq d_{min} \tag{2.7}$$

This constraint is expressed as a function of the robot configuration q. However, our controller is formulated as an optimization problem of the joint velocities $\dot{q}$. This means that in order to include collision avoidance in our control framework we need to express it as a function of $\dot{q}$. A sufficient (but not necessary) condition to ensure the satisfaction of (2.7) is to bound the distance rate of change as a function of the distance:

$$\dot{d}_i(q) \geq -K(d_i(q) - d_{min}), \tag{2.8}$$

where K is a symmetric positive-definite matrix, representing the convergence gain. Intuitively, this ordinary differential inequality makes the robot slow down as the distance to the obstacle is approaching d_{min}. The satisfaction of (2.8) ensures that:

$$d_i(q) - d_{min} \geq (d_i(q_0) - d_{min})e^{-Kt}, \tag{2.9}$$

where q_0 is the robot configuration at time zero. This means that if the distance constraint (2.7) is satisfied at time zero, then the right-hand side of (2.9) is always positive (and converging to zero as t increases), which means that (2.7) will always be satisfied (under the assumption of a static obstacle). It should be noted that this way of modeling collision avoidance does not account for the limited acceleration capabilities of the system [Rubrecht 2012], so it may demand unfeasible accelerations if K is too large. A proper tuning of the parameter K is thus fundamental for a successful implementation on real robots. While different methods to account for limited acceleration capabilities exist [Decré 2009, Rubrecht 2012, Del Prete 2018], it is not clear how to estimate appropriate bounds on the robot accelerations in Cartesian space, because they depend on the joint configuration. For this reason, we preferred to use this heuristic method, which is simpler to implement and works well in practice—after a proper tuning of K.

We can now express (2.8) as a linear function of the joint velocities, in order to use it in the SoT control framework:

$$-\frac{\partial d_i(q)}{\partial q}\dot{q} \leq -K(d_{min} - d_i(q)) \tag{2.10}$$

2.4.4.1 Distance Gradient

In order to implement (2.10) in our controller we need to compute the gradient of the distance functions $d_i(q)$ [Lefebvre 2005]. Consider a multi-body robot with n_c skin cells. Let $c_i(q)$ be the 3d position of the skin cell i at a robot configuration q, and $\mathcal{O}$ be the set of points occupied by the obstacle. By moving from $c_i(q)$ along the direction n_i, normal to the i-th skin cell, we may (or may not) meet the obstacle $\mathcal{O}$. In case we do, we call the first intersection point o_i. In case we do not, this means that the obstacle is not *visible* from the i-th skin cell, and we can assume o_i to be infinite. We can then represent the cell-obstacle distance as:

$$d_i(q) = \text{dist}(c_i(q), \mathcal{O}) = ||c_i(q) - o_i|| \tag{2.11}$$

The gradient of this distance with respect to the robot configuration is:

$$\frac{\partial d_i(q)}{\partial q} = n^T \frac{\partial c_i(q)}{\partial q},$$

where:

$$n = \frac{c_i(q) - o_i}{\|c_i(q) - o_i\|}$$

This shows that the distance gradient is simply a projection of the Jacobian associated to the skin cell on the direction normal to the skin cell.

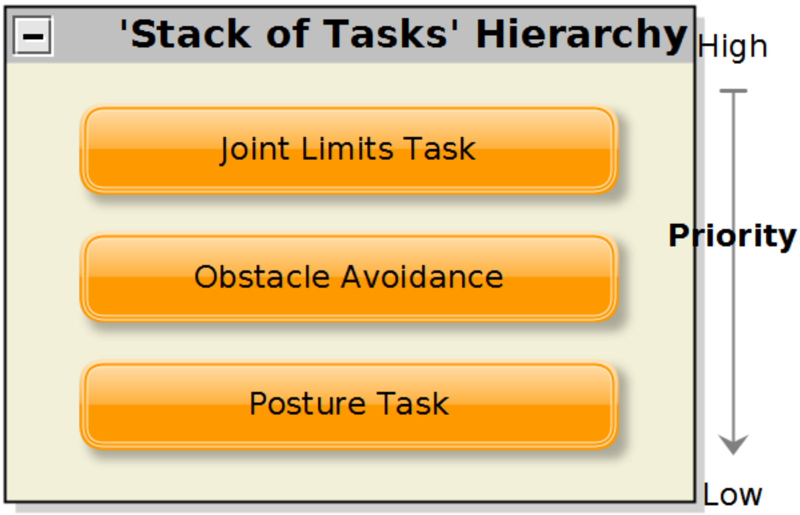


Figure 2.8: Stack order for trajectory tracking with obstacle avoidance.

2.4.4.2 Combining Trajectory Tracking and Obstacle Avoidance

Now that we have formulated collision avoidance as an inequality constraint, we need to decide how to combine it with the trajectory tracking task that the robot has to execute. The SoT framework allows us to specify multiple tasks that the robot has to achieve at the same time. The only thing that we have to decide is the priority order of the different tasks. Since we consider safety to be more important than trajectory tracking, we give higher priority to the collision avoidance task. We also include a joint limit task, which consists in not violating the joint position bounds. Fig. 2.8 shows the stack priority order.

T1 Joint Limits
$K_1(q_{min} - q) \leq \dot{q} \leq K_1(q_{max} - q)$
where q_{min} and q_{max} are the joint position bounds.

T2 Obstacle Avoidance
$\dot{d}(q) \geq K_2(d_{min} - d(q))$
where $d(q)$ is a vector containing all the distance measurements.

T3 Joint Trajectory Tracking
$\dot{q} = -K_3(q - q_{ref})$
where q_{ref} is the reference trajectory.

Once the tasks are specified, the SoT controller computes the desired joint velocities $\dot{q}$ that minimize the task tracking errors in a lexicographic sense (i.e. each task error is minimized under the constraint of not affecting the errors of the higher priority tasks).

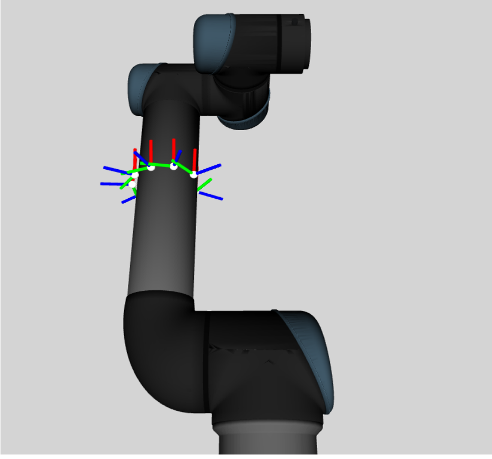


Figure 2.9: Skin cell sensors used in our experiments. They form a “ring” on the upper arm of the UR5 robot.

Table 2.1: Experiment Parameters.

	Meaning	Value
K_1	Joint Limits Gain	$0.2\ s^{-1}$
K_2	Obstacle Avoidance Gain	$0.1\ s^{-1}$
K_3	Posture Gain	$1\ s^{-1}$
d_{min}	Safety Distance	$0.058\ m$
dt	Controller Time Step	$0.01s$

2.5 Experiments

The presented Stack of Tasks (SoT) controller with collision avoidance constraints has been implemented and tested, both in simulation and on the UR5 robot setup with skin sensors (see Fig. 2.6). The values of the parameters used in our tests are summarized in Table 2.1.

2.5.1 Experiments with a UR5 Robot

The reactive collision avoidance has been experimentally tested on a UR5 robot with the skin sensor setup. The skin sensor network consists of approximately 300 cells, covering the entire surface of the UR5 robot arm. While using all the skin cells would increase the collision avoidance capabilities of the system, it would require a large number of inequalities in the solver. In our experience, the current HQP solver cannot handle such a large number of inequalities, probably because of numerical stability problems. For this reason, in our experiments we only used eight skin cells, located in the upper arm of the robot. These skin cells are positioned in a ring shape, as shown in Fig. 2.9; this symmetric configuration allows the robot to perceive obstacles from any direction.

We divided the experiments into two groups (called test 1, and test 2 in the following), according to the reference trajectory tracked by the controller. Though we carried out more tests, we decided to focus on only two of them to illustrate the performance of the controller. All reference trajectories started from or went to one of these three configurations: Home position, Pick position, and Place position. To show the capability of the system to perceive obstacles in any direction we repeated each test several times, placing the obstacle in different positions. A box has been used as the obstacle to disturb the robot movement. Moreover, we repeated each test with and without the collision avoidance task, to show the different behaviors of the system. The repetitiveness of these tests is meant to highlight the robustness of the controller. Additionally, a complete manipulation scenario is described in Section 2.5.1.3, to illustrate a real-world application of our framework.

The selected tests can be seen in the following videos:

- Test 1: Home to Pick
- Test 2: Pick to Place

2.5.1.1 Test 1

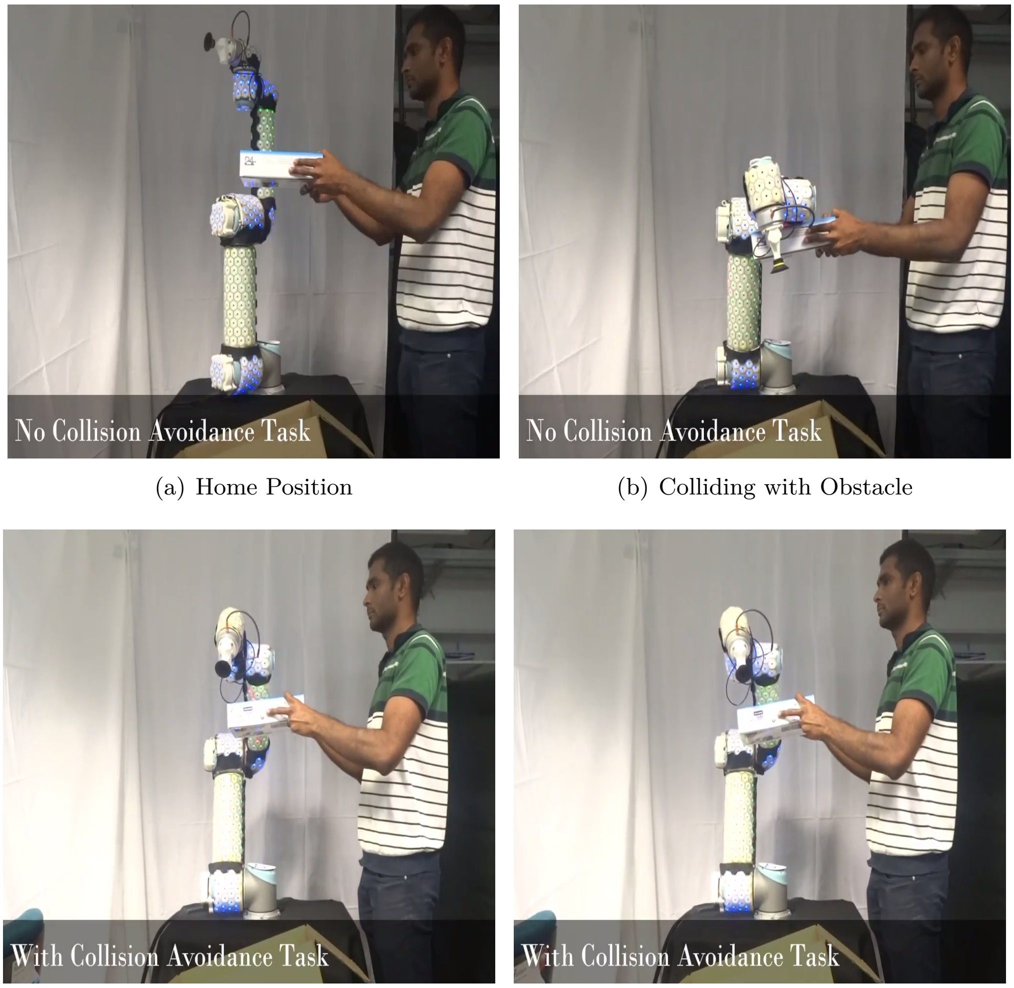
Fig. 2.10 shows some snapshots taken from the video of test 1, while the robot is executing a Home to Pick motion, with a fixed obstacle location. The plots in Fig. 2.11 show the trajectory execution *without* the collision avoidance task. As it can be seen both in the video and in the plots, the trajectory execution was not affected by the approaching obstacle. The distance measurement goes to zero slightly after 8 s, showing that the arm collided with the obstacle. As expected, the joint motion is smooth, without any disturbances.

It may be surprising to see that the tracking error gets large (above 1 rad) before the collision with the obstacle. The poor tracking is due to the fact that the proportional gain of the joint tracking task was small (see Table 2.1). This gain has been tuned to improve the joint tracking in the final application described in Section 2.5.1.3.

Fig. 2.12 shows the results of test 1 *with* the collision avoidance task. This time, thanks to the collision avoidance task, the robot did not collide with the obstacle. We can clearly see in the top plot of Fig. 2.12 that the distance measurement remains always above 1 cm. The three bottom plots show that the tracking error does not converge to zero, but keeps oscillating. This is because, given the obstacle location, the controller cannot reach the goal without colliding with it (i.e. the controller is stuck in local minima).

2.5.1.2 Test 2

Fig. 2.13 shows some snapshots taken from the video of test 2, while the robot is executing a Pick to Place motion, with a fixed obstacle location. The plots in Fig. 2.14 show the trajectory execution *without* the collision avoidance task. As it can be seen in the plots, the trajectory execution was not affected by the


No Collision Avoidance Task
No Collision Avoidance Task
With Collision Avoidance Task
With Collision Avoidance Task

(a) Home Position

(b) Colliding with Obstacle

(c) Avoiding Local Collisions

(d) Stuck in Local Minima

Figure 2.10: Test 1: Trajectory execution from home to pick location with and without collision avoidance task in the controller stack. (a) Initial state of the robot, i.e. Home position. (b) Trajectory execution without any collision avoidance task, leading to a collision with the obstacle. (c) Avoiding collision with the obstacle (d) The robot arm is stuck in local minimum.

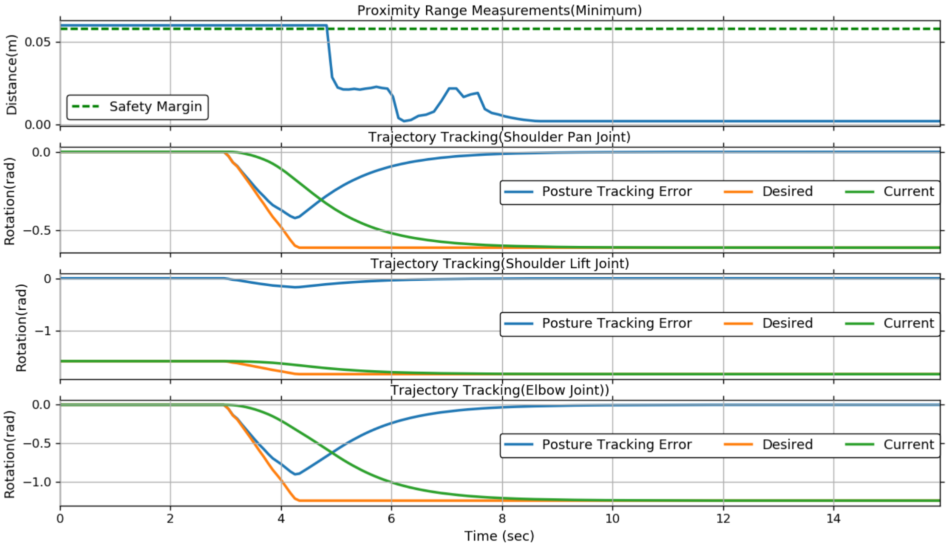


Figure 2.11: Test 1 without collision avoidance. These plots show a Home to Pick trajectory tracking. The top plot shows the minimum distance measurement from the 8 skin cells. The distance drops close to zero, showing that the obstacle touched the robot arm. The three bottom plots show the trajectory tracking for the three main joints.

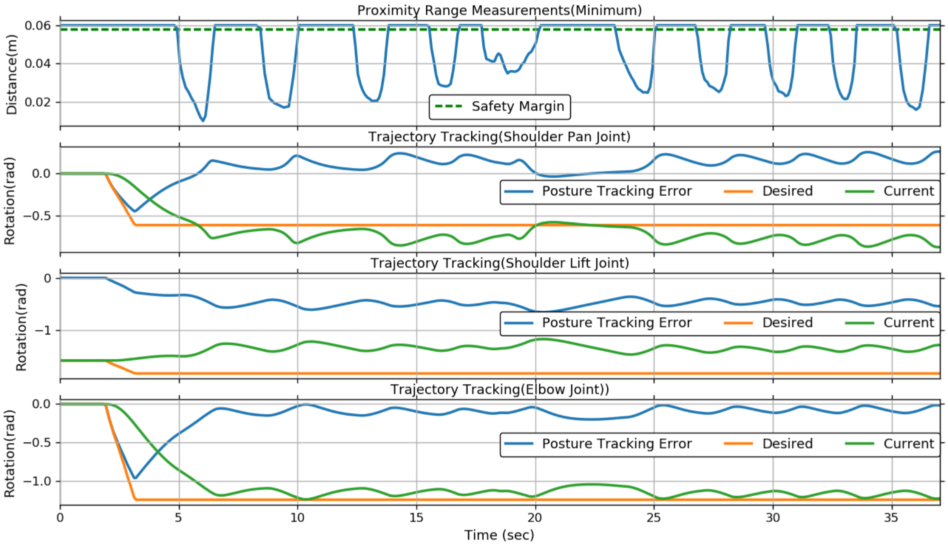


Figure 2.12: Test 1 with collision avoidance. Home to Pick Trajectory Execution. The top plot shows the minimum distance measurement from the 8 skin cells. The three bottom plots show the trajectory tracking for the three main joints.

approaching obstacle. The distance measurement gets to zero at around 3s, showing that the arm collided with the obstacle.

The results of test 2 with collision avoidance are plotted in Fig. 2.15. The differences with respect to the previous test are: i) the reference joint trajectory, and ii) the obstacle location. Contrary to test 1, the controller did not get stuck in an oscillating behavior when trying to avoid collisions. This is due to the different location of the obstacle, which allows the controller to reach the goal just by locally deforming the trajectory. In the bottom plots of Fig. 2.15 we can see that the joint tracking error converged to zero.

