Technische Universität München
Fakultät für Elektrotechnik und Informationstechnik

Safety Awareness for Rigid and Elastic Joint Robots: An Impact Dynamics and Control Framework

Nico Mansfeld, M.Sc.

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines Doktors der Ingenieurwissenschaften (Dr.-Ing.) genehmigten Dissertation.

Vorsitzender: Prof. Dr. Gordon Cheng

Prüfende der Dissertation:

1. Prof. Dr.-Ing. Sami Haddadin
2. Prof. Dr.-Ing. Alin Albu-Schäffer

Die Dissertation wurde am 16.11.2021 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 22.04.2022 angenommen.

Acknowledgments

I started my research endeavor approx. nine years ago at the German Aerospace Center (DLR), where I worked for seven years. For the last two years, I have been with the Technical University of Munich (TUM). I would like to take the opportunity to thank all persons that have supported me and made this thesis possible.

First of all, I would like to thank my supervisor and mentor Prof. Sami Haddadin, who has continuously encouraged and guided me in my academic research. Sami has always been a source of inspiration. He taught me how to formulate and approach interesting research questions and how to present the achieved results. We spent many evenings discussing research work and finalizing publications and presentations. My deepest gratitude goes to my former institute head Prof. Alin Albu-Schäffer and department head Dr. Christian Ott for their supervision, support, and the trust they have placed in me at DLR. I am grateful for the opportunity to have worked in this highly reputable international institution. My work environment has been very inspiring and has helped me to grow both professionally and personally.

I would like to thank the members of the DLR Human-Centered Robotics group Dr. Roman Weitschat, Rico Belder, Kai Krieger, Sven Parusel, Tim Rokahr, Antonio Gonzales Marin, Thomas Eiband, Dr. Matteo Saveriano, and Franz Steinmetz for their great teamwork in research projects and the development, maintenance, and demonstration of our robot systems. Sincere thanks go to Dr. Alexander Dietrich, Dr. Mehmet Can Özparpucu, Dr. Teodor Tomić, Dr. Gianluca Garofalo, and Fabian Beck for sharing their wisdom about control theory. Thank you also to Sebastian Wolf, Dr. Martin Pfanne, and Manuel Keppler for maintaining the DLR David system, supporting experiments, and discussions on intrinsically elastic joint robots. Many thanks go to Tilo Wüsthoff for creating great illustrations and bringing the design perspective into our conversations. Thank you to my former students Badis Djellab, Jaime Raldúa Veuthey, Julius Jankowski, Youssef Michel, and Guillermo Gómez Peña for their valuable inputs. I am also thankful to have worked with Robin Kirschner, Mazin Hamad, Marvin Becker, and Dr. Saeed Abdolshah on safety in human-robot interaction.

Special thanks go to my former DLR office mates Dr. Moritz Maier, Felix Huber, Dr. Roman Weitschat, Dr. Matteo Saveriano, Clemens Karl, Dr. Theodoros Stouraitis, Korbinian Nottensteiner, and Bao-Anh Dang-Vu for creating a pleasant working atmosphere and for all the engaging and fun discussions. I would also like to thank Dr. Katharina Hertkorn, Dr. Moritz Maier, Dr. Elisabeth Jensen, Robin Kirschner, and Dr. Luis Figueredo for proofreading the manuscript. I am deeply grateful to my family, my wife Nicole for her love as well as her continuous support and encouragement, and to our beloved daughters Nele and Julie, who make our lives richer every day.

Munich, November 2021 Nico Mansfeld

Abstract

In recent years, there has been a paradigm shift from the use of rigid, heavy-duty industrial robots requiring safeguarding measures to protect human co-workers to compliant lightweight robots that allow direct physical interaction. Modern tactile robots such as the Franka Emika Panda feature advanced sensing capabilities and soft control schemes. Intuitive app-based user interfaces make it simple even for non-experts to program complex robot applications. This technology is currently experiencing a strong push in the industry and opens up new fields of research and development. In the meantime, the next generation of robots is gaining maturity and increasing attention, which does not realize compliance via control, but intrinsic mechanical elasticity. Robots with intrinsically elastic joints mimic some musculoskeletal properties of humans or animals and can realize a previously unattained performance, robustness, and energy efficiency. However, there are still some hurdles to overcome for the economic and widespread use of modern compliant robots. In particular, a still unresolved problem is how to fulfill human safety and high performance requirements simultaneously. The current robot safety regulations are rather restrictive, and there is a lack of methods to systematically plan, implement, and verify collaborative applications that meet the demands of the application domain.

This thesis aims to make different robot system classes aware of their global safety properties and realize safe and efficient motions. The Safety Map concept is introduced, a map that captures human injury occurrence and robot inherent global or task-dependent safety properties in a unified manner, making it a novel, powerful, and convenient tool to quantitatively analyze the safety performance of a certain robot design. The kinematic, dynamic, and surface properties of stationary manipulators with rigid and compliant actuation as well as mobile manipulators are processed towards the Safety Map. The map representations of human injury data are derived by classifying and summarizing the most relevant impact studies from the biomechanics and robotics literature. The Safety Map and the previously developed Safe Motion Unit, which ensures biomechanically safe robot velocities along the trajectory, constitute a hierarchical tool stack that enables safe mechanism design, planning, and real-time control. It is independent of contact models, interpretation-free, applicable to arbitrary robots, and compatible with existing standards such as the ISO 10218.

An important physical quantity for assessing a robot's safety and performance characteristics is the maximum achievable endpoint velocity. While the theory is well understood

for rigid joint robots, it has been challenging to compute the maximum achievable velocity and the corresponding excitation trajectories for robots with intrinsic joint elasticity. In this thesis, the achievable speed gain of 1-DOF visco-elastic joints is determined in closed form. For complex, highly nonlinear multi-joint robots, several methods are proposed for approximating the maximum achievable endpoint velocity in real-time. The methods are verified via optimal control and previous real-world throwing experiments. In contrast to numerical optimal control approaches commonly used in literature, the developed methods require minimal computational effort, making them useful for quick hardware design iterations and motion planning. Gaining high speeds via suitable excitation is one problem in intrinsically elastic joint robots; the other is to dampen the inherently oscillatory system if undesired, potentially dangerous vibrations occur. The intrinsic mechanical damping is typically very low, so damping needs to be introduced via control. This works reviews existing and proposes novel vibration suppression schemes. The effectiveness of the controllers is demonstrated on the DLR David system in two benchmark experiments. Here, damping and impact shock absorption performance metrics are defined to quantitatively compare the tested controllers.

Collaborative and tactile robots often have more than six joints, which makes them more dexterous than classical industrial robots and allows them to fulfill auxiliary tasks in addition to the main task via so-called self-motions. This thesis investigates how the redundant degrees of freedom can be exploited to improve the performance in pHRI while maintaining safety. A control scheme is proposed that locally minimizes the robot effective mass perceived during potential contact. The reduction in effective mass allows increasing the robot speed while still satisfying the safety constraint within the SMU and Safety Map framework. In hierarchical redundancy resolution schemes, the task hierarchy is usually defined instantaneously and does not take the temporal dimension of task fulfillment into account. By design, the auxiliary tasks do not affect the primary task; however, it can occur that the primary task detriments the performance of the auxiliary tasks. The solution developed and demonstrated in this thesis is to temporarily slow down the main task while preserving the desired path. By reducing the execution speed, the null space tasks are given more time to optimize their performance criteria. Finally, interaction patterns are developed that enable kinematically reconfiguring the robot via self-motions. These patterns are intuitively interpretable to the robot user and straightforward to implement. It is shown how they can be integrated into remote and haptic teaching interfaces, and practical examples validate the proposed methods.

In summary, this thesis covers safety and performance analysis, motion planning, core safety control, and performance optimization, constituting a systematic framework for addressing the safety-performance tradeoff in pHRI.

Contents

Abbreviations and Symbols

In this thesis, scalar quantities are written as plain letters, e.g., λ, c, K. Vectors and matrices are represented by bold letters, e.g. $\boldsymbol{x}$, $\boldsymbol{M}(\boldsymbol{q})$. The total derivative w.r.t. time is indicated by a dot above the symbol, i.e., $\dot{\boldsymbol{x}} = \frac{d}{dt}\boldsymbol{x}$, $\ddot{\boldsymbol{x}} = \frac{d^2}{dt^2}\boldsymbol{x}$. The Euclidean norm of a vector $\boldsymbol{q}$ is denoted by $||\boldsymbol{q}||$, the dot product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ by $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$, and their cross product by $\boldsymbol{a} \times \boldsymbol{b}$. The estimate of the quantity $\boldsymbol{x}$ is denoted by $\hat{\boldsymbol{x}}$, and $\tilde{\boldsymbol{x}}$ denotes the error of $\boldsymbol{x}$.

All symbols are introduced in the text before they are used. In some sections, the argument is omitted for the sake of brevity. Several variables appear with different subscripts, superscripts, additional symbols, and dimensions. In the following list, the quantities are generally described without being further specified. The specific meaning becomes apparent when the respective variable is introduced in the text. Please note that the list of symbols is not complete, but it contains symbols that appear frequently or are of major importance in the thesis.

List of Symbols

c	Constant
d	Distance
i, j	Indices (for numbering)
l	Length
m	Number or mass
n	Number (e.g., number of joints)
t	Time
D	Damping ratio
H	Hamiltonian, performance criterion, or pseudo energy
J	Cost function
T	Kinetic Energy
U	Potential Energy
V	Total energy
ϵ	Small positive constant
$\mathcal{N}$	Null space of a matrix
Ω	Control region

$\boldsymbol{f}$	Vector function describing a task with coordinates $\boldsymbol{x}$
$\boldsymbol{g}$	Vector of gravity torques
$\boldsymbol{h}$	Vector of equality or inequality constraints
$\boldsymbol{p}$	Vector describing a point in space
$\boldsymbol{q}$	Vector of link-side joint positions
$\boldsymbol{u}$	Vector of control inputs or a Cartesian direction
$\boldsymbol{x}$	Vector of Cartesian coordinates
$\boldsymbol{B}$	Motor inertia matrix
$\boldsymbol{C}$	Coriolis and centrifugal matrix
$\boldsymbol{D}$	Damping matrix
$\boldsymbol{F}$	Vector of (generalized) Cartesian forces
$\boldsymbol{J}$	Jacobian matrix
$\boldsymbol{I}$	Identity matrix
$\boldsymbol{K}$	Stiffness or gain matrix
$\boldsymbol{M}$	Inertia Matrix
$\boldsymbol{N}$	Nullspace projection matrix
$\boldsymbol{Q}$	Modal decoupling matrix
$\boldsymbol{U}, \boldsymbol{S}, \boldsymbol{V}$	Matrix components of singular value decomposition
$\boldsymbol{W}$	Weighting matrix
$\boldsymbol{Z}$	Matrix describing the null space base of a Jacobian matrix
$\boldsymbol{\lambda}$	Vector of costates or eigenvalues
$\boldsymbol{\nu}$	Vector of translational Cartesian velocities
$\boldsymbol{\omega}$	Vector of angular Cartesian velocities
$\boldsymbol{\varphi}$	Vector of elastic deflections
$\boldsymbol{\sigma}$	Vector of stiffness adjuster positions
$\boldsymbol{\tau}$	Vector of joint or motor torques
$\boldsymbol{\theta}$	Vector of motor positions
$\boldsymbol{\Lambda}$	Reflected inertia matrix (e.g., in Cartesian directions $\boldsymbol{x}$)

List of Abbreviations

AIS	Abbreviated Injury Scale
adm	Admittance
aug	Augmented
cf.	*Confer* (compare)
dyn	Dynamic
e.g.	*Exempli gratia* (for example)
etc.	*Et cetera* (and so forth)
eq	Equilibrium
ext	External
i.e.	*Id est* (that is)
imp	Impedance
kin	Kinematic
max	Maximum
min	Minimum
ns	Null space

opt	Optimal
rel	Relative
rob	Robot
suc	Successive
symm	Symmetric
tor	Torque
vel	Velocity
w.r.t.	With respect to
DLR	Deutsches Zentrum für Luft- und Raumfahrt (German Aerospace Center)
DOF	Degree(s) of freedom
EE	End-effector
ESP	Elastic structure-preserving control
IEEE	Institute of Electrical and Electronics Engineers
LWR	DLR/KUKA Lightweight Robot
PD	Proportional derivative
pHRI	Physical human-robot interaction
OC	Optimal control
SEA	Series elastic actuation
SISO	Single-input, single-output
SME	Small and medium-sized enterprise
SMU	Safe Motion Unit
SVD	Singular value decomposition
TMP	Translational manipulability ellipsoid
TUM	Technical University of Munich
VIA	Variable impedance actuation
VSA	Variable stiffness actuation

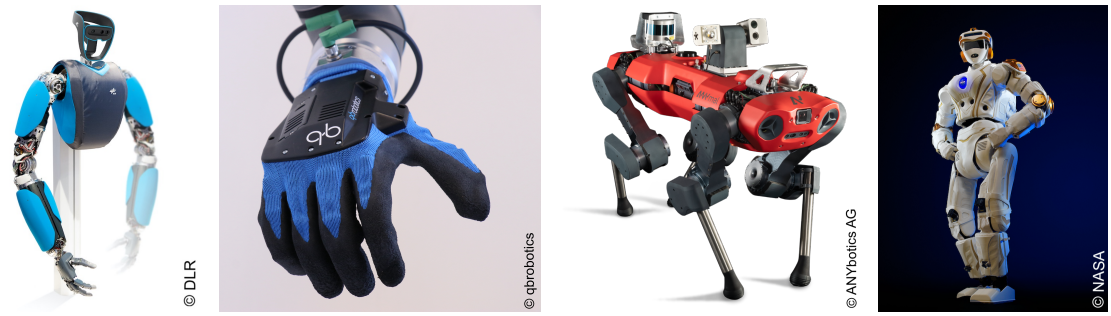
CHAPTER 1

Introduction

In physical human-robot interaction (pHRI), humans and robots share the same workspace. Because protective fences are no longer required, many industrial applications can be automated flexibly and cost-effectively. Classical collaborative robots (also referred to as cobots) such as the Universal Robot family feature lightweight mechanical design and joint-level position/velocity control. They are typically used for pick and place or machine tending tasks and provide rather basic physical interaction schemes, e.g., hand-guided pose teaching or surface polishing.

However, industrial applications that require sensitive physical interaction, particularly assembly tasks, can hardly be realized with cobots. For such tasks, so-called *soft robots* that feature advanced sensor technology and control algorithms are used nowadays. The KUKA iiwa, e.g., has integrated joint-torque sensing that enables implementing soft control concepts like impedance control. In contrast to classical collaborative robots, the iiwa can sense even slight contact with the environment and react with compliant and safe behavior. However, the robot's programming framework requires expert knowledge and does not offer tactile control for accurate force regulation yet, which makes it difficult for users to implement manipulation tasks. The Franka Emika Panda (see Fig. 1.1), on the other

Fig. 1.1: Physical human-robot interaction with the Franka Emika Panda.



(a) DLR David **(b)** qb SoftHand **(c)** ANYbotics ANYmal C **(d)** NASA Valkyrie

Fig. 1.2: Selection of robot systems with compliant actuation.

hand, is the first *tactile robot*. It has all aforementioned mechatronic and soft control features, is fully force-controlled, and comes with various physical interaction patterns such as haptic gestures or teaching modes. Also, this system provides an intuitive user interface and pre-programmed robot apps requiring minimal user configuration. This enables even non-experts to solve rather complex tasks. With the Panda, tactile robots and intuitive programming concepts were transferred from research labs to the industry, where it has now become a marketable and popular technology. The technology also opens up new and unforeseen fields of application inside and outside industrial shop floors. For example, tactile robots are increasingly used in medical applications, elderly care, and schools to acquaint and train the next generation of so-called "robonatives" with state-of-the-art robotics technology [1, 2].

Collaborative and tactile robots have already reached a certain level of maturity and real-world relevance. In the meanwhile, novel robot types have been proposed and are subject to research for several years. Current commercial robots realize compliance via soft control, as their mechanical structure is rather rigid [3, 4]. In robotics research, it has become increasingly popular to make robots intrinsically elastic by employing spring mechanisms in the joints [5–7]. A selection of such systems is depicted in Fig. 1.2. The intention of compliant actuation is to mimic some musculoskeletal properties of humans and animals. The capability to store and release energy can be exploited to perform explosive motions such as throwing and jumping, and in cyclic tasks like running, joint elasticity enhances the system's energy efficiency [8–11]. In pHRI, the motivation behind introducing compliant actuation is also to improve human safety and the robot's robustness to impacts.

Recently, the advances made in research and industry in the context of pHRI have partially been reflected in standardization [12, 13]. The current draft of the product-level (type C) standard ISO 10218-2 for safety in robot systems distinguishes between different collaboration types. For collaboration with so-called *Power and Force Limiting (PFL)*, which allows robots and humans to physically interact in a shared workspace, force and pressure thresholds for human pain onset were defined that should be respected during either intended or unintended contact. The norm proposes methods for measuring contact forces and computing safe robot speeds based on a simple contact model. To implement the suggestions made in the norm, several manufacturers have developed force and pressure sensing devices that estimate human contact forces in real robot applications. In Germany, such tools are used by the Technical Inspection Association (TÜV) and professional associations (BG/DGUV) to certify pHRI robots, tools, and applications.

For many applications in the industry, however, the current safety regulations can still be considered rather restrictive. In order to meet them, users often need to reduce the operational robot speed considerably, which increases the cycle time and makes pHRI applications often uneconomical. In other words, pHRI robots do not exploit their full potential in real-world applications, which hinders small and medium-sized enterprises (SMEs), start-ups, and large-scale industries from deploying this technology cost-effectively and quickly. In order to meet the demands of the application domain, the state of the art must advance through further research and development, which can then drive future revisions of the standards forward. Figure 1.3 shows a selection of key hardware, planning, and control technologies for achieving a certain performance and safety level in pHRI. A holistic human-centered solution for making pHRI safe and efficient (see right column in Fig. 1.3) requires thorough knowledge of the human's ergonomics as well as physical and perceived safety. The application should be optimized w.r.t. safety and monitored with reliable, possibly redundant sensors during task execution. The robot should be able to reason about the task context and the human and his/her intentions to plan actions and interact purposefully. Advanced task sharing and control methods shall fulfill even complex tasks successfully.

1.1 Problem Statement

One of the biggest hurdles in implementing pHRI in practice is running applications safely and at high performance. What is still missing for making robots truly aware of safety is a thorough understanding of human injury biomechanics and the transfer of this knowledge to safe control, planning, and mechanism design methods. This thesis aims to contribute several essential building blocks for making rigid and elastic joint robots aware of human safety and generating safe and efficient motions.

The first objective is to develop an impact dynamics framework for stationary rigid and elastic joint manipulators as well as mobile manipulators. The tool shall allow the characterization and quantitative comparison of entire robot designs and different types of robots in terms of their dynamic and safety performance. This enables to make minimal potential harm a key requirement not only in control, but also in planning and mechanism design. In this endeavor, human injury data, safety constraints, and the robots' safety-related parameters need to be processed towards a common coordinate system. An essential robot parameter is the maximum achievable operational velocity, which was shown to affect human injury probability during a collision significantly [14–18]. While the velocity capabilities of rigid joint robots have been understood well, an open question to be addressed in this thesis is how fast robots with intrinsic joint elasticity can become with suitable excitation. The achievable operational velocities shall be investigated and how to compute them efficiently. Provided the impact dynamics framework, the next objective is to deduce a planning scheme that generates safe and efficient robot trajectories. During task execution, a fundamental safety requirement for every robot is the ability to stop under any operating condition. This critical function is comparatively easy to implement in rigid joint robots and included in all cobots and tactile robots available on the market; see Fig. 1.3 (bottom left). However, the dynamics of intrinsically elastic joint robots are inherently oscillatory, which makes the problem more difficult for this robot type. Therefore, the fourth objective in this thesis is to effectively suppress undesired vibrations in

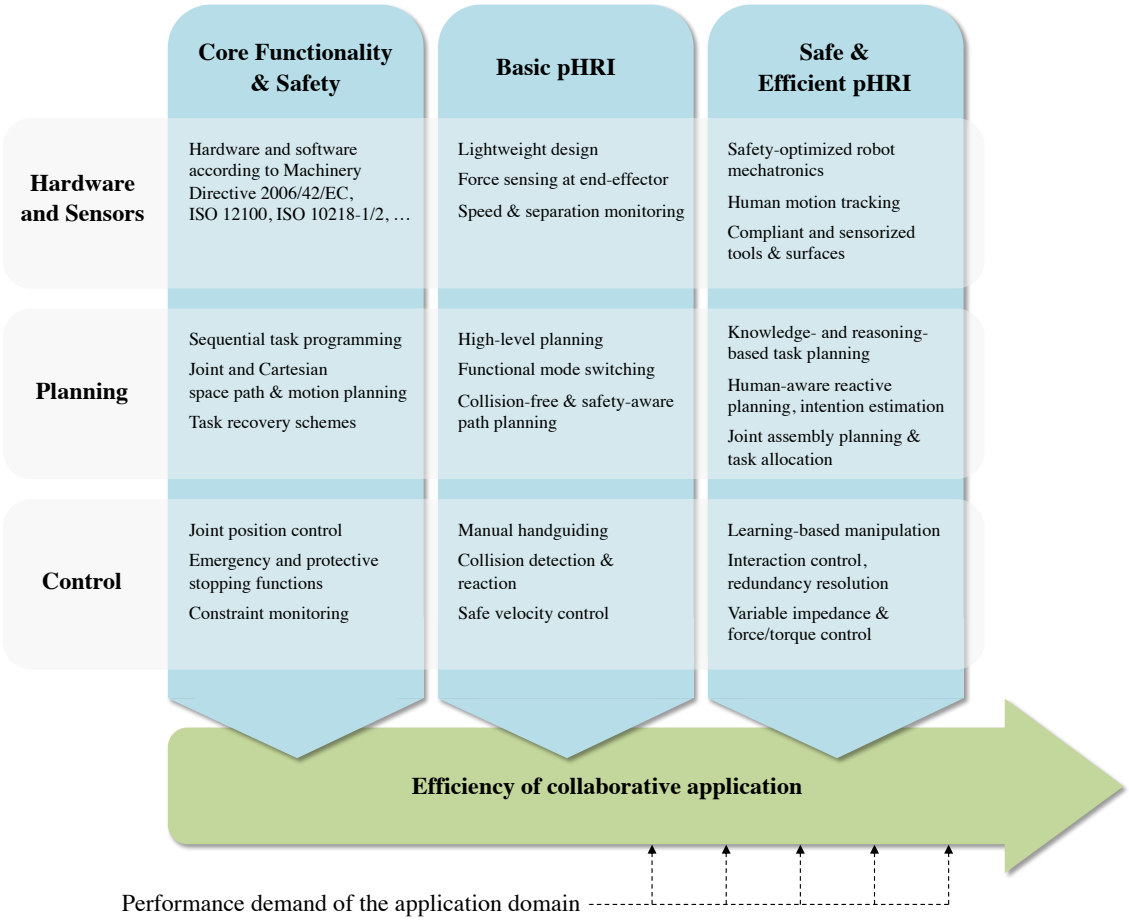


Fig. 1.3: Selection of key hardware, planning, and control technologies for achieving a certain efficiency and safety level in pHRI. It is distinguished between the core functionality that is included in every commercial robot system (left column), basic pHRI functionality realized by modern collaborative and tactile robots (middle column), and safe and efficient pHRI (right column), which still requires research, technology transfer, and revision of the current standards in order to meet the demands of the application domain.

intrinsically elastic joint robots via control. Once the system has been made controllable, the next problem is to ensure human safety during a potential collision. The local control problem of generating biomechanically safe velocities even in case of unforeseen collisions is treated in [12]. The so-called *Safe Motion Unit (SMU)* is based on a realistic and a-priori model-independent safety analysis. Even when there is no sensor to detect a human and potentially endangered body parts in the shared workspace, the SMU ensures safety. However, in that case, the controller assumes worst-case conditions (e.g., collisions with the most vulnerable body part), which may result in an undesirable speed reduction. This means that safety may come at the cost of additional cycle time. As collaborative robots typically have more degrees of freedom than necessary to solve common tasks in pHRI, the aim is to analyze how the cycle time of safe applications can be reduced by exploiting the robot's redundant degrees of freedom. In the context of redundancy resolution, it is well known that the kinematic reconfiguration of manipulators via self-motions is often non-predictable and non-intuitive. However, it is desirable (also from a safety perspective)

that the user can anticipate the robot's motion. The objective of this work is, therefore, to develop interaction patterns that allow the user to reconfigure the robot intuitively. From the objectives described above, seven research questions are derived that are addressed in this thesis:

Q1 How can entire robot designs be related to available human injury data and safety thresholds to assess the robot's safety characteristics on a global or task-dependent scale?

Q2 How can safe and efficient trajectories be generated for collaborative applications provided the global characterization of a robot's impact dynamics, the shared workspace, and safety thresholds?

Q3 How much operational speed can robots with multiple intrinsically elastic joints achieve compared to rigid joint robots, and how can the speed gain be computed efficiently?

Q4 What are the global safety properties of robots with intrinsically elastic joints, and how do elastic and rigid joint robots compare in terms of safety?

Q5 How can undesired vibrations in robots with intrinsic joint elasticity be suppressed via simple but effective control schemes?

Q6 How can the kinematics of redundant robots be exploited to maximize the task velocity while ensuring safety simultaneously?

Q7 How can the user kinematically reconfigure a redundant robot intuitively and interactively?

After the following summary of the state of the art, Sec. 1.3 describes how the research questions Q1 – Q7 are addressed in this thesis.

1.2 State of the Art

The following literature overview covers the most relevant topics related to this thesis, namely collision analysis in biomechanics and robotics, safe robot design, control, and planning for safe pHRI, and the design and control of intrinsically elastic robots.

1.2.1 Collision Analysis in Biomechanics and Robotics

Types of Experiments and Subjects

In order to understand the injury mechanisms of different human body parts during direct collisions, a large number of experiments and publications have been produced in biomechanics and forensics research during the last 50 years [12, 19–34]. These tests aim to investigate human collision behavior, determine and verify injury tolerances, develop mappings from impact conditions or measured physical quantities such as force or acceleration to human injury probability, and develop and verify anthropomorphic test devices that simulate the human kinematics and dynamics. In general, one can distinguish between three categories of direct impact tests [34]. In full-scale experiments, the subject is hit by

the entire machine or object. Component tests belong to the second category, where components such as car seats are isolated and tested under controlled laboratory conditions, e.g., in sled test setups. The third category regards single machine parts. In such tests, the impactors have the same or similar size and shape as single machine components and are propelled towards the subject. The advantage of these experiments is that impacts can be delivered precisely and repeatably under well-defined test conditions. In collision tests that investigate injury tolerances, the subjects are (for ethical reasons) typically animal or human cadavers (often referred to as post mortem test objects (PMTO)) or mechanical human surrogates. For the analysis of pain thresholds or minor injury such as contusions, experiments have also been conducted with human volunteers [13, 35–37].

Quantification and Classification of Human Injury

To evaluate and categorize injuries, the internationally established Abbreviated Injury Scale (AIS) is commonly used. It was defined in [38, 39] and subdivides the level of human injury into seven categories ranging from *none* to *fatal*; see Tab. 1.1. While observed human injury can be directly classified in terms of AIS, an important and non-trivial problem is how the contact conditions or measured sensor data during a collision can be mapped to the AIS scale. This mapping is required to estimate injury probability in crash-test experiments using anthropomorphic test devices or to account for safety in machine/robot design, planning, and task execution already prior to impact occurrence. There exist two approaches to establish the desired mapping. The first and most common approach relates physical quantities such as force or pressure to injury probability via contact models or injury severity indices; see Fig. 1.4 (upper path). The second approach, proposed in [12], is to relate collision parameters directly to medically observed injury without using intermediate physical quantities; see Fig. 1.4 (lower path). In the following, both approaches are described in more detail.

Tab. 1.1: Abbreviated Injury Scale.

AIS	Severity	Type of injury
0	None	None
1	Minor	Superficial injury
2	Moderate	Recoverable
3	Serious	Possibly recoverable
4	Severe	Not fully recoverable without care
5	Critical	Not fully recoverable with care
6	Fatal	Unsurvivable

Model-Based Relation of Output Quantities and Injury There exist several injury severity indices that are particularly defined for a certain body region. The Head Injury Criterion (HIC) [40] is an example that is widely used in automotive crash testing. It is

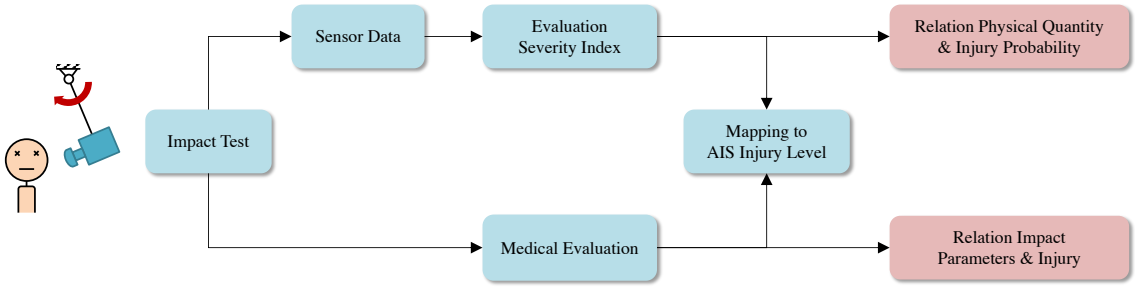


Fig. 1.4: From impact test to AIS injury level: Model-based (upper path) and medically oriented, data-driven approach (lower path).

defined as

$$\mathrm{HIC}_{36} = \max_{\Delta t} \left\{ \Delta t \left(\frac{1}{\Delta t} \int_{t_1}^{t_2} ||\ddot{\boldsymbol{x}}_H||_2 \mathrm{d}t \right)^{\left(\frac{5}{2}\right)} \right\} \leq 650\,, \tag{1.1}$$
$$\Delta t = t_2 - t_1 \leq \Delta t_{\max} = 36 \text{ ms}\,.$$

The human head acceleration resulting from an impact is denoted by $||\ddot{\boldsymbol{x}}_H||$ and measured in $\mathrm{g} = 9.81\,\mathrm{m/s^2}$. The optimization is carried out by varying the start and stop time t_1 and t_2 in the range 0 – 36 ms. The HIC weights the human head acceleration and impact duration. It takes into account that the human head can withstand high accelerations when the impact duration is low. The biomechanical literature contains a large variety of such indices. Collision data has to be acquired from impact test series to define a severity index for a certain type of interaction. Biomechanical analysis and interpretation, medical evaluation, and statistics are then employed to derive a relationship between measured physical *output* quantities such as force, deflection, or stress and resulting injury. This approach allows employing anthropomorphic crash-test dummies equipped with the respective sensors for injury assessment. After impact testing, the severity indices can be evaluated and linked to the AIS or another injury scale. Unfortunately, there are some problems with this approach. Firstly, it is difficult to measure quantities such as stress for complex impact geometries. Secondly, the consistency with medically observed injury is often insufficient, as a single physical quantity cannot capture the complex human impact response, especially when an injury is produced [12].