2.5.1.3 Real-World Scenario

After proper tuning, the presented collision avoidance framework has been used in a complete manipulation scenario. The scenario involved multiple sequences of pick-and-place operations of boxes and shaving cans, using a suction gripper on the end-effector. The robot controller still used the same ring of skin cells in the upper arm. The demonstration video can be seen at this link, with snapshots shown in Fig. 2.18.

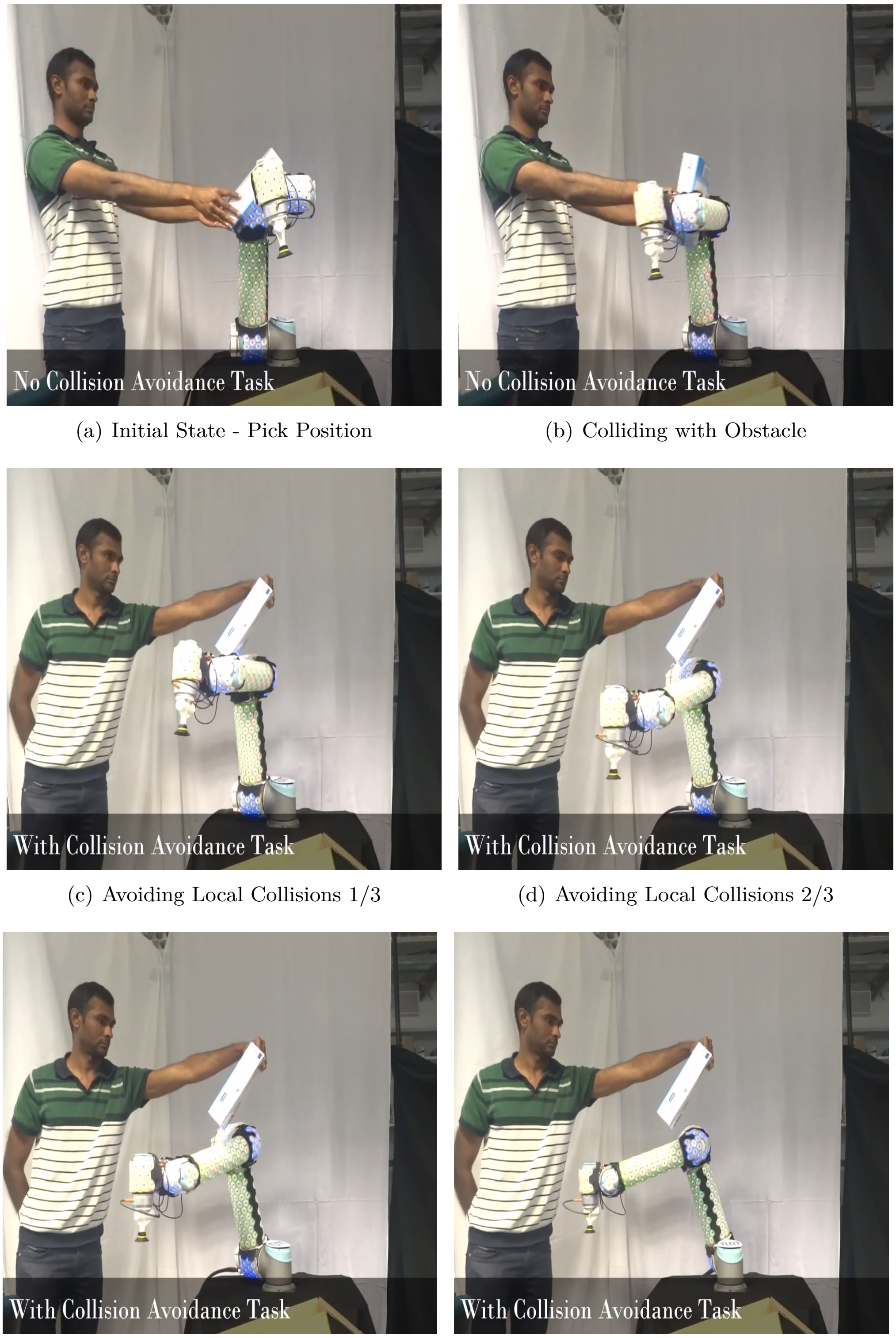
2.5.2 Simulations: Infeasible Constraints

Another interesting aspect of the controller is its ability to handle infeasible inequality constraints. This is due to the formulation of the HQP problem used in the SoT. Each task error is minimized (i.e. no task is formulated as a constraint), so that the problem is always feasible.

The collision avoidance task is formulated as the inequality constraint (2.10). If two (or more) skin cells on opposite sides of the arm perceive an object at the same time, their associated constraints may become conflicting, because they would require the arm to move in opposite directions. To illustrate this scenario, we performed two simulations in which we artificially generated the distance signals and observed the resulting controller behavior.

At first, we simulated the presence of an obstacle close to one of the skin cells, by lowering its distance measurement close to the safety margin d_{min}. While doing this, we kept the other skin cell measurements equal to 0.6, which is 10 times larger than d_{min}. The plot in Fig. 2.16 shows the change in velocity of a specific skin cell in response to the simulated approaching obstacle. The plot shows the feasibility of the goal, thus resulting in a behavior inline with the formulated collision avoidance constraint (2.7).

In the second simulation instead, we simulated multiple obstacles surrounding the arm, and thus making constraint (2.10) infeasible. We forced the distance measurement of one specific cell to cross the safety margin, while keeping the other measurements just 1mm above d_{min}. The plot in Fig. 2.17 shows the resulting skin cell velocity. Since the controller could not satisfy all the conflicting skin-cell velocity constraints, it found a compromise between them. Even if the constraints (2.10)


No Collision Avoidance Task
No Collision Avoidance Task
With Collision Avoidance Task
With Collision Avoidance Task
With Collision Avoidance Task
With Collision Avoidance Task

(a) Initial State - Pick Position

(b) Colliding with Obstacle

(c) Avoiding Local Collisions 1/3

(d) Avoiding Local Collisions 2/3

(e) Avoiding Local Collisions 3/3

(f) Final State - Place Position

Figure 2.13: Test 2: Trajectory execution from Pick to Place locations, with and without collision avoidance task. The robot avoiding the collision and reaching the goal can be seen in (c), (d), (e) and (f).

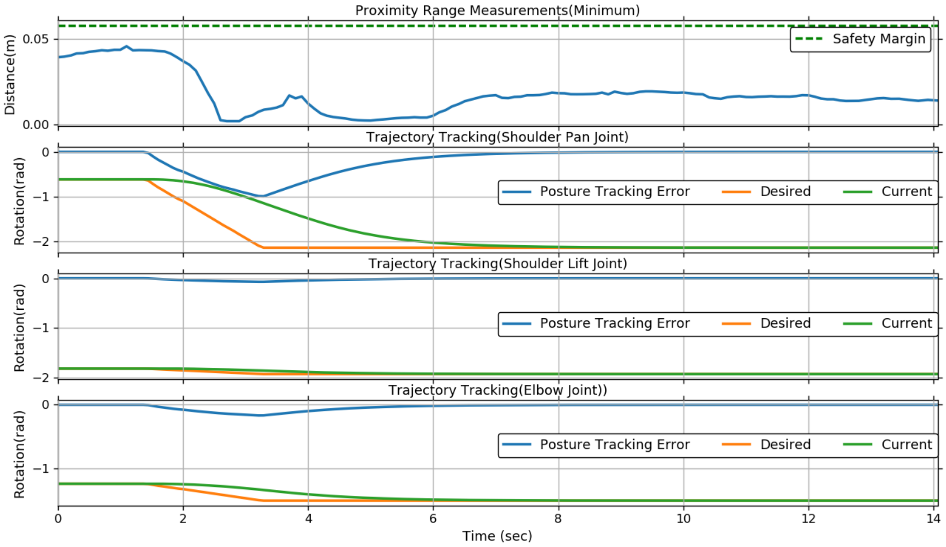


Figure 2.14: Test 2 without collision avoidance: Plots during Pick to Place trajectory tracking. The top plot shows the minimum distance measurement from the 8 skin cells. The distance drops close to zero, showing that the obstacle touched the robot arm. The three bottom plots show the trajectory tracking for the three main joints. There are no disturbances in the trajectory tracking when an obstacle is in collision..

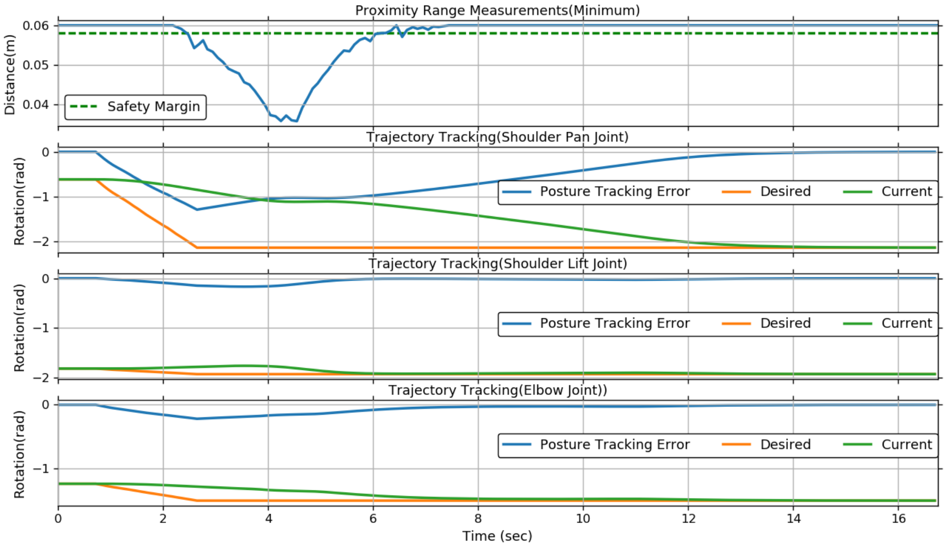


Figure 2.15: Test 2 with collision avoidance: Plot during a Pick to Place. The top plot shows the minimum distance measurement from the 8 skin cells. The three bottom plots show the trajectory tracking for the three main joints.

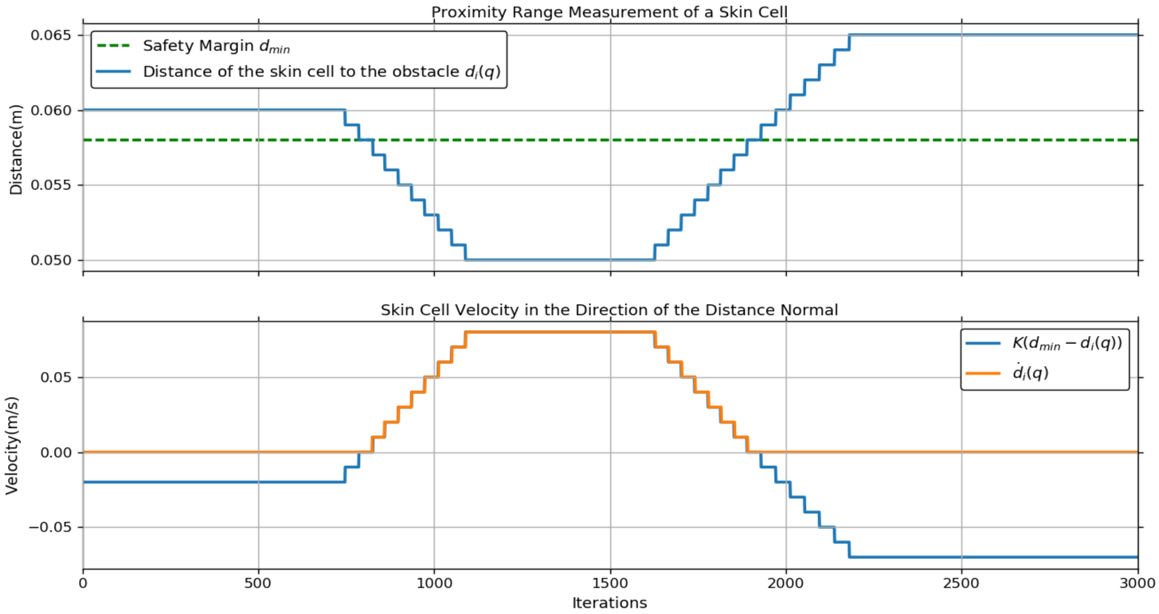


Figure 2.16: Plot showing the velocity of the skin cell i, when the other cells perceive no obstacles in the vicinity. In this case the inequality (2.10) is satisfied by the controller.

are temporarily violated, the resulting behavior is still reasonable and it does not lead to a crash of the solver.

2.6 Conclusions

This chapter presented a reactive collision-avoidance framework, developed in the factory-in-a-day EU project. Let us discuss briefly the implemented framework and the expected future work.

2.6.1 Skin Sensors for Distance Measurements

One of the key features of our framework is that it relies on the infra-red based proximity sensors to measure local distance information. The advantages of these sensors are:

- They allow for collision avoidance in unstructured environments, without the need for modeling the obstacles. Usually extrinsic sensors such as stereo camera or 3d camera are used to detect objects in the environment and an approximate model is used to fit the detected obstacle.

- Most collision avoidance methods estimate the robot-obstacle distance at run time. This distance is computed at every control cycle, which may be com-

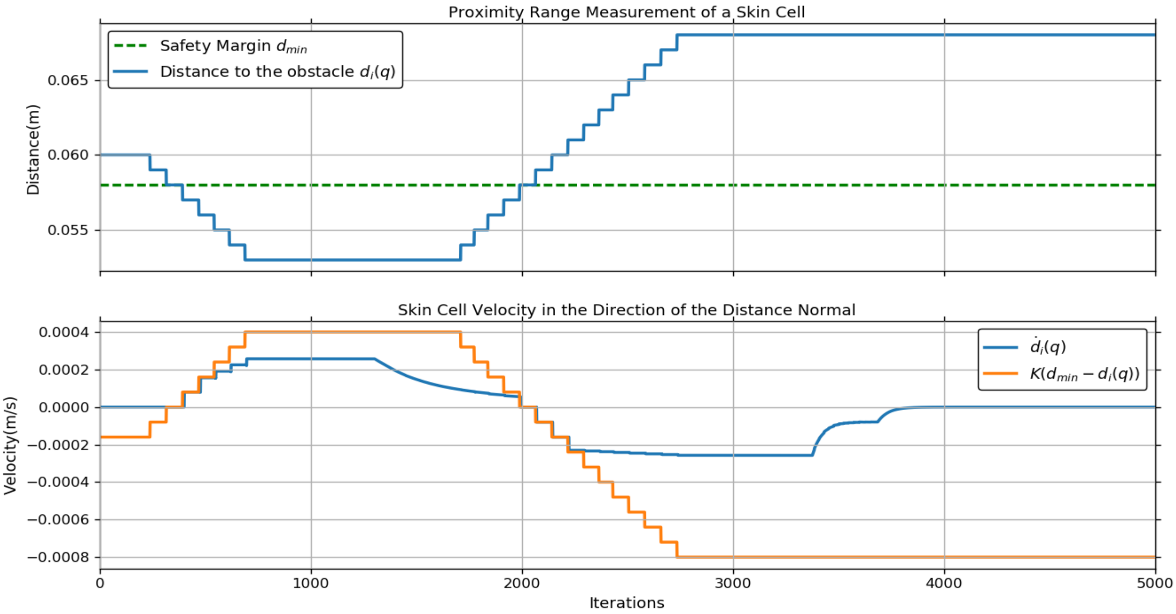
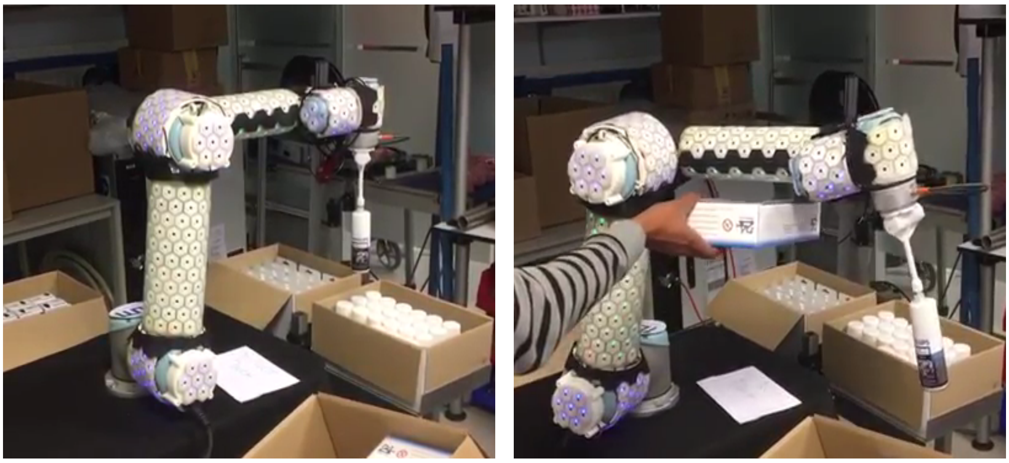


Figure 2.17: Plot showing the velocity of the skin cell i, when the other skin cells measure a distance of 6cm, which is just 2mm larger than the safety margin d_{min}. In this scenario, it is not possible to satisfy all the skin cell inequalities (2.10).



(a) Grasping with a suction gripper

(b) Avoiding Obstacles

Figure 2.18: Illustrating obstacle avoidance while picking and placing small objects using a suction gripper. The full scenario video can be seen in this video.

putational expensive. In the case of skin sensors, the distance is measured directly, assuming the skin cell poses are known.

Sensor Range Though the range of the skin sensors is limited to 6cm in the experimental prototype on the UR5 arm, after a proper tuning of the task gains, the robot has shown good collision avoidance behaviors. In the future, we hope to enhance the range of the skin cells to allow the robot to move faster and be more agile in avoiding obstacles. This would help increase the productivity of collaborative robots in industries.