Medically Oriented Relation of Input Parameters and Injury In [12], it was proposed to relate the collision *input* parameters to medically observed injury without using intermediate physical quantities such as force or pressure. In drop-test experiments on porcine soft-tissue, the authors established the mapping *impact mass, velocity, surface geometry* $\rightarrow$ *injury*. This approach is purely data-driven and interpretation-free, which means that more consistent results can be expected compared to model-driven injury prediction. The injury data was processed towards safety curves, which provide a maximum biomechanically safe velocity as a function of the instantaneous robot effective mass and surface curvature in the direction of motion. These representations were further developed into the Safe Motion Unit (SMU); see Fig. 1.5. The controller monitors the effective mass and the maximum allowable velocity of each point of interest on the robot structure along the trajectory. If the current task velocity exceeds the maximum biomechanically safe

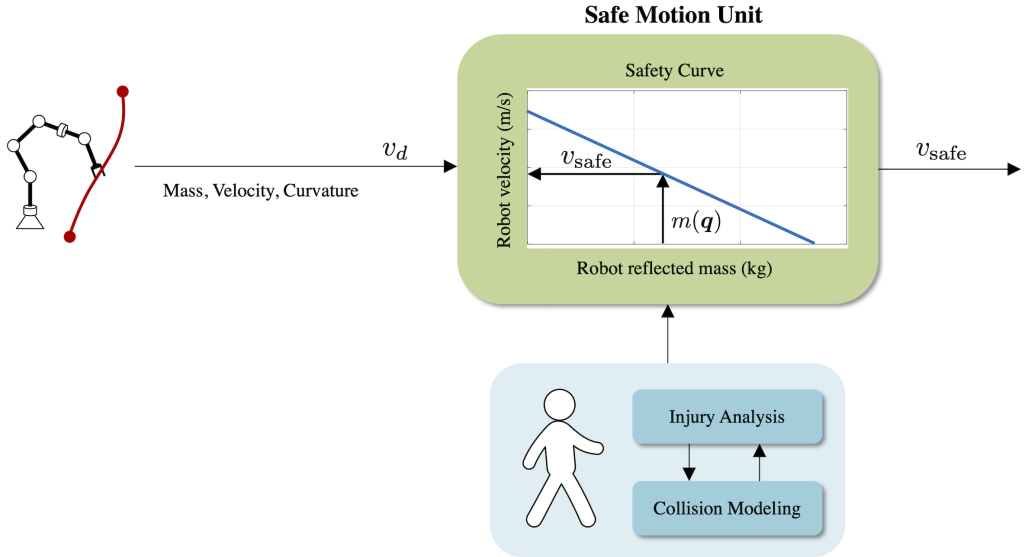


Fig. 1.5: The Safe Motion Unit (SMU) provides biomechanically safe robot velocities for control. The collision data obtained from collision experiments or simulations is processed towards so-called safety curves, which relate the current configuration-dependent robot reflected mass $m(\boldsymbol{q})$ and the surface geometry to the safe velocity v_{safe}. If v_{safe} is lower than the desired speed v_d, then the velocity is reduced in order to ensure safety.

speed, the controller reduces the velocity such that all safety constraints are met. As a result, human safety is ensured during task execution even in case of entirely unforeseen collisions.

Impact Experiments in Robotics

The first systematic experimental analysis of human safety in robotics was carried out in [15, 16, 41–43]. The authors conducted impact test series with lightweight and heavy-duty robots in unconstrained, constrained, and partially constrained collision scenarios. Blunt impacts were delivered to a Hybrid III dummy, which is commonly used for injury assessment in automobile crash-testing. The human injury probability was assessed in terms of well-established injury severity indices such as the HIC or Chest Compression Criterion (CC). Furthermore, the authors analyzed the influence of robot parameters (e.g., mass, velocity, and elasticity) on injury severity and the capability of collision detection and reaction schemes to alleviate the effect of unwanted collisions. An important observation made in the experiments was that collaborative and tactile robots could produce mild soft-tissue injuries such as contusions instead of more severe injuries like fractures. The biomechanical injury severity indices used in automobile crash-testing are thus of little relevance for pHRI. Robotic cutting and stabbing experiments with sharp tools were carried out on porcine soft-tissue in [44]. As sharp tools should generally be avoided in pHRI, the authors investigated edgy contact in [12]. So-called geometric impactor primitives (sphere, edge, cuboid etc.) with their corresponding parameters were identified and defined. Systematic drop-test experiments were then conducted to analyze the effect of the impactor mass, velocity, and curvature on the soft-tissue injury. The resulting injury of skin, muscles and tendons, and nerves and vessels of porcine soft tissue was assessed in

terms of the AO-classification [45]. The aforementioned safety curves were developed and embedded into the SMU controller based on the collected collision data. The first human voluntary soft-tissue experiments with an LWR and a healthy male adult were conducted in [37]. A volunteer was hit at various body locations by a sphere- and wedge-shaped impactor. The medical observation consisted of an injury severity assessment according to the AO-classification, biomechanical analysis, pain classification, and medical imaging. In [13, 46], the research in this direction was continued. Volunteer experiments were conducted with a pendulum test setup, where also different impactor masses, velocities, and curvatures were selected. The volunteers were hit at different body parts, the resulting injury and pain were observed, accompanied by immediate visual medical assessment and ultrasound imaging. Recent investigations on pain onset and pain tolerance of different human body parts were carried out in [36, 47, 48], for example. In addition to understanding the human collision biomechanics, these and other collision experiments conducted in the robotics context typically aim to provide standardization data for proper risk and safety assessment in pHRI.

1.2.2 Safe Mechanical Robot Design

To ensure collision safety in terms of robot mechanics, lightweight manipulator design is considered essential. By lowering the robot inertia, the kinetic energy upon impact and thus also the injury probability is reduced [49]. In addition to lightweight but relatively rigid manipulators, intrinsic joint elasticity and soft covering have been employed to improve collision safety. In compliant actuation, the motor inertia is largely decoupled from the link inertia, which means that the human only perceives the effective link inertia during a collision [50–52]. Besides systems in which elasticity is introduced deliberately, this decoupling property was also observed in so-called flexible joint robots such as the LWR family, where the constant joint stiffness of the harmonic drive gear and the torque sensor is orders of magnitude higher [14]. First works on the safety assessment of elastic joint robots were conducted in [17, 50, 52–55], where the influence of the elastic mechanism on the robot's model- or metrics-based safety rating was analyzed. These works typically considered the 1-DOF case or a limited number of collision scenarios in multi-joint robots. The influence of soft covering on collision safety was investigated in [56–58], for example. In [59], an airbag was proposed that covers the end-effector and absorbs impacts safely. For most robots, selecting the inertial, kinematic, and elastic properties is usually driven by certain design decisions and criteria such as workspace dimensions, maximum speed, payload, and cost. In contrast, the authors of [60–62] proposed to integrate quantitative safety (and performance) criteria already in the mechanical design phase in order to make robots inherently safe.

1.2.3 Planning and Control for Safety

In robot planning and control, it is generally distinguished between *pre-* and *post-collision* schemes to ensure human safety. In pre-collision methods, the goal is to minimize potential harm prior to impact, e.g., by avoiding potentially hazardous contact with the human. The objective of post-collision schemes is to properly detect and identify the collision and react purposefully and safely.

Pre-Collision Schemes

In pre-collision schemes, safety constraints can be employed on different hierarchy levels and levels of abstraction. AI-based frameworks use symbolic approaches to plan, execute, and reason about robot actions [63–68]. They allow specifying actions such as "place the cup on the table" in terms of objects that shall be manipulated and desired effects. Such action specifications need to be interpreted before the robot can carry out the desired motion [68, 69]. To cope with uncertainties, unmodeled and unwanted effects in real-world applications, and to reason about actions in high-level planning algorithms, a connection between the symbolic and control layer must be made [67, 70, 71]. By integrating representations of the human, contact scenarios and events as well as safety properties of the robot (e.g., sharp geometry at the end-effector) into the planning language, the robot can be made aware of safety in pHRI and plan and execute its actions under consideration of safety constraints [67].

In terms of trajectory planning and control, many metrics- and model-based approaches have been proposed [61, 72–78]. In [61, 73] a danger index/criterion was introduced that consists of a combination of several safety-relevant quantities such as the robot effective mass and velocity, and the distance between robot and human. In [73], a hierarchical motion generator was proposed that monitors the robot and human state and minimizes the danger criterion during task execution. The so-called danger field was introduced in [74], which quantifies how dangerous the current robot configuration and velocity are to the environment. The danger field can be computed in closed form and was integrated into a controller which generates safe motions. The reduction of the impact force was treated in [72, 79], where impact models were used to relate the instantaneous robot parameters to collision force. In [75], an impedance-based controller was proposed that fulfills performance requirements and energy- and power-based safety constraints at the same time. The passive variable impedance controller was realized via energy tanks. Also in [80], a controller was developed that keeps the energy of the robot within desired safety limits. In [76], a control strategy was developed that minimizes the dissipated kinetic energy in a potential inelastic impact in either constrained or unconstrained collisions, which is estimated through a contact model. For redundant robots, several redundancy resolution schemes have been proposed to improve safety. These methods range from collision avoidance [81] and self-collision avoidance [82] to collision detection and reaction [83], or the reduction of the impact force [84].

In [85, 86], the authors aim at realizing not only physically safe but also comfortable and socially acceptable motions and interaction behavior. A human-aware planning and manipulation framework was proposed, which explicitly considers the human posture and kinematics, vision field, and personal preferences. Another measure to ensure safety is to avoid any potentially dangerous contact with humans or the environment. Collision avoidance has been a vivid area of research for many years. [87–93]. Recent approaches in pHRI avoid collisions in real-time by monitoring the human and robot and taking their kinematic and dynamic properties as well as safety criteria into account [93–98].

Post-Collision Schemes

During either intended or unintended contact in pHRI, robots must react in a compliant manner. In collaborative robots with joint torque sensing, impedance control is widely used to define the robot's contact behavior properly and to achieve a soft interaction between

the human and the robot [3, 4]. Also hybrid force/impedance controllers were proposed that realize both compliant impedance behavior and accurate force tracking [99–101]. In order to prevent or mitigate injury caused by a collision with the robot, it is essential that the system can detect and handle contact quickly, effectively, and reliably. A thorough overview on this subject can be found in [102]. The five main phases in the so-called collision event pipeline [102] are:

1. **Detection** To detect collisions with the robot structure, many proprioceptive model-based and model-free observers have been proposed in the robotics literature [103–109] and also sensitive skins exist [110–112]. Improved collision detection sensitivity and accuracy can be obtained by fusing inertial measurement unit (IMU) measurements with proprioceptive sensing [113]. For contact detection, one needs to select proper collision detection thresholds. However, due to noise and varying dynamics of the motor torques, tuning these thresholds is difficult. Today, the generalized momentum observer [103, 104] is the most popular among the collision detection schemes and widely used both in research and the industry.

2. **Localization** Especially for collision reaction, it is important to know the point of contact (joint or link) along the robot structure. Solutions to the isolation problem are reported in [102] for one contact at a time and in [114] for multiple contacts.

3. **Isolation** In addition to contact location, the direction and magnitude of the external contact wrench are of interest. In [103, 104, 106] it is described how the contact wrench can be estimated based on the collision detection residual.

4. **Classification** To discriminate between intended and unintended collisions, [115] proposed to high-pass filter the estimated external torques, as high-frequency torque components can typically be associated with (undesired) fast rigid impacts. In [116], a learning-based approach for contact classification was proposed that was validated on an LWR.

5. **Reaction** Upon contact, the robot should react in a safe and compliant manner. The most simple collision reaction scheme is to stop the robot once the collision is detected. However, this reaction can lead to possibly hazardous clamping or blocking of the human. In [105, 117] several reaction schemes were developed that take all the contextual information obtained in steps $1 - 4$ into account. One possible reaction scheme is to retract the robot from the contact location in the opposite direction of impact.

1.2.4 Design and Control of Intrinsically Elastic Robots

As mentioned previously, robots with intrinsic joint elasticity have become increasingly popular due to their ability to absorb impacts and store and release spring potential energy to perform explosive, cyclic, and energy-efficient motions. Both in research and industry, many applications can benefit from compliant actuation, such as throwing objects in logistics, handling delicate objects, or walking on uneven terrain.

Design Principles

Overviews on design concepts and components for compliant actuation are provided in [5–7]. Elastic actuators can be sub-divided into systems with fixed and variable impedance. The Series Elastic Actuator (SEA) [118] is the most prominent compliant actuator with constant mechanical stiffness. Variable impedance actuators (VIAs) can realize variable impedance via mechanical reconfiguration; a subgroup without dedicated damping elements is variable stiffness actuators (VSAs). There exist three main principles to change the mechanical impedance in VIA systems [5]:

1. Variation of the spring preload

2. Variation of the transmission ratio between spring and link

3. Modification of the spring physical properties

The majority of joint designs proposed in literature belong to the first category. In biologically inspired antagonistic setups with two pairs of motor and nonlinear spring, the spring preload is changed via co-contraction [119, 120]. Another possibility is to use a nonlinear element between the spring and the link so that only one spring is required. In such actuators, one motor controls the spring stiffness, the other the link equilibrium position [121–123]. In the first category of VIA systems, the exploitable potential energy is reduced when the spring is preloaded. This is theoretically not the case in the second category, where the stiffness is modified by changing the transmission ratio between the output link and the spring [124, 125]. These systems often have many moving parts, which makes them more complex than actuators of spring preload type [6]. In the third subgroup, the spring stiffness is altered by changing the spring cross-section area, length, and/or the elastic modulus. The elastic modulus is a material property and difficult to change; however, mechanisms that change the length and cross-section area were proposed in [126–128]. The authors of [6] review the design process of VIA/VSA actuators from the desired use case to the selection of components. Common design contradictions in output power, potential energy capacity, stiffness range, efficiency, and accuracy are discussed. As many VSA setups with different design principles and characteristics exist, it is difficult for the end-user to select the actuator that best suits his/her application requirements. In [7], a datasheet was developed that systematically extends the specification of traditional electric actuators by the most important VSA parameters. It serves as a unified and compact description that enables comparing different setups quantitatively.

Although many novel and even unforeseen tasks can be realized with elastic joint robots, the control of these systems is more complex compared to rigid joint robots. This is because robots with compliant actuation have more state variables, i.e., motor and link coordinates instead of motor coordinates only, and they are underactuated and inherently oscillatory. In particular for cyclic or explosive tasks, suitable excitation trajectories need to be generated that systematically exploit the system's energy storage and release capabilities. Also, undesired vibrations in VSA robots with low mechanical damping have to be accounted for to ensure smooth link side position regulation and tracking as well as good shock absorption and braking performance.

Position Regulation

For so-called flexible joint robots like the DLR LWR, where the joint stiffness is at least an order of magnitude higher than in intrinsically elastic robots and usually modeled as a linear spring, control laws for regulating the link position were proposed in [129–132], for example. These concepts were generalized to joints with nonlinear elasticity in [133]. The above controllers use control-input-collocated variables for feedback; their performance depends on the joint stiffness. If the stiffness of the elastic mechanism is low, the link side damping performance is also limited. For intrinsically elastic joint robots, better damping performance can be achieved with the controllers [134–136], which also feed back non-collocated variables. In [136], a gain scheduling approach based on a set of linear quadratic regulators (LQRs) is used to control the position and stiffness at the link while damping undesired oscillations. In [135] a modal decoupling-based approach is used to transform the linearized system dynamics into a set of decoupled single-input single-output (SISO) systems. The aim is to modify the intrinsic robot dynamics as little as possible while introducing the desired damping behavior. In decoupled space, the closed-loop dynamics resemble two second-order critically damped systems with desired damping behavior.

Trajectory Tracking

A thorough overview of trajectory tracking controllers for flexible joint robots is given in [137] and [138]. Early works on tracking control can be found in [139–141]. Other relevant approaches consider the extension of adaptive control laws [142] to the flexible joint case [143], feedback linearization [139], [144], [145], integral manifold control [139], integrator backstepping [146], and cascaded structures [147]. More recently, the concept of elastic structure-preserving control (ESP, ESP+) was introduced for intrinsically elastic robots to perform link-side tracking while realizing the desired damping behavior [148, 149]. An overview of the concept, including rigorous stability analysis, is provided in [150]. The concept adds damping and feedforward terms to the link dynamics while leaving the plant's inertial properties and the structure of the nonlinear springs unchanged. The controller achieves closed-loop dynamics that structurally equal the dynamics of the original coordinates by introducing a coordinate transformation of the motor dynamics and new coordinates that reflect these damping and feedforward terms. The ESP and ESP+ were validated on the multi-joint DLR David system [151].

Impedance Control

Impedance control is widely used in soft robots such as the Franka Emika Panda or the LWR to achieve soft interaction control [134]. The joint elasticity in these robots is relatively high and typically caused by the flexibility of the torque sensor or the transmission. For robots with deliberate joint elasticity, where the stiffness is orders of magnitude lower, it was shown that the vibration damping performance of traditional impedance controllers is rather insufficient. In [152], a Cartesian impedance control law was proposed for robots with highly nonlinear, variable stiffness characteristics. The approach can realize a wide range in desired stiffness and damping and preserves the system's intrinsic inertial and elastic properties.

Vibration Suppression

Several control laws were proposed for suppressing undesired oscillations via energy dissipation. In [153], a phase plane approach was proposed to reduce the overall system energy via a simple state proportional controller that commands a desired motor velocity trajectory. Also, a state-triggered (bang-bang) version was described. The motor velocity switches to either minimum, maximum, or zero, depending on the current quadrant in each joint's deflection/velocity plane. Optimal control solutions and controllers of bang-bang type were proposed in [153] (model-free) and [154] (model-based) for 1-DOF linear or visco-elastic systems. The model-free controllers can be implemented joint-wise; the extension to n-DOF is therefore rather simple. In practice, however, control laws of bang-bang type are generally prone to undesired chattering effects, particularly in the vicinity of the robot's equilibrium position. This needs to be accounted for in real systems to avoid unwanted wear of the motors. In [155], a torque-based control law was proposed that regulates the system energy to the desired value. Depending on the desired energy, oscillations can be either damped out or induced. In the former case, an asymptotically stable equilibrium position in terms of the desired motor position is reached. Energy dissipation via active variation of the stiffness in VSA joints was addressed in [156, 157]. The control law proposed in [157] switches the desired stiffness between a minimum and maximum value while the motor position remains constant.

Cyclic and Explosive Motions

The generation of cyclic motions such as walking, running, or drumming in elastic joint robots has gained increasing attention [10, 11, 158–165]. The goal is to excite the natural modes of the system in a stable and coordinated fashion to perform energy-efficient gaits in legged robots or pick and place motions in manipulators, for example. In [10], a controller of bang-bang type was proposed that induces intrinsic oscillatory motions by switching the motor position based on a defined joint torque threshold. In [160], the parameters of multi-body legged robots were selected such that desired eigenmodes can be excited, and in [159], an energy-based limit cycle controller was developed. Significant progress has been made in understanding and unifying the theory of oscillatory normal modes in nonlinear multi-body systems very recently [160, 162–165].

In robotics literature, many authors investigated the optimality principles in visco-elastic and variable impedance joints [17, 52, 166–173]. In particular for fast motions, the goal is to understand how elastic joints can be excited (time-)optimally, how much operational velocity can be achieved, and how the system parameters affect the solution. While analytical solutions can be derived for the 1-DOF case, it has been difficult to extend this knowledge to n-DOF systems, as the dynamics of multi-joint robots are usually quite complex. Fast motions have, therefore, mainly been generated via numerical optimal control tools. In [8, 9, 174], it was investigated how the elastic energy of systems with two or more DOF can be exploited to realize explosive motions, in particular, the throwing of a ball as far as possible. The influence of coupling stiffness, nonlinear dynamics, and nonlinear elasticity with stiffness adjustment on unimodal and sequential-type motions was analyzed. Optimal trajectories were generated both for academic and real-world systems such as DLR David and a MACCEPA robot. Based on simulations and experimental results, basic power flow and energy transfer mechanisms like consecutive loading and unloading of the springs for sequential type motions were investigated. Furthermore, comparisons

were made to human studies on explosive motions. In [175], a learning framework was proposed that encodes sample (optimal control) trajectories into a Dynamic Movement Primitive (DMP) system [176]. The DMP system could reconstruct learned trajectories and extrapolate to other tasks with near-optimal performance.

1.3 Contribution

The main contributions of this thesis to the state of the art and their relation to the research questions Q1 – Q7 posed in Sec. 1.1 are described in the following. An overview of the objectives and their interrelationship is illustrated in Fig. 1.6. The proposed methods were experimentally validated on a DLR/KUKA LWR and DLR David, simulations of the proposed methods also on the PUMA 560, Franka Emika Panda, and systems consisting of a mobile platform and a Panda or LWR.

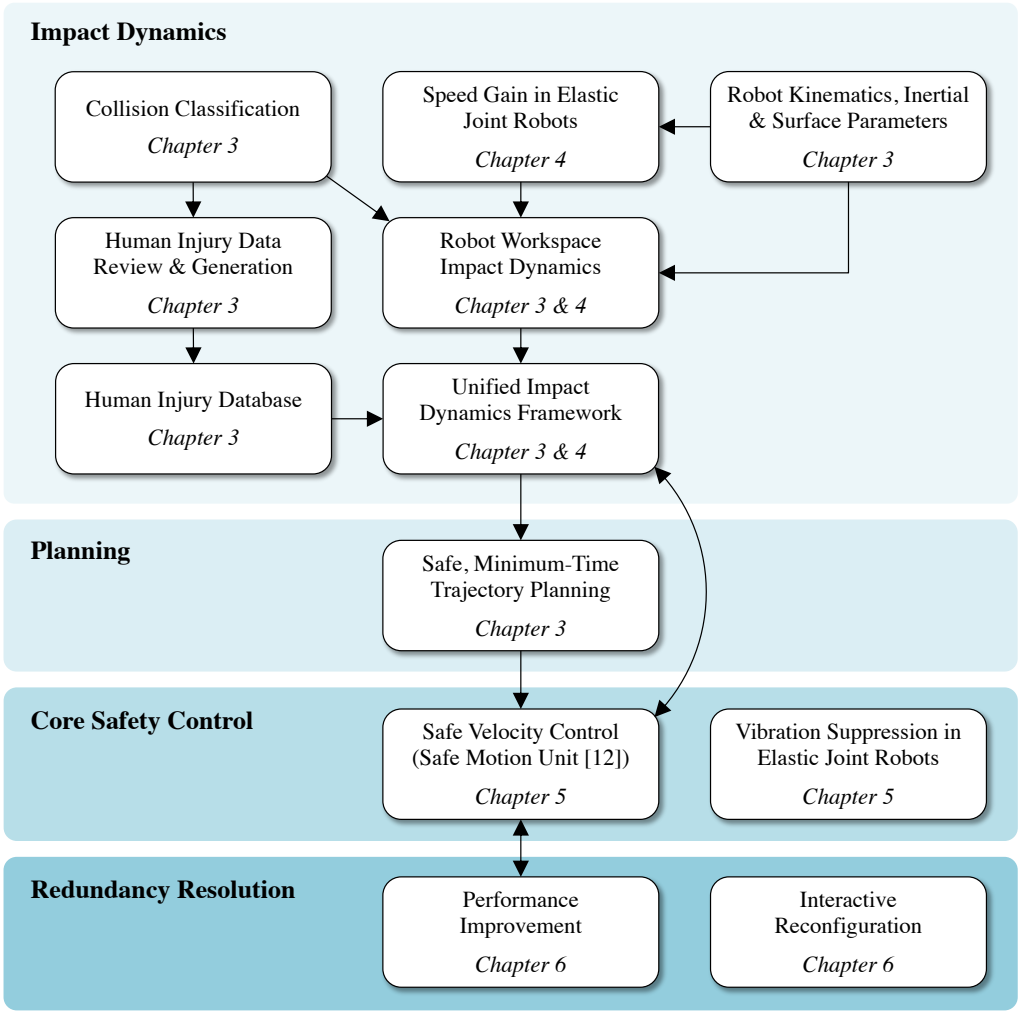


Fig. 1.6: Overview of the objectives of this thesis.

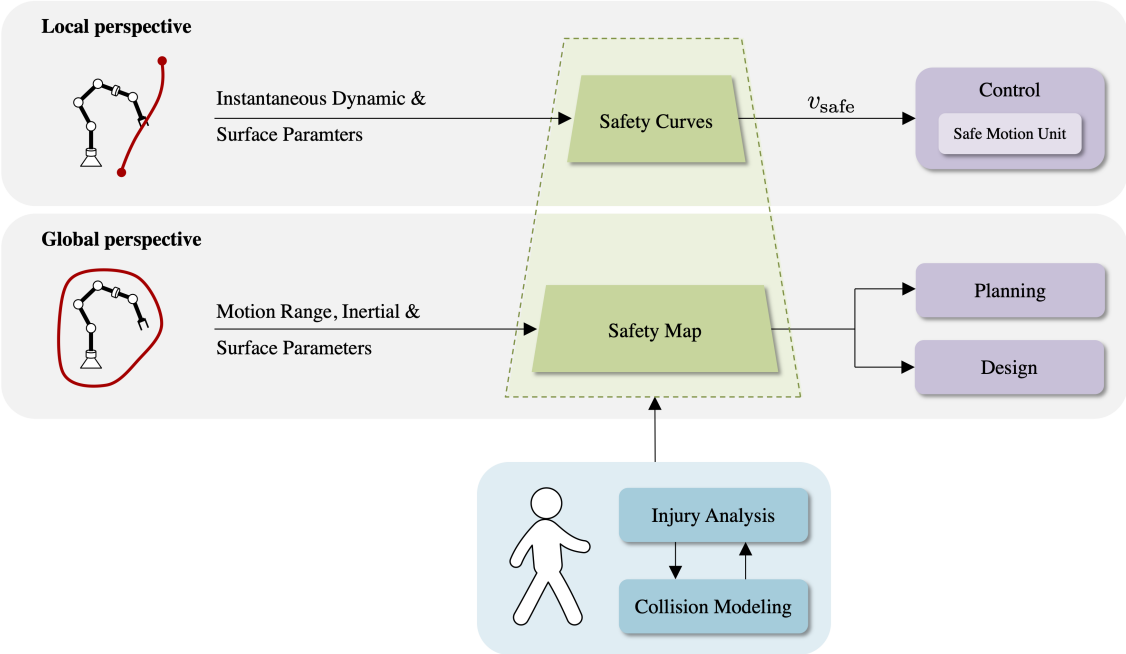


Fig. 1.7: Unified framework for biomechanics-based safety evaluation that can be utilized for safe robot control, planning, and mechanism design.

Impact Dynamics Framework and Safe Motion Planning (Q1 & Q2)

The Safety Map framework is developed, which is a biomechanics-based impact dynamics framework for safe robot hardware design and planning; see Fig. 1.7. In the Safety Map, the achievable robot reflected mass and velocity range for the entire workspace (or task-dependent subspaces) can be quantitatively compared to any available injury data for different contact geometries, collision cases, and human body parts. This gives the robot designer clear information about which kind of injury is most likely to occur during operation, thus guiding the hardware design process and providing valuable information to safe interaction control and motion planning algorithm development. The SMU and Safety Map constitute a hierarchical tool stack (see Fig. 1.7) that is applicable to arbitrary robots, can use all available biomechanics and simulation collision data, and is compatible with standards. In this thesis, the map representations of stationary manipulators (LWR, PUMA, Panda), mobile manipulators (Panda mounted on car-like and differential drive vehicle), and elastic joint robots (David) are derived and compared with each other. Furthermore, it is shown how the Safety Map can be linked not only to certain workspace areas but also to performance metrics; in particular, the robot's reachability map. Initial overviews on human injury data are extended by a thorough summary of the human head and chest. For this, a significant amount of relevant data from over 50 years of biomechanics injury research was classified, validated, processed, and linked to the Safety Map framework. Within the scope of a cooperation with an industrial partner, it is shown how the generation of injury data can be integrated into the risk assessment and reduction cycle for a particular pHRI application. Finally, examples show how the Safety Map can be used to assess the robot safety performance in a particular task and to generate safe, time-optimal trajectories.

Robots with Intrinsic Joint Elasticity: Speed Gain and Safety Characteristics (Q3 & Q4)

The maximum achievable velocity in elastic joint robots is investigated for 1-DOF and n-DOF systems under consideration of the most critical real-world constraints. The peak endpoint velocity is essential for robot design, motion and task planning to assess and optimize the robot's collision safety and performance. All potential system states are determined for a 1-DOF visco-elastic joint with constrained deflection that can be reached from equilibrium. The time-optimal trajectories are derived, and the influence of system parameters on the reachable states is analyzed. The results give clear indications on how to dimension the system parameters in order to perform specific tasks. The analysis is extended to n-DOF systems, where a sufficiently accurate approximation of the Cartesian speed gain in elastic joint robots is developed with minimal computational requirements. The methods are validated using experimental data from previous ball throwing experiments [8]. Furthermore, the theory is applied to robot global safety assessment. The safety characteristics of DLR David and a hypothetically rigid version of this robot are represented in the Safety Map, where both the rigid and the elastic joint manipulator can be quantitatively compared.

Vibration Suppression in Robots with Intrinsic Joint Elasticity (Q5)

New and existing strategies for suppressing vibrations in elastic joint robots are analyzed and compared in terms of performance and passivity. The analysis enables researchers and practitioners to select suitable system design parameters and control schemes to effectively dampen undesired oscillations, e.g., in emergency situations where the robot needs to stop as fast as possible. For 1-DOF visco-elastic joints, the system states that can reach equilibrium without violating constraints are derived as well as the analytic time-optimal braking trajectories. For control of bang-bang type, a modal decoupling-based framework is proposed that decouples both the dynamics and the control inputs. The framework allows applying simple SISO control laws to more complex (linearized) n-DOF systems. A selection of damping control laws is implemented on DLR David, and two safety-related experiments are carried out, namely a dynamic ball impact against the robot structure and an emergency stop during task execution. In the experiments, it is shown that the proposed methods effectively suppress even strong oscillations. Prior to this work, some of the tested controllers had not been validated in practice yet, or only on 1-DOF systems. Finally, impact response metrics are proposed to quantitatively compare the controller performance and assess the robot's impact absorption behavior.

Null Space Control for Safe and Efficient pHRI (Q6 & Q7)

It is investigated how the redundant degrees of freedom of pHRI robots can be exploited to improve collision safety and performance while meeting safety requirements simultaneously. A real-time capable null space controller is proposed, intended to be used in combination with the SMU. The SMU reduces the robot speed during execution to ensure safety if necessary. The controller proposed in this work minimizes the effective robot mass in the direction of motion. This allows to avoid a potential speed reduction of the SMU, which improves the cycle time of the application. Experiments with an LWR show the effectiveness of the approach, which is also extended to an 8-DOF system consisting

of an LWR and a linear axis. When optimizing particular performance or safety criteria via redundancy resolution, the null space scheme may eventually not fulfill the task due to the limited robot dynamics or the negative influence of the primary task. This thesis analyzes how the main and the auxiliary null space tasks can be synchronized temporally to fulfill both tasks at the cost of extra cycle time. For this, the geometric description of the main task is left unchanged, but the trajectory is temporarily slowed down to give the null space schemes more time to meet their objectives. Several time scaling schemes are proposed and validated on an LWR with one redundant degree of freedom. The concept is extended to multiple prioritized tasks and exemplified in simulation. Null space motions are typically non-intuitive and not suitable for interaction. In this thesis, null space interaction behaviors for redundant manipulators are developed, which allow the user to anticipate how the robot will move when it is kinematically reconfigured. The proposed tool for manual, interactive reconfiguration via self-motions can be integrated with little effort and solves real-world problems effectively. Two practical applications for an 8-DOF and 10-DOF mobile manipulator demonstrate the performance and ease of interaction.

1.4 Thesis Structure

This thesis is structured as follows. In Chapter 2, fundamentals on robot kinematics, dynamics, control, and optimization are summarized. Furthermore, the robots used for simulations and experiments are described. The impact dynamics framework Safety Map is introduced, elaborated, and applied to different robots in Chapter 3. In Chapter 4, the maximum achievable speed in elastic joint robots is investigated and verified with the optimal control solution and previous experimental results. The problem of suppressing undesired vibrations in robots with compliant actuation is treated in Chapter 5, where new and existing controllers are described and experimentally compared on DLR David. Redundancy resolution is considered in Chapter 6, where methods to improve safety, performance, and interaction are presented and validated for redundant robots such as the LWR. Finally, Chapter 7 concludes the thesis. Figure 1.8 provides and overview of the thesis structure and the relation to the main publications, which are listed in the next section.

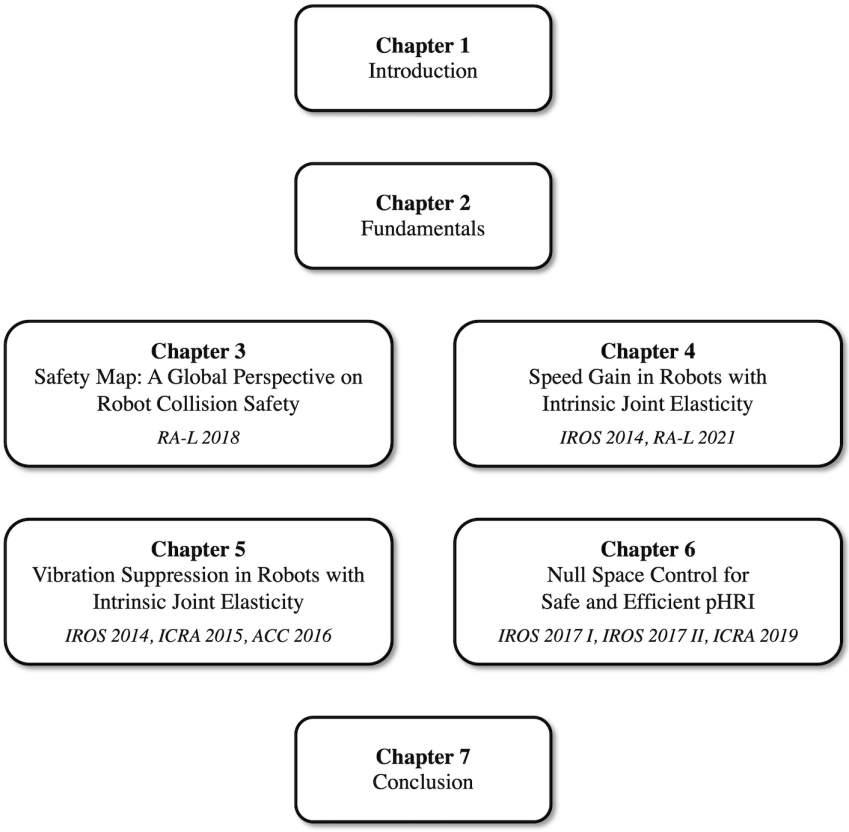


Fig. 1.8: Graphical overview of the chapters and their relation to the main publications.

1.5 Publications

In the following, the publications published or submitted to international, peer-reviewed conferences and journals in the scope of this thesis are listed. The content of the thesis mainly builds on the findings of the first-authored publications.

International Journals

- **N. Mansfeld**, R. Laha, L. F. C. Figueredo, R. J. Kirschner, S. Abdolshah and S. Haddadin, "S*: A Solution to the Safety-Performance Tradeoff in Robot Motion Planning", *to be submitted to The International Journal of Robotics Research (IJRR)*, 2022.

- R. J. Kirschner, H. Mayer, L. Burr, **N. Mansfeld**, S. Abdolshah and S. Haddadin, "Expectable Motion Unit: Avoiding Hazards From Human Involuntary Motions in Human-Robot Interaction", *IEEE Robotics and Automation Letters (RA-L)*, 2021.

- **N. Mansfeld**, M. Keppler and S. Haddadin, "Speed Gain in Elastic Joint Robots: An Energy Conversion-Based Approach", *IEEE Robotics and Automation Letters (RA-L)*, 2021.

- **N. Mansfeld**, M. Hamad, M. Becker, A. Gonzales Marin, and S. Haddadin, "Safety

Map: A Unified Representation for Biomechanics Impact Data and Robot Instantaneous Dynamic Properties", *IEEE Robotics and Automation Letters (RA-L)*, 2018.

International Conferences

- R. J. Kirschner, F. Martineau, **N. Mansfeld**, S. Abdolshah and S. Haddadin, "Manual Maneuverability: Metrics for Analysing and Benchmarking Kinesthetic Robot Guidance", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, Kyoto, Japan.

- **N. Mansfeld**, G. Gómez Peña, M. Hamad, A. A. Kurdas, S. Abdolshah and S. Haddadin, "Global Safety Characteristics of Wheeled Mobile Manipulators", *IEEE International Conference on Automation Science and Engineering (CASE)*, 2022, Mexico City, Mexico.

- A. A. Kurdas, M. Hamad, J. Vorndamme, **N. Mansfeld**, S. Abdolshah and S. Haddadin, "Online Payload Identification for Tactile Robots Using the Momentum Observer", *IEEE International Conference on Robotics and Automation (ICRA)*, 2022, Philadelphia, USA.

- T. Steinecker, A. A. Kurdas, **N. Mansfeld**, M. Hamad, R. J. Kirschner, S. Abdolshah and S. Haddadin, "Mean Reflected Mass: A Physically Interpretable Metric for Safety Assessment and Posture Optimization in Human-Robot Interaction", *IEEE International Conference on Robotics and Automation (ICRA)*, 2022, Philadelphia, USA.

- R. J. Kirschner, A. A. Kurdas, K. Karacan, P. Junge, S. A. B. Birjandi, **N. Mansfeld**, S. Abdolshah and S. Haddadin, "Towards a Reference Framework for Tactile Robot Performance and Safety Benchmarking", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, Prague, Czech Republic.

- **N. Mansfeld**, M. Keppler and S. Haddadin, "Speed Gain in Elastic Joint Robots: An Energy Conversion-Based Approach", *IEEE Robotics and Automation Letters (RA-L), presented at the IEEE International Conference on Robotics and Automation (ICRA)*, 2021, Xi'an, China.

- R. J. Kirschner, J. P. Jantalia, **N. Mansfeld**, S. Abdolshah and S. Haddadin, "CSM: Contact Sensitivity Maps for Benchmarking Robot Collision Handling Systems", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, Xi'an, China.

- R. J. Kirschner, **N. Mansfeld**, G. G. Peña, S. Abdolshah and Sami Haddadin, "Notion on the Correct Use of the Robot Effective Mass in the Safety Context and Comments on ISO/TS 15066", in *IEEE International Conference on Intelligence and Safety for Robotics (ISR)*, 2021, Nagoya, Japan.

- R. J. Kirschner, **N. Mansfeld**, S. Abdolshah and Sami Haddadin, "Experimental Analysis of Impact Forces in Constrained Collisions According to ISO/TS 15066", in *IEEE International Conference on Intelligence and Safety for Robotics (ISR)*, 2021, Nagoya, Japan.

- M. Hamad, **N. Mansfeld**, S. Abdolshah, and S. Haddadin, "The Role of Robot Payload in the Safety Map Framework", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, Macau.
- **N. Mansfeld**, Y. Michel, T. Bruckmann, and S. Haddadin, "Improving the Performance of Auxiliary Null Space Tasks via Time Scaling-Based Relaxation of the Primary Task", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, Montreal, Canada.
- G. Garofalo, **N. Mansfeld**, J. Jankowski, and C. Ott, "Sliding Mode Momentum Observers for Estimation of External Torques and Joint Acceleration", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, Montreal, Canada.
- **N. Mansfeld**, M. Hamad, M. Becker, A. Gonzales Marin, and S. Haddadin, "Safety Map: A Unified Representation for Biomechanics Impact Data and Robot Instantaneous Dynamic Properties", *IEEE Robotics and Automation Letters (RA-L), presented at the IEEE International Conference on Robotics and Automation (ICRA)*, 2018, Brisbane, Australia.
- **N. Mansfeld**, F. Beck, A. Dietrich, and S. Haddadin, "Interactive Null Space Control for Intuitively Interpretable Reconfiguration of Redundant Manipulators", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, Vancouver, Canada.
- **N. Mansfeld**, B. Djellab, J. Raldúa Veuthey, F. Beck, C. Ott, and S. Haddadin, "Improving the Performance of Biomechanically Safe Velocity Control for Redundant Robots through Reflected Mass Minimization", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, Vancouver, Canada.
- **N. Mansfeld**, C. Ott, and S. Haddadin, "Maximal Input Limits for Independent SISO Control in Modal Space Under Consideration of Actuator Constraints", in *American Control Conference (ACC)*, Boston, USA, 2016.
- M. Beetz, G. Bartels, A. Albu-Schäffer, F. Bálint-Benczèdi, R. Belder, D. Beßler, S. Haddadin, A. Maldonado, **N. Mansfeld**, T. Wiedemeyer, R. Weitschat, and J.-H. Worch, "Robotic Agents Capable of Natural and Safe Physical Interaction with Human Co-Workers", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, Hamburg, Germany.
- **N. Mansfeld** and S. Haddadin, "A comparison of braking strategies for elastic joint robots", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, Seattle, USA.
- **N. Mansfeld** and S. Haddadin, "Reaching desired states time-optimally from equilibrium and vice versa for visco-elastic joint robots with limited elastic deflection", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014, Chicago, USA.
- S. Haddadin, R. Weitschat, F. Huber, M. C. Özparpucu, **N. Mansfeld**, and A. Albu-Schäffer, "Optimal Control for Viscoelastic Robots and its Generalization in Real-Time", in *International Symposium on Robotics Research*, Singapore, 2013.

Book Chapters

- S. Wolf, T. Bals, M. Chalon, W. Friedl, M. Grebenstein, H. Höppner, M. Kühne, D. Lakatos, **N. Mansfeld**, M. C. Özparpucu, F. Petit, J. Reinecke, R. Weitschat, and A. Albu-Schäffer, "Soft Robotics with Variable Stiffness Actuators: Tough Robots for Soft Human Robot Interaction", *Soft Robotics: Transferring Theory to Applications*, Springer, 2015.

- S. Haddadin, R. Weitschat, F. Huber, M. C. Özparpucu, **N. Mansfeld**, and A. Albu-Schäffer, "Optimal Control for Viscoelastic Robots and its Generalization on Real-Time", *Springer Tracts on Advanced Robotics Robotics and Research (ISRR)*, Springer, 2016.

Workshops

- **N. Mansfeld**, M. Hamad, M. Becker, A. G. Marin, S. Haddadin, "Safety Map: A Tool for Global Robot Safety Evaluation and Safe Robot Design", *Workshop on Autonomous Robot Design at the IEEE International Conference on Robotics and Automation (ICRA)*, 2018, Brisbane, Australia.

- **N. Mansfeld** and S. Haddadin, "Sicherheit in der Mensch-Roboter-Interaktion", in *8. wissenschaftliche Fachtagung Verarbeitungsmaschinen und Verpackungstechnik*, 2015, Dresden.

- **N. Mansfeld**, M. C. Özparpucu, and S. Haddadin, "Braking elastic joint robots in near-minimum time", *7th International Workshop on Human-Friendly Robotics*, 2014, Pontedera, Italy.

- **N. Mansfeld** and S. Haddadin, "Collision Analysis and Safety Evaluation using a Collision Model for Blunt Robot-Human Impacts", *2013 DGR Days*, 2013, Munich, Germany.

- S. Haddadin and **N. Mansfeld**, "Human Injury Analysis in Robotics: An Overview", in *2013 International Conference on Intelligent Robots and Systems*, 2013, Tokyo, Japan.

Patents

- **N. Mansfeld** and S. Haddadin, "Steuern und/oder Regeln von Motoren eines Roboters", German patent no. 10 2015 106 227.5, filed on April 22, 2015.

CHAPTER 2

Fundamentals

This section briefly summarizes the fundamentals required for the theoretical concepts and practical implementations in this thesis. In Sec. 2.1, robot kinematics and dynamics are reviewed. Basic control concepts are described in Sec. 2.2 and 2.3, basics of optimization in Sec. 2.4. The robots that are used for simulations and the practical validation of control schemes are described in Sec. 2.5.

2.1 Robot Dynamic Modeling

In this section, the kinematics and dynamics of rigid and intrinsically elastic manipulators as well as mobile robots consisting of a platform and a manipulator are briefly described. A comprehensive overview on this topic can be found in [139, 177, 178], for example.

2.1.1 Kinematics

The generalized configuration coordinates associated with the robot joint positions are denoted by $\boldsymbol{q} \in \mathbb{R}^n$. The task coordinates are denoted by $\boldsymbol{x}(\boldsymbol{q}) \in \mathbb{R}^m$. Usually, the robot Cartesian tool center point (TCP) position and orientation constitute $\boldsymbol{x}(\boldsymbol{q})$. If the number of task dimensions equals the number of robot joints, i.e., $m = n$, the robot is considered non-redundant. Typical industrial robots are called non-redundant, as they are equipped with six joints to execute six-DOF position/orientation trajectories[1]. If $m < n$, then the robot is regarded as kinematically redundant. In addition to fulfilling the operational space task, up to $n - m$ additional tasks can be carried out to exploit the robot null space. Such motions are referred to as null space motions, internal motions, or self-motions.

The kinematic mapping from joint space to operational space is given by the (non-linear) forward kinematics $\boldsymbol{x} = \boldsymbol{f}_k(\boldsymbol{q})$. On a differential level, the joint space velocity $\dot{\boldsymbol{q}} \in \mathbb{R}^n$ is

[1] Of course, if the operational space task has less than six DOF, then such a manipulator is redundant w.r.t. this task.

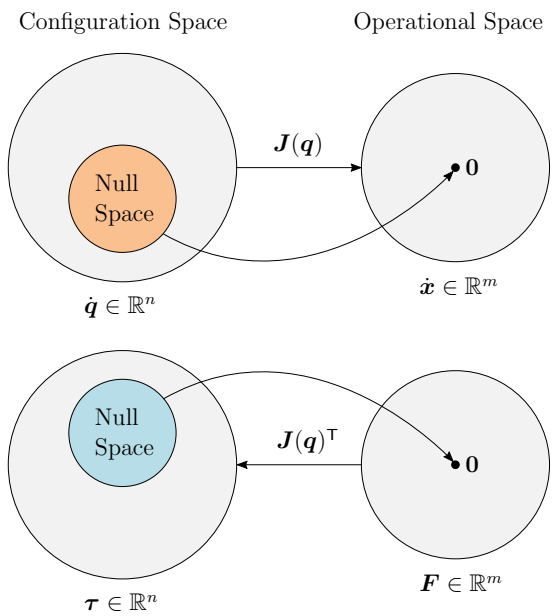


Fig. 2.1: Relationship between the operational space velocity $\dot{\boldsymbol{x}}$ and joint space velocity $\dot{\boldsymbol{q}}$ (upper), and the joint torque $\boldsymbol{\tau}$ and operational space force $\boldsymbol{F}$ (lower) for a redundant manipulator.

related to the operational space velocity $\dot{\boldsymbol{x}} \in \mathbb{R}^m$ via the Jacobian matrix $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{m \times n}$

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}\,, \tag{2.1}$$

$$\boldsymbol{J}(\boldsymbol{q}) = \frac{\partial \boldsymbol{f}_k(\boldsymbol{q})}{\partial \boldsymbol{q}}\,. \tag{2.2}$$

To map operational space velocities to joint velocities that can be commanded to low-level joint controllers, one can either solve the inverse kinematics problem by inverting the forward kinematics $\boldsymbol{x} = \boldsymbol{f}_k(\boldsymbol{q})$ or use the first- or second-order differential kinematics. Solving the direct inverse kinematics problem can be challenging because it is not always possible to derive a closed-form solution and multiple solutions or even infinite solutions (in case of redundancy) may exist. Especially for time-dependent tasks such as trajectory tracking, it is convenient to tackle the inverse differential kinematics problem. If the robot is redundant ($m < n$), then the general solution to (2.1) is

$$\dot{\boldsymbol{q}} = \boldsymbol{J}(\boldsymbol{q})^{\#}\dot{\boldsymbol{x}} + \Big(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^{\#}\boldsymbol{J}(\boldsymbol{q})\Big)\,\dot{\boldsymbol{q}}_0\,, \tag{2.3}$$

where $\boldsymbol{J}(\boldsymbol{q})^{\#}$ is a generalized inverse of the full row-rank Jacobian matrix. It satisfies $\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{\#} = \boldsymbol{I}$, where $\boldsymbol{I}$ is the $n \times n$ identity matrix. The first term in (2.3) represents the joint velocities that realize the desired task velocity; the second term projects the (possibly arbitrary) velocity $\dot{\boldsymbol{q}}_0$ onto the nullspace of the Jacobian matrix. The relationship between the operational space force[2] $\boldsymbol{F} \in \mathbb{R}^m$ at a contact location and the joint torque $\boldsymbol{\tau} \in \mathbb{R}^n$ can be expressed as

$$\boldsymbol{\tau} = \boldsymbol{J}(\boldsymbol{q})^\top \boldsymbol{F}\,. \tag{2.4}$$

[2]Please note that $\boldsymbol{F}$ may contain both forces and torques, e.g., three forces and three torque in case of an operational space wrench $\boldsymbol{F} \in \mathbb{R}^6$.

For redundant manipulators, the general solution to (2.4) is

$$\boldsymbol{\tau} = \boldsymbol{J}(\boldsymbol{q})^\mathsf{T}\boldsymbol{F} + \left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^\mathsf{T}(\boldsymbol{J}(\boldsymbol{q})^\#)^\mathsf{T}\right)\boldsymbol{\tau}_0 \,. \tag{2.5}$$

Here, $\boldsymbol{\tau}_0$ is a joint torque vector which is projected onto the nullspace of $\boldsymbol{J}(\boldsymbol{q})^\mathsf{T}$ via the $n \times n$ nullspace projector $\boldsymbol{N} = \boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^\mathsf{T}(\boldsymbol{J}(\boldsymbol{q})^\#)^\mathsf{T}$. The relationship between the joint space torque/velocity and operational space force/velocity is illustrated in Fig. 2.1.

2.1.2 Rigid Joint Dynamics

The dynamic equations for rigid joint robots are [177]

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_m + \boldsymbol{\tau}_f + \boldsymbol{\tau}_{\text{ext}} \,, \tag{2.6}$$

where the generalized coordinates $\boldsymbol{q} \in \mathbb{R}^n$ are associated with the link positions. $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ is the symmetric and positive definite inertia matrix, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} \in \mathbb{R}^n$ the centripetal and Coriolis vector, and $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$ the gravity vector. The motor torque is denoted by $\boldsymbol{\tau}_m \in \mathbb{R}^n$, the motor friction torque by $\boldsymbol{\tau}_f \in \mathbb{R}^n$, and the external torques by $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^n$. The Coriolis matrix is not unique [177], but it can be selected such that

$$\dot{\boldsymbol{M}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})^\mathsf{T} \,. \tag{2.7}$$

With this choice, the matrix $\dot{\boldsymbol{M}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - 2\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ becomes skew symmetric.

2.1.3 Elastic Joint Dynamics

In robotics research, one typically differentiates so-called flexible joint manipulators such as the DLR/KUKA LWR family [49] from robots with (deliberate) intrinsic joint elasticity such as DLR David [179]. The joint stiffness of the latter is at least an order of magnitude lower than the stiffness of flexible joint robots. The drive train in intrinsically elastic joint robots with constant stiffness is commonly called Series Elastic Actuator (SEA). There also exist systems with stiffness adjusting mechanisms. These are called variable impedance (VIA) or variable stiffness (VSA) robots. An overview of design principles for realizing series elastic and variable impedance actuation is given in [5]. A thorough introduction to the modeling of elastic joint robots can be found in [139, 180]. The dynamics of VSA robots like DLR David are [139]

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}, \boldsymbol{\sigma}) + \boldsymbol{\tau}_{\text{ext}} \,, \tag{2.8a}$$

$$\boldsymbol{B}\ddot{\boldsymbol{\theta}} + \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}, \boldsymbol{\sigma}) = \boldsymbol{\tau}_m - \boldsymbol{\tau}_f \,, \tag{2.8b}$$

where $\boldsymbol{\theta} \in \mathbb{R}^n$ and $\boldsymbol{q} \in \mathbb{R}^n$ represent the motor and link positions; see Fig. 4.1. The motor and link inertia matrix are denoted by $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$, the Coriolis matrix by $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$, and the gravity vector by $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$. The motor torque is denoted by $\boldsymbol{\tau}_m \in \mathbb{R}^n$, the motor friction torque by $\boldsymbol{\tau}_f \in \mathbb{R}^n$, and the possibly non-linear elastic joint torque by $\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}, \boldsymbol{\sigma}) \in \mathbb{R}^n$. The latter depends on the elastic deflection $\boldsymbol{\varphi} = \boldsymbol{\theta} - \boldsymbol{q}$ and the position of the stiffness adjusting mechanism $\boldsymbol{\sigma} \in \mathbb{R}^n$, which may also have underlying (second-order) dynamics. It is assumed that there are no stiffness couplings and no inertial couplings between the link-side and the motor-side dynamics, which means that they are only coupled via the elastic joint torque $\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}, \boldsymbol{\sigma})$ [139]. The $n \times n$ joint stiffness matrix is denoted by $\boldsymbol{K}_J = -\frac{\partial \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}, \boldsymbol{\sigma})}{\partial \boldsymbol{q}}$.

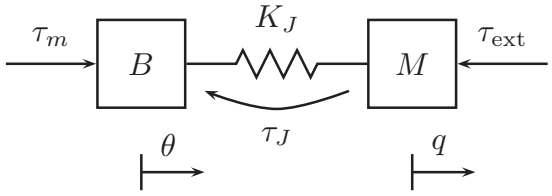


Fig. 2.2: 1-DOF elastic joint.

Singular Perturbation Model

Typically, the motor dynamics in intrinsically elastic manipulators are much faster than the link side dynamics[3]. Assuming that the motors can always provide enough torque to accelerate the motor shaft and compensate the elastic joint torque, one can model the motors as velocity sources, meaning the desired velocity can be reached instantaneously [8]. To verify this assumption, the dynamics are brought into singular perturbation form. Assume that the motor velocities are PI-controlled

$$\boldsymbol{\tau}_m = \boldsymbol{K}_P(\dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}}) + \boldsymbol{K}_I(\boldsymbol{\theta}_d - \boldsymbol{\theta})\,, \tag{2.9}$$

where $\boldsymbol{K}_P = \text{diag}\{k_{P,i}\} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{K}_I = \text{diag}\{k_{I,i}\} \in \mathbb{R}^{n\times n}, i = 1,\ldots,n$ are the controller gains. The proportional gain is typically very high, thus $\boldsymbol{\epsilon} = \boldsymbol{K}_P^{-1}$ is chosen. When inserting (2.8b) into (2.9) one gets

$$\boldsymbol{\epsilon}\,(\boldsymbol{B}\ddot{\boldsymbol{\theta}} + \boldsymbol{\tau}_J) = \dot{\boldsymbol{\theta}}_d - \dot{\boldsymbol{\theta}} + \boldsymbol{\epsilon}\,\boldsymbol{K}_I(\boldsymbol{\theta}_d - \boldsymbol{\theta})\,. \tag{2.10}$$

Setting $\boldsymbol{\epsilon} \to 0$ yields $\dot{\boldsymbol{\theta}}_d = \dot{\boldsymbol{\theta}}$. The reduced dynamics now become

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_J(\boldsymbol{\theta},\boldsymbol{q},\boldsymbol{\sigma}) + \boldsymbol{\tau}_{\text{ext}}\,, \tag{2.11a}$$

$$\boldsymbol{\theta} = \int \dot{\boldsymbol{\theta}}\;dt + \boldsymbol{\theta}_0\,. \tag{2.11b}$$

Please note that the singular perturbation model only affects the motor equations while the link dynamics remain unchanged.

2.1.4 Operational Space Dynamics

As a robot task is typically described in operational space, the joint space dynamics are not suitable to describe the dynamic end-effector behavior. For control and analysis of the end-effector task, the operational space dynamics can be expressed as [181]

$$\boldsymbol{\Lambda}(\boldsymbol{q})\ddot{\boldsymbol{x}} + \boldsymbol{\mu}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{p}(\boldsymbol{q}) = \boldsymbol{F}\,. \tag{2.12}$$

Here, $\boldsymbol{\Lambda}(\boldsymbol{q}) \in \mathbb{R}^{m\times m}$ is the operational mass matrix, $\boldsymbol{\mu}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^m$ the Coriolis and centrifugal terms, $\boldsymbol{p}(\boldsymbol{q}) \in \mathbb{R}^m$ the gravity vector, and $\boldsymbol{F} \in \mathbb{R}^m$ the wrench at the end-effector. The dynamic matrices can be obtained by successively inserting (2.1) and (2.4) into the

[3]This is of course not a valid assumption for typical flexible joint arms such as the LWR or Franka Emika Panda because the joint stiffness range is at least an order of magnitude larger.

joint space dynamics (5.54). For non-redundant manipulators, they are

$$\boldsymbol{\Lambda}(\boldsymbol{q}) = \boldsymbol{J}(\boldsymbol{q})^{-\mathsf{T}}\boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{-1}\,, \tag{2.13a}$$

$$\boldsymbol{\mu}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{J}(\boldsymbol{q})^{-\mathsf{T}}\left[\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{-1}\dot{\boldsymbol{J}}(\boldsymbol{q})\dot{\boldsymbol{q}}\right]\,, \tag{2.13b}$$

$$\boldsymbol{p}(\boldsymbol{q}) = \boldsymbol{J}(\boldsymbol{q})^{-\mathsf{T}}\boldsymbol{g}(\boldsymbol{q})\,. \tag{2.13c}$$

The achievable Cartesian stiffness of elastic joint robots was investigated in [182]. Assuming the joint stiffness matrix $\boldsymbol{K}_J$ is diagonal and no coupling terms exist, the Cartesian stiffness matrix is given by

$$\boldsymbol{K}_x = \left(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{K}_J^{-1}\boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\right)^{-1}\,. \tag{2.14}$$

Effective Mass and Inertia

The robot's inertial properties at the end-effector are described by the operational space mass matrix $\boldsymbol{\Lambda}(\boldsymbol{q})$. Typical end-effector tasks include translational or rotational motions along or about a Cartesian direction $\boldsymbol{u} \in \mathbb{R}^3$. For such tasks, the robot inertial properties can be expressed in terms of the so-called *effective mass/inertia*. Let us partition the Jacobian matrix $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{6\times n}$ at the end-effector as

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix}\boldsymbol{J}_\nu(\boldsymbol{q})\\ \boldsymbol{J}_\omega(\boldsymbol{q})\end{bmatrix}\,, \tag{2.15}$$

where $\boldsymbol{J}_\nu(\boldsymbol{q}) \in \mathbb{R}^{3\times n}$ is associated with translational and $\boldsymbol{J}_\omega(\boldsymbol{q}) \in \mathbb{R}^{3\times n}$ with angular motions. The operational space kinetic energy matrix for both non-redundant and redundant robots can be expressed as

$$\boldsymbol{\Lambda}(\boldsymbol{q}) = \left(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}^{\mathsf{T}}(\boldsymbol{q})\right)^{-1}\,. \tag{2.16}$$

The inverse of this matrix can be decomposed into

$$\boldsymbol{\Lambda}(\boldsymbol{q})^{-1} = \begin{bmatrix}\boldsymbol{\Lambda}_\nu(\boldsymbol{q})^{-1} & \overline{\boldsymbol{\Lambda}}_{v\omega}(\boldsymbol{q})\\ \overline{\boldsymbol{\Lambda}}_{v\omega}(\boldsymbol{q})^{\mathsf{T}} & \boldsymbol{\Lambda}_\omega(\boldsymbol{q})^{-1}\end{bmatrix}\,, \tag{2.17}$$

with

$$\boldsymbol{\Lambda}_\nu(\boldsymbol{q})^{-1} = \boldsymbol{J}_\nu(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}_\nu(\boldsymbol{q})^{\mathsf{T}}\,, \tag{2.18a}$$

$$\boldsymbol{\Lambda}_\omega(\boldsymbol{q})^{-1} = \boldsymbol{J}_\omega(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}_\omega(\boldsymbol{q})^{\mathsf{T}}\,, \tag{2.18b}$$

$$\overline{\boldsymbol{\Lambda}}_{v\omega}(\boldsymbol{q}) = \boldsymbol{J}_\nu(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}_\omega(\boldsymbol{q})^{\mathsf{T}}\,. \tag{2.18c}$$

The matrix $\boldsymbol{\Lambda}_\nu(\boldsymbol{q})^{-1}$ describes the translational response of the end-effector to a Cartesian force, the matrix $\boldsymbol{\Lambda}_\omega(\boldsymbol{q})^{-1}$ the rotational response to a moment, and the matrix $\overline{\boldsymbol{\Lambda}}_{v\omega}(\boldsymbol{q})$ the coupling of translational and rotational motions. The scalar mass perceived at the end-effector given a force in the Cartesian unit direction $\boldsymbol{u}$ is now given by

$$m_u(\boldsymbol{q}) = \left(\boldsymbol{u}^{\mathsf{T}}\boldsymbol{\Lambda}_\nu(\boldsymbol{q})^{-1}\boldsymbol{u}\right)^{-1}\,. \tag{2.19}$$

This quantity is referred to as the effective/reflected mass along direction $\boldsymbol{u}$. The effective inertia perceived at the end-effector given a moment about unit direction $\boldsymbol{u}$ is

$$I_u(\boldsymbol{q}) = \left(\boldsymbol{u}^\mathsf{T}\boldsymbol{\Lambda}_\omega(\boldsymbol{q})^{-1}\boldsymbol{u}\right)^{-1} . \tag{2.20}$$

The reflected mass and inertia in/about all Cartesian directions can be represented by so-called belted ellipsoids [181]

$$\frac{\boldsymbol{v}^\mathsf{T}\boldsymbol{\Lambda}_\nu(\boldsymbol{q})^{-1}\boldsymbol{v}}{\sqrt{\boldsymbol{v}^\mathsf{T}\boldsymbol{v}}} = 1 , \tag{2.21}$$

$$\frac{\boldsymbol{v}^\mathsf{T}\boldsymbol{\Lambda}_\omega(\boldsymbol{q})^{-1}\boldsymbol{v}}{\sqrt{\boldsymbol{v}^\mathsf{T}\boldsymbol{v}}} = 1 . \tag{2.22}$$

2.1.5 Mobile Manipulator Dynamics

The dynamics of a mobile robot consisting of a platform and a manipulator can be expressed as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{E}(\boldsymbol{q})\boldsymbol{\tau} - \boldsymbol{A}(\boldsymbol{q})^\mathsf{T}\boldsymbol{\lambda} , \tag{2.23}$$

where $\boldsymbol{q} \in \mathbb{R}^n, n = n_v + n_m$ are the generalized coordinates that are composed by the vehicle coordinates $\boldsymbol{q}_v \in \mathbb{R}^{n_v}$ and the manipulator coordinates $\boldsymbol{q}_m \in \mathbb{R}^{n_m}$

$$\boldsymbol{q} = \begin{bmatrix} \boldsymbol{q}_v \\ \boldsymbol{q}_m \end{bmatrix} . \tag{2.24}$$