2.6.2 Collision Avoidance as Inequalities

In our experiments, we have used only 8 skin cells covering the circumference of the upper arm. This was deliberately done to reduce the number of inequality constraints in the solver, which did not prove capable of handling a large number of inequalities. This problem has two potential solutions. One option would be to rely on an improved solver, capable of handling a large number of inequalities. However, to the best of our knowledge, hierarchical solvers are still an active research topic, and we are not aware of any reliable solver publicly available. Alternatively, we may try to improve our formulation, so as to limit the number of inequality constraints. For instance, we could model in the solver only the inequality constraints corresponding to the 8 skin cells that are currently measuring the smallest distances. We expect this simple approach to work well as long as no more than 8 skin cells are active at the same time (which is the case in our experiments). Other more complex solutions to reduce the number of inequality constraints may also be explored.

2.6.3 Reactive Replanning

Though the presented approach seems performed well in our tests, the controller tends to get stuck in local minima, which trigger the need for replanning. While it is possible to escape local minima by applying circular fields [Haddadin 2011b], the escape is not guaranteed because of the lack of global information. Extrinsic sensors (on top of the skin sensors) such as 3d cameras, are essential to reactively plan in case of unforeseen obstacles interrupting the task. Depth-based reactive planning [Flacco 2012, Dumonteil 2015] can also be useful to avoid local minima in an unstructured environment without building collision bodies.

2.6.4 Kinematic Redundancy

One of the main reasons for using task based control is to exploit the extra degrees of freedom in redundant robots to handle multiple tasks in parallel. The implemented controller has been tested on a fixed arm with 6 degrees of freedom, which do not provide much flexibility to reach a 3d pose with the end-effector. Mounting a manipulator on a mobile base makes the robot redundant, which increases the probability of success in case of collision avoidance. In the future, we would like to experiment on mobile robots with an arm equipped with a complete skin cell network, or at least a ring of skin cells covering the circumference of the arm. This will provide a way to verify whether improved collision avoidance behaviors are actually possible by exploiting the mobile base.

2.6.5 Final Integration

The collision avoidance component has been successfully integrated in the final project demonstration of Factory-in-a-day, which ran for more than 20 minutes, without any failure. We believe that the simplicity of the approach and its practical relevance make it an excellent candidate for industrial applications.

Chapter 3

Robustness to Inertial Parameter Errors for Legged Robots Balancing on Level Ground

Model-based control has become more and more popular in the legged robots community in the last ten years. The key idea is to exploit a model of the system to compute precise motor commands that result in the desired motion. This allows to improve the quality of the motion tracking, while using lower gains, leading so to higher compliance. However, the main flaw of this approach is typically its lack of robustness to modeling errors. In this chapter we focus on the robustness of inverse-dynamics control to errors in the inertial parameters of the robot. We assume these parameters to be known, but only with a certain accuracy. We then propose a computationally-efficient optimization-based controller that ensures the balance of the robot despite these uncertainties. We used the proposed controller in simulation to perform different reaching tasks with the HRP-2 humanoid robot, in the presence of various modeling errors.Comparisons against a standard inverse-dynamics controller through hundreds of simulations show the superiority of the proposed controller in ensuring the robot balance.

3.1 Introduction

The problem of balancing for real legged robots is still a challenge for the robotics community. Although our understanding of this problem has remarkably improved during the last 15 years, the robustness of the state-of-the-art control algorithms is far from satisfactory. For instance, during the recent DARPA Robotics Challenge Finals [Pratt 2015], all legged robots have moved extremely cautiously, and, despite that, sometimes they could not avoid falling. Another striking fact is the difference between what robots can do in simulation where they easily perform extremely dynamics tasks and what they can do in the real world where they struggle to execute slow movements in structured environments. The gap between simulation and real world can be explained through countless unmodeled uncertainties affecting these systems, such as poor torque control, model uncertainties, sensor noises and delays. In recent work of [Del Prete 2016a], an optimization-based controller tries to ensure the satisfaction of the physical constraints of the robot (force friction cones, joint acceleration limits and torque limits) despite errors in the joint torque tracking. In this work we move along the same line, designing a *robust* controller that can balance a legged robot despite bounded errors in its inertial parameters.

The chapter starts with a brief discussion about various control methodologies used in humanoid robots (Section 1.4). Section 3.2 presents robustness related work in optimization based control. In Section 3.3 we model the uncertainty in the inertial parameters of the robot through polytopes. Then we present the Task-Space Inverse Dynamics(TSID) controller with capture-point constraints [Ramos 2014] to ensure the balance of the robot in case of no modeling errors. Section 3.4 presents an extension of the standard capture-point inequalities that is robust to errors in the inertial parameters.We first formulate the associated robust optimization problem, and then use standard robust-optimization techniques to reformulate it in a tractable form. The Section 3.5 presents statistical results that compare in simulation the standard and the robust controller in a reaching task with the humanoid robot HRP-2. Regardless of the simulation conditions, our results empirically demonstrate the superiority of the proposed robust controller with respect to standard TSID. Finally, Section 3.6 draws the conclusions and discusses the future work.

3.2 Robustness in Humanoid Robots

Even though the problem of robustness is long-standing and well-identified, it remains largely unanswered for legged robots. Some approaches focus exclusively on the stability of the system rather than on the feasibility of the trajectories. For instance, adaptive control [Kelly 1989] and time-delay estimation [Jin 2008] try to estimate and compensate online for the major errors between nominal and real dynamic model. Virtual model control [Pratt 2001] does not rely on the dynamic model of the robot, which ensures robustness to errors in the inertial parameters [Dietrich 2013]. The main issue of these schemes is that they do not consider inequality constraints, which makes it hard to implement them on real systems, given the large number of bounds to which they are subject.

Other approaches are based on hand-tunable heuristics. For instance, a common heuristic in Task-Space Inverse Dynamics (TSID) [Del Prete 2015] which we adopt as well is to use a secondary task to keep the robot posture close to a reference one, in order to keep the movements far from the joint limits. Similarly, to avoid slipping/tipping, it was proposed to minimize the contact moments and the tangential contact forces in the null space of the main motion task [Righetti 2010]. Yet another common trick during locomotion is to maintain the center of pressure close to the center of the foot [Kajita 2003a]. The robotics literature is filled with these kinds of heuristics, which often are the main reason behind the successful implementations on real platforms. However, these heuristics can not ensure feasibility in the presence of any significant uncertainty and needs ad-hoc tuning depending on the situation.

Finally, another class of works which includes this work makes use of robust optimization techniques to formulate control and planning problems. Mordatch et al. [Mordatch 2015] considered several perturbed models of a humanoid robot to plan offline a trajectory that is robust to uncertainties, reporting success rate between 80% and 95% on a real platform. Another recent work [Luo 2015] has combined robust and time-scaling optimization to plan trajectories that are robust to bounded errors in friction coefficients and joint accelerations, whose magnitude can be estimated online through iterative learning. Nguyen and Sreenath [Nguyen 2015] have recently exploited control Lyapunov functions and Quadratic Programs (QPs) to ensure stability despite bounded uncertainties in the linearized system dynamics.

Contrary to [Mordatch 2015, Nguyen 2015], the uncertainties modeled in this work affect the parameters of the system, so they could be identified using set-membership identification techniques [Ramdani 2005]. The main contribution in this work is a novel formulation of the capture-point balance constraints, which can be included in the Task-Space Inverse Dynamics optimization problem to balance the robot despite bounded uncertainties in its inertial parameters. Contrary to previous approaches that dealt with uncertainties to inertial parameters, our approach allows us to include inequality constraints in the problem formulation. Thanks to this we can thus account for all the constraints to which legged robots are subject, ensuring the feasibility of the resulting trajectories.

3.3 Task-Space Inverse Dynamics with Capture-Point Balance Constraints

To design a controller that is robust to errors in the inertial parameters of the robot we have first to understand how these errors affect the control action. In this section we define the inertial parameters and we present a standard Task-Space Inverse Dynamics controller, which includes balance constraints. Throughout the presentation we explicitly show the dependency of the terms on the inertial parameters, while we omit the dependency on the robot configuration q and velocities v because they are constant values at each time step.

3.3.1 Inertial Parameters

We define the vector containing the 10 inertial parameters of link i as:

$$\phi_i = (m_i, m_i{}^i c_i, I_i^{xx}, I_i^{xy}, I_i^{xz}, I_i^{yy}, I_i^{yz}, I_i^{zz}),$$

where $m_i \in \mathbb{R}$ is the mass, $c_i \in \mathbb{R}^3$ is the CoM, $I_i \in \mathbb{R}^{3\times3}$ is the 3D rotational inertia matrix. Both c_i and I_i are expressed in the local reference frame of the link. Note that ϕ_i does not contain directly c_i, but it contains only its product with m_i. This is because the robot dynamics can be written in a linear form with respect to this parameterization of the inertial parameters [Traversaro 2015].

Now we can collect the inertial parameters of all the N links of the robot in a single vector:

$$\phi = (\phi_1, \ldots, \phi_N)$$

We assume that each link parameters ϕ_i are not known exactly, but we know that they lie inside a polytope U_i, i.e. $\phi_i \in U_i$. Hence also the vector ϕ lies inside a polytope:

$$\phi \in U = U_1 \times \cdots \times U_N$$

Note that since a polytope can be represented by a set of linear inequalities, the constraint $\phi_i \in U_i$ can be expressed under the form $A_i \phi_i \leq a_i$. Now that we defined the inertial parameters and the associated uncertainty polytopes, we can see how these uncertainties affect the controller.

3.3.2 Task-Space Inverse Dynamics

The controller that we consider in this work is an optimization-based inverse dynamics controller, which computes the desired torques taking into account the dynamcis of the robot. It has become a standard for the control of legged robots in recent years [Del Prete 2015, Herzog 2016, Sentis 2004, Saab 2011]. Table 3.1 shows TSID outperforming other control frameworks. Theortically, the kinematics and dynamics are decoupled. Kinematic level task prioritization is done first to compute acceleration and the torques are calculated to achieve the computed accelerations. Various formulations of the TSID optimization problem exist and are often equivalent or similar [Del Prete 2015]. We write it here as an optimization problem of the base and joint accelerations $\dot{v} \in \mathbb{R}^{n+6}$, the contact forces $f \in \mathbb{R}^k$, and the joint torques

Table 3.1: Comparison of Control Frameworks[Del Prete 2015]

Framework	Optimal	Efficient	Force	Under	Inequality	Output
			Control	actuated		
TSID[Del Prete 2015]	×	×	×	×		τ
UF[Peters 2007]		×	×			τ
WBCF[Sentis 2005]	×		×	(×)		τ
[Mistry 2011]			×	×		τ
SoT[Saab 2011]	×		×	×	×	τ
[De Lasa 2009]	×		×	×		τ
[Jeong 2009]	×	×				$\tau/\ddot{q}$
[Nakamura 1987]		×				$\dot{q}/\ddot{q}$
[Chiaverini 1997]		×				$\dot{q}$
[Siciliano 1991]	×	×				$\dot{q}$
[Baerlocher 1998]	×	×				$\dot{q}$
[Smits 2008]	×	×			×	$\dot{q}$

$\tau \in \mathbb{R}^n$ [Saab 2013]:

$$
\begin{aligned}
\underset{y=(\dot{v},f,\tau)}{\text{minimize}} \quad & ||Ay - a||^2 \\
\text{subject to} \quad & \begin{bmatrix} M(\phi) & -J_c^\top & -S^\top \\ J_c & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{v} \\ f \\ \tau \end{bmatrix} = \begin{bmatrix} -h(\phi) \\ -\dot{J}_c v \end{bmatrix} \\
& |\tau| \leq \tau^{max} && \text{(3.1a)} \\
& \dot{v}^{min} \leq \dot{v} \leq \dot{v}^{max} && \text{(3.1b)} \\
& f \in \mathcal{K} && \text{(3.1c)}
\end{aligned}
$$

where $J_c \in \mathbb{R}^{k\times(n+6)}$ is the constraint Jacobian, $M \in \mathbb{R}^{(n+6)\times(n+6)}$ is the mass matrix, $h \in \mathbb{R}^{n+6}$ contains the bias forces, $S \in \mathbb{R}^{n\times(n+6)}$ is the selection matrix, $\tau^{max} \in \mathbb{R}^n$ are the maximum joint torques, $\dot{v}^{min/max} \in \mathbb{R}^{n+6}$ are the acceleration bounds[1], and $\mathcal{K}$ is the force friction cone (which is typically linearized). The cost function represents the error of the task, which is typically an affine function of $\dot{v}$ (i.e. a task-space acceleration):

$$
\underbrace{\begin{bmatrix} J_{task} & 0 & 0 \end{bmatrix}}_{A} y - \underbrace{(\ddot{x}_{task}^{des} - \dot{J}_{task} v)}_{a} = \ddot{x}_{task} - \ddot{x}_{task}^{des}
$$

The task may be to track a predefined trajectory of a link, of the CoM of the robot, or to regulate the robot angular momentum.

3.3.3 Capture Point

Regardless of the task they are performing, legged robots must take care of balancing (i.e. avoiding to fall) at the same time. Balancing is fundamental for legged robots and it has been extensively studied [Collette 2008b, Morisawa 2012, Goswami 2004, Hyon 2006, Sherikov 2014]. This problem is particularly well understood for robots in contact with a flat terrain only. In this case, the dynamics of the robot CoM c is well approximated by a linear inverted pendulum. In this model the robot is approximated as a point mass (maintained at a constant height) supported by a variable-length leg link [Pratt 2006]. The resulting dynamics is:

$$
\ddot{c}^{xy}(\phi) = \omega(\phi)^2 (c^{xy}(\phi) - u),
$$

where $u \in \mathbb{R}^2$ is the ZMP, which is equivalent to the center of pressure [Wieber 2002], and $\omega(\phi) = \sqrt{\frac{g}{c^z(\phi)}}$. The same dynamics can also be obtained from the real dynamics of the robot CoM, by assuming that $\dot{c}^z = 0$ and the rate of change of the robot angular momentum is null [Wieber 2015]. Using this linear dynamics we can compute the point on the ground where the robot can put its

[1] The bounds of the joint positions and velocities are typically converted into joint-acceleration bounds [Rubrecht 2010]

ZMP to in order to stop its CoM:

$$\xi(\phi) = c^{xy}(\phi) + \frac{\dot{c}^{xy}(\phi)}{\omega(\phi)}$$

This point is known as the capture point [Pratt 2006], the divergent component of motion or the extrapolated CoM [Wieber 2015].

3.3.4 Capture-Point Balance Constraints

Originally, the capture point was used to decide where to make the robot step in order to recover from a push [Pratt 2006]. More recently, Ramos et al. [Ramos 2014] proposed use it to ensure the balance of the robot. The key idea is that, as long as the capture point remains inside the convex hull of the contact points (i.e. the so-called *support polygon* $\mathcal{S}$), the robot can balance without taking a step. To ensure the balance of the robot we can then add to (3.1) another set of inequalities to constrain the capture point to remain inside the support polygone:

$$B(\xi(\phi) + \Delta t \dot{\xi}(\phi)) \leq b,$$

where $\dot{\xi}(\phi) \in \mathbb{R}^2$ is the time derivative of the capture point, and the matrix B and the vector b define the support polygon (i.e. $Bx \leq b \iff x \in \mathcal{S}$). By expressing ξ and its derivative as functions of c^{xy} and its derivatives we get:

$$\begin{aligned} B\left(c^{xy}(\phi) + \frac{\dot{c}^{xy}(\phi)}{\omega(\phi)} + \Delta t \left(\dot{c}^{xy}(\phi) + \frac{\ddot{c}^{xy}(\phi)}{\omega(\phi)}\right)\right) \leq b \\ B\left(c^{xy}(\phi) + \alpha(\phi)\dot{c}^{xy}(\phi) + \frac{\Delta t}{\omega(\phi)}\ddot{c}^{xy}(\phi)\right) \leq b, \end{aligned}$$

where $\alpha(\phi) = \Delta t + \omega(\phi)^{-1}$. Finally we can express the CoM acceleration $\ddot{c}^{xy}$ as a function of the joint accelerations $\dot{v}$:

$$D(\phi)\dot{v} + B\left(c^{xy}(\phi) + \alpha(\phi)\dot{c}^{xy}(\phi) + \beta(\phi)\right) \leq b, \tag{3.2}$$

where:

$$\begin{aligned} D(\phi) &= \frac{\Delta t}{\omega(\phi)} B J_{com}(\phi) \\ \beta(\phi) &= \frac{\Delta t}{\omega(\phi)} \dot{J}_{com}(\phi) v \end{aligned}$$

These inequalities are linear with respect to the joint accelerations $\dot{v}$, so they can be added to the QP problem (3.1) to ensure the robot balance in case of no modeling uncertainties.

3.4 Robustness to Inertial Parameter Errors

In the previous section we saw that the inertial parameters appear in three different locations in the controller optimization problem (3.1): i) in the mass matrix M, ii) in the bias forces h, and iii) in the capture-point inequalities (3.2). Unfortunately M and h depend on ϕ in a highly-nonlinear way, so it is hard to deal with it. In this work we deal instead with the dependency of the capture-point inequalities (3.2) on the inertial parameters. More in details, many terms in (3.2) depend on ϕ, but we will focus on the dependency of the CoM xy coordinates on ϕ. In other words, we want to solve this optimization problem:

$$\begin{aligned}
\underset{y=(\dot{v},f,\tau)}{\text{minimize}} \quad & ||Ay-a||^2 \\
\text{subject to} \quad & \begin{bmatrix} M(\hat{\phi}) & -J_c^\top & -S^\top \\ J_c & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{v} \\ f \\ \tau \end{bmatrix} = \begin{bmatrix} -h(\hat{\phi}) \\ -\dot{J}_c v \end{bmatrix} && \text{(3.3a)} \\
& \text{(3.1a)}, \text{(3.1b)}, \text{(3.1c)} \\
& D(\hat{\phi})\dot{v} + Bc^{xy}(\phi) \leq \bar{b}(\hat{\phi}) \quad \forall \phi \in U, && \text{(3.3b)}
\end{aligned}$$

where $\hat{\phi}$ are the nominal inertial parameters (i.e. those used by the standard controller) and:

$$\bar{b}(\hat{\phi}) = b - B(\alpha(\hat{\phi})\dot{c}^{xy}(\hat{\phi}) + \beta(\hat{\phi}))$$

Problem (3.3) is not tractable because it has an infinite number of inequality constraints due to the capture-point inequalities that need to be satisfied for all the possible values of ϕ. In order to solve (3.3) we need to reformulate it in a tractable form. To do that, we will start by analyzing the relationship between c^{xy} and ϕ (which is linear). Then we will show how to reformulate the robust capture-point inequalities (3.3b) in a tractable form.

3.4.1 Dependency of CoM on Inertial Parameters

The CoM of the robot is the average of the CoM of all its links, weighted by their respective masses:

$$\begin{aligned}
c^{xy} &= P\frac{\sum_{i=1}^N m_i(p_i + {}^wR_i {}^ic_i)}{m_{tot}} \\
&= \sum_{i=1}^N \underbrace{m_{tot}^{-1} P \begin{bmatrix} p_i & {}^wR_i & 0_{3\times 6} \end{bmatrix}}_{F_i} \phi_i && \text{(3.4)} \\
&= \begin{bmatrix} F_1 & \dots & F_N \end{bmatrix} \phi = F\phi,
\end{aligned}$$

where $P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, $p_i \in \mathbb{R}^3$ is the position of the reference frame of link i expressed in the world frame, ${}^wR_i \in \mathbb{R}^{3\times 3}$ is a rotation matrix from link i reference frame to the world frame, and m_{tot} is the total mass of the robot. From (3.4) we can see that the robot CoM is the ratio of two linear functions of the inertial parameters because m_{tot} is clearly a linear function of ϕ. However, given that we can easily know the robot total mass, we can assume that the uncertainty in m_{tot} be negligible. In the context of robustness we can thus consider c^{xy} as a linear function of ϕ.

3.4.2 Robust Capture-Point Inequalities

Now we want to reformulate the robust capture-point inequalities into a tractable form. We can start by rewriting the i-th line of (3.3b) using (3.4):

$$D_i \dot{v} + B_i F \phi \leq \bar{b}_i \qquad \forall \phi \in U,$$

where D_i, B_i and $\bar{b}_i$ are the i-th lines of the associated matrix/vector, and we dropped the dependency on the nominal inertial parameters $\hat{\phi}$ for the sake of readability. We can get rid of the quantifier operator $\forall$ by replacing the uncertain term with its maximum:

$$D_i \dot{v} + \max_{\phi \in U} (B_i F \phi) \leq \bar{b}_i \tag{3.5}$$

We could compute the maximum of $B_i G \phi$ under the constraint of ϕ belonging to the polytope U by solving a Linear Program (LP) for each capture-point inequality. However, that would be too computationally expensive for a controller that typically has to run at 1 kHz because of the size of the LP (i.e. $10N$ variables and even more constraints). Luckily we show now that we can solve this LP by solving N LPs of much smaller size.

$$\max_{\phi \in U} B_i F \phi = \max_{\phi \in U} \sum_{j=1}^{N} B_i F_j \phi_j = \sum_{j=1}^{N} \max_{\phi_j \in U_j} B_i F_j \phi_j \tag{3.6}$$

Thanks to this reformulation, rather than maximizing a linear function of the robot CoM, we can maximize a linear function of each link CoM. This boils down to finding, for each link, the CoM position that maximizes the dot product with the vector B_i. If the polytope of possible CoM positions has not many vertices, this optimization can be performed by enumeration. This means that we can compute the dot product of B_i with all the vertices of the CoM polytope and then take the one that resulted in the largest value. Since the vertices of the CoM polytope of each link can be computed offline before starting the controller, this operation is extremely computationally efficient. If we assume that each CoM polytope has n_v vertices and that the support polygone has n_s sides, the computation of $\max_{\phi \in U} B_i F \phi$ for all i requires $n_s n_v N$ dot products of 3D vectors. For a typical scenario of $n_s = 6$, $n_v = 10$ and $N = 30$, this gives 1800 dot products. On a standard computer this would take only a few microseconds, so it is suitable for real-time control on a real robot.

Once this quantity has been computed, the robust capture-point inequalities (3.3b) can be written as standard linear inequalities and problem (3.3) can be

Table 3.2: Simulation parameters.

Symbol	Meaning	Value
Δt	Simulation time step	0.002 s
$\dot{v}_j^{max}$	Max joint acceleration	$10\,\mathrm{rad\,s^{-2}}$
v_j^{max}	Max joint velocity	$9.14\,\mathrm{rad\,s^{-1}}$
μ	Force friction coefficient	0.3
w_{reach}	Reaching weight	1
w_{post}	Posture weight	10^{-2}
w_{force}	Force minimization weight	10^{-5}

solved by a standard QP solver.

3.5 Tests

In this section, we present a series of simulation results that try to answer to the following question: what improvement can we get in terms of fall prevention by using the robust controller? We tested the proposed controller on a typical humanoid tasks (i.e. whole-body reaching) with the 30-degree-of-freedom humanoid robot HRP-2. We carried out several batches of tests, each batch differing for the simulated uncertainties. Each batch was composed by 100 tests, which is not enough for being a statistically significant sampling, but was dictated by the computation time of our simulation environment (about 6 hours for 100 tests). Each test consisted in trying to perform the reaching motion with the two controllers (classic and robust) until the robot either fell or reached the end of the motion. The inertial parameter errors changed at each test, but they were the same for the two controllers. We then measured the number of times each controller drove the robot to a fall and the average distance between the final end-effector position and the desired target.

3.5.1 Simulation Environment

To assess the proposed controller we developed a dedicated simulation environment based on a state-of-the-art algorithm for frictional contacts in multibody systems [Kaufman 2008]. We integrated the equations of motion of the system with a first-order Euler scheme with fixed time step Δt. Our choice of not using an off-the-shelf simulator is motivated by our desire to completely understand and control the simulation environment. The inertial parameters (masses and centers of mass) of the model used by the controller did not match those of the model used by the simulator. The random inertial-parameter errors were generated using uniform distribution. For masses, the maximum error was expressed in terms of percentage of the real values. For centers of mass, the maximum error was instead expressed in meters. In each test we specify which uncertainties were simulated. Table 3.2 lists all the simulation and controller parameters.

Table 3.3: Results of Test 1. For each controller we show the number of falls (Falls), the average time to complete the motion (Task Time) and the average distance of the end-effector to the target at the end of the motion (Task Error).

Uncertainties		Classic Controller			Robust Controller		
Max Mass Error [%]	Max CoM Error [mm]	Falls [%]	Task Time [s]	Task Error [mm]	Falls [%]	Task Time [s]	Task Error [mm]
10	10	31	4.4	49	3	4.5	60
10	20	33	4.3	52	1	4.5	69
10	40	45	4.3	55	3	4.7	102
20	10	38	4.2	49	11	4.5	77
20	20	49	4.5	51	9	4.55	103
20	40	45	4.5	59	14	4.72	122

3.5.2 Task Description

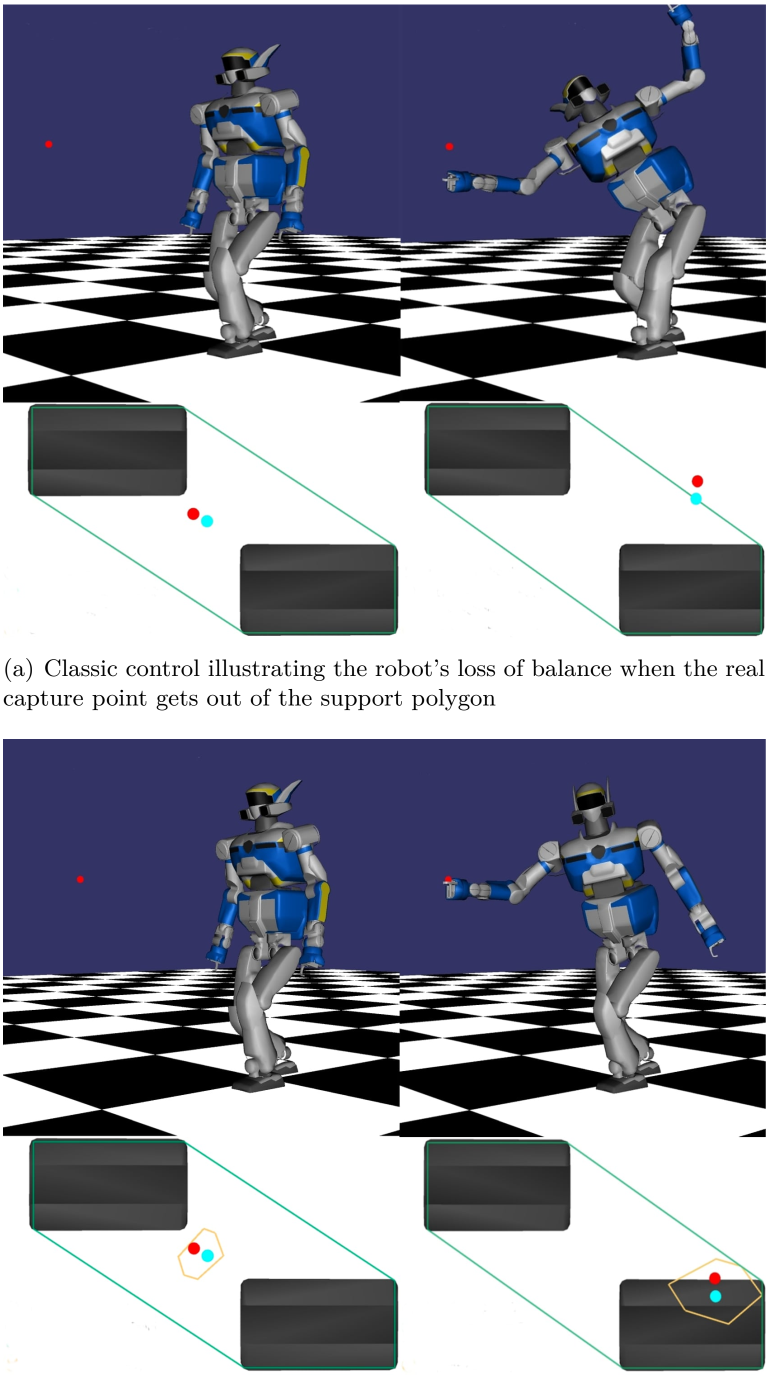
The control objective was defined by three tasks that the robot had to perform at the same time. Since the tasks are in conflict, we weighted them with hand-tuned parameters, according to their importance. The three tasks, in order of decreasing priority, are:

- reach the desired target with the right end-effector (weight w_{reach})
- maintain initial joint posture (weight w_{post})
- minimize contact moments and tangential forces [Righetti 2013b] (weight w_{force})

We carried out two sets of simulations. In both cases HRP-2 executed a reaching motion that made its capture point reach the boundaries of its support polygon.

3.5.3 Test 1

In this test we set the right end-effector target far in front of the robot. Fig. 3.1 shows some screen shots of the simulations. To reach the target the robot must move its CoM (and hence also its capture point) close to the boundaries of its support polygon. Table 3.3 presents the results. Regardless of the magnitude of the inertial parameter errors, the robust controller managed to prevent the robot from falling almost always, while with the standard controller the robot fell more than 30% of the times. However, since the target was far away from the robot, the robust controller did not manage to reach it because that would have required violating the robust balance constraints.



(a) Classic control illustrating the robot's loss of balance when the real capture point gets out of the support polygon

(b) Robust control illustrating the robot right end effector reaching close to the goal without losing balance

● Real Capture Point ● Estimated Capture Point ● Support Polygon ● Capture Point Polytope

Figure 3.1: Screenshots of HRP-2 executing Test 1 to reach the ball target with the robust controller.

Table 3.4: Results of Test 2. For each controller we show the number of falls (Falls), the average time to complete the motion (Task Time) and the average distance of the end-effector to the target at the end of the motion (Task Error).

Max Mass Error [%]	Max CoM Error [mm]	Falls [%]	Task Time [s]	Task Error [mm]	Falls [%]	Task Time [s]	Task Error [mm]
10	10	29	3.94	2	0	3.4	2
10	20	35	3.4	2	2	3.0	2
10	40	42	3.86	4	0	2.6	4
20	10	43	3.6	3	0	2.8	4
20	20	45	3.5	3	0	2.5	4
20	40	45	3.0	5	0	2.1	5

3.5.4 Test 2

In this test we moved the right end-effector target closer to the robot, so that HRP-2 can reach it without moving its CoM close to the support polygon boundaries. However, we increased the desired speed of reaching (by increasing the gains of the reaching task). This affected the velocity of the CoM, which in turns affected the capture point, making it reach the boundaries of the support polygon. The difference with respect to Test 1 is that in this case also the robust controller can reach the target. Table 3.4 summarizes the results.

Similarly to Test 1, the classic controller leads the robot to a fall in more than 30% of the cases. However, contrary to Test 1, this time the robust controller also manages to reach the target, because it is located closer to the robot. This test shows that being robust does not necessarily implies that we have to sacrifice performance. A video result of the same is available here.

3.6 Conclusions

This chapter presented a novel optimization-based inverse-dynamics controller that can balance a legged robot despite bounded uncertainties in its inertial parameters. The controller is based on the state-or-the-art control framework Task-Space Inverse Dynamics. In particular, this work is based on the capture-point inequalities [Ramos 2014], which can be included in the controller formulation to ensure the balance of the robot on a level ground. We extended these capture-point inequalities to be robust to bounded uncertainties in the inertial parameters of the robot. The resulting optimization problem is still a Quadratic Program with the same number of variables and inequalities. Moreover, the time required for the additional computation of the robust controller is negligible in this context (i.e. a few microseconds).

We tested the robust controller in simulations with the HRP-2 robot, trying to reach a target position with its right end-effector while balancing. We performed several batches of 100 simulations each, introducing different errors in the inertial parameters and varying the position of the target position and the required speed of motion. Comparisons against a classic TSID controller have shown impressive improvements in terms of fall prevention.

In the derivation of the robust controller we saw that the inertial parameters appear in different terms of the optimization problem. In this preliminary work we focused only on how the uncertainties affect the CoM position. We believe it should be possible to extend this analysis to the other terms in the capture-point inequalities: CoM velocity, CoM altitude, CoM Jacobian and its time derivative. Extending it also to the mass matrix and the bias forces is an interesting future direction, but it seems more challenging because of nonlinearities.

Another issue of the presented approach is that it is rather conservative and this can lead to poor performance, which can be unacceptable on a real system. Modeling uncertainties with probability distributions (rather than with polytopes) may lead to a less conservative behavior of the system, and it is thus an interesting future direction. In [Del Prete 2016a], the proposed controller was robust to joint-torque tracking errors. Integrating the two controllers together seems to be feasible and it would provide robustness to both kinds of uncertainties. In this preliminary work we focused on simulations to validate the controller formulation and to test it with different parameter errors. Of course, we plan also to test the generated movements on the real HRP-2 robot, to quantify how much it can benefit from this robustness.

Chapter 4
Conclusions

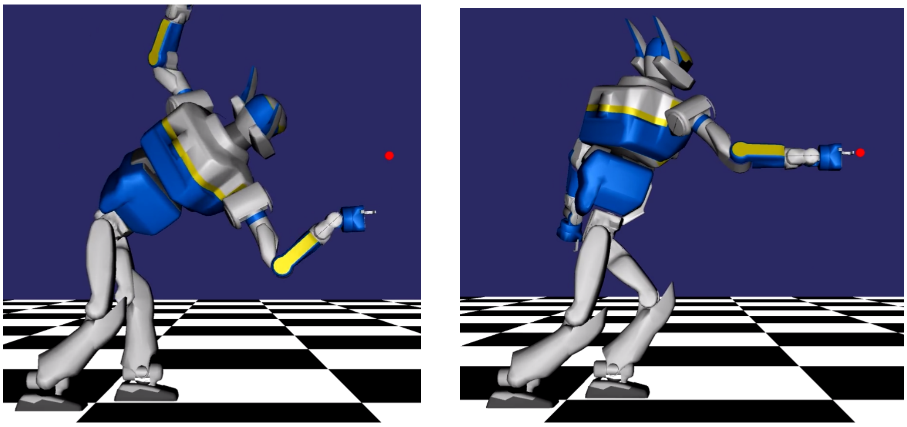
This thesis investigated the development of robust and reactive control methodologies for industrial and legged robots. Both the contributions in this thesis focuses on handling uncertainties and variability in robotic control in two different platforms exposed to different environment settings. This chapter concludes this thesis with a brief overview of the contributions and the expected future work.

4.1 Dynamic Obstacle Avoidance

The first contribution of this thesis is a framework that augments robot manipulators with dynamic obstacle avoidance functionalities. The goal is to make industrial robots more capable of collaborating with human beings. The framework uses proximity skin sensors to perceive its environment, which thus does not need to be modeled. The dense proximity information around the arm allows the system to react to obstacles approaching from any direction. The reactive controller relies on the state-of-the-art hierarchical QP solver, which makes it efficient enough to be used in real time. The tasks are programmed systematically to execute the planned trajectory, while avoiding potential collisions thanks to the higher-priority collision-avoidance task. The controller has been validated on a UR5 robot, shown in Fig. 2.6.

The integration and installation of advanced functionalities, such as the presented dynamic obstacle avoidance solution, poses three main challenges from the software point of view. The first is the integration of different components such as the skin driver, path planner and robot motion control. We addressed this challenge by adhering to the software development paradigm of the ROS-Industrial initiative. All the components presented in the work have been successfully integrated with ROS.

A second challenge is the quality assurance and robustness of the integrated robot software. This is crucial in production environments, and is specially important in collaborative applications, where safety needs to be guaranteed. For this purpose an Automated testing Framework (ATF) has been developed [Weisshardt 2016] as a part of the FiaD project, which allows for the systematic testing of robot software components, including unit testing, simulation-in-the-loop testing, and eventually hardware-in-the-loop testing. The tests can be automated and integrated in a centralized continuous integration system. Preliminary tests have already been conducted with the components of the robot software system of this work, and the integrated prototype applications will be tested with ATF.



(a) Classic Controller

(b) Robust Controller

Figure 4.1: Reaching Task under Inertial Uncertainties

Finally, the third challenge is the deployment of the software. One of the main barriers to transfer solutions based on robot frameworks such as ROS to industry, and specially SMEs, is how cumbersome they are to deploy. As a part of the FiaD project, a robot deployment toolbox has been developed [Lüdtke 2017], based on ROS, which can also be integrated with ATF. The deployment tools will also be evaluated on the RBE17 prototype.

The developed reactive control scheme has been demonstrated in the final project meeting (see this video) in the context of developing collaborative robot applications that can be deployed in industries quickly and with ease.