The symmetric, positive definite mass matrix of the coupled system is denoted by $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$, the Coriolis matrix by $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$, and the gravity torque by $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$. The input transformation matrix is denoted by $\boldsymbol{E}(\boldsymbol{q}) \in \mathbb{R}^{n\times r}$, the input torque vector by $\boldsymbol{\tau} \in \mathbb{R}^n$. The system is subject to m constraints. The matrix associated with the constraints is denoted by $\boldsymbol{A}(\boldsymbol{q}) \in \mathbb{R}^{m\times n}$ and $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the vector of constraint forces. The vehicle's so-called *steering system* is given by

$$\dot{\boldsymbol{q}}_v = \boldsymbol{S}(\boldsymbol{q}_v)^{\mathbb{S}}\dot{\boldsymbol{q}}_v , \tag{2.25}$$

where ${}^{\mathbb{S}}\dot{\boldsymbol{q}}_v$ is the input velocity [183,184]. For a differential drive vehicle, the input is given by the velocities of the two wheels, for example. The mapping from the input velocity to the differential coordinates $\dot{\boldsymbol{q}}_{\boldsymbol{v}}$ is given by the constraint auxiliary matrix $\boldsymbol{S}(\boldsymbol{q}_v)$, which includes the vehicle's motion constraints (e.g., on the wheels) and fulfills $\boldsymbol{S}(\boldsymbol{q})^\mathsf{T}\boldsymbol{A}(\boldsymbol{q})^\mathsf{T} = \boldsymbol{0}$. The superscript $\mathbb{S}$ indicates the constraint-free space. The velocity vector of the full system is given by

$$\dot{\boldsymbol{q}} = \boldsymbol{S}_I(\boldsymbol{q})^{\mathbb{S}}\dot{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{S}(\boldsymbol{q}_v) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} {}^{\mathbb{S}}\dot{\boldsymbol{q}}_v \\ \dot{\boldsymbol{q}}_m \end{bmatrix} , \tag{2.26}$$

where $\boldsymbol{S}_I(\boldsymbol{q})$ denotes the extended constraint auxiliary matrix. The constraint-free velocities are mapped to operational space via

$$\dot{\boldsymbol{x}} =^{\mathbb{S}} \boldsymbol{J}(\boldsymbol{q})^{\mathbb{S}}\dot{\boldsymbol{q}} . \tag{2.27}$$

The Jacobian matrix is given by

$${}^{\mathbb{S}}\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_v(\boldsymbol{q})\boldsymbol{S}(\boldsymbol{q}_v) & \boldsymbol{J}_m(\boldsymbol{q}) \end{bmatrix} , \tag{2.28}$$

where $\boldsymbol{J}_v(\boldsymbol{q}) \in \mathbb{R}^{3\times n_v}$ and $\boldsymbol{J}_m(\boldsymbol{q}) \in \mathbb{R}^{3\times n_m}$ are the Jacobian matrices associated with the vehicle and the manipulator, respectively. The dynamics of the coupled system can be derived with the macro/mini structure approach proposed in [181]. In constraint-free space, the dynamics can be expressed by

$$ {}^{\mathbb{S}}\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + {}^{\mathbb{S}}\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}, {}^{\mathbb{S}}\dot{\boldsymbol{q}}_v) + {}^{\mathbb{S}}\boldsymbol{g}(\boldsymbol{q}) = {}^{\mathbb{S}}\boldsymbol{E}(\boldsymbol{q})\boldsymbol{\tau}, \tag{2.29} $$

where

$$ {}^{\mathbb{S}}\boldsymbol{M}(\boldsymbol{q}) = \boldsymbol{S}_I(\boldsymbol{q}_v)^{\mathsf{T}}\boldsymbol{M}(\boldsymbol{q})\boldsymbol{S}_I(\boldsymbol{q}_v)\,, \tag{2.30} $$

$$ {}^{\mathbb{S}}\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}, {}^{\mathbb{S}}\dot{\boldsymbol{q}}_v) = \boldsymbol{S}_I(\boldsymbol{q}_v)^{\mathsf{T}}\left(\boldsymbol{M}(\boldsymbol{q})\begin{bmatrix}\dot{\boldsymbol{S}}_I(\boldsymbol{q}_v)^{\mathbb{S}}\dot{\boldsymbol{q}}_v \\ \boldsymbol{0}\end{bmatrix} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}\right)\,, \tag{2.31} $$

$$ {}^{\mathbb{S}}\boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{g}(\boldsymbol{q})\,, \tag{2.32} $$

$$ {}^{\mathbb{S}}\boldsymbol{E}(\boldsymbol{q}) = \boldsymbol{S}_I(\boldsymbol{q}_v)^{\mathsf{T}}\boldsymbol{E}(\boldsymbol{q})\,. \tag{2.33} $$

The 6×6 Cartesian mass matrix inverse can be obtained from

$$ {}^{\mathbb{S}}\boldsymbol{\Lambda}(\boldsymbol{q})^{-1} = {}^{\mathbb{S}}\boldsymbol{J}(\boldsymbol{q})\,{}^{\mathbb{S}}\boldsymbol{M}(\boldsymbol{q})^{-1}\,{}^{\mathbb{S}}\boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\,. \tag{2.34} $$

The reflected mass in Cartesian unit direction $\boldsymbol{u} \in \mathbb{R}^3$ and the inertia about $\boldsymbol{u}$ that take the vehicle's motion constraints into account are

$$ m_u = \left(\boldsymbol{u}^{\mathsf{T}}\,{}^{\mathbb{S}}\boldsymbol{\Lambda}_\nu(\boldsymbol{q})^{-1}\,\boldsymbol{u}\right)^{-1}\,, \tag{2.35} $$

$$ I_u(\boldsymbol{q}) = \left(\boldsymbol{u}^{\mathsf{T}}{}^{\mathbb{S}}\boldsymbol{\Lambda}_\omega(\boldsymbol{q})^{-1}\boldsymbol{u}\right)^{-1}\,, \tag{2.36} $$

where ${}^{\mathbb{S}}\boldsymbol{\Lambda}_\nu(\boldsymbol{q})^{-1}$ is the upper 3×3 translational and ${}^{\mathbb{S}}\boldsymbol{\Lambda}_\omega(\boldsymbol{q})^{-1}$ the lower 3×3 rotational part of ${}^{\mathbb{S}}\boldsymbol{\Lambda}(\boldsymbol{q})^{-1}$, respectively.

2.2 Soft Robot Control

Whenever a robot acts in partially unknown dynamic environments or the vicinity of humans, it should react to contact in a compliant manner. Compliance can either be realized via soft mechanism design or control. Elastic joint robots realize compliance via passive mechanical elements, while for rigid joint robots, compliance can be achieved via soft control schemes. In the following, impedance and admittance control are briefly described. These control concepts are widely used in lightweight and tactile robots nowadays.

2.2.1 Impedance Control

As perceived from the environment, physical systems appear either as *admittances* which accept effort input (force) and yield flow output (motion) or as *impedances* which accept flow input (motion) and yield effort output (force) [3]. Two physical systems complement each other constantly if one system is an admittance, the other must be an impedance and vice versa [3]. A passive environment of a manipulator can be physically described as an admittance. In order to properly define the robot's contact behavior, impedance control

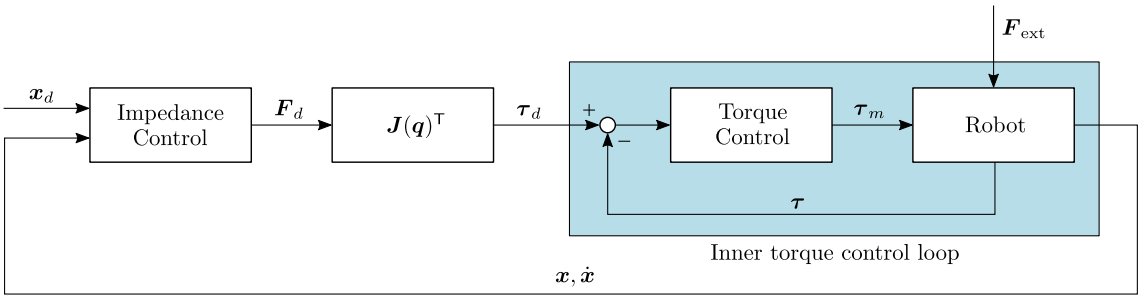


Fig. 2.3: Operational space impedance control [188].

is frequently used, which alters the mechanical impedance of the system. The classical impedance control law proposed in [3] is

$$\boldsymbol{\tau}_d = -\boldsymbol{J}(\boldsymbol{q})^\mathsf{T} \left(\boldsymbol{K}_x \tilde{\boldsymbol{x}}(\boldsymbol{q}) + \boldsymbol{D}_x \dot{\boldsymbol{x}}\right) + \boldsymbol{g}(\boldsymbol{q}) \,, \tag{2.37a}$$

$$\tilde{\boldsymbol{x}}(\boldsymbol{q}) = \boldsymbol{x}(\boldsymbol{q}) - \boldsymbol{x}_d \,, \tag{2.37b}$$

where $\boldsymbol{K}_x \in \mathbb{R}^{m\times m}$ is the desired Cartesian diagonal positive definite stiffness matrix, $\boldsymbol{D}_x \in \mathbb{R}^{m\times m}$ the desired diagonal positive definite damping matrix, $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$ the gravity compensation term, and $\boldsymbol{x}_d \in \mathbb{R}^m$ the desired end-effector position in Cartesian space. In addition to desired stiffness and damping, it is also possible to include a desired inertial behavior $\boldsymbol{\Lambda}_x(\boldsymbol{q}) \in \mathbb{R}^{m\times m}$ into the control law. However, this would require feedback from the external wrench. For the sake of robustness and ease of implementation, this term is usually dropped, which means that the robot's natural inertia is preserved. Due to the lack of an integrator, the impedance control law is of PD type. A steady-state error $\tilde{\boldsymbol{x}}(\boldsymbol{q})$ is therefore inevitable when external forces or model uncertainties are present. Controller (2.37) is passive[4] w.r.t. input-output pair $\{\boldsymbol{\tau}_J + \boldsymbol{\tau}_{\text{ext}}, \dot{\boldsymbol{q}}\}$ for rigid joint manipulators. For flexible joint robots, however, it lacks this property for the input-output pair $\{\dot{\boldsymbol{q}}, -\boldsymbol{\tau}_J\}$. To ensure stability, it was proposed in [4, 186, 187] to replace $\boldsymbol{q}$ in (2.37) by the static equilibrium position of $\boldsymbol{q}$, denoted by $\bar{\boldsymbol{q}}(\boldsymbol{\theta})$, which depends only on the motor position. In order to implement the impedance control law (2.37), a low-level motor torque controller is required; see Fig. 2.3. In the LWR, which is used for most experiments in this thesis, a full state feedback controller [4] is used, which alters the motor kinetic energy by shaping the motor inertia to a lower value while preserving the overall structure of the motor dynamics.

2.2.2 Admittance Control

An admittance is the inversion of an impedance. It accepts effort input (force) and yields flow output (motion). In admittance control, an external wrench $\boldsymbol{F}_{\text{ext}}$ measured by a sensor is mapped to the desired position or velocity. For example, this mapping can be realized via a second-order system with desired mass/inertia and damping. If the output of the admittance controller is a Cartesian position $\boldsymbol{x}_d \in \mathbb{R}^m$ with $m < n$, then the redundancy must be resolved with an inverse kinematics scheme to obtain the desired joint configuration $\boldsymbol{q}_d$; see Fig. 2.4. A stiff inner position control loop then regulates this

[4]Passivity is an input-output property of a physical system defined in terms of energy dissipation and conversion [185]. A system is stable if a bounded input energy yields a bounded output energy.

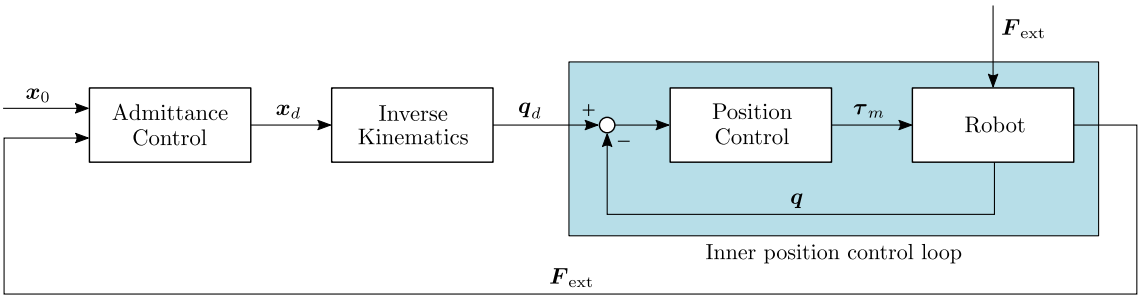


Fig. 2.4: Operational space admittance control with $\boldsymbol{x} \in \mathbb{R}^m$ and $m < n$ [188].

position. In contrast to impedance control, the system will only react to external contact if the sensor measures an external force. If the external torque is measured or estimated in each joint [102], then a force exerted at any link induces a robot motion. However, external forces at the end-effector will only induce a robot movement if a force sensor is placed at the flange. The stiff position control will keep the robot configuration if the user pushes the robot at another link. Admittance control is often used in systems that do not provide a low-level joint torque interface [189], e.g., in mobile platforms. It enables using such systems in combination with impedance-controlled robots. In DLR Justin, e.g., the velocity-controlled mobile platform and the torque-controlled upper body were combined in a whole-body control framework [190]. Some restrictions need to be considered in admittance control, e.g., low stiffness in the inner position control loop can destabilize the system [189]; also the dynamics between the sensor and actuator may cause contact instability [191].

2.3 Redundancy Resolution

In this section, possible schemes to resolve a robot's redundant degrees of freedom are described. First, the properties of the generalized weighted Jacobian matrix inverse and nullspace projections are reviewed. Then, possible performance criteria are described and how they can be optimized locally. Finally, hierarchical approaches to redundancy resolution are summarized that can handle a stack of nullspace tasks.

2.3.1 Weighted Pseudoinverse and Nullspace Projections

The general solution to the differential inverse kinematics problem (2.3) can be rewritten as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\dot{\boldsymbol{x}} + \underbrace{\left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\boldsymbol{J}(\boldsymbol{q})\right)}_{\boldsymbol{N}_{\text{vel}}(\boldsymbol{q})}\dot{\boldsymbol{q}}_0 \,. \tag{2.38}$$

Here, the generalized inverse $\boldsymbol{J}(\boldsymbol{q})^{\#}$ of the (full row rank) Jacobian matrix that fulfills

$$\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{\#} = \boldsymbol{I} \,, \tag{2.39}$$

was replaced by the weighted generalized inverse

$$\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+} = \boldsymbol{W}^{-1}\boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\left(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{W}^{-1}\boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}\right)^{-1} \,. \tag{2.40}$$

This matrix satisfies $\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+} = \boldsymbol{I}$ and (2.39) if the inversion of the right term in (2.38) is feasible. The non-singular, invertible weighting matrix is denoted by $\boldsymbol{W} \in \mathbb{R}^{n\times n}$. It can be shown that the weighted pseudoinverse solves the optimization problem [192]

$$\min_{\dot{\boldsymbol{q}}} \frac{1}{2}\dot{\boldsymbol{q}}^\top \boldsymbol{W}\dot{\boldsymbol{q}}\,, \tag{2.41a}$$

$$\text{s.t.} \quad \dot{\boldsymbol{x}} - \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0}\,. \tag{2.41b}$$

When selecting $\boldsymbol{W} = \boldsymbol{I}$, the generalized inverse is given by the Moore-Penrose pseudoinverse ($\boldsymbol{J}(\boldsymbol{q})^\dagger := \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{I}+} = \boldsymbol{J}(\boldsymbol{q})^\top(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^\top)^{-1}$), which minimizes the Euclidean norm $\dot{\boldsymbol{q}}^\top\boldsymbol{q}$ of joint velocities. The choice $\boldsymbol{W} = \boldsymbol{M}(\boldsymbol{q})$ minimizes the kinetic energy $\frac{1}{2}\dot{\boldsymbol{q}}^\top\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}$, the corresponding inverse is called the dynamically consistent pseudoinverse. In the second term in (2.38) the $n \times n$ matrix

$$\boldsymbol{N}_{\text{vel}}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\boldsymbol{J}(\boldsymbol{q})\,, \tag{2.42}$$

projects the (possibly arbitrary) joint velocity $\dot{\boldsymbol{q}}_0$ onto the nullspace of $\boldsymbol{J}(\boldsymbol{q})$. This projection yields zero operational space velocity because $\boldsymbol{J}(\boldsymbol{q})\boldsymbol{N}_{\text{vel}}(\boldsymbol{q})\dot{\boldsymbol{q}}_0 = \boldsymbol{0}$. For joint torque control, (2.5) can be rewritten as

$$\boldsymbol{\tau} = \boldsymbol{J}(\boldsymbol{q})^\top\boldsymbol{F} + \underbrace{\left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^\top(\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+})^\top\right)}_{\boldsymbol{N}_{\text{tor}}(\boldsymbol{q})}\boldsymbol{\tau}_0\,, \tag{2.43}$$

where the nullspace projection matrix is denoted by $\boldsymbol{N}_{\text{tor}}(\boldsymbol{q})$. In the thesis, mainly torque control is considered. For the sake of brevity, $\boldsymbol{N}_{\text{tor}}(\boldsymbol{q})$ is replaced by $\boldsymbol{N}(\boldsymbol{q})$ in the following. It is common to select the mass matrix as the weighting matrix ($\boldsymbol{W} = \boldsymbol{M}(\boldsymbol{q})$) to make the nullspace projection dynamically consistent. The *static consistency* and *dynamic consistency* property of nullspace projections were defined in [181]. A nullspace projector is statically consistent if it does not generate interfering forces in the higher priority operational space task in static equilibrium. The condition for static consistency is

$$(\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+})^\top\boldsymbol{N}(\boldsymbol{q}) = \boldsymbol{0}\,, \tag{2.44}$$

when $\dot{\boldsymbol{q}} = \boldsymbol{0}, \ddot{\boldsymbol{q}} = \boldsymbol{0}$. It can be shown that static consistency is achieved for every nullspace projector, independent of the weighting matrix [193]. Dynamically consistent projections also do not generate accelerations in the operational space task. The requirement for dynamic consistency is [181]

$$\boldsymbol{J}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{N}(\boldsymbol{q}) = \boldsymbol{0}\,. \tag{2.45}$$

Next, common choices for the nullspace torque $\boldsymbol{\tau}_0$ are described.

2.3.2 Local Optimization

A classical approach to redundancy resolution is to locally optimize a scalar, configuration-dependent performance criterion $H(\boldsymbol{q})$. Here, the joint torque $\boldsymbol{\tau}_0$ (cf. (2.5), similar procedure for velocity control) is selected in direction of the antigradient of $H(\boldsymbol{q})$

$$\boldsymbol{\tau}_0 = -K_H\nabla H(\boldsymbol{q})\,, \tag{2.46}$$

where K_H is the scalar step size and $\nabla H(\boldsymbol{q})$ is the gradient of $H(\boldsymbol{q})$ at the joint configuration $\boldsymbol{q}$. The total desired robot joint torque is

$$\boldsymbol{\tau}_d = \boldsymbol{\tau}_{\text{prim}} - K_H \left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^\mathsf{T} (\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+})^\mathsf{T} \right) \nabla H(\boldsymbol{q}) \,, \tag{2.47}$$

where $\boldsymbol{\tau}_{\text{prim}} \in \mathbb{R}^n$ is the primary controller torque, e.g., the output of the impedance controller (2.37). Common choices for the performance criterion are the manipulability measure [194]

$$H(\boldsymbol{q}) = \sqrt{\det(\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^\mathsf{T})} \,, \tag{2.48}$$

for avoiding singular configurations, or the distance from mechanical joint limits

$$H(\boldsymbol{q}) = -\frac{1}{2n} \sum_{i=1}^{n} \left(\frac{q_i - \bar{q}_i}{q_{i,M} - q_{i,m}} \right)^2 \,, \tag{2.49}$$

where $q_{i,M}$ and $q_{i,m}$ denote the maximum and minimum joint limit and $\bar{q}_i$ the middle value of the joint position range of joint $i = 1, \ldots, n$.

2.3.3 Task Hierarchy

When considering multiple, i.e., up to $r < n - m$ nullspace objectives that shall be accomplished, it is possible to formulate them either as constraints or arrange them in a task hierarchy. In such a hierarchy, each individual task shall not affect the higher priority tasks but act in the respective null space. There exist two common task hierarchy schemes, namely 1) successive projections and 2) augmented projections. Both approaches are briefly described in the following. Afterwards, the extended Jacobian approach is summarized.

Successive Projections

Let us assign the index $i = 1$ and torque $\boldsymbol{\tau}_1 \in \mathbb{R}^n$ to the main task. In the successive nullspace projection approach [195], the torque $\boldsymbol{\tau}_2 \in \mathbb{R}^n$ of the task on the second (nullspace) level is projected onto the nullspace of the primary task via

$$\boldsymbol{\tau}_2^p = \boldsymbol{N}_2^{\text{suc}}(\boldsymbol{q}) \boldsymbol{\tau}_2 \,, \tag{2.50}$$

where $\boldsymbol{\tau}_2^p$ is the projected nullspace torque and $\boldsymbol{N}_2^{\text{suc}}(\boldsymbol{q})$ is the nullspace projector, which is given by

$$\boldsymbol{N}_2^{\text{suc}}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}_1(\boldsymbol{q})^\mathsf{T} (\boldsymbol{J}_1(\boldsymbol{q})^{\boldsymbol{W}+})^\mathsf{T} \,. \tag{2.51}$$

For the remaining $2 < i < r$ tasks the joint torque is obtained by

$$\boldsymbol{\tau}_i^p = \boldsymbol{N}_i^{\text{suc}}(\boldsymbol{q}) \boldsymbol{\tau}_i \,. \tag{2.52}$$

The successive nullspace projectors are given by the recursive algorithm

$$\boldsymbol{N}_i^{\text{suc}}(\boldsymbol{q}) = \boldsymbol{N}_{i-1}^{\text{suc}}(\boldsymbol{q}) \left(\boldsymbol{I} - \boldsymbol{J}_{i-1}(\boldsymbol{q})^\mathsf{T} (\boldsymbol{J}_{i-1}(\boldsymbol{q})^{\boldsymbol{W}+})^\mathsf{T} \right) \,. \tag{2.53}$$

The total torque that is commanded to the robot is now given by the sum of the primary and all nullspace torques

$$\boldsymbol{\tau} = \boldsymbol{\tau}_1 + \sum_{i=1}^{r} \boldsymbol{\tau}_i^p \,. \tag{2.54}$$

Augmented Projections

In the augmented projection approach introduced in [196], the nullspace projection on the second level ((2.50) – (2.51)) is same as in the successive projection scheme. For levels 3 to r the projected torque is obtained by

$$\boldsymbol{\tau}_i^p = \boldsymbol{N}_i^{\text{aug}}(\boldsymbol{q})\boldsymbol{\tau}_i\,. \tag{2.55}$$

Here, the nullspace projector $\boldsymbol{N}_i^{\text{aug}}(\boldsymbol{q})$ can be expressed as

$$\boldsymbol{N}_i^{\text{aug}}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}_{i-1}^{\text{aug}}(\boldsymbol{q})^{\mathsf{T}}(\boldsymbol{J}_{i-1}^{\text{aug}}(\boldsymbol{q})^{\boldsymbol{W}+})^{\mathsf{T}}\,. \tag{2.56}$$

The augmented Jacobian matrix is denoted $\boldsymbol{J}_{i-1}^{\text{aug}}(\boldsymbol{q})$ and can be assembled as follows

$$\boldsymbol{J}_{i-1}^{\text{aug}}(\boldsymbol{q}) = \begin{pmatrix} \boldsymbol{J}_1(\boldsymbol{q}) \\ \boldsymbol{J}_2(\boldsymbol{q}) \\ \vdots \\ \boldsymbol{J}_{i-1}(\boldsymbol{q}) \end{pmatrix}\,. \tag{2.57}$$

As in the previous task hierarchy scheme, the total joint torque is given by (2.54), where the projected torque on each level can be determined with (2.55). There exist several recursive algorithms [196, 197] to reduce the computational effort required to determine the augmented projections (2.56). Augmented projections strictly comply with the task hierarchy; however, algorithmic singularities may arise [193]. Successive projections, on the other hand, are computationally efficient and avoid singularities but do not ensure strict compliance with the task hierarchy. A thorough comparison of successive and augmented projections is provided in [193].

Extended Jacobian

Besides using a task hierarchy, one can formulate the nullspace objectives as constraints. In the *extended Jacobian matrix* approach, one can add up to $r \leq n - m$ independent constraints of the form

$$\boldsymbol{h}(\boldsymbol{q}) = \boldsymbol{0}, \tag{2.58}$$

to the original end-effector task [198,199]. The constraints must be differentiable w. r. t. the joint position $\boldsymbol{q}$. For a motion that fulfills the main task and (2.58), one gets

$$\begin{pmatrix} \boldsymbol{f}_k(\boldsymbol{q}) \\ \boldsymbol{h}(\boldsymbol{q}) \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}(\boldsymbol{q}) \\ \boldsymbol{0} \end{pmatrix}\,, \tag{2.59}$$

where $\boldsymbol{f}_k(\boldsymbol{q})$ are the forward kinematics of the main task. Differentiating (2.59) w. r. t. time yields

$$\begin{pmatrix} \boldsymbol{J}(\boldsymbol{q}) \\ \dfrac{\partial \boldsymbol{h}(\boldsymbol{q})}{\partial \boldsymbol{q}} \end{pmatrix} \dot{\boldsymbol{q}} = \begin{pmatrix} \dot{\boldsymbol{x}} \\ \boldsymbol{0} \end{pmatrix}\,, \tag{2.60}$$

where

$$\boldsymbol{J}_{\text{e}}(\boldsymbol{q}) \in \mathbb{R}^{(m+r)\times n} = \begin{pmatrix} \boldsymbol{J}(\boldsymbol{q}) \\ \dfrac{\partial \boldsymbol{h}(\boldsymbol{q})}{\partial \boldsymbol{q}} \end{pmatrix}\,, \tag{2.61}$$

is defined as the extended Jacobian matrix. Typically, $n - m$ additional constraints are formulated such that $\boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})$ becomes invertible [198]. In addition to kinematic singularities, algorithmic singularities may occur when using the extended Jacobian matrix. In this case, $\boldsymbol{J}_e(\boldsymbol{q})$ becomes singular even in joint configurations for which $\boldsymbol{J}(\boldsymbol{q})$ has full rank [198]. Possible approaches to cope with such singularities are described in [200], for example.

2.4 Basics of Optimization

In this section, the basics of static optimization and optimal control are briefly reviewed. A thorough summary of the theory on optimization and numerical solution techniques can be found in [201–203], for example.

2.4.1 Static Optimization

In static optimization problems, the aim is to minimize the objective function $J(\boldsymbol{x})$, also referred to as cost function, where $\boldsymbol{x} \in \mathbb{R}^n$ is the optimization variable. The problem can formally be summarized as

$$\min_{\boldsymbol{x}\in\mathcal{X}} J(\boldsymbol{x}) \quad \text{s.t.} \quad \mathcal{X} = \{\boldsymbol{x} \in \mathbb{R}^n \,|\, \boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}, \boldsymbol{h}(\boldsymbol{x}) \leq \boldsymbol{0}\}\,. \tag{2.62}$$

Two types of constraints can be considered, namely equality constraints

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}, \quad \boldsymbol{g} : \mathbb{R}^n \to \mathbb{R}^m,\ m < n\,, \tag{2.63}$$

and inequality constraints

$$\boldsymbol{h}(\boldsymbol{x}) \leq \boldsymbol{0}, \quad \boldsymbol{h} : \mathbb{R}^n \to \mathbb{R}^q\,. \tag{2.64}$$

The equality and inequality constraints define the admissible set $\mathcal{X} \subset \mathbb{R}^n$ of the optimization variable $\boldsymbol{x}$. The state $\boldsymbol{x}^*$ is called global minimum if

$$J(\boldsymbol{x}^*) \leq J(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{X}\,. \tag{2.65}$$

A local minimum is achieved if there exists a $\varepsilon > 0$ such that

$$J(\boldsymbol{x}^*) \leq J(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X} \cap B_\varepsilon(\boldsymbol{x}^*)\,, \tag{2.66}$$

where $B_\varepsilon(\boldsymbol{x}^*)$ is a small, open ball around $\boldsymbol{x}^*$. To formulate the necessary optimality conditions, the Lagrange function

$$L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{\lambda}^\mathsf{T}\boldsymbol{g}(\boldsymbol{x}) + \boldsymbol{\mu}^\mathsf{T}\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i g_i(\boldsymbol{x}) + \sum_{i=1}^{q} \mu_i h_i(\boldsymbol{x})\,, \tag{2.67}$$

is defined. Here, $\boldsymbol{\lambda} \in \mathbb{R}^m$ are the so-called Lagrange multipliers, and $\boldsymbol{\mu} \in \mathbb{R}^q$ are the Kuhn-Tucker multipliers. The first-order optimality conditions are as follows. If $\boldsymbol{x}^*$ corresponds to a minimum in $J(\boldsymbol{x})$ under consideration of the equality and inequality constraints, then

there exist multipliers $\boldsymbol{\lambda}^* \in \mathbb{R}^m$ and $\boldsymbol{\mu}^* \in \mathbb{R}^q$ such that

$$\nabla_{\boldsymbol{x}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\boldsymbol{x}} g_i(\boldsymbol{x}^*) + \sum_{i=1}^{q} \mu_i^* \nabla_{\boldsymbol{x}} h_i(\boldsymbol{x}^*) = \boldsymbol{0}\,, \tag{2.68a}$$

$$\nabla_{\boldsymbol{\lambda}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \boldsymbol{g}(\boldsymbol{x}^*) = \boldsymbol{0}\,, \tag{2.68b}$$

$$\boldsymbol{h}(\boldsymbol{x}^*) \leq \boldsymbol{0}\,, \tag{2.68c}$$

$$\boldsymbol{\mu}^* \leq \boldsymbol{0}\,, \tag{2.68d}$$

$$\sum_{i=1}^{q} \mu_i^* h_i(\boldsymbol{x}^*) = 0\,. \tag{2.68e}$$

These optimality conditions are also referred to as Karush-Kuhn-Tucker (KKT) conditions. The second-order optimality conditions are

$$\boldsymbol{\delta x}^{\mathsf{T}} \nabla^2_{\boldsymbol{xx}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \boldsymbol{\delta x} \geq 0\,, \tag{2.69}$$

with

$$\boldsymbol{\delta x} \in \mathcal{Y} = \{\boldsymbol{\delta x} \,|\, \nabla g_1(\boldsymbol{x}^*)^{\mathsf{T}} \boldsymbol{\delta x} = 0, \ldots, \nabla g_m(\boldsymbol{x}^*)^{\mathsf{T}} \boldsymbol{\delta x} = 0\,, \tag{2.70a}$$

$$\nabla h_1^{sa}(\boldsymbol{x}^*)^{\mathsf{T}} \boldsymbol{\delta x} = 0, \ldots, \nabla h_{q_{sa}}^{sa}(\boldsymbol{x}^*)^{\mathsf{T}} \boldsymbol{\delta x} = 0\,, \tag{2.70b}$$

$$\nabla h_1^{ma}(\boldsymbol{x}^*)^{\mathsf{T}} \boldsymbol{\delta x} = 0, \ldots, \nabla h_{q_{ma}}^{ma}(\boldsymbol{x}^*)^{\mathsf{T}} \boldsymbol{\delta x} = 0\}\,, \tag{2.70c}$$

where $\boldsymbol{h}^{sa}$ are strictly active inequality constraints ($h_i = 0, \mu_i \neq 0$) and $\boldsymbol{h}^{ma}$ momentary active inequality constraints ($h_i = 0, \mu_i = 0$). A sufficient optimality condition is

$$\boldsymbol{\delta x}^{\mathsf{T}} \nabla^2_{\boldsymbol{xx}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \boldsymbol{\delta x} > 0 \quad \boldsymbol{\delta x} \in \mathcal{Y}\,. \tag{2.71}$$

Depending on the form of the cost function and constraints, one can distinguish between different classes of optimization problems. The most common classes are:

- Linear Programming (LP): Both the cost function and constraints are linear.
- Quadratic Programming (QP): The cost function is quadratic, the constraints are linear.
- Non-linear Programming (NLP): The cost function or at least one constraint is non-linear.
- Integer Programming (IP): All variables are discrete.
- Mixed-Integer Programming (MIP): Both continuous and discrete variables are considered.