4.2 Robust Balance Control

The second contribution is a strategy to model inertial parameter uncertainties in a balance controller for legged robots. The specific controller that we considered in this work relies on the capture point to ensure the balance of the system. We investigated how errors in the masses and centers of mass of the robot links affect the estimation of the capture point, and thus the robot balance. This has allowed us to derive new *robust* capture-point inequality constraints, that we integrated in a standard Task-Space Inverse Dynamics control framework. Our simulations with the HRP-2 humanoid robot show that these new capture-point constraints make the controller much more robust to inertial errors of the robot model.

While the presented results are extremely encouraging, this work represents only a first step towards the achievement of a completely reliable balance control for legged robots. In the future, several issues of the presented controller need to be addressed in order to improve its performance, both from a theoretical and a practical stand point.

From a theoretical stand point, the proposed controller cannot guarantee the balance of the robot even if the inertial parameter uncertainties lie inside the given polytopes. This is because the inertial parameters appear in different terms of the optimization problem, but we have focused only on how they affect the CoM position. Our decision was based on our intuition, which suggested us that this was the most important term to take into account to achieve a robust behavior. Our simulations seem to confirm our intuition. However, this leaves us with a controller that cannot actually provide any guarantee on the balance of the system, and thus it is not reliable. We believe that it should be possible to account also for the effect of inertial parameter errors in several other terms of the capture-point inequalities: CoM velocity, CoM altitude, CoM Jacobian and its time derivative. However, this would still be insufficient to provide a theoretical guarantee of stability. This is because the inertial parameters also affect the mass matrix and the bias forces of the robot dynamics equation. While trying to account for all these uncertainties is an interesting future direction, it seems extremely challenging due to the highly nonlinear dependence of these quantities on the inertial parameters.

From a pragmatic stand point, during our tests we noticed that the presented approach can lead to extremely conservative behaviors. This problem is directly connected to our *worst-case* approach to model uncertainties: errors are supposed to lie inside polytopes, and we try to make the system robust to any possible realization of these errors. Another well-known way to model uncertainties is to rely on probability distributions rather than on uncertainty sets. Probability distributions provide a much richer descriptions of the errors, and thus allow the controller to make smarter choice when looking for a trade-off between safety and performance. These intuitions have also been confirmed by a recent work on robustness to joint-torque tracking errors [Del Prete 2016a], which compared the worst-case and the stochastic approaches to ensure robustness. All of this makes us believe that by using probability distributions to model inertial parameter errors we may obtain less conservative behaviors, while maintaining a high level of robustness.

Another interesting direction of future work would be to extend the presented controller to account for other kinds of uncertainties. For instance, it could be possible to integrate our controller with the one presented in [Del Prete 2016a], so as to ensure robustness to both inertial parameter errors and joint-torque tracking errors.

Bibliography

[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. (Cited in page 21.)

[Ali 2010] Muhammad A Ali, H Andy Park and CS George Lee. *Closed-form inverse kinematic joint solution for humanoid robots.* In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, pages 704–709. IEEE, 2010. (Cited in page 19.)

[Arbulu 2007] Mario Arbulu and Carlos Balaguer. *Real-time gait planning for Rh-1 humanoid robot, using local axis gait algorithm.* In Humanoid Robots, 2007 7th IEEE-RAS International Conference on, pages 563–568. IEEE, 2007. (Cited in page 11.)

[Arisumi 2008] Hitoshi Arisumi, Sylvain Miossec, Jean-Rémy Chardonnet and Kazuhito Yokoi. *Dynamic lifting by whole body motion of humanoid robots.* In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, pages 668–675. IEEE, 2008. (Cited in page 24.)

[Atkeson 1994] Christopher G Atkeson*et al. Using local trajectory optimizers to speed up global optimization in dynamic programming.* Advances in neural information processing systems, pages 663–663, 1994. (Cited in page 23.)

[Atkeson 2000] Christopher G Atkeson, Joshua G Hale, Frank Pollick, Marcia Riley, Shinya Kotosaka, S Schaul, Tomohiro Shibata, Gaurav Tevatia, Ales Ude, Sethu Vijayakumar*et al. Using humanoid robots to study human behavior.* IEEE Intelligent Systems and their applications, vol. 15, no. 4, pages 46–56, 2000. (Cited in page 11.)

[Atkeson 2003] Christopher G Atkeson and Jun Morimoto. *Nonparametric representation of policies and value functions: A trajectory-based approach.* Advances in neural information processing systems, pages 1643–1650, 2003. (Cited in page 23.)

[Atkeson 2016] Christopher G Atkeson, BPW Babu, N Banerjee, D Berenson, CP Bove, X Cui, M DeDonato, R Du, S Feng, P Franklin*et al. What happened at the DARPA robotics challenge, and why.* submitted to the DRC Finals Special Issue of the Journal of Field Robotics, 2016. (Cited in page 15.)

[Avanzini 2014] Giovanni Buizza Avanzini, Nicola Maria Ceriani, Andrea Maria Zanchettin, Paolo Rocco and Luca Bascetta. *Safety control of industrial* *robots based on a distributed distance sensor.* IEEE Transactions on Control Systems Technology, vol. 22, no. 6, pages 2127–2140, 2014. (Cited in pages 31 and 37.)

[Baerlocher 1998] Paolo Baerlocher and R Boulic. *Task-priority formulations for the kinematic control of highly redundant articulated structures.* Intelligent Robots and Systems, no. 2, 1998. (Cited in page 62.)

[Baginski 1998] Boris Baginski. *Motion planning for manipulators with many degrees of freedom-the BB-method.* 1998. (Cited in page 30.)

[Baillieul 1990] J Baillieul and DP Martin. *Resolution of kinematic redundancy.* In Proceedings of Symposia in Applied Mathematics, volume 41, pages 49–89, 1990. (Cited in page 19.)

[Balan 2006] Lucian Balan and Gary M Bone. *Real-time 3D collision avoidance method for safe human and robot coexistence.* In Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference On, pages 276–282. IEEE, 2006. (Cited in page 31.)

[Bascetta 2010] Luca Bascetta, Gianantonio Magnani, Paolo Rocco, Roberto Migliorini and Marco Pelagatti. *Anti-collision systems for robotic applications based on laser time-of-flight sensors.* In Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on, pages 278–284. IEEE, 2010. (Cited in page 32.)

[Bejczy 1974] Antal K Bejczy. *Robot arm dynamics and control.* 1974. (Cited in page 21.)

[Bélanger-Barrette 2015] Mathieu Bélanger-Barrette. *ROBOTIQ Collaborative Robot eBook.* October, vol. 15, 2015. (Cited in page 10.)

[Berryman 2003] Sylvia Berryman. *Ancient automata and mechanical explanation.* Phronesis, vol. 48, no. 4, pages 344–369, 2003. (Cited in page 2.)

[Bertsekas 1995] Dimitri P Bertsekas, Dimitri P Bertsekas, Dimitri P Bertsekas and Dimitri P Bertsekas. Dynamic programming and optimal control, volume 1. Athena scientific Belmont, MA, 1995. (Cited in page 23.)

[Brock 2002] Oliver Brock and Oussama Khatib. *Elastic strips: A framework for motion generation in human environments.* The International Journal of Robotics Research, vol. 21, no. 12, pages 1031–1052, 2002. (Cited in page 31.)

[Broquere 2008] Xavier Broquere, Daniel Sidobre and Ignacio Herrera-Aguilar. *Soft motion trajectory planner for service manipulator robot.* In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, pages 2808–2813. IEEE, 2008. (Cited in pages 34 and 42.)

[Broquere 2010] Xavier Broquere, Daniel Sidobre and Khoi Nguyen. *From motion planning to trajectory control with bounded jerk for service manipulator robots*. In Robotics and Automation (ICRA), 2010 IEEE International Conference on, pages 4505–4510. IEEE, 2010. (Cited in page 34.)

[Buchli 2009] Jonas Buchli, Mrinal Kalakrishnan, Michael Mistry, Peter Pastor and Stefan Schaal. *Compliant quadruped locomotion over rough terrain*. In Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on, pages 814–820. IEEE, 2009. (Cited in page 21.)

[Calinon 2010] Sylvain Calinon, Irene Sardellitti and Darwin G Caldwell. *Learning-based control strategy for safe human-robot interaction exploiting task and robot redundancies*. In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, pages 249–254. IEEE, 2010. (Cited in page 29.)

[Castain 1984] Ralph H Castain and Richard P Paul. *An on-line dynamic trajectory generator*. The International Journal of Robotics Research, vol. 3, no. 1, pages 68–72, 1984. (Cited in page 34.)

[Ceriani 2013] Nicola Maria Ceriani, Giovanni Buizza Avanzini, Andrea Maria Zanchettin, Luca Bascetta and Paolo Rocco. *Optimal placement of spots in distributed proximity sensors for safe human-robot interaction*. In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 5858–5863. IEEE, 2013. (Cited in page 37.)

[Chan 1995] Tan Fung Chan and Rajiv V Dubey. *A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators*. Robotics and Automation, IEEE Transactions on, vol. 11, no. 2, pages 286–292, 1995. (Cited in page 43.)

[Cheng 2007] Gordon Cheng, Sang-Ho Hyon, Jun Morimoto, Aleš Ude, Joshua G Hale, Glenn Colvin, Wayco Scroggin and Stephen C Jacobsen. *CB: A humanoid research platform for exploring neuroscience*. Advanced Robotics, vol. 21, no. 10, pages 1097–1114, 2007. (Cited in page 11.)

[Chiaverini 1994] Stefano Chiaverini, Bruno Siciliano and Olav Egeland. *Review of the Damped Least-Squares Inverse Kinematics with Experiments on an Industrial Robot Manipulator*. Control Systems Technology, IEEE Transactions on, vol. 2, no. 2, pages 123–134, 1994. (Cited in page 19.)

[Chiaverini 1997] S. Chiaverini. *Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators*. IEEE Transactions on Robotics and Automation, vol. 13, no. 3, pages 398–410, jun 1997. (Cited in pages 19 and 62.)

[Collette 2008a] Cyrille Collette, Alain Micaelli, Claude Andriot and Pierre Lemerle. *Robust balance optimization control of humanoid robots with multiple non coplanar grasps and frictional contacts.* In Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, pages 3187–3193. IEEE, 2008. (Cited in page 22.)

[Collette 2008b] Cyrille Collette, Alain Micaelli, Claude Andriot and Pierre Lemerle. *Robust balance optimization control of humanoid robots with multiple non coplanar grasps and frictional contacts.* Proceedings - IEEE International Conference on Robotics and Automation, pages 3187–3193, 2008. (Cited in page 63.)

[De Lasa 2009] Martin De Lasa and Aaron Hertzmann. *Prioritized optimization for task-space control.* In Intelligent Robots and Systems, IEEE/RSJ International Conference on, volume 3, pages 5755–5762. Ieee, oct 2009. (Cited in page 62.)

[de Lasa 2010] Martin de Lasa, Igor Mordatch and Aaron Hertzmann. *Feature-based locomotion controllers.* ACM Transactions on Graphics (TOG), vol. 29, no. 4, page 131, 2010. (Cited in page 43.)

[De Luca 2004] Alessandro De Luca and Raffaella Mattone. *An adapt-and-detect actuator FDI scheme for robot manipulators.* In Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on, volume 5, pages 4975–4980. IEEE, 2004. (Cited in page 36.)

[De Luca 2006] Alessandro De Luca, Alin Albu-Schaffer, Sami Haddadin and Gerd Hirzinger. *Collision detection and safe reaction with the DLR-III lightweight manipulator arm.* In Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, pages 1623–1630. IEEE, 2006. (Cited in pages 35 and 36.)

[De Luca 2009] Alessandro De Luca, Fabrizio Flacco, Antonio Bicchi and Riccardo Schiavi. *Nonlinear decoupled motion-stiffness control and collision detection/reaction for the VSA-II variable stiffness device.* In Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on, pages 5487–5494. IEEE, 2009. (Cited in page 35.)

[De Luca 2012] Alessandro De Luca and Fabrizio Flacco. *Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration.* In Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on, pages 288–295. IEEE, 2012. (Cited in pages 30 and 35.)

[Dean-Leon 2016a] E. Dean-Leon, F. Bergner, K. Ramirez-Amaro and G. Cheng. *From multi-modal tactile signals to a compliant control.* In 2016 IEEE-RAS

16th International Conference on Humanoid Robots (Humanoids), pages 892–898, Nov 2016. (Cited in page 40.)

[Dean-Leon 2016b] E. Dean-Leon, K. Ramirez-Amaro, F. Bergner, I. Dianov, P. Lanillos and G. Cheng. *Robotic technologies for fast deployment of industrial robot systems.* In IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, pages 6900–6907, Oct 2016. (Cited in page 41.)

[Dean-Leon 2017] E. Dean-Leon, B. Pierce, P. Mittendorfer, F.Bergner, K. Ramirez-Amaro, W. Burger and Gordon Cheng. *TOMM: Tactile Omnidirectional Mobile Manipulator.* In 2017 IEEE International Conference on Robotics and Automation (ICRA), Accepted, May 2017. (Cited in page 40.)

[Decré 2009] Wilm Decré, Ruben Smits, Herman Bruyninckx and Joris De Schutter. *Extending iTaSC to support inequality constraints and non-instantaneous task specification.* pages 964–971, 2009. (Cited in pages 34 and 46.)

[Del Prete 2014] Andrea Del Prete, Francesco Romano, Lorenzo Natale, Giorgio Metta, Giulio Sandini and Francesco Nori. *Prioritized optimal control.* In Robotics and Automation (ICRA), 2014 IEEE International Conference on, pages 2540–2545. IEEE, 2014. (Cited in page 24.)

[Del Prete 2015] Andrea Del Prete, Francesco Nori, Giorgio Metta and Lorenzo Natale. *Prioritized Motion-Force Control of Constrained Fully-Actuated Robots: "Task Space Inverse Dynamics".* Robotics and Autonomous Systems, vol. 63, pages 150–157, 2015. (Cited in pages 60 and 62.)

[Del Prete 2016a] Andrea Del Prete and Nicolas Mansard. *Robustness to Joint-Torque Tracking Errors in Task-Space Inverse Dynamics.* IEEE Transaction on Robotics (accepted for publication), 2016. (Cited in pages 59, 71, and 75.)

[Del Prete 2016b] Andrea Del Prete, Nicolas Mansard, Oscar E Ramos, Olivier Stasse and Francesco Nori. *Implementing torque control with high-ratio gear boxes and without joint-torque sensors.* International Journal of Humanoid Robotics, vol. 13, no. 01, page 1550044, 2016. (Cited in pages 13 and 21.)

[Del Prete 2018] Andrea Del Prete. *Joint Position and Velocity Bounds in Discrete-Time Acceleration / Torque Control of Robot Manipulators.* Robotics and Automation Letters, vol. 3, no. 1, 2018. (Cited in pages 34 and 46.)

[Dellin 2012] Christopher M Dellin and Siddhartha S Srinivasa. *A framework for extreme locomotion planning.* In Robotics and Automation (ICRA), 2012 IEEE International Conference on, pages 989–996. IEEE, 2012. (Cited in page 24.)

[Desai 1999] Jaydev P Desai and Vijay Kumar. *Motion planning for cooperating mobile manipulators.* Journal of robotic systems, vol. 16, no. 10, pages 557–579, 1999. (Cited in page 25.)

[Dietrich 2013] Alexander Dietrich, Christian Ott and Alin Albu-Schäffer. *Multi-objective compliance control of redundant manipulators: Hierarchy, control, and stability.* In 2013 IEEE/RSJ international conference on intelligent robots and systems, pages 3043–3050. IEEE, 2013. (Cited in page 60.)

[Dimitrov 2011] Dimitar Dimitrov, Alexander Sherikov and Pierre-Brice Wieber. *A sparse model predictive control formulation for walking motion generation.* In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pages 2292–2299. IEEE, 2011. (Cited in page 24.)

[Dumonteil 2015] Gautier Dumonteil, Guido Manfredi, Michel Devy, Ambroise Confetti and Daniel Sidobre. *Reactive planning on a collaborative robot for industrial applications.* In Informatics in Control, Automation and Robotics (ICINCO), 2015 12th International Conference on, volume 2, pages 450–457. IEEE, 2015. (Cited in page 57.)

[El Khoury 2013] Antonio El Khoury, Florent Lamiraux and Michel Taïx. *Optimal motion planning for humanoid robots.* In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 3136–3141. IEEE, 2013. (Cited in page 24.)

[Englsberger 2014] Johannes Englsberger, Alexander Werner, Christian Ott, Bernd Henze, Maximo A Roa, Gianluca Garofalo, Robert Burger, Alexander Beyer, Oliver Eiberger, Korbinian Schmid*et al.* *Overview of the torque-controlled humanoid robot TORO.* In Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on, pages 916–923. IEEE, 2014. (Cited in page 14.)

[Erdem 2011] Esra Erdem, Kadir Haspalamutgil, Can Palaz, Volkan Patoglu and Tansel Uras. *Combining high-level causal reasoning with low-level geometric reasoning and motion planning for robotic manipulation.* In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pages 4575–4581. IEEE, 2011. (Cited in page 35.)

[Escande 2014a] Adrien Escande, Nicolas Mansard and Pierre-Brice Wieber. *Hierarchical Quadratic Programming: Fast Online Humanoid-Robot Motion Generation.* International Journal of Robotics Research, vol. 33, no. 7, pages 1006–1028, 2014. (Cited in page 20.)

[Escande 2014b] Adrien Escande, Nicolas Mansard and Pierre-Brice Wieber. *Hierarchical quadratic programming: Fast online humanoid-robot motion generation.* The International Journal of Robotics Research, page 0278364914521306, 2014. (Cited in page 43.)

[Faverjon 1987] B. Faverjon and P. Tournassoud. *A local based approach for path planning of manipulators with a high number of degrees of freedom.* In Robotics and Automation. Proceedings. 1987 IEEE International Conference on, volume 4, pages 1152–1159, Mar 1987. (Cited in pages vii, 32, 33, and 37.)

[Featherstone 2010] Roy Featherstone. *Exploiting sparsity in operational-space dynamics.* The International Journal of Robotics Research, pages 1–21, 2010. (Cited in page 21.)

[Featherstone 2014] Roy Featherstone. Rigid body dynamics algorithms. Springer, 2014. (Cited in page 18.)

[Fink 2012] Julia Fink. *Anthropomorphism and human likeness in the design of robots and human-robot interaction.* In International Conference on Social Robotics, pages 199–208. Springer, 2012. (Cited in page 11.)

[Flacco 2012] Fabrizio Flacco and Alessandro De Luca. *A Depth Space Approach to Human-Robot Collision Avoidance.* 2012. (Cited in pages 30, 31, 32, and 57.)

[Fryman 2012] Jeff Fryman and Bjoern Matthias. *Safety of industrial robots: From conventional to collaborative applications.* In Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on, pages 1–5. VDE, 2012. (Cited in page 8.)

[Geoffroy 2014] Perle Geoffroy, Nicolas Mansard, Maxime Raison, Sofiane Achiche and Emo Todorov. *From inverse kinematics to optimal control.* In Advances in Robot Kinematics, pages 409–418. Springer, 2014. (Cited in page 24.)

[Geppert 2004] Linda Geppert. *Qrio, the robot that could.* Ieee Spectrum, vol. 41, no. 5, pages 34–37, 2004. (Cited in page 11.)

[Geravand 2013] Milad Geravand, Fabrizio Flacco and Alessandro De Luca. *Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture.* In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 4000–4007. IEEE, 2013. (Cited in page 35.)

[Gienger 1999] M Gienger, K Löffler and F Pfeiffer. *Design and control of a biped walking and jogging robot.* In Proceedings of the 2nd International Conference on Climbing and Walking Robots (CLAWAR), pages 49–58, 1999. (Cited in page 11.)

[Gienger 2005] Michael Gienger, Herbert Janssen and Christian Goerick. *Task-oriented whole body motion for humanoid robots.* In Humanoid Robots, 2005 5th IEEE-RAS International Conference on, pages 238–244. IEEE, 2005. (Cited in page 20.)

[Golz 2015] Saskia Golz, Christian Osendorfer and Sami Haddadin. *Using tactile sensation for learning contact knowledge: Discriminate collision from physical interaction.* In Robotics and Automation (ICRA), 2015 IEEE International Conference on, pages 3788–3794. IEEE, 2015. (Cited in page 35.)

[Goswami 2004] a. Goswami and V. Kallem. *Rate of change of angular momentum and balance maintenance of biped robots.* IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004, vol. 4, 2004. (Cited in page 63.)

[Gouaillier 2009] David Gouaillier, Vincent Hugel, Pierre Blazevic, Chris Kilner, Jérôme Monceaux, Pascal Lafourcade, Brice Marnier, Julien Serre and Bruno Maisonnier. *Mechatronic design of NAO humanoid.* In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, pages 769–774. IEEE, 2009. (Cited in page 12.)

[Grey 2013] Michael X Grey, Neil Dantam, Daniel M Lofaro, Aaron Bobick, Magnus Egerstedt, Paul Oh and Mike Stilman. *Multi-process control software for hubo2 plus robot.* In Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on, pages 1–6. IEEE, 2013. (Cited in pages 11 and 13.)

[Guizzo 2015] Erico Guizzo and Evan Ackerman. *The hard lessons of DARPA's robotics challenge [News].* IEEE Spectrum, vol. 52, no. 8, pages 11–13, 2015. (Cited in page 14.)

[Haddadin 2008] Sami Haddadin, Alin Albu-Schaffer, Alessandro De Luca and Gerd Hirzinger. *Collision detection and reaction: A contribution to safe physical human-robot interaction.* In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, pages 3356–3363. IEEE, 2008. (Cited in page 36.)

[Haddadin 2010] Sami Haddadin, Holger Urbanek, Sven Parusel, Darius Burschka, Jürgen Roßmann, Alin Albu-Schäffer and Gerd Hirzinger. *Real-time reactive motion generation based on variable attractor dynamics and shaped velocities.* In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, pages 3109–3116. IEEE, 2010. (Cited in page 29.)

[Haddadin 2011a] Sami Haddadin, Alin Albu-Schaffer, Fahed Haddadin, Jurgen Rosmann and Gerd Hirzinger. *Study on soft-tissue injury in robotics.* IEEE Robotics & Automation Magazine, vol. 18, no. 4, pages 20–34, 2011. (Cited in page 36.)

[Haddadin 2011b] Sami Haddadin, Rico Belder and Alin Albu-Schäffer. *Dynamic motion planning for robots in partially unknown environments.* IFAC Proceedings Volumes, vol. 44, no. 1, pages 6842–6850, 2011. (Cited in pages 30 and 57.)

[Haddadin 2012] Sami Haddadin, Simon Haddadin, Augusto Khoury, Tim Rokahr, Sven Parusel, Rainer Burgkart, Antonio Bicchi and Alin Albu-Schäffer. *On making robots understand safety: Embedding injury knowledge into control.* The International Journal of Robotics Research, vol. 31, no. 13, pages 1578–1602, 2012. (Cited in page 35.)

[Hanafusa 1981] Hideo Hanafusa, Tsuneo Yoshikawa and Yoshihiko Nakamura. *Analysis and control of articulated robot with redundancy.* In IFAC, 8th Triennal World Congress, volume 4, pages 1927–1932, 1981. (Cited in page 43.)

[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. (Cited in page 34.)

[Herdt 2010] Andrei Herdt, Holger Diedam, Pierre-Brice Wieber, Dimitar Dimitrov, Katja Mombaur and Moritz Diehl. *Online walking motion generation with automatic footstep placement.* Advanced Robotics, vol. 24, no. 5-6, pages 719–737, 2010. (Cited in page 24.)

[Hereid 2016] Ayonga Hereid, Eric A Cousineau, Christian M Hubicki and Aaron D Ames. *3D dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics.* In Robotics and Automation (ICRA), 2016 IEEE International Conference on, pages 1447–1454. IEEE, 2016. (Cited in page 14.)

[Herzog 2013] Alexander Herzog, Ludovic Righetti, Felix Grimminger, Peter Pastor and Stefan Schaal. *Experiments with a hierarchical inverse dynamics controller on a torque-controlled humanoid.* arXiv preprint, 2013. (Cited in page 22.)

[Herzog 2016] Alexander Herzog, Nicholas Rotella, Sean Mason, Felix Grimminger, Stefan Schaal and Ludovic Righetti. *Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid.* Autonomous Robots, vol. 40, no. 3, pages 473–491, 2016. (Cited in page 62.)

[Hirai 1998] Kazuo Hirai, Masato Hirose, Yuji Haikawa and Toru Takenaka. *The development of Honda humanoid robot.* In Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on, volume 2, pages 1321–1326. IEEE, 1998. (Cited in pages 11 and 13.)

[Hirose 2007] Masato Hirose and Kenichi Ogawa. *Honda humanoid robots development.* Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol. 365, no. 1850, pages 11–19, 2007. (Cited in page 11.)

[Hirzinger 2002] Gerd Hirzinger, Norbert Sporer, A Albu-Schaffer, M Hahnle, Rainer Krenn, A Pascucci and Markus Schedl. *DLR's torque-controlled light* *weight robot III-are we reaching the technological limits now?* In Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on, volume 2, pages 1710–1716. IEEE, 2002. (Cited in page 14.)

[Hockstein 2007] Neil G Hockstein, CG Gourin, RA Faust and David J Terris. *A history of robots: from science fiction to surgical robotics.* Journal of robotic surgery, vol. 1, no. 2, pages 113–118, 2007. (Cited in page 3.)

[Hofmann 2009] Andreas Hofmann, Marko Popovic and Hugh Herr. *Exploiting angular momentum to enhance bipedal center-of-mass control.* In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, pages 4423–4429. IEEE, 2009. (Cited in page 22.)

[Hollerbach 1980] John M Hollerbach. *A recursive lagrangian formulation of maniputator dynamics and a comparative study of dynamics formulation complexity.* IEEE Transactions on Systems, Man, and Cybernetics, vol. 10, no. 11, pages 730–736, 1980. (Cited in page 21.)

[Hutter 2012] Marco Hutter, Michael Gehring, Michael Bloesch, C David Remy, Roland Yves Siegwart and Mark A Hoepflinger. *StarlETH: A compliant quadrupedal robot for fast, efficient, and versatile locomotion.* 2012. (Cited in page 14.)

[Hyon 2006] SH Hyon and Gordon Cheng. *Gravity compensation and full-body balancing for humanoid robots.* In Humanoid Robots, 2006 6th IEEE-RAS International Conference on, volume 00, pages 214–221, 2006. (Cited in page 63.)

[Inaba 2003] Masayuki Inaba, Ikuo Mizuuchi, Ryosuke Tajima, Tomoaki Yoshikai, Daisuke Sato, Koichi Nagashima and Hirochika Inoue. *Building spined muscle-tendon humanoid.* Robotics Research, pages 113–127, 2003. (Cited in page 11.)

[Jacobson 1968] David H Jacobson. *New second-order and first-order algorithms for determining optimal control: A differential dynamic programming approach.* Journal of Optimization Theory and Applications, vol. 2, no. 6, pages 411–440, 1968. (Cited in page 23.)

[Jarquín 2013] Gerardo Jarquín, Adrien Escande, Gustavo Arechavaleta, Thomas Moulard, Eiichi Yoshida and Vicente Parra-Vega. *Real-time smooth task transitions for hierarchical inverse kinematics.* In Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on, pages 528–533. IEEE, 2013. (Cited in page 20.)

[Jeong 2009] JW Jeong. *A Task-priority Based Framework for Multiple Tasks in Highly Redundant Robots.* Intelligent Robots and Systems, IEEE/RSJ International Conference on., pages 5886–5891, 2009. (Cited in page 62.)

[Jin 2008] M. Jin, S. H. Kang and P. H. Chang. *Robust compliant motion control of robot with nonlinear friction using time-delay estimation.* Industrial Electronics, IEEE Transactions on, vol. 55, no. 1, 2008. (Cited in page 60.)

[Kahn 1969] Michael Edwin Kahn and B Roth. The near minimum-time control of open-loop articulated kinematic chains. Stanford University, Computer Science Department, 1969. (Cited in page 21.)

[Kajita 2003a] Shuuji Kajita and F. Kanehiro. *Biped walking pattern generation by using preview control of zero-moment point.* In Robotics and Automation (ICRA), 2003 IEEE International Conference on, 2003. (Cited in page 60.)

[Kajita 2003b] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Kazuhito Yokoi and Hirohisa Hirukawa. *Biped walking pattern generation by using preview control of zero-moment point.* In Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on, volume 2, pages 1620–1626. IEEE, 2003. (Cited in page 24.)

[Kajita 2003c] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Kazuhito Yokoi and Hirohisa Hirukawa. *Resolved momentum control: Humanoid motion planning based on the linear and angular momentum.* In Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on, volume 2, pages 1644–1650. IEEE, 2003. (Cited in page 22.)

[Kalakrishnan 2011] Mrinal Kalakrishnan, Sachin Chitta, Evangelos Theodorou, Peter Pastor and Stefan Schaal. *STOMP: Stochastic trajectory optimization for motion planning.* In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pages 4569–4574. IEEE, 2011. (Cited in page 25.)

[Kalman 1960] R E Kalman. *A New Approach to Linear Filtering and Prediction Problems 1.* vol. 82, no. Series D, pages 35–45, 1960. (Cited in page 23.)

[Kanehiro 2008] Fumio Kanehiro, Florent Lamiraux, Oussama Kanoun, Eiichi Yoshida and Jean-Paul Laumond. *A local collision avoidance method for non-strictly convex polyhedra.* Proceedings of robotics: science and systems IV, 2008. (Cited in page 33.)

[Kaneko 2002] Kenji Kaneko, Fumio Kanehiro, Shuuji Kajita, Kazuhiko Yokoyama, Kazuhiko Akachi, Toshikazu Kawasaki, Shigehiko Ota and Takakatsu Isozumi. *Design of prototype humanoid robotics platform for HRP.* In Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on, volume 3, pages 2431–2436. IEEE, 2002. (Cited in page 11.)

[Kaneko 2004] Kenji Kaneko, Fumio Kanehiro, Shuuji Kajita, Hirohisa Hirukawa, Toshikazu Kawasaki, Masaru Hirata, Kazuhiko Akachi and Takakatsu Isozumi. *Humanoid robot HRP-2.* In In IEEE Int. Conf. Rob. Aut, pages 1083–1090, 2004. (Cited in pages 11 and 13.)

[Kaneko 2008] Kenji Kaneko, Kensuke Harada, Fumio Kanehiro, Go Miyamori and Kazuhiko Akachi. *Humanoid robot HRP-3.* In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, pages 2471–2478. IEEE, 2008. (Cited in page 11.)

[Kaneko 2009] Kenji Kaneko, Fumio Kanehiro, Mitsuharu Morisawa, Kanako Miura, Shin'ichiro Nakaoka and Shuuji Kajita. *Cybernetic human HRP-4C.* In Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on, pages 7–14. IEEE, 2009. (Cited in page 11.)

[Kaneko 2011] Kenji Kaneko, Fumio Kanehiro, Mitsuharu Morisawa, Kazuhiko Akachi, Go Miyamori, Atsushi Hayashi and Noriyuki Kanehira. *Humanoid robot hrp-4-humanoid robotics platform with lightweight and slim body.* In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pages 4400–4407. IEEE, 2011. (Cited in page 11.)

[Kanoun 2009] Oussama Kanoun, Florent Lamiraux, Pierre-Brice Wieber, Fumio Kanehiro, Eiichi Yoshida and Jean-Paul Laumond. *Prioritizing linear equality and inequality systems: application to local motion planning for redundant robots.* In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, pages 2939–2944. IEEE, 2009. (Cited in page 20.)

[Kanoun 2011a] Oussama Kanoun. *Kinematic Control of Redundant Manipulators: Generalizing the Task-Priority Framework to Inequality Task.* ... , IEEE Transactions on, pages 1–9, 2011. (Cited in page 43.)

[Kanoun 2011b] Oussama Kanoun, Florent Lamiraux and P-B Wieber. *Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task.* Robotics, IEEE Transactions on, vol. 27, no. 4, pages 785–792, 2011. (Cited in page 43.)

[Karinne Ramirez-Amaro 2016] Karinne Ramirez-Amaro, Emmanuel Dean-Leon, Ilya Dianov, Florian Bergner and Gordon Cheng. *General Recognition Models Capable of Integrating Multiple Sensors for Different Domains.* In IEEE-RAS International Conference on Humanoid Robots 2016, Cancun, Mexico, Nov 2016. IEEE. (Cited in page 40.)

[Kato 1973] Ichiro Kato. *Development of WABOT 1.* Biomechanism, vol. 2, pages 173–214, 1973. (Cited in page 11.)

[Kaufman 2008] Danny M. Kaufman, Shinjiro Sueda, Doug L. James and Dinesh K. Pai. *Staggered projections for frictional contact in multibody systems.* ACM Transactions on Graphics, vol. 27, no. 5, page 1, 2008. (Cited in page 67.)

[Kelly 1989] R. Kelly and R. Carelli. *On adaptive impedance control of robot manipulators.* Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on, 1989. (Cited in page 60.)

[Khatib 1986] Oussama Khatib. *Real-time obstacle avoidance for manipulators and mobile robots.* The international journal of robotics research, vol. 5, no. 1, pages 90–98, 1986. (Cited in page 29.)

[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, feb 1987. (Cited in pages 18 and 21.)

[Khatib 2004] Oussama Khatib, Luis Sentis, Jaeheung Park and James Warren. *Whole-body dynamic behavior and control of human-like robots.* International Journal of Humanoid Robotics, vol. 1, no. 01, pages 29–43, 2004. (Cited in page 22.)

[Khatib 2008] Oussama Khatib, Peter Thaulad, Taizo Yoshikawa and Jaeheung Park. *Torque-position transformer for task control of position controlled robots.* In Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, pages 1729–1734. IEEE, 2008. (Cited in page 13.)

[Killpack 2016] Marc D Killpack, Ariel Kapusta and Charles C Kemp. *Model predictive control for fast reaching in clutter.* Autonomous Robots, vol. 40, no. 3, pages 537–560, 2016. (Cited in page 37.)

[Kim 2002] Jung-Hoon Kim, Seo-Wook Park, Ill-Woo Park and Jun-Ho Oh. *Development of a humanoid biped walking robot platform KHR-1-initial design and its performance evaluation.* feedback, vol. 1, page 12, 2002. (Cited in page 11.)

[Kim 2011] KangGeon Kim, Young-Su Cha, Jung-Min Park, Ji-Yong Lee and Bum-Jae You. *Providing services using network-based humanoids in a home environment.* IEEE Transactions on Consumer Electronics, vol. 57, no. 4, pages 1628–1636, 2011. (Cited in page 11.)

[Knabe 2013] Coleman Knabe, Mike Hopkins and Dennis W Hong. *Team charli: Robocup 2012 humanoid adultsize league winner.* In RoboCup 2012: Robot Soccer World Cup XVI, pages 59–64. Springer, 2013. (Cited in page 12.)

[Koch 2012] Kai Henning Koch, Katja Daniela Mombaur and Philipp Souères. *Studying the effect of different optimization criteria on humanoid walking motions.* In International Conference on Simulation, Modeling, and Programming for Autonomous Robots, pages 221–236. Springer, 2012. (Cited in page 24.)

[Kormushev 2011] Petar Kormushev, Barkan Ugurlu, Sylvain Calinon, Nikolaos G Tsagarakis and Darwin G Caldwell. *Bipedal walking energy minimization by reinforcement learning with evolving policy parameterization.* In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pages 318–324. IEEE, 2011. (Cited in page 14.)

[Kröger 2011] Torsten Kröger. *Opening the door to new sensor-based robot applications—The Reflexxes Motion Libraries.* In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pages 1–4. IEEE, 2011. (Cited in page 42.)

[Kulić 2007] Dana Kulić and Elizabeth Croft. *Pre-collision safety strategies for human-robot interaction.* Autonomous Robots, vol. 22, no. 2, pages 149–164, 2007. (Cited in page 29.)

[Kwon 2007] Woong Kwon, Hyun K Kim, Joong Kyung Park, Chang Hyun Roh, Jawoo Lee, Jaeho Park, Won-Kuk Kim and Kyungshik Roh. *Biped humanoid robot Mahru III.* In Humanoid Robots, 2007 7th IEEE-RAS International Conference on, pages 583–588. IEEE, 2007. (Cited in page 11.)