For each class, there exist algorithms that are tailored to the specific optimization problem [203].

2.4.2 Optimal Control

Optimal control aims at determining a control input for a dynamic system that minimizes the cost functional

$$J\left(\boldsymbol{x}(t), \boldsymbol{u}(t), t_f\right) = \vartheta\left(\boldsymbol{x}(t_f), t_f\right) + \int_0^{t_f} \phi\left(\boldsymbol{x}(t), \boldsymbol{u}(t), t\right)\ dt\,, \tag{2.72}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the system state, $\boldsymbol{u}(t) \in \mathbb{R}^m$ the control input, and t_f the terminal time. The so-called Mayer term $\vartheta\left(\boldsymbol{x}(t_f), t_f\right)$ weights the final state and time, the integral Lagrange term $\phi\left(\boldsymbol{x}(t), \boldsymbol{u}(t), t\right)$ considers the timely evolution of state, control input, and time. The dynamics are described in terms of a system of first order differential equations

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0\,, \tag{2.73}$$

where $\boldsymbol{x}_0$ is the initial state. Several types of constraints that are inherent to every real-world system can be taken into account in the optimal control formalism. Boundary state constraints are formulated by

$$\boldsymbol{g}(\boldsymbol{x}(t_f), t_f) = \boldsymbol{0}\,, \tag{2.74}$$

where $\boldsymbol{g} \in \mathbb{R}^l$. Algebraic path inequality constraints are denoted by

$$\boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \leq \boldsymbol{0}, \qquad \forall t \in [0, t_f]\,. \tag{2.75}$$

Under consideration of (2.75), the optimal control input must satisfy $\boldsymbol{u}(t) \in \mathcal{U}(\boldsymbol{x}, t)$, where

$$\mathcal{U}(\boldsymbol{x}, t) = \{\boldsymbol{u}(t) \,|\, \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \leq \boldsymbol{0}\}\,, \tag{2.76}$$

is the admissible control region. To describe the necessary optimality conditions, a Hamiltonian defined as

$$H = \phi(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \boldsymbol{\lambda}(t)^\mathsf{T} \boldsymbol{f}\left(\boldsymbol{x}(t), \boldsymbol{u}(t)\right)\,, \tag{2.77}$$

is used. Here, $\boldsymbol{\lambda}(t) \in \mathbb{R}^n$ are Lagrange multipliers, which are also referred to as costates. The inequality constraints can be integrated into the Hamiltonian to form the extended Hamiltonian

$$\tilde{H} = \phi\left(\boldsymbol{x}(t), \boldsymbol{u}(t), t\right) + \boldsymbol{\lambda}(t)^\mathsf{T} \boldsymbol{f}\left(\boldsymbol{x}(t), \boldsymbol{u}(t)\right) + \boldsymbol{\mu}(t)^\mathsf{T} \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t))\,, \tag{2.78}$$

where $\boldsymbol{\mu}(t) \in \mathbb{R}^q$. The necessary first order conditions for all $t \in [0, t_f]$ are

$$\dot{\boldsymbol{x}} = \frac{\partial \tilde{H}}{\partial \boldsymbol{\lambda}} = \boldsymbol{f}\,, \tag{2.79a}$$

$$\dot{\boldsymbol{\lambda}} = -\frac{\partial \tilde{H}}{\partial \boldsymbol{x}} = -\frac{\partial \phi}{\partial \boldsymbol{x}} - \frac{\partial \boldsymbol{f}^\mathsf{T}}{\partial \boldsymbol{x}} \boldsymbol{\lambda} - \frac{\partial \boldsymbol{h}^\mathsf{T}}{\partial \boldsymbol{x}} \boldsymbol{\mu}\,, \tag{2.79b}$$

$$\mu_i h_i = 0, \quad i = 1, \ldots, q\,, \tag{2.79c}$$

$$\frac{\partial H}{\partial \boldsymbol{u}} = \frac{\partial \phi}{\partial \boldsymbol{u}} + \frac{\partial \boldsymbol{f}^\mathsf{T}}{\partial \boldsymbol{u}} \boldsymbol{\lambda} + \frac{\partial \boldsymbol{h}^\mathsf{T}}{\partial \boldsymbol{u}} \boldsymbol{\mu} = 0\,. \tag{2.79d}$$

Equations (2.79a) and (2.79b) form a canonical system of $2n$ differential equations for the dynamic system of order n [201]. A second-order necessary optimality condition to confirm local minima is the Legendre-Clebsch condition

$$\frac{\partial^2 H}{\partial^2 \boldsymbol{u}} \geq \boldsymbol{0}, \qquad \forall t \in [0, t_f]\,. \tag{2.80}$$

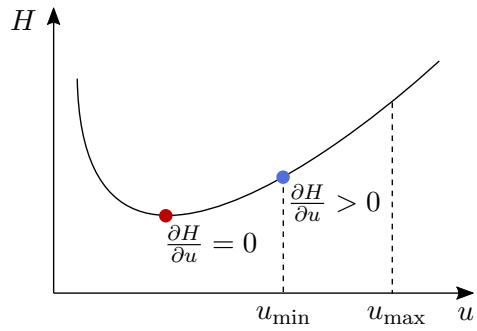


Fig. 2.5: Minimization of the Hamiltonian when the control region is bounded. The minimum and maximum value of the control input are denoted by $u_{\min}$ and $u_{\max}$. The (local) minimum in the Hamiltonian is represented by a red dot, the minimum value reachable value for the admissible control region by a blue dot.

The transversality conditions at the final time instant are

$$\left[\frac{\partial \phi}{\partial \boldsymbol{x}(t_f)} + \frac{\partial \boldsymbol{g}^\mathsf{T}}{\partial \boldsymbol{x}(t_f)}\boldsymbol{\nu} - \boldsymbol{\lambda}(t_f)\right]^\mathsf{T} = \mathbf{0}\,, \tag{2.81a}$$

$$H(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f), \boldsymbol{\lambda}(t_f), t_f) + \frac{\partial \varphi}{\partial t_f} + \frac{\partial \boldsymbol{g}^\mathsf{T}}{\partial t_f}\boldsymbol{\nu} = \mathbf{0}\,, \tag{2.81b}$$

with $\boldsymbol{\nu} \in \mathbb{R}^l$. Together with the boundary conditions (2.73) and (2.74) there exist $2n+l+1$ conditions. $2n$ conditions are required to solve the boundary value problem formed by the canonical system; the other $l+1$ conditions are necessary to obtain the multipliers $\boldsymbol{\nu}$ and the final time t_f. The boundary value problem can be solved analytically only for rather simple problems. If the dynamic system is complex or state constraints are present, numerical tools, e.g., shooting methods [202], are typically used to determine a solution. In problems where $\frac{\partial^2 H}{\partial^2 \boldsymbol{u}}$ is regular, the optimal control input can be obtained via (2.79d). The resulting control trajectory depends on the timely evolution of states and costates. However, if the problem is singular, then an alternative approach must be taken [204].

Singular Optimal Control Problems

An optimal control problem is singular if $\frac{\partial^2 H}{\partial^2 \boldsymbol{u}} = 0$. A comprehensive summary of this problem type can be found in [204]. In the most frequent subclass of singular problems, the Hamiltonian is linear in the control put. In this case the Hamiltonian can be split into two parts

$$H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t) = H_1(\boldsymbol{x}, \boldsymbol{\lambda}, t) + \boldsymbol{H}_2(\boldsymbol{x}, \boldsymbol{\lambda}, t)^\mathsf{T}\boldsymbol{u}(t)\,. \tag{2.82}$$

The first term does not depend on the control input, while the second is linear in $\boldsymbol{u}(t)$. When evaluating (2.79d), the control input vanishes, which implies that (2.79d) cannot be used to determine $\boldsymbol{u}(t)$. However, if the optimal control input is bounded, then the Minimum Principle of Pontryagin [205] provides a solution to the optimal control problem.

Minimum Principle of Pontryagin

If the admissible control region $\mathcal{U}$ is bounded, then it is possible that no local minimum in the Hamiltonian is reachable; see Fig. 2.5. There exist three cases: 1) the local minimum

is located inside the control region, 2) the partial derivative of the Hamiltonian w.r.t. the control input at the boundary of the control region is $\frac{\partial H}{\partial \boldsymbol{u}} < 0$, or 3) $\frac{\partial H}{\partial \boldsymbol{u}} > 0$. For the sake of brevity, only the last case is depicted in Fig. 2.5. The Minimum Principle of Pontryagin requires the optimal control input to minimize the Hamiltonian in every time step. This means that (2.79d) is replaced by

$$H(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u}, t) = \min_{\boldsymbol{U} \in \mathcal{U}(\boldsymbol{x}, t)} H(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{U}, t) \,. \tag{2.83}$$

For each control input $u_i, i = 1, \ldots, n$ that is bounded by $u_{i,\min} \leq u_i \leq u_{i,\max}$ the switching law is

$$u_i = \begin{cases} u_{i,\min}, & \sigma_i > 0 \\ u_{i,\max}, & \sigma_i < 0 \\ u_{i,\text{sing}}, & \sigma_i \equiv 0 \end{cases} , \tag{2.84}$$

where σ_i are the so-called switching functions

$$\sigma_i = \frac{\partial H}{\partial u_i} \,. \tag{2.85}$$

The control is of bang-bang type, i.e., the control input is either $u = u_{i,\min}$ or $u = u_{i,\max}$, if the switching functions only contain isolated zeros. If there exists a time interval $[t_1, t_2] \subseteq [0, t_f], t_1 \neq t_2$ in which σ_i equals zero, then singular solution pieces, so-called singular arcs, occur. In some cases such singular arcs can be difficult or even impossible to determine [206].

2.5 Considered Robots

This section briefly describes the robots that are used for simulations and experiments in this thesis. The robots are illustrated in Fig. 2.6.

(a) DLR/KUKA LWR mounted on linear axis

(b) Franka Emika Panda

(c) DLR David

Fig. 2.6: Robots considered in this work: (a) DLR/KUKA Lightweight Robot with/without linear axis, (b) Franka Emika Panda, and (c) DLR David.

2.5.1 DLR/KUKA Lightweight Robot

The development of lightweight robots has been a central research topic at the Institute of Robotics and Mechatronics of the German Aerospace Center (DLR) for more than 20 years. Three generations of the DLR Lightweight Robot (LWR) were developed. The latest generation, the LWR III, has seven degrees of freedom and a maximum reach of 936 mm; see Fig. 2.6 (a). It weighs 14 kg and has a payload to weight ratio of 1:1, meaning the robot is powerful enough to carry its own weight. The electronics and cabling are fully integrated; motor and joint position and joint torque sensors are available. The lightweight design and sensory capabilities enable the robot to detect and react to contact with the environment, enabling safe physical interaction with the human. With 1 kHz control frequency, advanced torque-based control concepts like joint or Cartesian impedance control can be realized (see above). The LWR is widely used in robotics research and is often part of complex mobile, flying, and multi-arm robot systems. In 2004 the LWR III was licensed to KUKA AG. KUKA continued the development and released the LWR IV (2008), the LWR IV+ (2010), and the iiwa (2013). Throughout the thesis, the LWR IV+ is used for various simulations and experiments. Following variants are considered:

- Stationary LWR
- LWR mounted on linear axis (see Fig. 2.6 (b))
- LWR mounted on omnidirectional mobile platform

The robot's flexible joint dynamics are provided in Sec. 2.1.

2.5.2 Franka Emika Panda

The Franka Emika Panda is a tactile, collaborative robot that was first presented to the public in 2016; see Fig. 2.6 (b). Panda features seven revolute joints and joint torque sensing in each joint. The robot's maximum reach is 855 mm, the workspace coverage is 94.5 %, the nominal payload is 3 kg, and the weight is 18 kg. As with the LWR, real-time torque, position, and velocity control are possible at 1 kHz sampling rate, which allows for adjustable stiffness/compliance control and unified force/impedance control. A user interface called *Pilot* is located at the last link. The buttons allow to execute and customize applications, e.g., teaching motions via manual guidance. The *Franka Desk* a visual programming environment runs on browsers of computers or tablets. It allows to intuitively program the robot via apps that require only minimal parameterization by the user. Nowadays, the Franka Emika Panda is the most popular platform in robotics research. In this thesis, the safety characteristics of the following variants of Panda are investigated:

- Stationary Panda
- Panda mounted on car-like mobile platform
- Panda mounted on differential drive mobile platform

The dynamics of these systems are provided in Sec. 2.1.

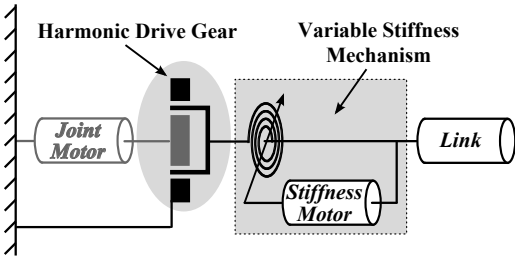


Fig. 2.7: The variable stiffness mechanism of the FSJ is located in series between the harmonic drive gear box of the main actuator and the link. © 2008 IEEE [122].

2.5.3 DLR David

DLR David [151] is an anthropomorphic robot that has variable stiffness actuation in each joint; see Fig. 2.6 (c). The design goal of David was to achieve human-like size, weight, dexterity, and performance. The motivation for introducing deliberate joint elasticity was the improvement of mechanical robustness, safety in human-robot interaction, and the inherent capability to store and release energy. This, in turn, can be utilized to outperform rigid robots in terms of energy efficiency and peak velocity. Its weight is 26 kg, the total number of DOF is 41, and the control frequency is 3 kHz. David's arm consists of four Floating Spring Joints (FSJ) [123] with two motors each. The first motor positions the joint, the second (smaller) motor changes the joint stiffness; see Fig. 2.7. The torque/deflection curve of an FSJ is progressive, i.e., the joint stiffness increases with the elastic deflection. The available spring potential energy depends on the stiffness preset; for a soft stiffness preset, more energy can be exploited than for a stiff preset. In David's forearm, three Bidirectional Antagonistic Variable Stiffness Joints (BAVS) [207] control the 2-DOF wrist. The tendon-driven hand has 19 DOF; its design was based on an abstraction of the principal functionalities of the human hand [179]. The antagonistic finger actuation was shown to be both highly dynamic and robust against impacts. The 42 motors and 42 nonlinear compliance mechanisms used to actuate the hand are located in David's forearm. In this work, vibration suppression schemes are developed and tested on David, cf. Chapter 5. Furthermore, the robot's maximum endpoint velocity capabilities and safety characteristics are derived and compared to those of rigid joint robots; cf. Chapter 4.

CHAPTER 3

Safety Map: A Global Perspective on Robot Collision Safety

In close physical human-robot interaction, it is primary to ensure human safety during operation. In particular, the intrinsic safety characteristics of a robot in terms of potential human injury have to be understood well. Then, minimal potential harm can be made an essential requirement at an early stage of the robot design. In this chapter, the *Safety Map* is introduced, a map that captures human injury occurrence and robot inherent global or task-dependent safety properties for the desired granularity in a unified manner. To derive the Safety Map representation and relate entire robot designs to available biomechanics safety data, the robot's reflected mass and maximum velocity are analyzed in task-dependent workspace sets. This is done for the PUMA 560, the KUKA LWR IV+, and the Franka Emika Panda. The approach is then extended to mobile manipulators that consist of a mobile vehicle and the Franka Emika Panda. Both the principal and the quantitative effects of the platform parameters on the mobile robot's collision safety are investigated. Regarding human injury data, the initial injury data literature overview provided in [37] is extended by a thorough summary of the human head and chest. It is shown how the generation of injury data can be embedded into the risk assessment and reduction cycle of a particular application. Furthermore, examples are given of how the Safety Map can be utilized for safety assessment and optimization as well as motion generation. In summary, the Safety Map concept serves as a global safety assessment framework for entire robot designs without the need for simplifications. This makes it a valuable tool not only for safety-oriented planning and control but in particular for safer robot design.

The remainder of this chapter is organized as follows. In Sec. 3.1, the Safety Map concept is described in detail. In Sec. 3.2, the collision model that is used to classify and compare collision experiments is introduced. The results of the literature review on injury data for the human head and chest are provided in Sec. 3.3. In Sec. 3.4 it is described how injury data can be systematically generated for particular applications. In Sec. 3.5, the robot kinematic and dynamic parameters are processed towards the Safety Map representation; examples for the PUMA 560, LWR IV+, and Franka Emika Panda are given. The extension to mobile manipulators is considered in Sec. 3.7. Section 3.8 addresses applications of the Safety Map. Finally, Sec. 3.9 concludes the chapter.

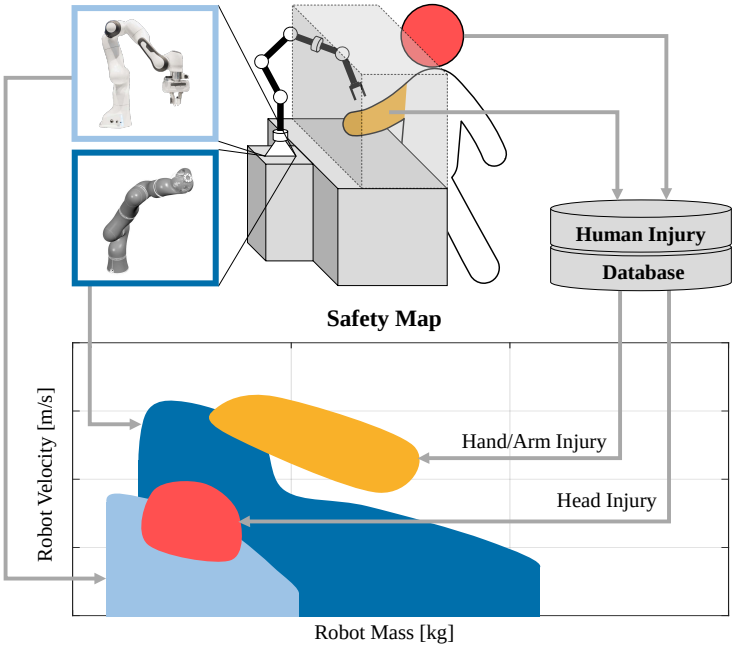


Fig. 3.1: Safety Map concept. The global or task-specific (gray interaction area) mass/velocity ranges of different robots (here: DLR LWR and Franka Emika Panda) and injury occurrence of human body parts are represented in a unified manner in the Safety Map. The injury data associated with the considered body parts are obtained from an injury database. © 2018 IEEE [208].

3.1 Definition

In previous works, trajectories or "representative" configurations were related to human injury probability or safety metrics in order to locally avoid unwanted injury via planning or control. In [12], it was proposed to associate instantaneous robot collision behavior, i.e., reflected mass, velocity, and contact geometry to observed human injury for a realistic and a-priori model-independent safety analysis. In contrast to other approaches, no intermediate physical quantities such as force or pressure had to be associated with injury. Then, so-called safety curves can be derived that provide a maximum biomechanically safe velocity as a function of instantaneous inertial robot properties. These representations were further developed into the safe velocity controller *Safe Motion Unit (SMU)* that limits the instantaneous robot speed by respecting the safety curves, thereby ensuring human safety even in case of entirely unforeseen collisions [12]. In this thesis, this idea is further employed to deduce a global perspective of a robot's collision safety. It is proposed to

- **relate entire robot designs**, i.e., the mass/velocity pairs for the reachable workspace, respectively a task-dependent subset, **to**
- **human injury data**, which may
  - originate from different types of experiments and disciplines (robotics, forensics, biomechanics, simulations etc.),
  - consider different body parts,

  - impactor curvatures (blunt, edgy, sharp, ...), and
  - collision cases (constrained, unconstrained),

- **in the same "coordinate system"**, namely the plane spanned by the robot reflected mass and velocity.

This global representation is denoted Safety Map, the concept is illustrated in Fig. 3.1.

The representations of human injury data and robot dynamic properties shall be independent, meaning there is no direct dependency of injury data on robot data and vice versa. This allows comparing different robot designs with the same injury data. Ideally, the injury representation is available in a purely data-driven functional relation, directly linking robot input parameters to human injury or pain. However, also injury criteria based on physical quantities such as force or pressure can be represented in the mass/velocity plane if a model or functional relation is available that provides the desired mapping.

By aggregating the representations of human injury data and dynamic robot properties, the Safety Map enables a robot/task designer to assess the considered robot(s) in combination with the task specification in terms of safety already at a very early planning stage. For example, in Fig. 3.1 the mass/velocity ranges of the two exemplary robots intersect with the head injury data, which means that both robots may harm the human head during unforeseen collisions. Hand/arm injury, however, can only be produced by the second robot[1]. Please note that the injury representations in Fig. 3.1 reflect the mass/velocity range in which the biomechanics collision experiments were carried out. Injury likely occurs for higher masses and velocities, i.e., above and on the right of the red/yellow injury areas. For the sake of readability, however, only the mass/velocity ranges considered in the experiments are depicted.

In Sec. 3.8, it is described in more detail how the Safety Map can be utilized for safety assessment and safe robot/task design. In the following, it is elaborated how injury data and robot dynamic properties can be systematically processed towards the Safety Map.

3.2 Collision Experiment Classification

In this section, an impact experiment classification is provided that governs all contact conditions that are relevant from a robotics perspective. The classification enables to systematically store the experimental data in a database and to process the generated data and insights towards safe control, planning, and mechanism design schemes; cf. Sec. 3.4. The collision model depicted in Fig. 3.2 follows the approach taken in [12] and is based on the idea that any mechanical system (here: impactor/robot and subject/human) can be represented by an instantaneous scalar mass, velocity, and surface properties in a certain Cartesian direction during contact. Furthermore, it is distinguished between different subject types, collision scenarios, and experimental setups. These are explained in the following subsections.

3.2.1 Impactor

The impactor/robot is modeled in terms of its instantaneous mass m_r, velocity $\dot{x}_r$, curvature c_r, and elastic surface properties EP_r. In biomechanics drop or pendulum test

[1]Please note that these are no general conclusions as for illustrative reasons the data in Fig. 3.1 is fictitious.

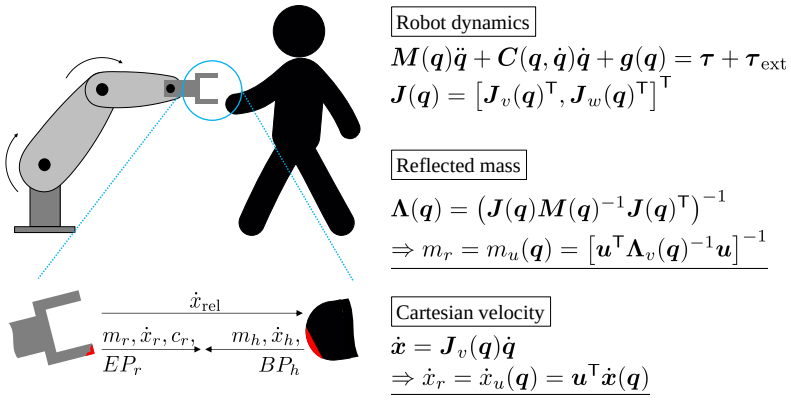


Fig. 3.2: Collision model for representing the instantaneous dynamic properties of the impactor/robot and subject/human. The scalar mass and velocity in the normalized Cartesian direction $\boldsymbol{u} \in \mathbb{R}^3$ are denoted by $m_u(\boldsymbol{q}) \in \mathbb{R}$ and $\dot{x}_u(\boldsymbol{q}) \in \mathbb{R}$. © 2018 IEEE [208].

setups, the impactors usually have a simple shape; the inertial properties can be measured easily [22,209]. For robots, the so-called reflected mass in a certain Cartesian direction constitutes the mass perceived during a collision [181]. The essential equations for calculating the Cartesian reflected mass and velocity are described in Sec. 2.1.4 and summarized in Fig. 3.2. The impactor surface may either be blunt, edgy, or sharp. In most publications, simple geometric shapes such as cylinders or flat circular plates are used, designed to deliver precise impacts to a desired location on the subject. In [12, 37] principal geometric primitives were identified and clustered; see Fig. 3.3. With these primitives, it is possible to classify most impactors used in biomechanics and robotics impact experiments.

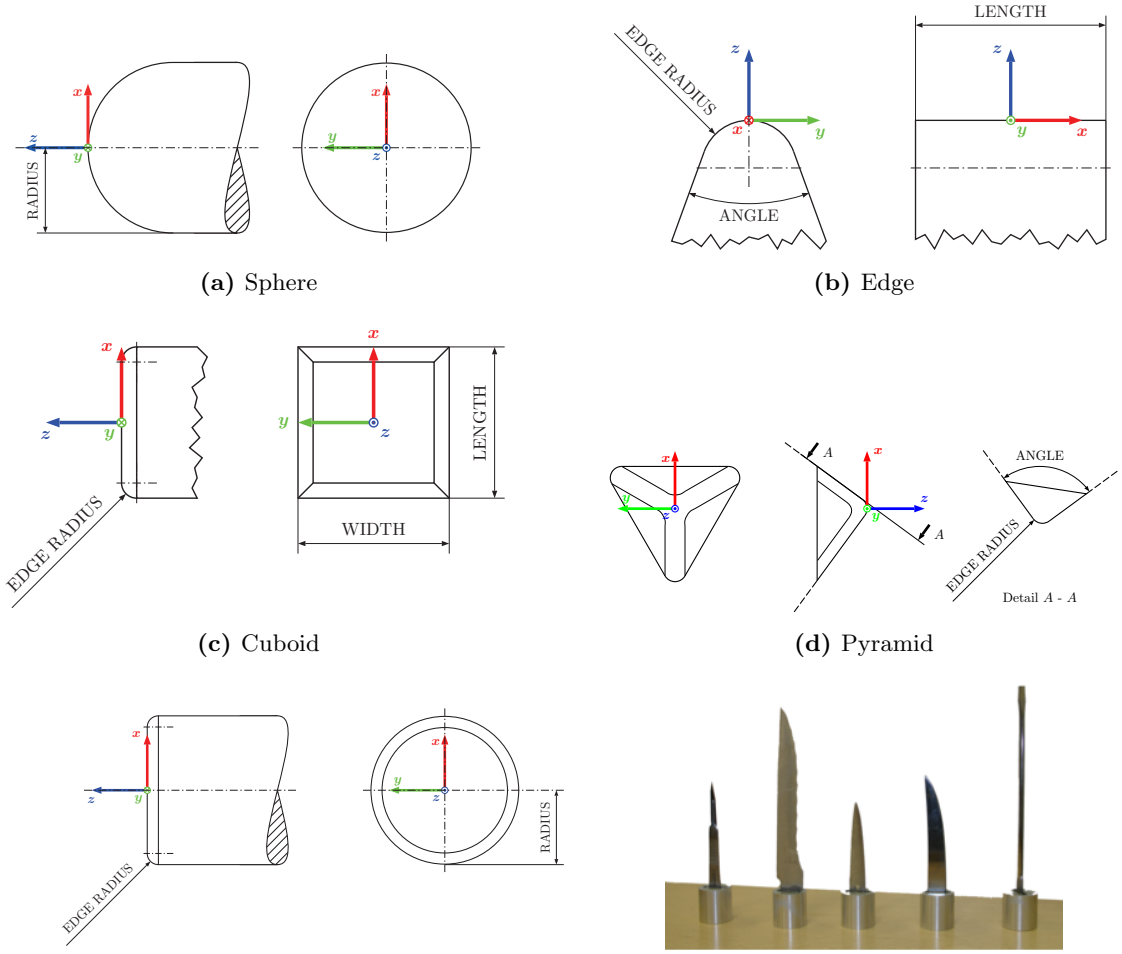
3.2.2 Subject

The subject type can be distinguished between an animal or human cadaver (in vitro), a mechanical human surrogate, a human volunteer (in vivo), or a mathematical collision model. Depending on the subject type, different aspects of the human biomechanical response can be studied [34]. To investigate human injury tolerances, the subjects are (for ethical reasons) typically animal or human cadavers (often referred to as post mortem test objects). In automobile crash-testing, mechanical human surrogates such as the Hybrid III dummy are commonly used for injury assessment. To analyze pain threshold or minor injury such as contusions, experiments have also been conducted with human volunteers [13, 37, 210].

In the impact model illustrated in Fig. 3.2, the subject is characterized by the impact location BP_h, the instantaneous mass m_h, and the velocity $\dot{x}_h$. The impact location of the human is a distinctive landmark of the musculoskeletal system, such as the frontal bone or maxilla in the human head [211]. It is assumed that the impactor's Cartesian direction of motion coincides with the surface normal of the respective body part. This agrees with the experimental design in almost all biomechanics and robotics publications on injury analysis. The relative velocity between subject and impactor[2] is denoted by $\dot{x}_{\text{rel}} = |\dot{x}_r - \dot{x}_h|$.

To estimate the effective human mass at the contact location (if not reported), one can a)

[2]Only robot and human velocities are considered that result in a collision.



(a) Sphere **(b)** Edge

(c) Cuboid **(d)** Pyramid

(e) Flat circular **(f)** Sharp tools

Fig. 3.3: Impactor primitives and their parameters.

use a model of the human body based on the geometrical and inertial properties [212,213], or b) fit the parameters of a mathematical collision model, e.g., a fully (in)elastic impact in a mass-spring-mass system, by conducting suitable impact experiments. For method a), the body segment inertial properties (BSIPs), namely mass, the moment of inertia, center of mass position, and the dimensions have to be known. Given only the weight and the height of a subject, mean values can be derived from cadaver measurements [212, 214]. For living subjects, magnetic resonance imaging (MRI) can be used for estimating the BSIPs [213]. By applying multi-body dynamics, one can then determine the effective mass analogous to the robot reflected mass; cf. Fig. 3.2 (right). Method b) is based on a mathematical collision model. The effective human mass can be estimated using the signals recorded during real impact experiments. Assume that a simple linear mass-spring-mass system represents the collision between robot and human. The maximal force during the collision is

$$F_{c,\max} = \sqrt{\frac{m_r m_h}{m_r + m_h}} \sqrt{K_h} \dot{x}_{\text{rel}} \,, \tag{3.1}$$

where K_h is the contact stiffness. After measuring the contact force and assuming a certain

contact stiffness we can rearrange (3.1) and estimate the human reflected mass via

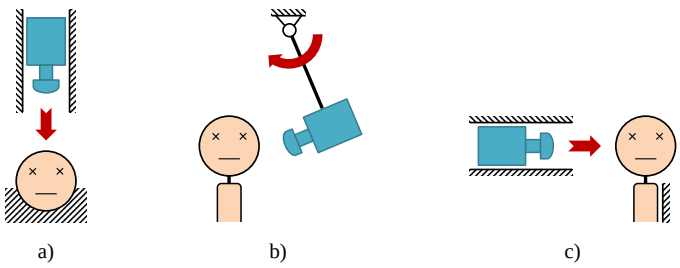
$$\hat{m}_h = \frac{F_{c,\max}^2 m_r}{K_h \dot{x}_{\text{rel}}^2 m_r - F_{c,\max}^2} \,. \tag{3.2}$$

3.2.3 Collision Scenario

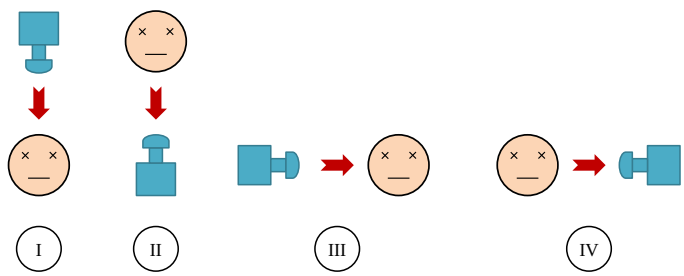
Regarding the collision scenario, it is distinguished between collisions where the subject is either *unconstrained*, *constrained*, or *partially constrained* [14]; see Fig. 3.4 (a). The latter is characterized only by a part of the subject being clamped which is not directly in contact with the impactor. The following abbreviations are used in the remainder of this chapter to classify the collision scenario: U: unconstrained, C: constrained, PC: partially constrained.