[Lacevic 2010] Bakir Lacevic and Paolo Rocco. *Kinetostatic danger field-a novel safety assessment for human-robot interaction.* In Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, pages 2169–2174. IEEE, 2010. (Cited in pages 30 and 32.)

[Laffranchi 2009] Matteo Laffranchi, Nikolaos G Tsagarakis and Darwin G Caldwell. *Safe human robot interaction via energy regulation control.* In Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on, pages 35–41. IEEE, 2009. (Cited in page 34.)

[Lasota 2014] Przemyslaw A Lasota, Gregory F Rossano and Julie A Shah. *Toward safe close-proximity human-robot interaction with standard industrial robots.* In Automation Science and Engineering (CASE), 2014 IEEE International Conference on, pages 339–344. IEEE, 2014. (Cited in page 31.)

[Lefebvre 2005] O. Lefebvre, F. Lamiraux and D. Bonnafous. *Fast computation of robot-obstacle interactions in nonholonomic trajectory deformation.* In Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on, pages 4612–4617. IEEE, 2005. (Cited in page 46.)

[Lengagne 2013] Sébastien Lengagne, Joris Vaillant, Eiichi Yoshida and Abderrahmane Kheddar. *Generation of whole-body optimal dynamic multi-contact motions.* The International Journal of Robotics Research, vol. 32, no. 9-10, pages 1104–1119, 2013. (Cited in page 24.)

[Levine 2012] Sergey Levine, Jack M. Wang, Alexis Haraux, Zoran Popović and Vladlen Koltun. *Continuous character control with low-dimensional embeddings.* ACM Transactions on Graphics, vol. 31, no. 4, pages 1–10, jul 2012. (Cited in page 24.)

[Liegeois 1977] A Liegeois. *Automatic supervisory control of the configuration and behavior of multibody mechanisms.* IEEE Transactions on Systems, Man, and Cybernetics, no. 12, pages 868–871, 1977. (Cited in page 19.)

[llya Dianov 2016] llya Dianov, Karinne Ramirez-Amaro, Pablo Lanillos, Emmanuel Dean-Leon, Florian Bergner and Gordon Cheng. *Extracting general task structures to accelerate the learning of new tasks.* In IEEE-RAS International Conference on Humanoid Robots 2016, Cancun, Mexico, Nov 2016. IEEE. (Cited in page 40.)

[Lohmeier 2009] Sebastian Lohmeier, Thomas Buschmann and Heinz Ulbrich. *Humanoid robot LOLA.* In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, pages 775–780. IEEE, 2009. (Cited in page 11.)

[Lüdtke 2017] Mathias Lüdtke. *Integration Platform and Deployment Environment.* https://github.com/ipa-mdl/ipde_utils, 2017. (Cited in page 74.)

[Lumelsky 2000] Vladimir Lumelsky, Michael S Shur and Sigurd Wagner. Sensitive skin, volume 18. World Scientific, 2000. (Cited in page 37.)

[Luo 2015] Jingru Luo and Kris Hauser. *Robust Trajectory Optimization Under Frictional Contact with Iterative Learning.* In Robotics, Science and Systems (RSS), 2015. (Cited in page 61.)

[Mansard 2007] Nicolas Mansard and François Chaumette. *Task sequencing for high-level sensor-based control.* Robotics, IEEE Transactions on, vol. 23, no. 1, pages 60–72, 2007. (Cited in page 20.)

[Mansard 2009a] N. Mansard, O. Khatib and a. Kheddar. *A Unified Approach to Integrate Unilateral Constraints in the Stack of Tasks.* IEEE Transactions on Robotics, vol. 25, no. 3, pages 670–685, June 2009. (Cited in page 42.)

[Mansard 2009b] Nicolas Mansard and François Chaumette. *Directional redundancy for robot control.* Automatic Control, IEEE Transactions on, vol. 54, no. 6, pages 1179–1192, 2009. (Cited in page 43.)

[Mansard 2009c] Nicolas Mansard, Olivier Stasse, Paul Evrard and Abderrahmane Kheddar. *A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks.* Advanced Robotics, . . . , 2009. (Cited in page 43.)

[Meguenani 2016] Anis Meguenani, Vincent Padois, Jimmy Da Silva, Antoine Hoarau and Philippe Bidaud. *Energy based control for safe human-robot physical interaction.* In International Symposium on Experimental Robotics, pages 809–818. Springer, 2016. (Cited in page 34.)

[Metta 2010] Giorgio Metta, Lorenzo Natale, Francesco Nori, Giulio Sandini, David Vernon, Luciano Fadiga, Claes Von Hofsten, Kerstin Rosander, Manuel Lopes, José Santos-Victor*et al. The iCub humanoid robot: An open-systems platform for research in cognitive development.* Neural Networks, vol. 23, no. 8, pages 1125–1134, 2010. (Cited in page 11.)

[Michalos 2015] George Michalos, Sotiris Makris, Panagiota Tsarouchi, Toni Guasch, Dimitris Kontovrakis and George Chryssolouris. *Design considerations for safe human-robot collaborative workplaces.* Procedia CIrP, vol. 37, pages 248–253, 2015. (Cited in page 9.)

[Miossec 2006] Sylvain Miossec, Kazuhito Yokoi and Abderrahmane Kheddar. *Development of a software for motion optimization of robots-application to the kick motion of the hrp-2 robot.* In Robotics and Biomimetics, 2006. ROBIO'06. IEEE International Conference on, pages 299–304. IEEE, 2006. (Cited in page 24.)

[Mistry 2010] Michael Mistry, Jonas Buchli and Stefan Schaal. *Inverse dynamics control of floating base systems using orthogonal decomposition.* In Robotics and Automation (ICRA), 2010 IEEE International Conference on, pages 3406–3412. IEEE, 2010. (Cited in page 22.)

[Mistry 2011] Michael Mistry and Ludovic Righetti. *Operational Space Control of Constrained and Underactuated Systems.* In Proceedings of robotics: science and systems, 2011. (Cited in page 62.)

[Mittendorfer 2012a] P. Mittendorfer and G. Cheng. *3D surface reconstruction for robotic body parts with artificial skins.* In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 4505–4510, Oct 2012. (Cited in page 39.)

[Mittendorfer 2012b] Philipp Mittendorfer and Gordon Cheng. *Uniform cellular design of artificial robotic skin.* In Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on, pages 1–5. VDE, 2012. (Cited in page 39.)

[Mittendorfer 2015] Philipp Mittendorfer, Eiichi Yoshida and Gordon Cheng. *Realizing whole-body tactile interactions with a self-organizing, multi-modal artificial skin on a humanoid robot.* Advanced Robotics, vol. 29, no. 1, pages 51–67, 2015. (Cited in page 39.)

[Mizuuchi 2007] Ikuo Mizuuchi, Yuto Nakanishi, Yoshinao Sodeyama, Yuta Namiki, Tamaki Nishino, Naoya Muramatsu, Junichi Urata, Kazuo Hongo, Tomoaki Yoshikai and Masayuki Inaba. *An advanced musculoskeletal humanoid kojiro.* In Humanoid Robots, 2007 7th IEEE-RAS International Conference on, pages 294–299. IEEE, 2007. (Cited in page 11.)

[Monje 2011] Concepción A Monje, Santiago Martínez, Alberto Jardón, Paolo Pierro, Carlos Balaguer and Delia Muñoz. *Full-Size humanoid robot TEO: Design attending mechanical robustness and energy consumption.* In Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on, pages 325–330. IEEE, 2011. (Cited in page 11.)

[Mordatch 2015] Igor Mordatch, Kendall Lowrey and Emanuel Todorov. *Ensemble-CIO : Full-Body Dynamic Motion Planning that Transfers to Physical Humanoids.* In Robotics and Automation (ICRA), IEEE International Conference on, 2015. (Cited in page 61.)

[Morisawa 2012] Mitsuharu Morisawa, Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kanako Miura and Kazuhiro Yokoi. *Balance control based on Capture Point error compensation for biped walking on uneven terrain.* IEEE-RAS International Conference on Humanoid Robots, pages 734–740, 2012. (Cited in page 63.)

[Moro 2013] Federico L Moro, Michael Gienger, Ambarish Goswami, Nikos G Tsagarakis and Darwin G Caldwell. *An attractor-based whole-body motion control (WBMC) system for humanoid robots.* In Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on, pages 42–49. IEEE, 2013. (Cited in page 22.)

[Murray 1994] Richard M. Murray, Zexiang Li and S Shankar Sastry. A Mathematical Introduction to Robotic Manipulation. CRC press, 1994. (Cited in page 18.)

[Nakamura 1986] Yoshihiko Nakamura and Hideo Hanafusa. *Inverse kinematic solutions with singularity robustness for robot manipulator control.* ASME, Transactions, Journal of Dynamic Systems, Measurement, and Control, vol. 108, pages 163–171, 1986. (Cited in page 19.)

[Nakamura 1987] Yoshihiko Nakamura, Hideo Hanafusa and Tsuneo Yoshikawa. *Task-priority based redundancy control of robot manipulators.* The International Journal of Robotics Research, vol. 6, no. 2, pages 3–15, 1987. (Cited in pages 20 and 62.)

[Nakamura 1990] Yoshihiko Nakamura. Advanced robotics: redundancy and optimization. Addison-Wesley Longman Publishing Co., Inc., 1990. (Cited in page 19.)

[Nakanishi 2012] Yuto Nakanishi, Yuki Asano, Toyotaka Kozuki, Hironori Mizoguchi, Yotaro Motegi, Masahiko Osada, Takuma Shirai, Junichi Urata, Kei Okada and Masayuki Inaba. *Design concept of detail musculoskeletal humanoid "Kenshiro"-toward a real human body musculoskeletal simulator.* In Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on, pages 1–6. IEEE, 2012. (Cited in page 11.)

[Negrello 2016] Francesca Negrello, Manolo Garabini, Manuel G Catalano, Przemyslaw Kryczka, Wooseok Choi, Darwin G Caldwell, Antonio Bicchi and Nikolaos G Tsagarakis. *Walk-man humanoid lower body design optimization for enhanced physical performance.* In Robotics and Automation (ICRA), 2016 IEEE International Conference on, pages 1817–1824. IEEE, 2016. (Cited in page 14.)

[Nelson 1995] Bradley Nelson and Pradeep K. Khosla. *Strategies for Increasing the Tracking Region of an Eye-in-Hand System by Singularity and Joint Limit Avoidance*. Journal of Robotics Research, vol. 14, pages 418–423, 1995. (Cited in page 43.)

[Nelson 2012] Gabe Nelson, Aaron Saunders, Neil Neville, Ben Swilling, Joe Bondaryk, Devin Billings, Chris Lee, Robert Playter and Marc Raibert. *Petman: A humanoid robot for testing chemical protective clothing*. Journal of the Robotics Society of Japan, vol. 30, no. 4, pages 372–377, 2012. (Cited in page 13.)

[Nguyen 2015] Quan Nguyen and Koushil Sreenath. *Optimal Robust Control for Bipedal Robots through Control Lyapunov Function based Quadratic Programs*. In Robotics, Science and Systems (RSS), 2015. (Cited in page 61.)

[Nishiwaki 2007] Koichi Nishiwaki, James Kuffner, Satoshi Kagami, Masayuki Inaba and Hirochika Inoue. *The experimental humanoid robot H7: a research platform for autonomous behaviour*. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol. 365, no. 1850, pages 79–107, 2007. (Cited in page 11.)

[Noda 2014] Shintaro Noda, Masaki Murooka, Shunichi Nozawa, Yohoei Kakiuchi, Kei Okada and Masayuki Inaba. *Generating Whole-body Motion Keep Away From Joint Torque , Contact Force , Contact Moment Limitations enabling Steep Climbing with a Real Humanoid Robot*. pages 1775–1781, 2014. (Cited in page 24.)

[Nori 2015] Francesco Nori, Silvio Traversaro, Jorhabib Eljaik, Francesco Romano, Andrea Del Prete and Daniele Pucci. *iCub whole-body control through force regulation on rigid non-coplanar contacts*. Frontiers in Robotics and AI, vol. 2, page 6, 2015. (Cited in page 13.)

[Nunez 2012] J Victor Nunez, Abraham Briseno, Dora A Rodriguez, J Manuel Ibarra and Victor M Rodriguez. *Explicit analytic solution for inverse kinematics of bioloid humanoid robot*. In Robotics Symposium and Latin American Robotics Symposium (SBR-LARS), 2012 Brazilian, pages 33–38. IEEE, 2012. (Cited in page 19.)

[Ogren 2000] P Ogren, N Egerstedt and Xiaoming Hu. *Reactive mobile manipulation using dynamic trajectory tracking*. In Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on, volume 4, pages 3473–3478. IEEE, 2000. (Cited in page 29.)

[Ogura 2006] Yusuke Ogura, Hiroyuki Aikawa, Kazushi Shimomura, Akitoshi Morishima, Hun-ok Lim and Atsuo Takanishi. *Development of a new humanoid* *robot WABIAN-2*. In Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, pages 76–81. IEEE, 2006. (Cited in page 11.)

[Orin 1979] DE Orin, RB McGhee, M Vukobratović and G Hartoch. *Kinematic and kinetic analysis of open-chain linkages utilizing Newton-Euler methods*. Mathematical Biosciences, vol. 43, no. 1-2, pages 107–130, 1979. (Cited in page 21.)

[Orin 2008] David E Orin and Ambarish Goswami. *Centroidal momentum matrix of a humanoid robot: Structure and properties*. In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, pages 653–659. IEEE, 2008. (Cited in page 21.)

[Orin 2013] David E Orin, Ambarish Goswami and Sung-Hee Lee. *Centroidal dynamics of a humanoid robot*. Autonomous Robots, vol. 35, no. 2-3, pages 161–176, 2013. (Cited in page 21.)

[Ott 2008] 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. (Cited in page 21.)

[Ott 2010] Christian Ott, Christoph Baumgärtner, Johannes Mayr, Matthias Fuchs, Robert Burger, Dongheui Lee, Oliver Eiberger, Alin Albu-Schäffer, Markus Grebenstein and Gerd Hirzinger. *Development of a biped robot with torque controlled joints*. In Humanoid Robots (Humanoids), 2010 10th IEEE-RAS International Conference on, pages 167–173. IEEE, 2010. (Cited in page 14.)

[Paden 1985] Bradley Evan Paden. *Kinematics and control of robot manipulators*. 1985. (Cited in page 19.)

[Pan 2013] Jia Pan, Ioan A Şucan, Sachin Chitta and Dinesh Manocha. *Real-time collision detection and distance computation on point cloud sensor data*. In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 3593–3599. IEEE, 2013. (Cited in page 32.)

[Park 2005a] Ill-Woo Park, Jung-Yup Kim, Jungho Lee and Jun-Ho Oh. *Mechanical design of humanoid robot platform KHR-3 (KAIST humanoid robot 3: HUBO)*. In Humanoid Robots, 2005 5th IEEE-RAS International Conference on, pages 321–326. IEEE, 2005. (Cited in page 11.)

[Park 2005b] Ill-Woo Park, Jung-Yup Kim, Seo-Wook Park and Jun-Ho Oh. *Development of humanoid robot platform khr-2 (kaist humanoid robot 2)*. International Journal of Humanoid Robotics, vol. 2, no. 04, pages 519–536, 2005. (Cited in page 11.)

[Park 2014] Chonhyon Park, Jia Pan and Dinesh Manocha. *High-dof robots in dynamic environments using incremental trajectory optimization.* International Journal of Humanoid Robotics, vol. 11, no. 02, page 1441001, 2014. (Cited in page 25.)

[Part 1985] SI Part. *Impedance control: An approach to manipulation.* Journal of dynamic systems, measurement, and control, vol. 107, page 17, 1985. (Cited in page 21.)

[pau 1981] *Robot manipulators: mathematics, programming and control.* 1981. (Cited in page 19.)

[Peiper 1968] Donald L Peiper. *The kinematics of manipulators under computer control.* Technical Report, DTIC Document, 1968. (Cited in page 19.)

[Peters 2007] Jan Peters, Michael Mistry, Firdaus E. Udwadia, Jun Nakanishi and Stefan Schaal. *A unifying framework for robot control with redundant DOFs.* Autonomous Robots, vol. 24, no. 1, pages 1–12, oct 2007. (Cited in page 62.)

[Phan 2011] Samson Phan, Zhan Fan Quek, Preyas Shah, Dongjun Shin, Zubair Ahmed, Oussama Khatib and Mark Cutkosky. *Capacitive skin sensors for robot impact monitoring.* In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pages 2992–2997. IEEE, 2011. (Cited in page 36.)

[Polverini 2014] Matteo Parigi Polverini, Andrea Maria Zanchettin and Paolo Rocco. *Real-time collision avoidance in human-robot interaction based on kinetostatic safety field.* In Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, pages 4136–4141. IEEE, 2014. (Cited in page 30.)

[Pontryagin 1987] Lev Semenovich Pontryagin. Mathematical theory of optimal processes. CRC Press, 1987. (Cited in page 23.)

[Popovic 2004] Marko Popovic, Andreas Hofmann and Hugh Herr. *Angular momentum regulation during human walking: biomechanics and control.* In Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on, volume 3, pages 2405–2411. IEEE, 2004. (Cited in page 22.)

[Pratt 2001] Jerry Pratt, Chee-meng Chew, Ann Torres, Peter Dilworth and Gill Pratt. *Virtual model control: An intuitive approach for bipedal locomotion.* The International Journal of Robotics Research, vol. 20, no. 2, pages 129–143, 2001. (Cited in page 60.)

[Pratt 2006] Jerry Pratt, J. Carff, S. Drakunov and Ambarish Goswami. *Capture Point: A Step toward Humanoid Push Recovery.* 2006 6th IEEE-RAS International Conference on Humanoid Robots, 2006. (Cited in pages 63 and 64.)

[Pratt 2008] Jerry Pratt and Ben Krupp. *Design of a bipedal walking robot.* In Proc. of SPIE, volume 6962, pages 69621F1–69621F13, 2008. (Cited in page 14.)

[Pratt 2015] Gill Pratt. *http://www.theroboticschallenge.org/*, 2015. (Cited in page 59.)

[Quinlan 1993] Sean Quinlan and Oussama Khatib. *Elastic bands: Connecting path planning and control.* In Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on, pages 802–807. IEEE, 1993. (Cited in page 30.)

[Radford 2015] Nicolaus A Radford, Philip Strawser, Kimberly Hambuchen, Joshua S Mehling, William K Verdeyen, A Stuart Donnan, James Holley, Jairo Sanchez, Vienny Nguyen, Lyndon Bridgwater*et al. Valkyrie: NASA's first bipedal humanoid robot.* Journal of Field Robotics, vol. 32, no. 3, pages 397–419, 2015. (Cited in page 12.)

[Raghavan 1993] Madhusudan Raghavan and Bernard Roth. *Inverse kinematics of the general 6R manipulator and related linkages.* Journal of Mechanical Design, vol. 115, no. 3, pages 502–508, 1993. (Cited in page 19.)

[Ramdani 2005] Nacim Ramdani and Philippe Poignet. *Robust Dynamic Experimental Identification of Robots With Set Membership Uncertainty.* IEEE/ASME Transactions on Mechatronics, vol. 10, no. 2, pages 253–256, 2005. (Cited in page 61.)

[Ramos 2014] Oscar E. Ramos, Nicolas Mansard and Philippe Soueres. *Whole-body Motion Integrating the Capture Point in the Operational Space Inverse Dynamics Control.* In Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on, 2014. (Cited in pages 60, 64, and 70.)

[Raunhardt 2007] Daniel Raunhardt and Ronan Boulic. *Progressive clamping.* In Robotics and Automation, 2007 IEEE International Conference on, pages 4414–4419. IEEE, 2007. (Cited in page 43.)

[Richalet 1978] Jacques Richalet, A Rault, JL Testud and J Papon. *Model predictive heuristic control: Applications to industrial processes.* Automatica, vol. 14, no. 5, pages 413–428, 1978. (Cited in page 23.)

[Righetti 2010] Ludovic Righetti, Jonas Buchli, Michael Mistry and Stefan Schaal. *Inverse Dynamics With Optimal Distribution of Ground Reaction Forces for Legged Robots.* In Emerging Trends in Mobile Robotics - Proceedings of the 13th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines, pages 580–587, Singapore, 2010. World Scientific Publishing Co. Pte. Ltd. (Cited in page 60.)

[Righetti 2011a] Ludovic Righetti, Jonas Buchli, Michael Mistry and Stefan Schaal. *Control of legged robots with optimal distribution of contact forces.* 2011 11th

IEEE-RAS International Conference on Humanoid Robots, pages 318–324, oct 2011. (Cited in page 22.)

[Righetti 2011b] Ludovic Righetti, Jonas Buchli, Michael Mistry and Stefan Schaal. *Inverse dynamics control of floating-base robots with external constraints: A unified view.* 2011 IEEE International Conference on Robotics and Automation, pages 1085–1090, may 2011. (Cited in page 22.)

[Righetti 2013a] Ludovic Righetti, Jonas Buchli, Michael Mistry, Mrinal Kalakrishnan and Stefan Schaal. *Optimal distribution of contact forces with inverse-dynamics control.* The International Journal of Robotics Research, vol. 32, no. 3, pages 280–298, 2013. (Cited in page 21.)

[Righetti 2013b] Ludovic Righetti, Jonas Buchli, Michael Mistry, Mrinal Kalakrishnan and Stefan Schaal. *Optimal distribution of contact forces with inverse dynamics control.* The International Journal of Robotics Research, no. January, jan 2013. (Cited in page 68.)

[Robotics 2014] PAL Robotics. *REEM-C: Robotics Research.* 2014. (Cited in page 11.)

[Rubrecht 2010] Sébastien Rubrecht, Vincent Padois, Philippe Bidaud and Michel De Broissia. *Constraints Compliant Control : constraints compatibility and the displaced configuration approach.* In IEEE/RSJ Intenational Conference on Intelligent Robots and Systems (IROS), 2010. (Cited in page 63.)

[Rubrecht 2012] Sébastien Rubrecht, Vincent Padois, Philippe Bidaud, Michel Broissia and Max Da Silva Simoes. *Motion safety and constraints compatibility for multibody robots.* Autonomous Robots, vol. 32, no. 3, pages 333–349, jan 2012. (Cited in pages 34 and 46.)

[Rybski 2012] Paul Rybski, Peter Anderson-Sprecher, Daniel Huber, Chris Niessl and Reid Simmons. *Sensor fusion for human safety in industrial workcells.* In Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, pages 3612–3619. IEEE, 2012. (Cited in page 31.)

[Saab 2011] Layale Saab, Nicolas Mansard, Francois Keith, J-Y. Fourquet and Philippe Soueres. *Generation of dynamic motion for anthropomorphic systems under prioritized equality and inequality constraints.* Robotics and Automation, IEEE International Conference on, pages 1091–1096, may 2011. (Cited in page 62.)

[Saab 2013] Layale Saab, Oscar E. Ramos, Nicolas Mansard, Philippe Soueres and Jean-yves Fourquet. *Dynamic Whole-Body Motion Generation under Rigid Contacts and other Unilateral Constraints.* IEEE Transactions on Robotics, vol. 29, no. 2, pages 346–362, 2013. (Cited in pages 22 and 63.)

[Salini 2009] Joseph Salini, Sébastien Barthélemy and Philippe Bidaud. *Lqp controller design for generic whole body motion.* Climbing and Walking Robots and the Support Technologies for Mobile Machines, page 24, 2009. (Cited in page 20.)

[Salini 2011] Joseph Salini, Vincent Padois and Philippe Bidaud. *Synthesis of complex humanoid whole-body behavior: A focus on sequencing and tasks transitions.* 2011 IEEE International Conference on Robotics and Automation, pages 1283–1290, may 2011. (Cited in page 22.)

[Samson 1991] Claude Samson, Bernard Espiau and Michel Le Borgne. Robot control: The task function approach. Oxford University Press, 1991. (Cited in page 43.)

[Schäffer 2008] AA Schäffer, O Eiberger, M Grebenstein, S Haddadin, C Ott, T Wimböck, S Wolf and G Hirzinger. *Soft robotics, from torque feedback-controlled lightweight robots to intrinsically compliant systems.* IEEE Robotics and Automation Magazine, vol. 15, no. 3, pages 20–30, 2008. (Cited in page 21.)

[Schiavi 2009] Riccardo Schiavi, Antonio Bicchi and Fabrizio Flacco. *Integration of active and passive compliance control for safe human-robot coexistence.* In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, pages 259–264. IEEE, 2009. (Cited in page 32.)

[Schultz 2010] Gerrit Schultz and Katja Mombaur. *Modeling and optimal control of human-like running.* IEEE/ASME Transactions on mechatronics, vol. 15, no. 5, pages 783–792, 2010. (Cited in page 24.)

[Sciavicco 1987] L Sciavicco and B Siciliano. *Solving the inverse kinematic problem for robotic manipulators.* In RoManSy 6, pages 107–114. Springer, 1987. (Cited in page 19.)

[Sentis 2004] Luis Sentis and Oussama Khatib. *Task-oriented control of humanoid robots through prioritization.* International Journal of Humanoid Robotics, pages 1–16, 2004. (Cited in page 62.)

[Sentis 2005] Luis Sentis and Oussama Khatib. *Synthesis of whole-body behaviors through hierarchical control of behavioral primitives.* International Journal of Humanoid Robotics, vol. 2, no. 4, pages 505–518, 2005. (Cited in page 62.)

[Sherikov 2014] Alexander Sherikov, Dimitar Dimitrov and Pierre-brice Wieber. *Whole body motion controller with long-term balance constraints.* In Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on, 2014. (Cited in page 63.)

[Shin 2011] Dongjun Shin, Zhan Fan Quek, Samson Phan, Mark Cutkosky and Oussama Khatib. *Instantaneous stiffness effects on impact forces in human-friendly robots.* In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pages 2998–3003. IEEE, 2011. (Cited in page 37.)

[Siciliano 1991] Bruno Siciliano and J-JE Slotine. *A general framework for managing multiple tasks in highly redundant robotic systems.* In Advanced Robotics, 1991.'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on, pages 1211–1216. IEEE, 1991. (Cited in pages 19, 20, 43, 44, and 62.)

[Siciliano 1999] Bruno Siciliano. *The Tricept robot: Inverse kinematics, manipulability analysis and closed-loop direct kinematics algorithm.* Robotica, vol. 17, no. 4, pages 437–445, 1999. (Cited in page 19.)

[Singh 1996] Leena Singh, Harry Stephanou and John Wen. *Real-time robot motion control with circulatory fields.* In Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on, volume 3, pages 2737–2742. IEEE, 1996. (Cited in page 30.)

[Slovich 2012] M Slovich, N Paine, K Kemper, A Metzger, A Edsinger, J Weber and L Sentis. *Hume: A bipedal robot for human-centered hyperagility.* In Dynamic Walking conference, 2012. (Cited in page 14.)

[Smits 2008] Ruben Smits, Tinne De Laet, Kasper Claes, Herman Bruyninckx and Joris De Schutter. *iTASC: a tool for multi-sensor integration in robot manipulation.* IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, vol. 2, pages 426–433, aug 2008. (Cited in page 62.)

[Soenke Kock 2006] Peter J. Eriksson Mats Myhr Soenke Kock Jan Bredahl and Kevin Behnisch. *Taming the robot: Better safety without higher fences.* In ABB Review, pages 11–14. ABB, 2006. (Cited in page 31.)

[Spong 1992] Mark W Spong. *Remarks on robot dynamics: canonical transformations and Riemannian geometry.* In Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on, pages 554–559. IEEE, 1992. (Cited in page 21.)

[Spong 2006] Mark W Spong, Seth Hutchinson and Mathukumalli Vidyasagar. Robot modeling and control, volume 3. wiley New York, 2006. (Cited in page 18.)

[Stentz 2015] Anthony Stentz, Herman Herman, Alonzo Kelly, Eric Meyhofer, G Clark Haynes, David Stager, Brian Zajac, J Andrew Bagnell, Jordan Brindza, Christopher Dellin*et al. Chimp, the cmu highly intelligent mobile* *platform.* Journal of Field Robotics, vol. 32, no. 2, pages 209–228, 2015. (Cited in page 12.)

[Sussmann 1997] Hector J Sussmann and Jan C Willems. *300 years of optimal control: from the brachystochrone to the maximum principle.* IEEE Control Systems Magazine, vol. 17, no. 3, pages 32–44, 1997. (Cited in page 22.)

[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. (Cited in page 24.)

[Tassa 2014] Yuval Tassa, Nicolas Mansard and Emo Todorov. *Control-limited differential dynamic programming.* In Robotics and Automation (ICRA), 2014 IEEE International Conference on, pages 1168–1175. IEEE, 2014. (Cited in page 24.)

[Tellez 2008] Ricardo Tellez, Francesco Ferro, Sergio Garcia, Esteban Gomez, Enric Jorge, Dario Mora, Daniel Pinyol, Joan Oliver, Oriol Torres, Jorge Velazquez*et al.* *Reem-B: An autonomous lightweight human-size humanoid robot.* In Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on, pages 462–468. IEEE, 2008. (Cited in page 11.)

[Tevatia 2000] Gaurav Tevatia and Stefan Schaal. *Inverse kinematics for humanoid robots.* In Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on, volume 1, pages 294–299. IEEE, 2000. (Cited in page 20.)

[Todorov 2012] Emanuel Todorov, Tom Erez and Yuval Tassa. *Mujoco: A physics engine for model-based control.* In Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, pages 5026–5033. IEEE, 2012. (Cited in page 24.)

[Traversaro 2015] Silvio Traversaro, Andrea Del Prete, Serena Ivaldi and Francesco Nori. *Inertial parameters identification and joint torques estimation with proximal force / torque sensing.* In Robotics and Automation (ICRA), 2015 IEEE International Conference on, 2015. (Cited in page 61.)

[Tsagarakis 2013] Nikos G Tsagarakis, Stephen Morfey, Gustavo Medrano Cerda, Li Zhibin and Darwin G Caldwell. *Compliant humanoid coman: Optimal joint stiffness tuning for modal frequency control.* In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 673–678. IEEE, 2013. (Cited in pages 11 and 14.)

[Uicker 1969] JJ Uicker. *Dynamic behavior of spatial linkages.* Journal of Engineering for Industry, vol. 91, no. 1, pages 251–258, 1969. (Cited in page 21.)

[Vannoy 2008] John Vannoy and Jing Xiao. *Real-time adaptive motion planning (RAMP) of mobile manipulators in dynamic environments with unforeseen changes*. IEEE Transactions on Robotics, vol. 24, no. 5, pages 1199–1212, 2008. (Cited in page 31.)

[Vick 2013] Axel Vick, Dragoljub Surdilovic and J Kruger. *Safe physical human-robot interaction with industrial dual-arm robots*. In Robot Motion and Control (RoMoCo), 2013 9th Workshop on, pages 264–269. IEEE, 2013. (Cited in page 36.)

[Vogel 2013] Christian Vogel, Christoph Walter and Norbert Elkmann. *A projection-based sensor system for safe physical human-robot collaboration*. In Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on, pages 5359–5364. IEEE, 2013. (Cited in page 31.)

[Vukobratović 1972] Miomir Vukobratović and J Stepanenko. *On the stability of anthropomorphic systems*. Mathematical biosciences, vol. 15, no. 1-2, pages 1–37, 1972. (Cited in page 13.)

[Wahl 2010] Friedrich M Wahl. *Online Trajectory Generation : Basic Concepts for Instantaneous Reactions to Unforeseen Events*. vol. 26, no. 1, pages 94–111, 2010. (Cited in page 34.)

[Wallén 2008] Johanna Wallén. The history of the industrial robot. Linköping University Electronic Press, 2008. (Cited in pages 4 and 5.)

[Warren 1989] Charles W Warren. *Global path planning using artificial potential fields*. In Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on, pages 316–321. IEEE, 1989. (Cited in pages 29 and 30.)

[Weisshardt 2016] F. Weisshardt and F. Koehler. *Automatic Testing Framework for Benchmarking Applications*. In Proceedings of ISR 2016: 47st International Symposium on Robotics, pages 1–6, June 2016. (Cited in page 73.)

[Wensing 2013] Patrick M Wensing and David E Orin. *Generation of dynamic humanoid behaviors through task-space control with conic optimization*. In Robotics and Automation (ICRA), 2013 IEEE International Conference on, pages 3103–3109. IEEE, 2013. (Cited in page 22.)

[Whitman 2013] Eric C Whitman. *Coordination of multiple dynamic programming policies for control of bipedal walking*. PhD thesis, US National Science Foundation, 2013. (Cited in page 24.)

[Whitney 1969] Daniel E Whitney. *Resolved motion rate control of manipulators and human prostheses*. Man-Machine Systems, IEEE Transactions on, 1969. (Cited in page 19.)

[Wieber 2002] Pierre-Brice Wieber. *On the stability of walking systems.* In Proceedings of the international workshop on humanoid and human friendly robotics, 2002. (Cited in page 63.)

[Wieber 2015] Pierre-Brice Wieber, Russ Tedrake and Scott Kuindersma. *Modeling and Control of Legged Robots.* In Bruno Siciliano and Oussama Khatib, editors, Springer Handbook of Robotics, 2nd Ed, chapter 48. 2015. (Cited in pages 63 and 64.)

[Wiener 1949] Norbert Wiener. Extrapolation, interpolation, and smoothing of stationary time series, volume 7. MIT press Cambridge, MA, 1949. (Cited in page 23.)

[Wisse 2013a] M Wisse. *Factory in a Day Project (FiaD)*, 2013. (Cited in page 7.)

[Wisse 2013b] Martijn Wisse. *Factory in a Day Project (FiaD).* http://www.factory-in-a-day.eu/, Oct. 2013. (Cited in page 27.)

[Yang 2006] Yuandong Yang*et al. Elastic roadmaps: Globally task-consistent motion for autonomous mobile manipulation in dynamic environments.* 2006. (Cited in pages 31 and 32.)

[Yang 2010] Yuandong Yang and Oliver Brock. *Elastic roadmaps—motion generation for autonomous mobile manipulation.* Autonomous Robots, vol. 28, no. 1, pages 113–130, 2010. (Cited in pages 31 and 32.)

[Yi 2015] Seung-Joon Yi, Stephen G McGill, Larry Vadakedathu, Qin He, Inyong Ha, Jeakweon Han, Hyunjong Song, Michael Rouleau, Byoung-Tak Zhang, Dennis Hong*et al. Team thor's entry in the darpa robotics challenge trials 2013.* Journal of Field Robotics, vol. 32, no. 3, pages 315–335, 2015. (Cited in page 12.)

[Yoshida 2006] Eiichi Yoshida, Oussama Kanoun, Claudia Esteves and Jean-Paul Laumond. *Task-driven support polygon reshaping for humanoids.* In Humanoid Robots, 2006 6th IEEE-RAS International Conference on, pages 208–213. IEEE, 2006. (Cited in page 20.)

[You 2005] Bum-Jae You, Y Choi, M Jeong, Doik Kim, Y Oh, C Kim, J Cho, Minchul Park and S Oh. *Network-based humanoids' Mahru'and'Ahra'.* In Proceedings of the International Conference on Ubiquitous Robots and Ambient Intelligence, pages 376–379, 2005. (Cited in page 11.)

[Zanchettin 2016] Andrea Maria Zanchettin, Nicola Maria Ceriani, Paolo Rocco, Hao Ding and Björn Matthias. *Safety in human-robot collaborative manufacturing environments: Metrics and control.* IEEE Transactions on Automation Science and Engineering, vol. 13, no. 2, pages 882–893, 2016. (Cited in page 31.)

[Zhao 2014] Ran Zhao, Daniel Sidobre and Wuwei He. *Online via-points trajectory generation for reactive manipulations*. In Advanced Intelligent Mechatronics (AIM), 2014 IEEE/ASME International Conference on, pages 1243–1248. IEEE, 2014. (Cited in page 34.)

[Zucker 2013] Matt Zucker, Nathan Ratliff, Anca D Dragan, Mihail Pivtoraiko, Matthew Klingensmith, Christopher M Dellin, J Andrew Bagnell and Siddhartha S Srinivasa. *CHOMP: Covariant Hamiltonian optimization for motion planning*. The International Journal of Robotics Research, vol. 32, no. 9-10, pages 1164–1193, 2013. (Cited in page 25.)