3.2.4 Test Setup

In biomechanics collision experiments, several types of setups have been used. Drop tests are the most common experiments, where free-fall due to gravity accelerates the impactor, that has a certain mass and impact geometry [12, 22, 209, 215]. The initial drop height determines the impactor velocity; the subject is usually constrained. Alternatively, impacts can be delivered horizontally to the subject. In [13,216], for example, a pendulum was used where the initial deflection adjusts the impact velocity. The impactor can also be propelled by a pneumatic cylinder [24, 217, 218]. Typically, the impactor is accelerated towards the subject. However, it is also possible to propel the subject towards the impactor [33, 219].



(a) Typical test setups and collision scenarios. a) Drop test setup with constrained subject, b) pendulum setup with unconstrained subject, and c) piston setup with partially constrained subject.



(b) Classification of test setups.

Fig. 3.4: Test setups in biomechanics impact experiments (a) and their classification (b).

From the biomechanics experiments, four principal setups are identified; see Fig. 3.4 (b). The free-fall principle is used in setups I and II, where I) the impactor or II) the subject is accelerated. In setups III and IV, the impact is delivered horizontally, where either III) the subject or IV) the impactor is at rest. Different human collision scenarios are possible for each setup, i.e., the body part may be constrained, unconstrained, or partially constrained.

Influence of Gravity in Collision Experiments The results from drop test experiments (setups I and II) are likely to be more conservative than the results of experiments in which the impactor is moving horizontally (setups III and IV). To the best of the author's knowledge, the influence of gravity has not been analyzed so far. In the following, a preliminary analysis based on a simplified collision model with and without gravity (see Fig. 3.5) is provided. A clamped human body part with constant contact stiffness K_h is considered. The selected stiffnesses $K_h = 10$, 100, and 1000 N/mm are in the range of human body parts [32]. The impactor is modeled as a blunt mass with a certain velocity. The maximum contact force $F_{\max,g}$ with gravity and without gravity ($F_{\max}$) is evaluated in simulation. Then, the ratio $F_{\max,g}/F_{\max}$ is calculated, which depends on both the initial velocity and mass. In Figure 3.6 it can be observed that the maximum contact force including gravity is larger than the force without gravity, which agrees well with intuition.

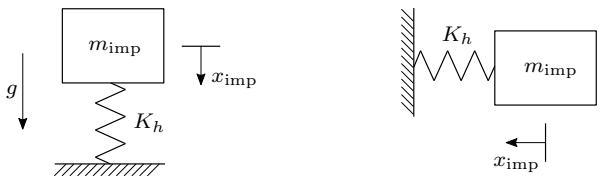


Fig. 3.5: 1-DOF collision model including gravity (left) and without gravity (right).

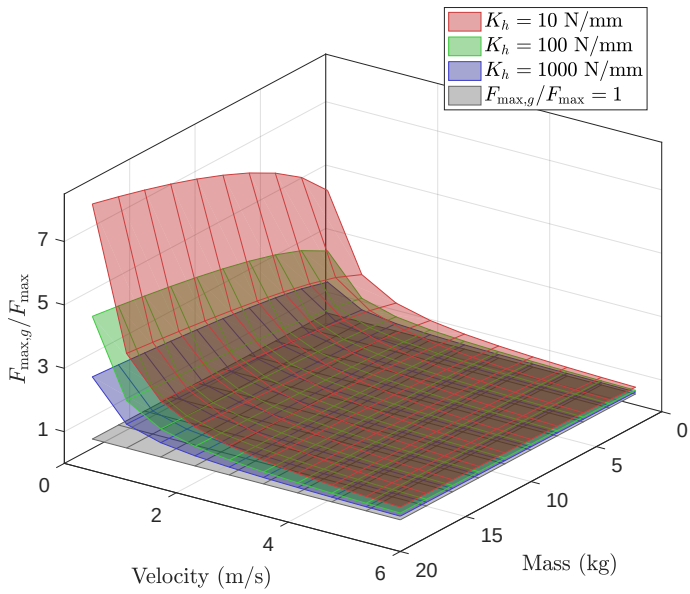


Fig. 3.6: Influence of gravity on the maximum contact force for a simplified collision model: Ratio of the maximum force with and without the influence of gravity for the stiffnesses $K_h = 10$, 100, and 1000 N/mm, depending on the initial impactor velocity and mass.

For this simplified model, the force ratio increases when the velocity decreases, the contact stiffness decreases, or the impactor mass increases. The velocity and contact stiffness have a more substantial influence on maximum force than the impactor mass. Theoretically, the ratio $F_{\max,g}/F_{\max}$ can be used to determine the maximum force for setups I/II in Fig. 3.4 when experiments were conducted with setups III/IV and vice versa. However, this only holds for the considered simplified model, which is not grounded by experimental analysis. Furthermore, the maximum contact force does not necessarily correlate well with resulting injury; cf. [12]. Therefore, the presented analysis is preliminary and requires verification by real-world experiments. Considering that collision setups I/II likely provide more conservative results than setups III/IV, it is not differentiated between collisions with and without the influence of gravity in the remainder of the thesis.

3.3 Synopsis of Human Injury Data

The proposed classification of collision experiments allows representing a large variety of experiments by a limited number of parameters. In the following, a summary of the most relevant biomechanics and robotics collision experiments on the human head and chest is provided; see Tab. 3.1. This summary is a result of an extensive literature study and extends the initial literature survey reported in [37]. For the data-driven approach of relating collision input parameters to resulting injury, the collected data is of great value because it allows comparing the results from different experiments, determining whether a particular robot may produce injury, or verifying mathematical collision models. Most of the available biomechanical literature stems from automotive injury analysis, focusing on more severe injuries. The head and chest are usually of particular interest, which is why much collision data has been generated for these two body parts. In Tab. 3.1, the experimental conditions for each impact series are summarized. The graphical representation of the relationship between impact parameters and injury severity is illustrated in Fig. 3.7 for a selection of the listed experiments. The aim is to represent the injury data in the mass/velocity plane, which will allow us to compare this data with the robot's safety properties in the Safety Map, as will be shown in Sec. 3.6.

3.3.1 Head

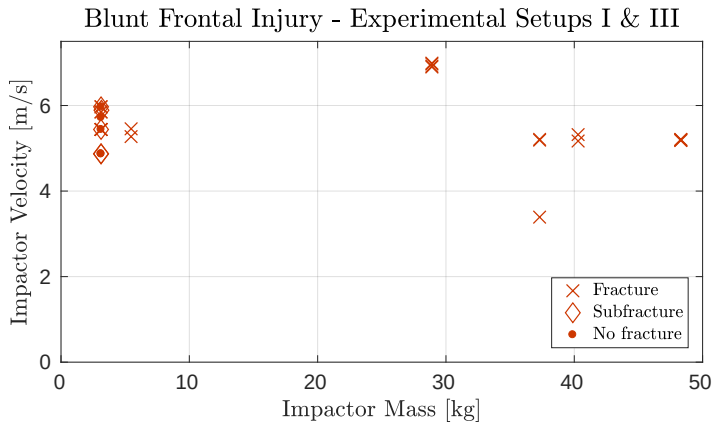
In the upper half of Tab. 3.1, an overview of data from facial and cranial bone injury analysis is provided. In Fig. 3.7 (a), the results of experiments on the frontal bone are illustrated, where a flat impactor is accelerated towards the subject (setups I and III) [21, 215, 216]. The collision input parameters mass and velocity are related to injury severity, which is skull fracture or subfractures (e.g., hairline cracks) in the figure. The impact velocity is $> 4\,\mathrm{m/s}$ in most experiments, which is higher than typical speeds in pHRI ($\leq 2\,\mathrm{m/s}$).

3.3.2 Chest

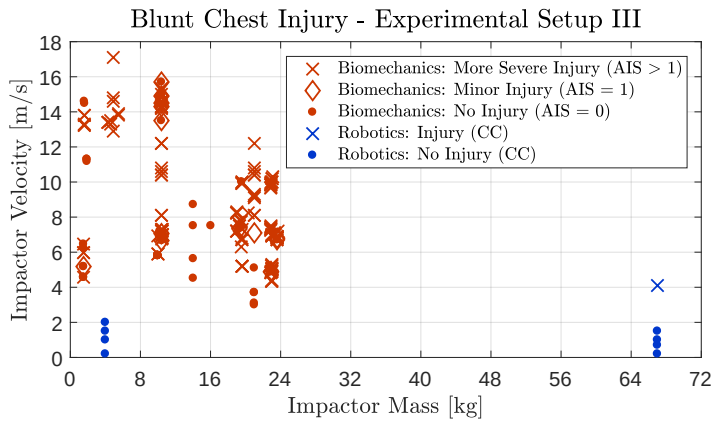
A significant amount of experiments on chest injury analysis was conducted in [23, 25, 224–230]. In order to better understand thoracic trauma in frontal impacts, a more recent extensive crash-test program was established [231]. In robotics, series of chest crash-test experiments were conducted in [15, 16], where several lightweight and heavy-duty robots

Tab. 3.1: Overview of selected impact experiments from biomechanics and robotics literature for the human head and chest. Following abbreviations are used: U: unconstrained, C: constrained, PC: partially constrained. © 2018 IEEE [208].

Body Part	Exp. Setup	Case	Impactor		Subject	Mass (kg)	Velocity (m/s)	Reference
			Primitive	Parameters				
Frontal	III	PC	Flat circular	35 mm radius	Cadaver	28.9 – 48.3	3.39 – 6.99	[216]
Frontal	II	U	Edge	10 mm radius	Cadaver	14.5	3.0 – 4.2	[209]
Frontal	I	U	Edge	25.4 mm & 7.9 mm radius	Cadaver	4.54	1.44 – 3.22	[22]
Frontal, Occipital	II	U	Flat rectangular	Padded, 120 mm x 80 mm	Cadaver	5	2.8 – 7.0	[220]
Frontal	III	U, PC	Sphere	Rigid, 120 mm radius	HIII Dummy	4.0, 67.0, 1980.0	0.2 – 4.2	[15]
Frontal	II	PC	Sphere	Padded, 76.2 mm – 203.2 mm radius	Cadaver	4.54 – 6.49	2.95 – 3.54	[221]
Frontal, Temporo-Parietal	II	PC	Edge	Padded, 3.2 mm – 25.4 mm radius	Cadaver	3.18 – 6.49	2.2 – 4.37	[221]
Frontal, Temporo-Parietal, Occipital	II	PC	Flat	Padded	Cadaver	3.31 – 5.9	2.23 – 5.43	[221]
Frontal, Zygoma, Mandible, Maxilla	III	U, PC	Flat circular	Padded, 14.3 mm & 32.7 mm radius	Cadaver	0.9 – 7.3	2.6 – 8.5	[21]
Frontal, Parietal, Occipital	II	U	Flat	-	Cadaver	3.74 – 6.64	4.1 – 6.9	[222]
Frontal, Temporo-Parietal, Zygoma, Maxilla, Mandible	I	PC	Flat circular	Padded, 14.3 mm radius	Cadaver	1.08 – 3.82	2.99 – 5.97	[215]
Temporo-Parietal	I	C	Flat circular	12.7 mm radius	Cadaver	10.6	2.7	[31]
Temporo-Parietal	I	C	Flat rectangular	50 mm x 100 mm	Cadaver	12	4.3	[31]
Nose	I	C	Flat circular	14.3 mm radius	Cadaver	3.2	1.58 – 3.16	[223]
Nose	III	U	Edge	12.5 mm	Cadaver	32, 64	2.8 – 7.1	[30]
Chest	IV	PC	Flat circular	Padded, 15.24 cm diameter, 161.29 – 193.55 cm^2 surface	Cadaver	14.03 - 19.55	4.52 - 10.06	[224, 225]
Chest	III	U	Flat circular	15.24 cm diameter, 1.28 cm edge radius	Cadaver	1.63 - 23.59	6.26 - 14.31	[23]
Chest	III	PC	Flat circular	15.24 cm diameter, 1.28 cm edge radius	Cadaver	9.98	5.36 - 6.26	[25]
Chest	IV	U	Flat circular	Padded, 15.24 cm diameter, 1.28 cm edge radius	Volunteer	10.01	2.40 - 4.60	[226]
Chest	III	U, C	Flat circular	Rigid/padded, 6.45 cm^2 surface	Cadaver	1.51, 10.01	4.02 - 10.01	[226]
Chest	III	U, C	Flat circular	15.24 cm diameter 1.28 cm edge radius	Cadaver	1.59 23.04	4.34 - 13.23	[227]
Chest	III	U, PC	Flat circular	15.24 cm diameter 1.28 cm edge radius	Porcine (anesthetized)	21.00	3.00 - 12.20	[228]
Chest	II	PC	Flat circular	15.00 cm diameter 1.27 cm edge radius	Swine (anesthetized)	4.90, 10.40, 21.00 21.00	8.10 - 31.60	[229, 230]
Chest	IV	C	Flat circular	–	Cadaver	11.05 - 26.19	6.44 - 16.61	[231]
Chest	III	U, C	Sphere	12.0 cm radius	HIII Dummy	4, 67, 1980	0.20 - 4.20	[15]
Chest	I	PC	Sphere	12.5 mm radius	Volunteer	3.68 - 3.79	0.19 - 1.31	[37]
			Edge	0.2 mm edge radius, 20.0 cm length		4.28 - 3.57	0.19 - 0.84	



(a)



(b)

Fig. 3.7: Summary of the relationship between mass, velocity, and injury (red) for selected data on (a) the human frontal bone [21, 215, 216] and (b) the human chest [15, 23, 25, 226–230]. For the robotics experiments with the Hybrid III dummy (blue), the results for the Chest Compression Criterion (CC) are illustrated. © 2018 IEEE [208].

were used. Both dynamic unconstrained (with KUKA KR6 and KR500 robot) and quasistatic constrained (with LWR III) frontal chest impacts were carried out on a Hybrid III dummy. An overview of the most relevant impact experiments on the chest, which were conducted in both biomechanics and robotics, is provided in the lower half of Tab. 3.1. In Figure 3.7 (b), the relationship between collision input parameters and injury severity is illustrated for selected chest impact experiments. In the figure, red markers represent results from biomechanics injury; the robotics data is shown in blue.

3.4 Application-Oriented Generation of Injury Data

Previously, it was explained how collision data could be classified and processed towards the Safety Map. For possible collisions in pHRI, already a significant amount of impact data has been generated [12, 13, 15, 16, 46, 232]. However, to understand the human impact behavior for a wide range of collision scenarios, applications, and types of human injury, there is still much data to be collected. In [12], a generic approach for generat-

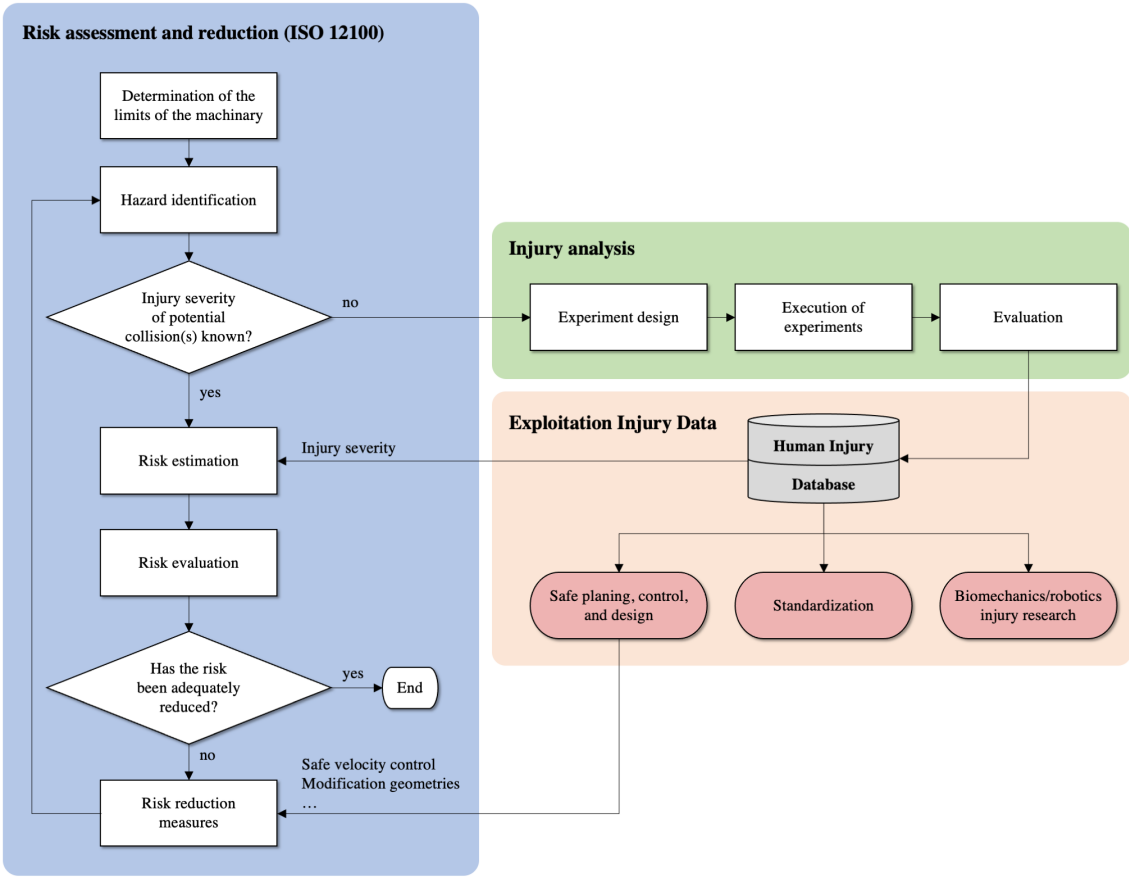


Fig. 3.8: Integration of experimental injury analysis into the risk assessment and reduction cycle.

ing collision data was proposed. The authors suggested decomposing the robot structure, in particular the end-effector, into a set of geometric primitives; see Fig. 3.3. Then, the achievable reflected mass and velocity range of typical collaborative robots was determined and discretized. Impact tests were conducted and evaluated for every possible (discretized) combination of mass, velocity, and primitive parameters. While generic contact primitives can represent a wide range of geometries located on the robot structure, more complex geometries need to be tested individually. Furthermore, an open question is how the injury data generation and safety assessment can be tailored to a particular application to estimate possible risks and take risk reduction measures quickly and target-oriented. In this thesis, a systematic approach is taken to generate and exploit injury data in the context of the risk assessment and reduction cycle described in ISO 12100 and ISO 10218; see Fig. 3.8. This approach is explained in the following.

The first step in the risk assessment is to determine the boundaries of the machinery. Then, possible hazards are identified, e.g., clamping situations or collisions with sharp objects. If the injury severity of a potential collision is unknown, then suitable impact experiments need to be conducted. This can be done with either the considered robot or a test setup that resembles the real contact conditions, e.g., a drop test or pendulum. The results of the impact tests are summarized and collected in a database. Information about the possible injury severity from this database can be extracted and provided to the risk

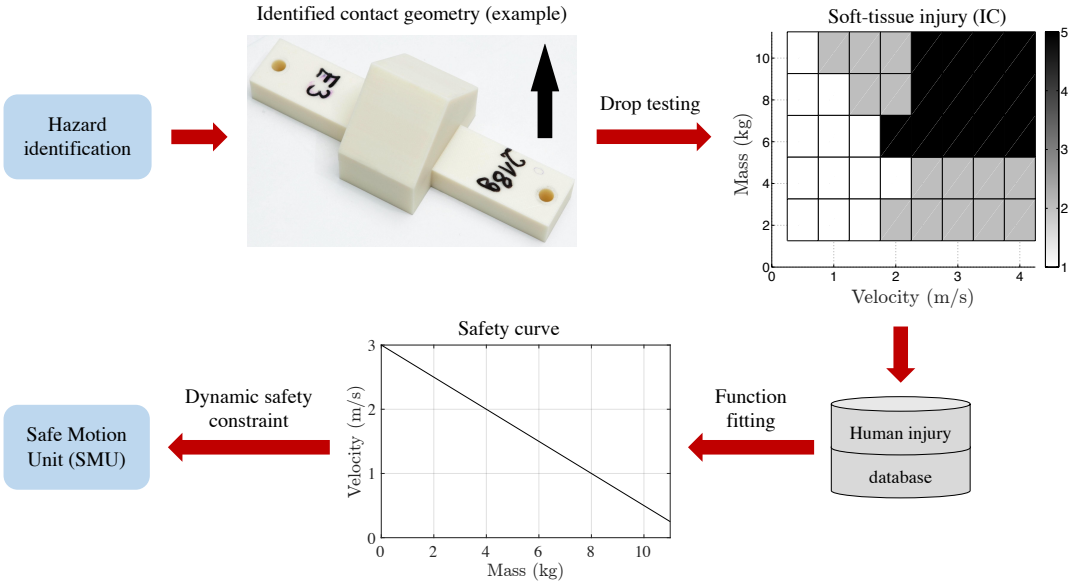


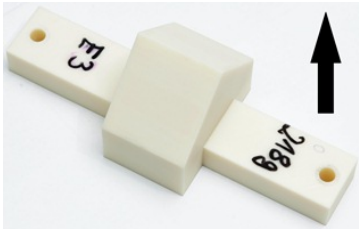
Fig. 3.9: From hazard identification to safe motion generation in an anonymized industrial use case. One of the contact geometries identified potentially hazardous is the illustrated edge with 50 mm length and 0 mm edge radius. The black arrow indicates the impact direction. This geometry is then tested via drop test experiments, and the medical evaluation processed towards a database, from which a safety curve is automatically generated. Finally, the safety curve is embedded into the Safe Motion Unit, which generates safe motions in the pHRI application.

estimation, which assesses both the severity of the harm and the likelihood of occurrence. Given the information provided by the risk assessment, the risk evaluation determines whether risk reduction measures are required. If this is the case, then again, the generated collision data and planning, control, or design optimization schemes based on this data can be employed to improve safety and start the risk assessment cycle from the beginning.

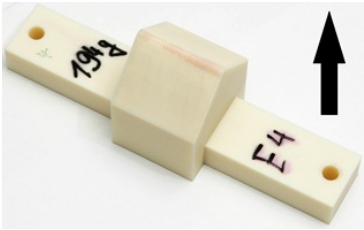
Together with an industrial partner, this process is completely run through for a real-world pHRI application; see Fig. 3.9. In the hazard identification phase, the contact geometries illustrated in Fig. 3.10 are identified. These are tested in systematic drop test experiments, which follow the same protocol as the tests conducted in [12]. The impactor mass range is 2 - 10 kg with 2 kg increment, the velocity range is 0.5 - 4 m/s with 0.5 m/s increment. In Fig. 3.9, the results of the medical soft-tissue assessment are reported for an edge-shaped geometry with 50 mm length and 0 mm edge radius. The medical data, sensor data, and experimental conditions are stored in a SQL database. From this database, drop test reports are automatically generated that summarize the results of a test series. These reports serve to document the experiments, and they have also been used to communicate the results to standardization committees. Another implemented export function regards the automatic generation of safety curves for real-time safe velocity control. Together with the industrial partner, the safety curves for the considered contact geometries (see Fig. 3.10) are embedded into the Safe Motion Unit, which generates biomechanically safe motions in the partner's pHRI application.

(a) Corner 5 mm edge radius.

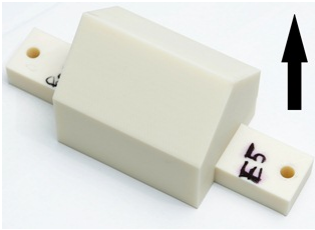
(b) Cylinder 25 mm radius, 0 mm edge radius.



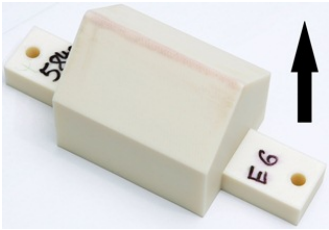
(c) Edge 50 mm length, 0 mm edge radius.



(d) Edge 50 mm length, 4 mm edge radius.



(e) Edge 100 mm length, 0 mm edge radius.



(f) Edge 100 mm length, 4 mm edge radius.

Fig. 3.10: Overview of impactors that were tested in drop test experiments in order to investigate the potential injury severity in an industrial use case. The impactors were made with a 3D resin printer and have high rigidity. The black arrow indicates the impact direction.

3.5 Global Robot Dynamic Properties

Having collected, classified, and processed human injury data, it is now described how the kinematic and dynamic characteristics of a robot can be mapped to a mass/velocity range in the Safety Map in order to represent the robot properties on a global or local, task-dependent, scale. One seeks the reflected mass and maximum velocity range for all reachable poses, i.e., Cartesian positions and orientations, and in every Cartesian direction. One main idea of the concept is to calculate the global dynamic properties of a robot design for the desired granularity only once. Afterwards, the data associated with task-dependent subsets of the robot workspace can be extracted. Specific trajectories or single static configurations can also be analyzed by interpolating the data, thus allowing for different degrees of granularity in the safety analysis. The procedure for computing the global robot dynamic properties consists of four steps:

Step 1 Discretize the workspace and determine all reachable poses of a robot, in other words, its reachability map.

Step 2 For each (discretized) reachable pose, determine the set of reachable null space configurations if the robot is redundant or select only one robot configuration with using a specified redundancy resolution method.

Step 3 Generate a grid of Cartesian directions.

Step 4 Calculate the Cartesian reflected mass and maximum velocity for each feasible pose, null space configuration, and Cartesian direction.

In the following, the four steps are explained in more detail. All considered quantities are illustrated in Fig. 3.11.

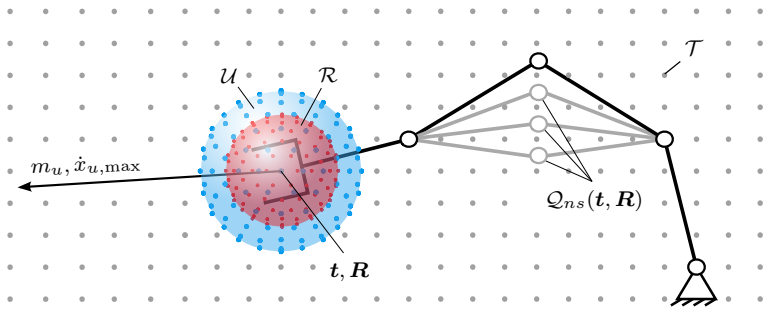


Fig. 3.11: Determining the global robot dynamic properties. The evenly-spaced Cartesian position grid is denoted by $\mathcal{T}$, the current end-effector position and orientation by $\boldsymbol{t}$ and $\boldsymbol{R}$, and the set of possible null space positions associated with this pose by $\mathcal{Q}_{ns}(\boldsymbol{t}, \boldsymbol{R})$. The set of discretized end-effector orientations (red sphere) for which the reachability is analyzed (for each position) is denoted by $\mathcal{R}$, the set of Cartesian unit directions (blue sphere) for evaluating the reflected robot mass and velocity by $\mathcal{U}$. The current reflected mass and velocity in direction $\boldsymbol{u}$ are denoted by m_u and $\dot{x}_{u,\max}$, respectively. © 2018 IEEE [208].

3.5.1 Workspace Discretization

One seeks the robot's so-called *versatile workspace*, i.e., all (discretized) reachable combinations of Cartesian position $\boldsymbol{t} \in \mathbb{R}^3$ and orientation $\boldsymbol{R} \in SO(3)$, which is defined as [233]

$$\mathcal{W}_v = \left\{ (\boldsymbol{t}, \boldsymbol{R}) \mid \exists \boldsymbol{q} \in \mathcal{Q} : \boldsymbol{f}_k(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0} & 1 \end{bmatrix} \right\} \subset SE(3)\,. \tag{3.3}$$

Here, $\mathcal{Q}$ is the robot configuration space, and $\boldsymbol{f}_k(\boldsymbol{q})$ is the forward kinematics. Many algorithms exist to determine the versatile workspace. In the algorithm proposed in [233], the workspace is discretized into an evenly-spaced, orthogonal position grid with desired granularity. The position set is defined as $\mathcal{T} = \{\boldsymbol{t}_1, \ldots, \boldsymbol{t}_{n_t}\} \subset \mathbb{R}^3$. For each position, a certain number n_r of discretized end-effector orientations $\mathcal{R} = \{\boldsymbol{R}_1, \ldots, \boldsymbol{R}_{n_r}\} \subset SO(3)$ is defined. Forward and/or inverse kinematics can be utilized to compute the reachable poses and the associated joint configurations.

3.5.2 Null Space Configurations for Redundant Robots

Reachability map algorithms usually provide one joint configuration for a certain Cartesian pose. If the robot is redundant, then also the desired number $n_{q_{ns}}$ of possible null space positions can be determined[3]. For each Cartesian position $\boldsymbol{t} \in \mathcal{T}$ and orientation $\boldsymbol{R} \in \mathcal{R}$ the set of discretized, achievable null space configurations is denoted by $\mathcal{Q}_{ns}(\boldsymbol{t}, \boldsymbol{R})$. For a six-DOF position/orientation task, the LWR has one redundant degree of freedom if the configuration is non-singular. The possible null space positions associated with a certain pose can be determined by successfully integrating the Jacobian matrix's one-dimensional kernel. Details on the procedure for the LWR are reported in Sec. 6.1. The analysis of the self-motion manifold for robots with more DOF is treated in [234, 235].

[3]Also non-redundant robots may have several possible joint configurations for a desired end-effector pose. However, for the sake of brevity, a thorough analysis of such configurations is omitted in this work.

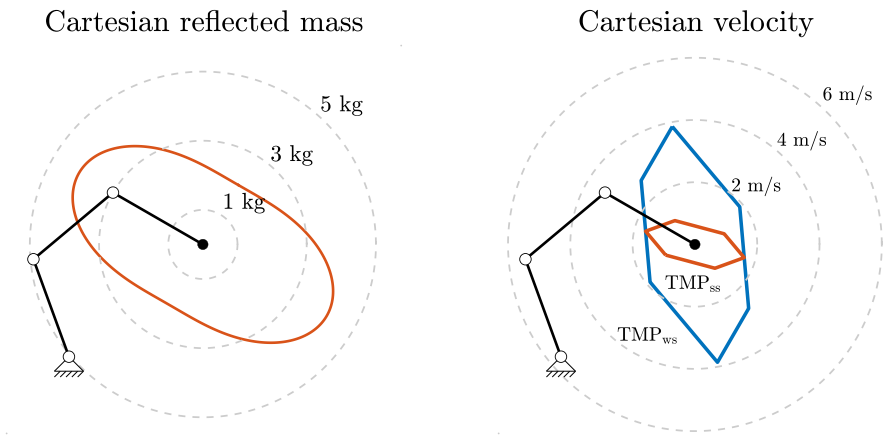


Fig. 3.12: 3R planar robot: Cartesian belted ellipsoid of reflected robot mass (left) and velocity polytopes (right), where TMP$_{ws}$ denotes the weak sense and TMP$_{ss}$ the strong sense polytope. The length of each robot link is 0.5 m, at the distal end of each link a 1 kg point mass is located. The maximum symmetric velocity of each motor is 2 rad/s and the joint configuration is $\boldsymbol{q}^{\mathsf{T}} = [110, -70, -70]\,°$. © 2021 IEEE [236].

3.5.3 Cartesian Direction Grid

The robot reflected mass and maximum velocity are calculated for a discretized number n_u of Cartesian unit directions. For this, a uniform grid on the surface of the unit sphere $\mathcal{S}^2 = \left\{\boldsymbol{x} \in \mathbb{R}^3 : ||\boldsymbol{x}|| = 1\right\}$ is generated, where the set of distributed points is defined as $\mathcal{U} = \{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n_u}\} \subset \mathcal{S}^2$.

3.5.4 Reflected Mass

For all reachable poses, for all possible null space positions, and for all Cartesian directions, i.e., for at most $n_{\text{tot}} \leq n_t \times n_r \times n_{q_{ns}} \times n_u$ configurations, the reflected mass $m_u(\boldsymbol{q})$ and maximum Cartesian velocity $\dot{x}_{u,\max}$ at the robot flange, respectively the tool center point (TCP), are evaluated. The reflected mass $m_u(\boldsymbol{q})$ in a particular unit direction $\boldsymbol{u}$ is given by (2.19); see also Fig. 3.2. For the 3R robot, the reflected mass ellipsoid, i.e., the representation of the reflected mass for all Cartesian directions, is illustrated in Fig. 3.12 (left).

3.5.5 Maximum Endpoint Velocity

The manipulability ellipsoid and polytope for rigid joint robots are well-established tools for analyzing a robot's Cartesian velocity capabilities for a given configuration [194, 237]. From a computational point of view, ellipsoids are more tractable than polytopes. However, only polytopes are considered in this work, as they represent all achievable velocities, while the ellipsoids only constitute a subset of these velocities. Consider the motor velocity constraints

$$\dot{\boldsymbol{q}}_{\min} \leq \dot{\boldsymbol{q}} \leq \dot{\boldsymbol{q}}_{\max}\,, \tag{3.4}$$

where $\dot{\boldsymbol{q}}_{\min}$ and $\dot{\boldsymbol{q}}_{\max}$ are the minimum and maximum motor velocity, respectively. For the sake of simplicity, it is assumed that these bounds are symmetric. The $2n$ motor velocity bounds form a hyperrectangle in joint space, which has 2^n vertices. The translational

manipulability polytope in the so-called *weak sense*, denoted by $\mathrm{TMP}_{\mathrm{ws}}$, can be obtained by transforming the joint-space hyperrectangle to Cartesian space via $\boldsymbol{\nu} = \boldsymbol{J}_{\nu}(\boldsymbol{q})\dot{\boldsymbol{q}}$, where $\boldsymbol{\nu} \in \mathbb{R}^3$ denotes the translational velocity. Weak sense means that the angular velocity $\boldsymbol{\omega} = \boldsymbol{J}_{\omega}(\boldsymbol{q})\dot{\boldsymbol{q}}$ may be non-zero. In case of redundancy ($m < n$) the hyperrectangle is mapped to a space of lower dimension. As a result, the boundary of the Cartesian polytope is defined by less than 2^n vertices because there are internal ones. For a 3R planar robot that performs a translational task ($n = 3, m = 2$), we obtain $2^3 = 8$ vertices in joint space and typically six vertices in Cartesian space which defines the polytope; see Fig. 3.12 (right). To compute the TMP in the so-called *strong sense* (i.e., purely translational motions), denoted by $\mathrm{TMP}_{\mathrm{ss}}$, in case of redundancy, one needs to determine the motor speeds that are part of the null space of $\boldsymbol{J}_{\omega}(\boldsymbol{q})$ and fulfill the velocity constraints at the same time, i.e., $\{\dot{\boldsymbol{q}} \mid \dot{\boldsymbol{q}} \in \mathcal{N}(\boldsymbol{J}_{\omega}(\boldsymbol{q})) \ \& \ |\dot{\boldsymbol{q}}| \leq \dot{\boldsymbol{q}}_{\max}\}$. An algorithm for determining these motor velocities is provided in [238], which was originally proposed for force polytope analysis. The maximum possible velocity (either weak or strong sense) in a certain Cartesian direction $\boldsymbol{u} \in \mathbb{R}^3$ can be determined with the line clipping algorithm proposed in [239], for example. Please note that the strong sense TMP is a subset of the weak sense TMP.

3.6 Safety Map of Stationary Manipulators

3.6.1 Results

The Safety Map representations of the six-DOF PUMA 560 and the seven-DOF LWR IV+ are illustrated in Fig. 3.13. A uniform 5 cm distance between the positions is selected to generate the Cartesian position grid. For the sake of clarity in presentation, only one end-effector orientation is considered. For both robots, the end-effector axes $\boldsymbol{x}_{EE}$, $\boldsymbol{y}_{EE}$, and $\boldsymbol{z}_{EE}$ are aligned with the Cartesian axes $\boldsymbol{x}_0$, $\boldsymbol{y}_0$, and $\boldsymbol{z}_0$ as follows: $\boldsymbol{x}_{EE} = -\boldsymbol{x}_0$, $\boldsymbol{y}_{EE} = \boldsymbol{y}_0$, $\boldsymbol{z}_{EE} = -\boldsymbol{z}_0$. One wants to analyze the reflected mass in the principal motion directions; in Fig. 3.13 $\boldsymbol{u} = \boldsymbol{x}_0$, $\boldsymbol{u} = \boldsymbol{y}_0$, and $\boldsymbol{u} = \boldsymbol{z}_0$ are selected. In addition to the global mass/velocity range, the dynamic properties of both robots are analyzed for a typical (dexterous) workspace area of $60 \times 20 \times 40$ cm size.

For the PUMA 560, the inverse kinematics algorithm [240] is used where an elbow-up and so-called "lefty" configuration are selected in the algorithm's preferences. For the considered problem, 19 837 reachable poses were identified. In Fig. 3.13 (middle row), the accumulated mass/velocity ranges of the robot for translational motions in Cartesian X-, Y-, and Z-direction are illustrated. For the LWR's inverse kinematics algorithm [241], a default elbow-up configuration is selected to resolve the system's redundant DOF. 9138 positions were determined for the entire workspace and 15 null space configurations were computed for each non-singular configuration. In Fig. 3.13 (lower row) the robot's global X, Y, and Z mass/velocity ranges including null space motions are illustrated. Please note that the maximum possible velocity is shown. Of course, the robot can always travel with lower speed, meaning the area below the illustrated mass/velocity ranges is also feasible.

The boundary of the robot's Safety Map representation is mainly defined by singular or near-singular configurations. When the robot approaches the workspace boundary, the reflected mass in the direction of the robot structure becomes very high while the maximum velocity is low. If the robot is stretched out, but the structure does not point in the direction of a Cartesian axis, then the reflected mass in X-, Y-, or Z-direction is in a "normal" range. At the same time, the maximum velocity is still very low due to the

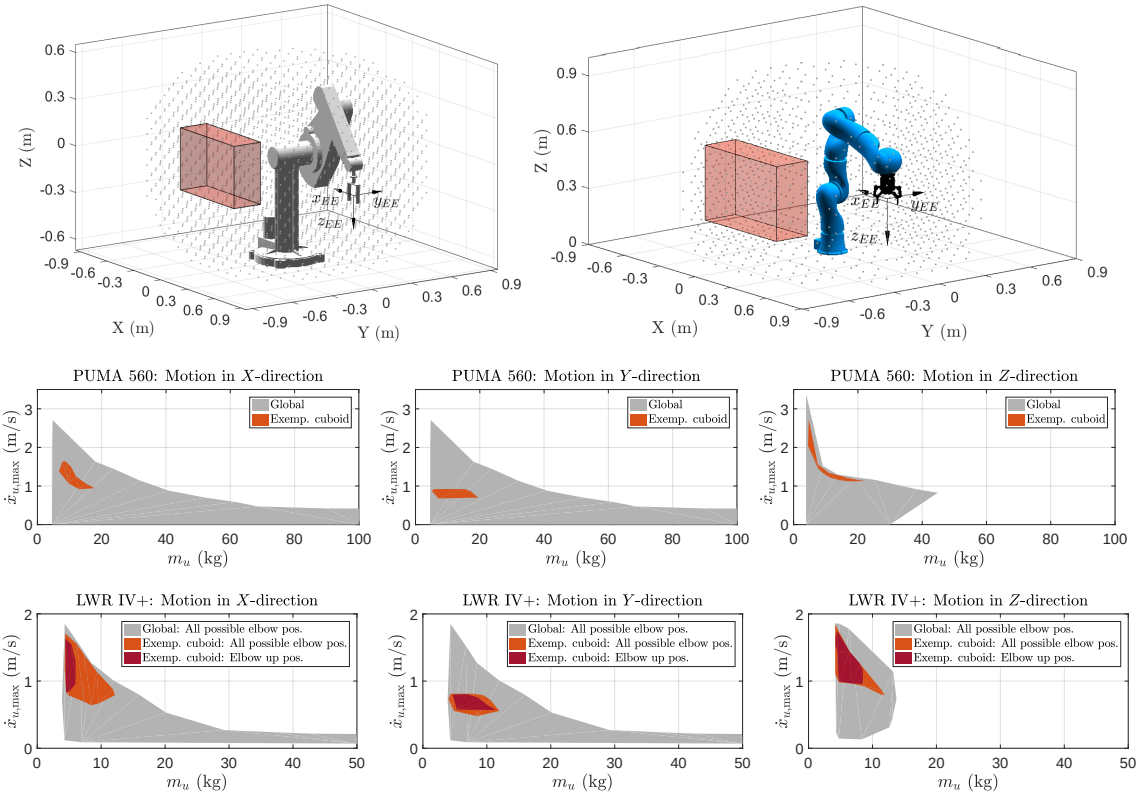
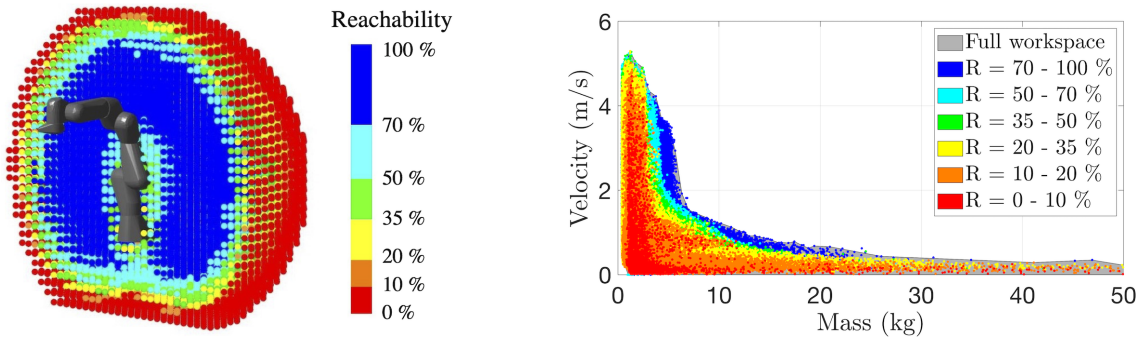


Fig. 3.13: Safety Map representation of the PUMA 560 and the LWR IV+ for motions in Cartesian X-, Y-, and Z-direction. In the upper row, the feasible Cartesian positions for the considered end-effector orientation are illustrated as dots, an exemplary subset of the workspace, a cuboid with $60 \times 20 \times 40$ cm, is colored red. In the middle row, the mass/velocity representation of the entire PUMA 560 workspace and of the exemplary cuboid are depicted. For the LWR IV+ results depicted in the lower row, it is also differentiated between the standard elbow-up configuration and the discretized null space configurations for the exemplary cuboid. © 2018 IEEE [208].

robot configuration being singular. High velocities can be achieved either in singular or non-singular configurations. The results for the exemplary cube indicate the reflected mass and maximum velocity ranges that can be expected in a typical workspace area. These results will be later used in the use case described in Sec. 3.8.

For the Franka Emika Panda, it is shown in Fig. 3.14 how the reachability map can be linked to the Safety Map. To generate the Panda's reachability map, a uniform Cartesian position grid with 5 cm distance and 20 equally distributed $SO(3)$ end-effector orientations were selected for each position. The built-in inverse kinematics algorithm of Matlab 2020b was used to check the reachability of each pose, where both self-collisions and the joint position limits were accounted for. The robot's weak sense polytope was used to determine the achievable Cartesian velocities. The colors in Fig. 3.14 (a) represent the relative number of reachable poses for every Cartesian position. The accumulated mass/velocity range for a certain reachability index is depicted in Fig. 3.14 (b). This shows how the Safety Map can be linked not only to certain workspace areas but also to performance metrics.



(a) Reachability Map **(b)** Safety Map

Fig. 3.14: Reachability map (a) and Safety Map (b) of the Franka Emika Panda. In the reachability map, the colors indicate the reachability index, i.e., the relative number of end-effector orientations the robot can reach for every workspace position. The reachability index ranges from 0 % (red) to 100 % (blue). In (b), the reachability indices are linked to mass/velocity ranges in the Safety Map.

3.6.2 Influence of Payload

Previously, the robot mass/velocity range was calculated for the robot flange or TCP. When attaching an end-effector/payload to the system, its mass and inertia influence the robot's inertial properties, the geometry the maximum achievable Cartesian endpoint velocity. Ideally, one would like to compute the robot's Safety Map representation only once and then shift/transform the mass/velocity range according to the specific tool parameters with only little computational effort.

When expressed w.r.t. the operational point, the overall kinetic energy matrix is given by the sum of the kinetic energy matrices of robot and end-effector [243]. The reflected mass at the load in direction $\boldsymbol{u}$ increases with the specific load mass and inertia. If the flange (partially) rotates, the tool geometry may influence the operational speed and the maximum reachable velocity. An example is provided in Fig. 3.15, where the influence of a payload on the reflected mass and operational velocity (weak and strong sense) of the PUMA 560 is illustrated. The load mass is $m_L = 2$ kg, the inertia tensor about the main axis is $\boldsymbol{I}_L = \text{diag}\{0.2, 0.3, 0.4\}$ kgm^2, and the location of the center of mass w.r.t. the robot flange is $\boldsymbol{r}_{L,\text{COM}}^{\mathsf{T}} = [5, 10, 15]$ cm.

The figure shows that the strong sense velocity (red) remains the same after adding the payload, whereas the maximum possible weak sense velocity increases (blue). In both cases, the reflected mass increases after adding the payload. A thorough analysis on this topic, particularly the efficient calculation of the Safety Map update, is subject to future work.

3.6.3 Remark on ISO/TS 15066 Reflected Mass Model

In the current ISO/TS 15066, the simplified model

$$m_u = \frac{M}{2} + m_L \,, \tag{3.5}$$

is used to determine the robot reflected mass. Here, M is the summed mass of all the moving parts, and the m_l is the payload. In contrast to the theoretically correct model proposed in [181], (3.5) neither depends on the robot configuration nor on the Cartesian

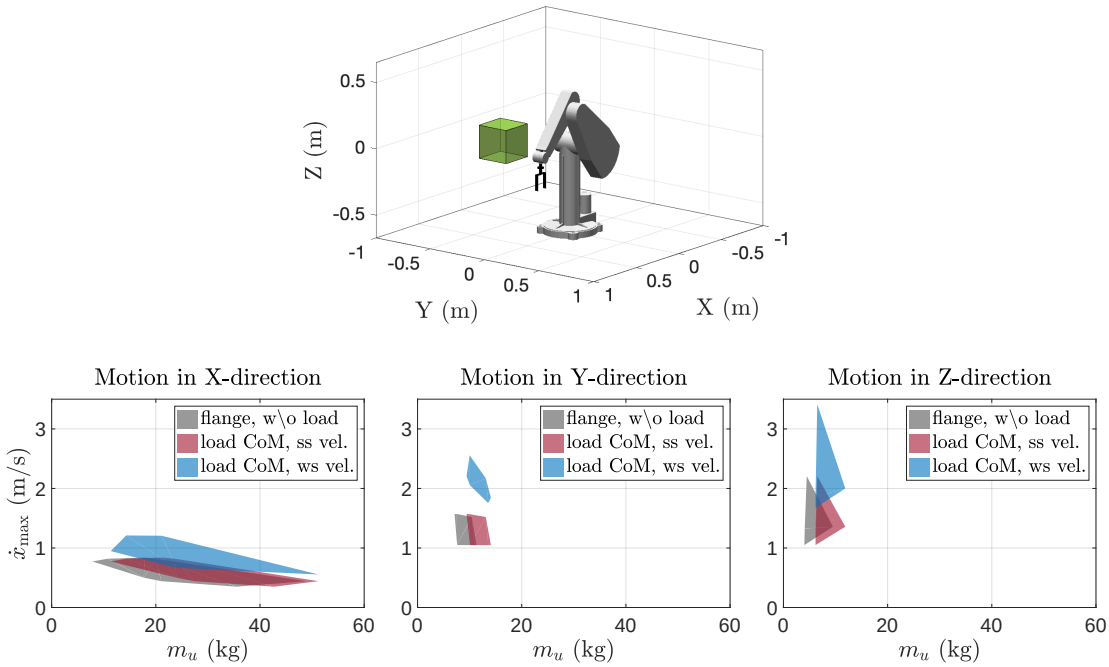


Fig. 3.15: Influence of a payload on the robot's endpoint reflected mass and maximum velocity. The results for the flange are shown in gray, for the load in blue (weak sense) and red (strong sense). © 2019 IEEE [242].

direction $\boldsymbol{u}$. To analyze whether the ISO model still provides a reasonable estimation of the reflected mass for the LWR IV+ and the Franka Emika Panda, the mass obtained by (3.5) is compared to the robot's workspace effective mass distribution. In Fig. 3.16, the relative number of robot positions associated to a certain effective mass range and the reflected mass obtained by (3.5) (LWR: 6.3 kg, Panda: 5.45 kg) are shown. Like in the previous Safety Map results, only one end-effector orientation is considered for the LWR. The flange is pointing downwards, with the end-effector frame being axis-aligned with the world coordinate frame. For Panda, 20 uniformly distributed Cartesian directions are considered for each position. No payload is taken into account. For the LWR, it can be observed that in approx. 60 % of the reachable workspace, the reflected mass is lower than the one obtained by ISO/TS; see Fig. 3.16 (a). For the Panda, the actual reflected mass is lower than the simplified ISO/TS estimate in 97 % of the workspace; see Fig. 3.16 (b). This implies that implementation of the SMU (which is also part of the ISO/TS 15066) using Khatib's reflected mass model typically outputs higher safe velocities than an implementation based on the ISO model. The ISO model thus deteriorates the performance of pHRI applications in terms of cycle time in many cases. However, in Fig. 3.16 it is also shown that the model underestimates the reflected robot mass in some workspace areas, which may lead to potentially hazardous contact situations. To sum up, a reasonably accurate estimation/calculation of the robot reflected mass (possibly supported by measurements [244]) is inevitable for enabling safe and efficient HRI. The current ISO model (3.5) can be regarded as over-simplified and needs to be updated.

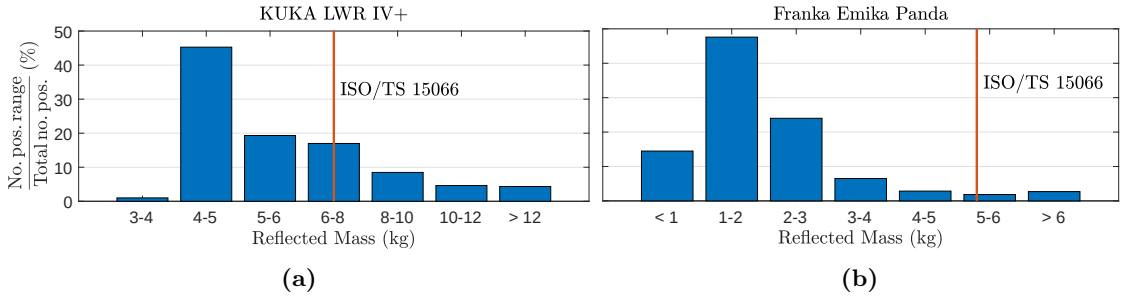


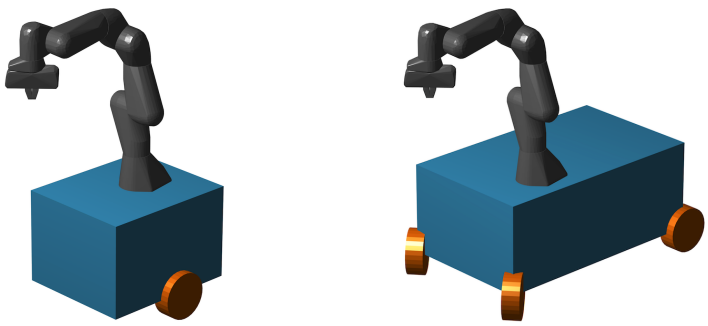
Fig. 3.16: LWR IV+ and Panda: Comparison of workspace reflected mass distribution and effective mass obtained by ISO/TS 15066. © 2021 IEEE [244].

3.7 Safety Map of Wheeled Mobile Manipulators

In the previous section, the safety characteristics of stationary robots were investigated. In the following, the achievable Cartesian reflected mass and velocity range of two common types of mobile platforms, i.e., a differential drive and car-like vehicle, and the combination of these platforms with a seven-DOF Franka Emika Panda (see Fig. 3.17) is analyzed. Both the principal and the quantitative effects of the platform parameters on the mobile robot's collision safety are investigated for four collision scenarios; see Fig. 3.18. Finally, the mass/velocity data is processed towards the Safety Map, which allows assessing the safety performance of these systems and comparing them to stationary manipulators, elastic joint robots (cf. Chapter 4), or other mobile systems, for example.

3.7.1 Considered Systems

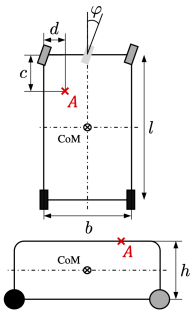
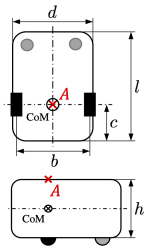
The differential drive and the car-like wheeled platforms that are shown in Tab. 3.2 together with their parameters are considered in this work. The drive wheels (black) of the car-like vehicle are coupled and thus rotate at the same speed; the angles of the (also coupled) steering wheels govern the vehicle's steering angle φ. In the differential drive vehicle, the drive wheels rotate independently from each other. Both platforms satisfy the conditions of pure rolling and non-slipping. At the location A on the surface of the vehicles, the seven-DOF Franka Emika robot (total weight of 18 kg) is mounted; see Fig. 3.17. The



(a) Differential drive **(b)** Car-like

Fig. 3.17: Considered mobile manipulators consisting of a differential drive (a) and a car-like (b) platform and the Franka Emika Panda.

Tab. 3.2: Considered platforms and their parameters. The active wheels are shown in black, the passive wheels in gray. The location of the manipulator base is indicated by A.

Car-like	Parameters	Differential drive	Parameters
	**Wheels (active & passive)** Mass: 2 kg Radius: 0.1 m Steering angle φ: ± 25 ° Max. translational velocity: 1.666 m/s Inertia about axis of rot.: 0.005 kgm^2 **Platform body** Dimensions: $b = 0.6$ m $c = 0.25$ m $d = 0.15$ m $l = 1$ m $h = 0.4$ m Mass: 50 kg Inertia about center of mass: 5.7 kgm^2		**Active wheels (passive wheels neglected)** Mass: 5 kg Radius: 0.065 m Max. translational velocity: 1.666 m/s Inertia about axis of rot.: 0.006 kgm^2 **Platform body** Dimensions: $b = 0.5$ m $c = 0.25$ m $d = 0.55$ m $l = 0.75$ m $h = 0.393$ m Mass: 124 kg Inertia about center of mass: 8.94 kgm^2

combination of platform and manipulator forms the mobile manipulator. The kinematics and dynamics of the coupled system are described in Chapter 2.

3.7.2 Collision Cases

The following four collision cases (see Fig. 3.18) are considered, which are relevant in real-world applications:

Case a) When the mobile robot is transporting goods, typically the vehicle is moving while the manipulator is at rest[4]. In this case, the platform may collide against the human, typically at the lower extremities.

Case b) The mobile platform is at rest; the manipulator is moving. This case typically occurs when the mobile robot stops at a workstation and manipulates objects there. For this case, collisions with the upper body are more likely to occur than collisions with the lower body.

Case c) Like a) with the difference that not the vehicle, but the manipulator may collide with the human, typically against the upper body.

Case d) If the vehicle and the manipulator move simultaneously, then the system can exploit its maximum performance in terms of achievable velocity. This can be required to perform explosive motions such as the throwing of objects, for example.

3.7.3 Derivation of Reflected Mass and Velocity Range

The general procedure for determining the achievable mass and velocity range is the same for all cases a) - d). The different scenarios can be examined by narrowing down the system parameters accordingly. The cases can be distinguished between 1) the contact location(s) and 2) whether the two subsystems vehicle and manipulator are (actively) moving or not. The procedure for deriving the Safety Map representation is as follows.

[4]In terms of instantaneous collision dynamics, there is no difference between a moving and a resting manipulator in case a).

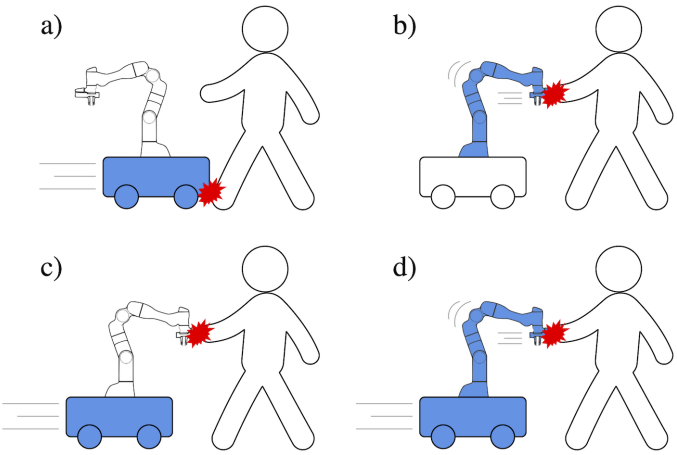


Fig. 3.18: Possible collisions between a human and a mobile robot. In case a) the platform collides against the human, in b) - d) the manipulator against the human. In a) the platform and potentially also the manipulator moves actively, in b) only the manipulator, in c) only the platform, and in d) both the manipulator and the platform.

Step 1 Select the point(s) of interest (POI) on the robot structure. In case a) six POI on the vehicle body are considered, namely the four corners and the center in the front and back. In case b) - d), the manipulator end-effector is the only POI. The POI(s) define the contact Jacobian matrix that is required to determine the reflected mass and the velocity polytope described in the previous section and Chapter 2.

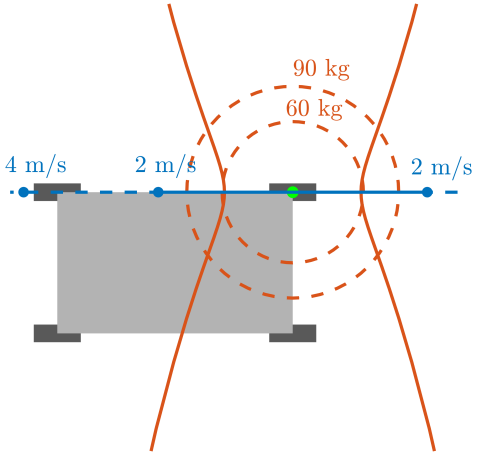
Step 2 Set the velocity constraints. In case a) and c), the manipulator is at rest, i.e., the velocity of every joint is zero. The car-like platform can travel at a velocity within its speed limits; in the differential drive, this holds for both (independent) wheels. In case b) the platform velocity is zero, the achievable Cartesian velocity is governed by the manipulator's velocity polytope described in Sec. 3.5. In case d), both the vehicle and the manipulator contribute to the achievable end-point velocity.

Step 3 Sample the robot configurations for the desired parameter range and discretization[5]. For each POI, all combinations of (discretized) platform parameters and the robot's reachability map are determined. In this thesis, also the ISO 9283 cube is investigated.

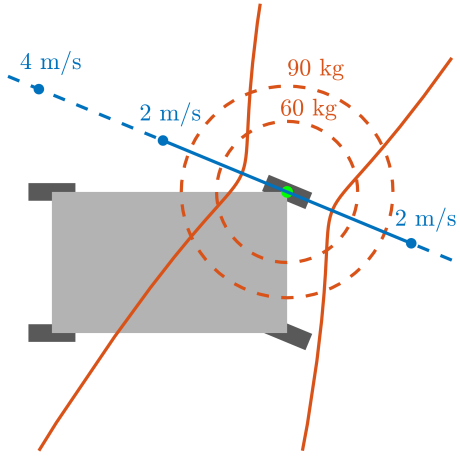
Step 4 Evaluate the reflected mass and velocity range for every configuration and in the possible directions of motion. For collisions with the platform, the direction of motion can be constrained by the vehicle's motion constraints. In contrast, the manipulator can typically move in all (discretized) Cartesian directions if the configuration is non-singular.

Step 5 Accumulate the calculated mass/velocity data in the Safety Map and relate the robot representation to injury data and/or other robots' mass/velocity range.

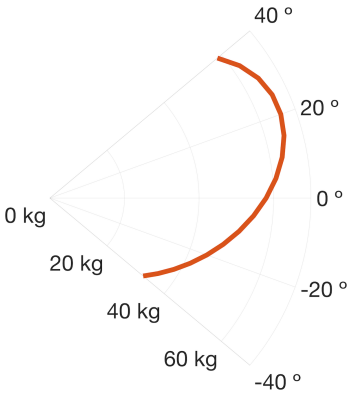
[5]The discretization of the robot's workspace and the end-effector orientation is the same as in the previous section. For the car-like vehicle, $\varphi \in [-25, 25]$ ° steering wheel angle with 5 ° increment is selected. For the differential drive, the directions on the unit circle used to evaluate the reflected mass and velocity have 5 ° increment.



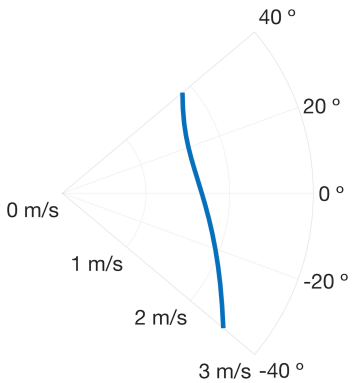
(a) Reflected mass and velocity range for $\varphi = 0\,^\circ$ steering angle.



(b) Reflected mass and velocity range for $\varphi = 25\,^\circ$ steering angle.



(c) Reflected mass in the direction of motion over steering angle, $\varphi \in [-40, 40]\,^\circ$.



(d) Maximum velocity in the direction of motion over steering angle, $\varphi \in [-40, 40]\,^\circ$.

Fig. 3.19: Achievable reflected mass (red) and velocity (blue) for the car-like vehicle at the center of the top right wheel (green circle). In (a) and (b), the reflected mass belted ellipsoid and the achievable velocity range are illustrated for $\varphi = 0\,^\circ$ and $25\,^\circ$ steering wheel angle. In (c) and (d), the reflected mass and maximum velocity in the direction of motion direction are shown for the range $\varphi \in [-40, 40]\,^\circ$ in steering wheel angle for the considered point of interest (green circle).

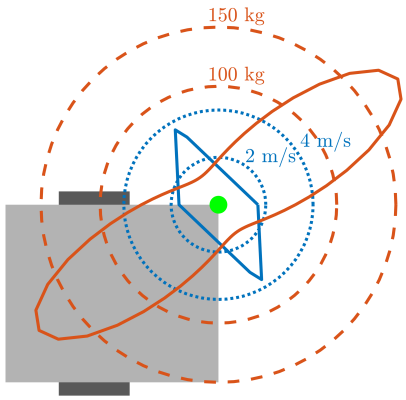


Fig. 3.20: Achievable reflected mass (red) and velocity (blue) at the top right corner (green circle) of the differential drive vehicle.

3.7.4 Results

In this section, the results for cases a) - d) are provided. For each case, the qualitative and quantitative effects of different robot parameters on the reflected mass and endpoint velocity are described. The accumulated mass/velocity data is summarized in the Safety Maps illustrated in Fig. 3.22. There, also the current ISO/TS 15066 thresholds for the considered body parts are shown. However, please note that these thresholds only serve as an example. The goal of this work is to derive the safety-relevant robot dynamic properties, which can be employed for impact mitigation schemes and safety assessment of real-world use cases in future work.

Case a)

In case a), the platform collides against the human, typically at the lower extremities. In the following, first, the achievable mass/velocity range of the two vehicles is investigated without an attached manipulator, thereafter the effect of the manipulator on the safety characteristics of the coupled system.

Vehicle only For the car-like vehicle, the reflected mass and velocity for a certain steering wheel angle and POI are illustrated in Fig. 3.19 (a) and (b). For $\varphi = 0\,°$, the motion is purely translational, the reflected mass in the direction of motion (along blue line) equals the platform mass. For $\varphi = 25\,°$, the motion is both translational and rotational; the reflected mass is slightly smaller and also governed by the platform inertia. For directions other than the direction of motion, it can be observed that the reflected mass becomes very large (perpendicular to the wheel, even infinite), which is due to the vehicle's motion constraints. However, this work focuses on the reflected mass and velocity range that the robot can achieve in the direction of motion. This analysis is provided in Fig. 3.19 (c) and (d), where the dependency of both quantities on the steering wheel angle is shown. Results for the differential drive are illustrated in Fig. 3.20. Those POIs on the platform that are not located on the axis of rotation of the active wheels can move in every direction by varying the velocities of the two wheels. For rotational motions, POIs close to the wheels typically reach relatively low velocities while the reflected mass is high. Distal POIs (see Fig. 3.20) can reach high velocities but relatively low reflected mass in the direction of

motion. This is also reflected in the Safety Map representations of both systems, which are depicted in Fig. 3.22 (left column, red area).

Combination of Vehicle and Manipulator Mounting a manipulator on the mobile platform adds extra mass and inertia to the system. The increase in reflected mass can be observed in Fig. 3.22 (left column, red vs. blue area), the achievable operational velocity remains the same.

Case b)

In case b), only the manipulator moves and possibly collides against the human while the vehicle is at rest. In Fig. 3.22 (second column), it can be observed that the difference between stationary manipulator (red) and mobile manipulator (blue) in terms of achievable mass/velocity range is negligible. As the platform is at rest, there is no change in operational velocity, but the platform's inertial properties affect the reflected mass at the end-effector. However, the reflected mass of the Franka Emika Panda (and other tactile or collaborative robots) is mainly governed by the last few links, which is why the extra degrees of freedom provided by the platform have only little influence on the reflected mass. This holds for the two considered platforms with 50 and 124 kg mass, but also for a broader platform mass range; cf. [181]. For simplicity, one can therefore neglect the platform in calculating the reflected mass for non-singular configurations, which is also conservative from a safety point of view.

Case c)

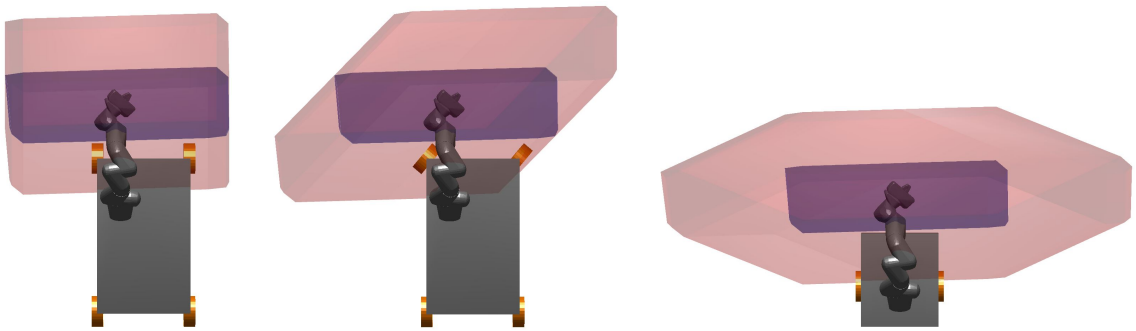
In both cases a) and c), only the vehicle is actuated. In case c), however, the manipulator acts as a lever arm during rotational motions (especially in stretched-out configurations), which is why the achievable translational velocity at the end-effector is higher in case c) than the one in case a). In case c), the reflected mass range at the manipulator's end-effector is the same as in case b).

Case d)

In case d), both the platform and manipulator are moving. In terms of a potential collision with the manipulator's end-effector, d) is the general case, i.e., the achievable velocity and reflected mass ranges obtained in cases b) and c) are subsets of the one obtained in d). In this case, the maximum possible velocity is comparatively high because the vehicle and manipulator speed sum up. In Fig. 3.21, it is shown how the velocity polytope of the stationary manipulator (purple) increases when a mobile platform is added.

3.8 Applications

This section describes how the Safety Map can be integrated into the robot and task design workflow as a safety evaluation and optimization tool as well as using it to generate safe trajectories. The given examples regard only stationary robots; however, they can be applied or extended to other robot types quite straightforward.



(a) Car-like, $\varphi = 0\ ^\circ$ **(b)** Car-like, $\varphi = 40\ ^\circ$ **(c)** Differential drive

Fig. 3.21: Cartesian velocity polytope for two types of mobile robots (red) compared to the polytope of the stationary manipulator (purple). In (a) and (b), the influence of the steering angle on the velocity polytope is shown for the car-like platform; the differential drive is illustrated in (c).

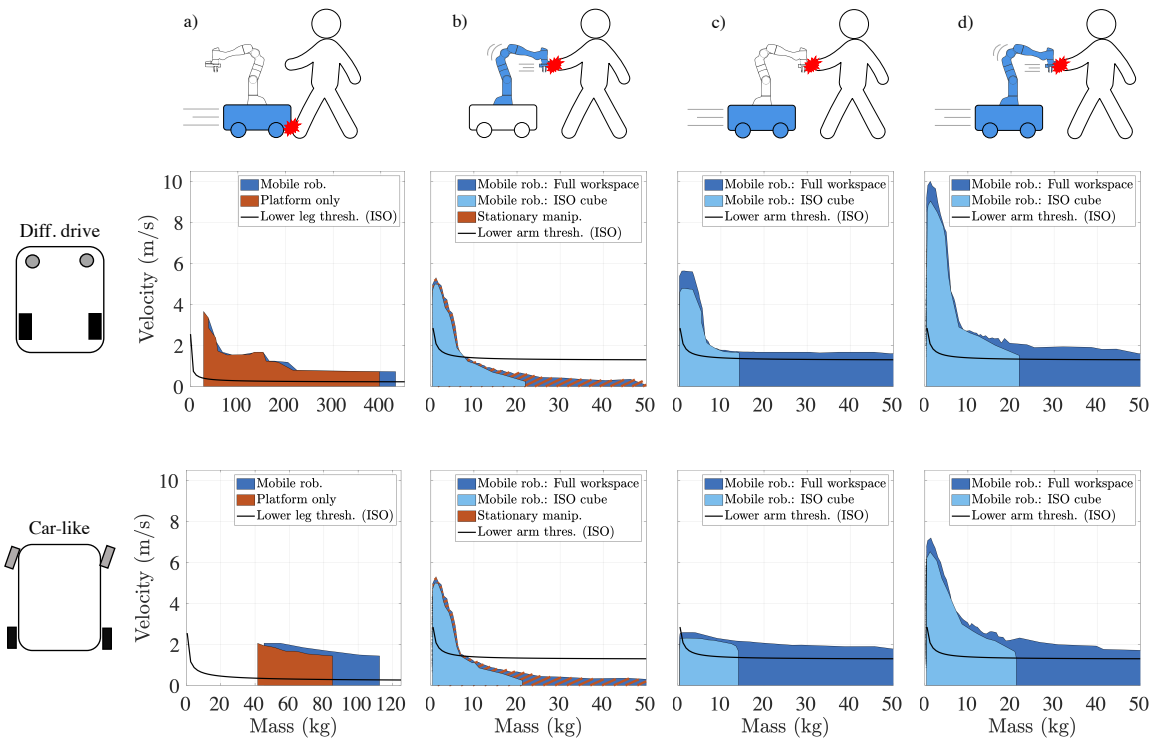


Fig. 3.22: Safety map representations for two types of mobile platforms including a Franka Emika Panda and the four collision cases a) - d) mentioned above. The dark blue area represents the achievable mass/velocity range of the mobile robot, where the entire workspace of the manipulator is considered. The mass/velocity range for Panda's ISO cube is illustrated in light blue. For collision case a) the mobile robot (blue) is compared to the platform without attached manipulator (red), for case b) the mass/velocity range of both the mobile robot (blue) and the stationary manipulator without platform (red) are illustrated for comparison. For collisions with the end-effector, the lower arm injury threshold from ISO/DIS 10218 is shown, for impacts with the platform the threshold on lower leg injury is depicted.

3.8.1 Safety Assessment

Global Suppose one is interested in the robot's global safety performance without having a specific application at hand. In that case, the Safety Map can be utilized to analyze, e.g., whether a) the robot is in principle capable of producing a specific type of injury, b) where the most dangerous areas in the reachable workspace are located, or c) how the safety properties compare with other performance indices such as manipulability/dexterity in certain workspace areas (see Panda results in Fig. 3.14). Please note that the framework allows for different degrees of granularity in the safety assessment. The robot dynamic properties can be calculated either with fine or coarse grids, which makes short iterations in robot design and safety evaluation possible.

Task-Oriented To assess certain robot applications in terms of safety, the following steps have to be carried out to determine the Safety Map:

Step 1 Extract the task-dependent mass/velocity data (workspace area or trajectory) from the global robot dynamic properties.

Step 2 Assign contact primitives with their parameters to points of interest on the robot structure (usually the end-effector).

Step 3 Identify collision scenarios (constrained/unconstrained) and human body parts that may be hit during collisions by analyzing the shared workspace.

Step 4 Select the corresponding injury data and relevant thresholds from the current standards.

This procedure is now applied to a pick and place task with two robots. The PUMA 560 and LWR IV+ shall perform translational motions (strong sense) in the Cartesian X-, Y-, or Z-direction in the exemplary cuboid depicted in Fig. 3.13 (upper row, same workspace

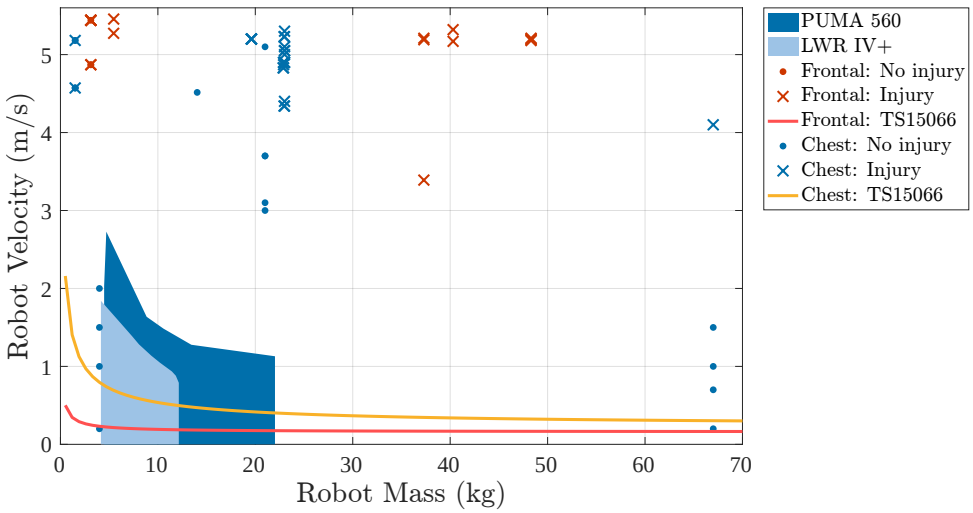
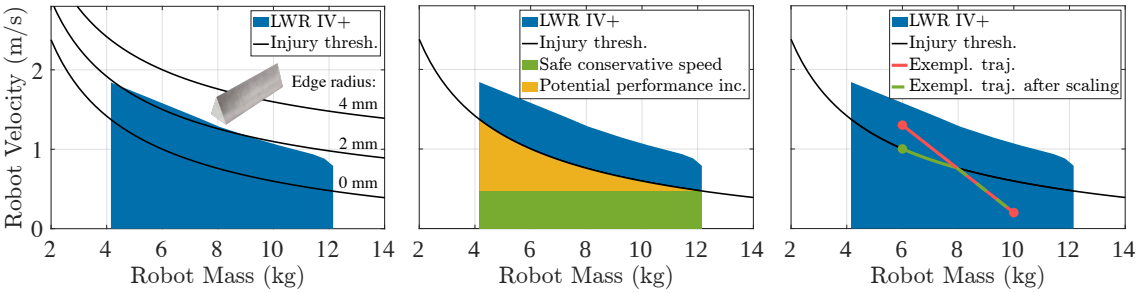


Fig. 3.23: Safety Map for a pick and place use case. The mass/velocity range of the PUMA 560 and LWR IV+ for the considered workspace area are illustrated in dark/light gray, head and chest biomechanics injury data in red and blue. © 2018 IEEE [208].



(a) Influence of contact geometry parameters on the injury threshold

(b) Safe conservative speed for considered workspace area (green) and potential performance increase (yellow).

(c) Exemplary trajectory before (red) and after scaling (green) to biomechanically safe velocity via SMU.

Fig. 3.24: Applications of the Safety Map. The blue area represents the LWR IV+ mass/velocity range for the cube depicted in Fig. 3.13, the injury threshold is fictitious.

for both robots). In Fig. 3.23, the combined robot mass/velocity ranges are illustrated for both robots. The robot data is compared to injury data for blunt, unconstrained impacts against the human frontal bone (see Fig. 3.7 (left)), blunt chest injury data (see Fig. 3.7 (right)), and the current ISO/TS 15066 thresholds. Please note that the TS 15066 relates force/pressure limits (assuming constantly 1 cm^2 contact area) to thresholds in the mass/velocity plane via a (over-)simplified contact model; cf. Sec. 3.6.3. For the considered task, the PUMA 560 can reach higher velocities than the LWR, and also the reflected mass is typically larger. In Fig. 3.23, the ranges of achievable mass and velocity for both robots remain well below the critical values of the biomechanics data[6]. The fact that the considered types of injury are unlikely to occur for these robot mass/velocity ranges was already shown in [15]. However, to the best of the author's knowledge, the global systematic analysis and comparison of robot mass and velocity range and the relation to real biomechanics injury data has not been made until now. While no severe injury is to be expected according to the well-established biomechanics data, the TS 15066 threshold can be violated by both robots.

3.8.2 Safe Mechanism Design

After the map representations of different robots and injury of different human body parts were developed and compared, the next question is how the information provided by the Safety Map can be used to ensure or improve safety and performance in a particular application. The long-term research goal is to systematically integrate the Safety Map into the (partially automatic) robot and task design process, which is subject to future work. In this thesis, illustrative examples are given that show how the Safety Map can be utilized for ensuring safety; see Fig. 3.24. For this, the LWR and the same workspace area that was used previously are considered.

As human injury severity is usually related to the contact geometry, one can analyze how robot/tool surface modifications affect safety. In Fig. 3.24 (a), three fictitious safety curves for an edge-shaped impactor are illustrated. Each safety curve represents the injury

[6]Please note that raw biomechanics data is illustrated without classification in terms of injury/no injury in terms of a safety curve.

threshold of a certain edge radius. The larger the edge radius, the safer the contact. For 0 mm edge radius, the injury threshold intersects the LWR's map representation, which means that this safety constraint needs to be taken into account when planning the task. For 4 mm edge radius, the considered type of injury can be excluded, which means that the robot can always travel at maximum speed without further safety measures being necessary. However, such mechanical modifications are not possible or wanted in every industrial application.

3.8.3 Safe Motion Generation

In Fig. 3.24 (b), the safety curve is evaluated at the maximum robot reflected mass that the robot can reach in the considered workspace area. The resulting velocity ensures safety for any path within the workspace area; see Fig. 3.24 ((b), green). However, this simple selection might deteriorate performance because the full performance potential in terms of achievable safe velocity is not exploited; see the yellow area in Fig. 3.24 (b). If a robot trajectory was planned already, then the Safety Map provides an intuitive visualization of whether the trajectory is safe or not. For example, in Fig. 3.24, the initial trajectory is represented by a red line. Part of this line is above the injury threshold, which means that the safety constraint is violated. The green line represents the trajectory that was scaled in time to satisfy the constraint. Technically, this time scaling can be done without the Safety Map by checking the compliance with the safety constraint along the trajectory and reducing speed when necessary. The following paragraph describes how the Safety Map can also be utilized also for safe trajectory planning.

Especially in industrial settings, it is desired to execute a task safely and as fast as possible. The quantities computed for the robots' Safety Map representation, i.e., the workspace grid, the Cartesian velocity and reflected mass range, and the safety constraints provide all the essential information needed to plan safe and time-efficient trajectories. In the following, a basic example is provided that shows how such trajectories can be generated. The goal is to compute a minimum-time trajectory from a start to a goal pose that takes the safety constraint into account given by a safety curve. Consider a planar, gravity-free 2R robot; see Fig. 3.25. The length of each robot link is 0.5 m, at the distal end of each link a 2 kg point mass is located. The distance between the robot's workspace grid positions is 10 cm; see Fig. 3.25. For every grid position, a corresponding joint configuration is determined via the robot's inverse kinematics. Then, for each position, the maximum safe velocity in the direction of the adjacent grid positions is determined via the safety curve depicted in Fig. 3.25 (upper right). These velocities are represented by gray lines in the workspace grid. The longer a line, the higher the maximum safe velocity in the considered direction. For the sake of simplicity, joint-space acceleration, velocity, and position constraints are omitted. Provided the distances and the speed limitations between the grid positions, and assuming that the velocity is constant when traveling from one position to another, we may now determine the time required to traverse through the grid. This information can be used to find a time-optimal path from start to goal. This is done with the well-known A* algorithm [245]. In Fig. 3.25, the red line represents the minimum-time solution obtained via B-spline interpolation. It is compared to the minimum-distance trajectory, which is shown in blue. Both trajectories are subject to the safety-curve constraint, i.e., safety is ensured at every time instant. The bottom right figure shows that the minimum-time trajectory is faster than the minimum-distance trajectory and results

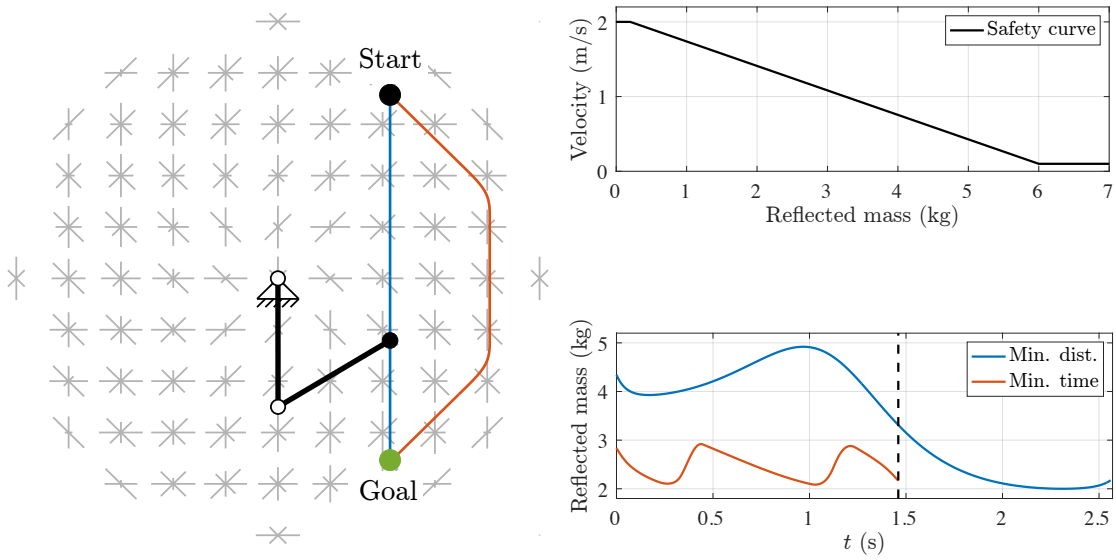


Fig. 3.25: Planning of safe, minimum-time trajectories with the Safety Map. The workspace grid of a 2R planar robot is shown on the left. For every grid position, the maximum safe velocity in the direction of the adjacent positions is determined via the safety curve depicted in the upper right plot. These velocities are represented by gray lines in the workspace grid (left). The longer a line, the higher the safe velocity in the considered direction. The blue line represents the minimum-distance trajectory from start to goal, planned via the A* algorithm; the red line represents the minimum-time trajectory. Both trajectories are subject to the safety-curve constraint, i.e., safety is ensured at every time instant. The bottom right figure shows that the minimum-time trajectory is faster than the minimum-distance trajectory and results in lower reflected mass.

in lower reflected mass.

This simple example shows how the Safety Map can be used not only to assess safety but also to generate safe motions. This example regarded only translational motions of a planar, non-redundant 2R robot. Future work could address the extension from 2D to 6D (translational and rotational motion), redundant robots, the avoidance of obstacles, and real-time trajectory generation.

3.9 Summary

In this thesis, the Safety Map concept was proposed, a global map that serves as a common unified representation for injury biomechanics data and robot collision behavior. The Safety Map is a novel tool for robot developers that can be utilized for injury analysis and safer robot design already at an early concept phase of the design and development process. The Safety Map enables the user to address the following (among other) questions:

- Is the considered robot capable of producing a particular type of injury during unforeseen collisions in my application?
- Where are the most dangerous areas in the reachable robot workspace?
- How do the robot safety characteristics compare with other performance indices? For example, how dangerous is the robot in its most dexterous workspace?

- How does the robot compare to other robots in terms of safety?

The mass and velocity range of the entire robot workspace, or task-dependent subspaces, can be quantitatively compared to any available injury data for different contact primitives, collision cases, and human body parts. This gives the designer clear information on which kind of injury is most likely to occur during operation, thus guiding the hardware design process and giving valuable information to safe interaction control and motion planning algorithm development. In fact, the Safety Map can also be directly employed as a cost map for robot safety-oriented motion planning or as a global cost function for optimal control. In this work, the approach was validated using the dynamics of the six-DOF PUMA 560 and the seven-DOF LWR IV+ and Franka Emika Panda. The Safety Map representation was determined for all robots and related to biomechanics injury data, which was classified, validated, and processed during a thorough biomechanics literature survey. Furthermore, the framework was extended to mobile manipulators that consist of a mobile platform and a Franka Emika Panda. The principal and the quantitative effect of the system parameters on collision safety was investigated for four use cases relevant to real-world applications. Finally, examples showed how the Safety Map could be used for safety assessment and optimization as well as trajectory generation.

CHAPTER 4

Speed Gain in Robots with Intrinsic Joint Elasticity

Many robotic systems with intrinsic joint compliance have been developed in recent years, e.g., legged robots, hands, prostheses, manipulators, and humanoids. The motivation for deliberately introducing joint elasticity in manipulators is to improve mechanical robustness, safety in human-robot interaction, and the inherent capability to store and release energy, which can be utilized to outperform rigid robots in terms of energy efficiency and peak velocity. In this chapter, the (configuration-dependent) maximum achievable endpoint velocity of elastic joint robots is investigated, an important characteristic for assessing and optimizing the performance and safety properties of a robot design. In Sec. 4.1, the set of reachable velocities and deflections is derived analytically for a 1-DOF visco-elastic joint considering important real-world constraints on the motor velocity and elastic deflection. Furthermore, the time-optimal trajectories for all reachable states are determined. In Sec. 4.2, the 1-DOF results are extended to n-DOF manipulators with compliant actuation. Several hypotheses are made on how and to what extent the motor velocity and the elastic energy can be exploited and converted to link kinetic energy. The hypotheses are exemplified using a 3R planar robot. For verification, the theory is compared to the optimal control solution. Here, a reasonable agreement in terms of maximum achievable TCP velocity is observed, and it is shown that the computation time for the proposed estimation methods is orders of magnitude lower than for the one of a state-of-the-art optimal control solver. It is also shown how the results can be used to accelerate the computation of the optimal control solution. Also, the results of the presented approach are compared with those from real-world throwing experiments, which were recently conducted on the elastic DLR David system [8]. Finally, the methods are applied to derive and quantitatively compare the safety properties of DLR David and a hypothetically rigid version of this robot in terms of the Safety Map framework described previously.

4.1 1-DOF Visco-Elastic Joint: Speed Gain and Time-Optimal Control

In the robotics literature, many authors investigated the optimality principles for 1-DOF visco-elastic and variable impedance joints [17,52,166–169]. Optimal excitation trajectories and the influence of parameters such as mass, nonlinear/variable stiffness, and damping were derived for different types of oscillators. However, typically only the constraints on the system's inputs (motor velocity or torque) have been considered. In the author's previous work [52], the maximum possible velocity was determined for an elastic joint with constant stiffness and limited elastic deflection, which is typically the most important constraint of the elastic mechanism. This research is followed up here. For the case of a 1-DOF visco-elastic joint with limited deflection and motor velocity, the maximum possible velocity is determined depending on the system parameters and constraints. Furthermore, the more general problem of determining all system states (velocity and deflection) that can be reached from equilibrium and the generation of the corresponding time-optimal trajectories are addressed.

4.1.1 Problem Definition

The dynamics of the considered visco-elastic joint (see Fig. 4.1) can be expressed as

$$M\ddot{q} = \tau_g + \tau_{\text{ext}} - \tau_J \,, \tag{4.1a}$$

$$B\ddot{\theta} = \tau_m - \tau_f - \tau_J \,, \tag{4.1b}$$

$$\tau_J = K_J(\theta - q) + D_J(\dot{\theta} - \dot{q}) \,, \tag{4.1c}$$

where $B \in \mathbb{R}$ is the motor inertia, $\theta \in \mathbb{R}$ the motor position, $D_J \in \mathbb{R}$ the constant joint damping, $K_J \in \mathbb{R}$ the joint stiffness, $M \in \mathbb{R}$ the link inertia, $q \in \mathbb{R}$ the link position, and $\tau_m, \tau_f, \tau_g, \tau_{\text{ext}} \in \mathbb{R}$ the motor, motor friction, link gravity, and external torques, respectively. For the sake of simplicity and clarity of the analysis, $\tau_f \approx 0$ and $\tau_{\text{ext}} \approx 0$ are assumed. Furthermore, the joint is considered to move horizontally, i.e., $\tau_g = 0$. The following symmetric constraints on the spring deflection and motor velocity are taken into account

$$|\varphi| = |\theta - q| \leq \varphi_{\max} \,, \tag{4.2a}$$

$$|\dot{\theta}| \leq \dot{\theta}_{\max} \,. \tag{4.2b}$$

The maximum elastic deflection depends on the spring energy storage capacity and usually on the selected stiffness setup if the stiffness can be adjusted. In the DLR FSJ joint [122], e.g., a stiff setup allows for a lower maximum deflection than a soft stiffness setup by

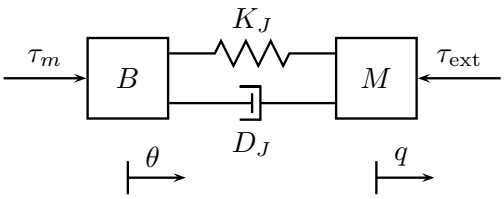


Fig. 4.1: 1-DOF visco-elastic joint.

design. In some VSA actuators, $\varphi_{\max}$ is constant, irrespective of the selected stiffness. The potential energy stored in an elastic joint is given by

$$U_S(\varphi) = \int_0^{\varphi} \tau(\tilde{\varphi})\ d\tilde{\varphi}\,. \tag{4.3}$$

Typically, not the total potential elastic energy can be utilized for energy storage and release. The available energy can be limited by a) spring pretension (e.g., in antagonistic setups), b) the stiffness setup, and c) the maximum possible elastic deflection. Here, it is assumed that the motor can apply the maximum velocity (4.2b) under all possible operating conditions. As described in Sec. 2.1, one can bring the dynamics into singular perturbation form when assuming that the motor dynamics are significantly faster than the link side dynamics. The reduced dynamics are found to be

$$\ddot{q} = 2D\omega(\dot{\theta} - \dot{q}) + \omega^2(\theta - q)\,, \tag{4.4a}$$

$$\theta = \int \dot{\theta}\ dt + \theta_0\,, \tag{4.4b}$$

with θ_0 being the initial motor position, $\omega = \sqrt{K_J/M}$ the undamped eigenfrequency, and $D = D_J/\left(2\sqrt{K_J M}\right)$ the damping ratio. Damping is typically undesired in elastic joints designed for performance increase and kept as low as possible, because it can deteriorate the performance. Therefore, underdamped joints ($D < 1$) are considered in this work. To describe the first order differential equations, the system state $\boldsymbol{x} = [\dot{q}, \varphi]^\mathsf{T}$ is selected. Now (4.4) can be grouped as

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + Bu = \begin{pmatrix} -2D\omega & \omega^2 \\ -1 & 0 \end{pmatrix} \boldsymbol{x} + \begin{pmatrix} 2D\omega \\ 1 \end{pmatrix} u\,, \tag{4.5}$$

where $u = \dot{\theta}$ is the control input.

Reachable States

The reachable set $\mathcal{R}$ is defined as the set of states $\boldsymbol{x}$ that can be reached from the origin $\boldsymbol{x}(0) = [0, 0]^\mathsf{T}$ without violating the deflection constraint (4.2a) or the motor velocity constraint (4.2b). Together with the initial and terminal conditions the problem is defined as

$$\dot{q}(0) = 0, \quad \dot{q}(t_f) = \dot{q}_d\,, \tag{4.6a}$$

$$\varphi(0) = 0, \quad \varphi(t_f) = \varphi_d\,, \tag{4.6b}$$

where $\dot{q}_d$ and φ_d are the desired link velocity and deflection at the final time t_f. The desired state shall be located within the reachable set, i.e., $\boldsymbol{x}_d = [\dot{q}_d, \varphi_d]^\mathsf{T} \in \mathcal{R}$.

Optimal Control Formulation

Intuitively speaking, the joint shall be accelerated as fast as possible. The suitable cost function is simply $\min J(u) = t_f$, subject to dynamics (4.5). The control input is the limited motor velocity $u = \dot{\theta}$, $|u| \leq u_{\max} = \dot{\theta}_{\max}$. The Hamiltonian (cf. Sec. 2.4) is defined as

$$H = \lambda_1 \left(2D\omega(u - x_1) + \omega^2 x_2\right) + \lambda_2(u - x_1)\,, \tag{4.7}$$

where λ_1 and λ_2 are the costates [202]. In addition to the bounded input, also a path constraint is considered, namely the limited elastic deflection (4.2a). Formally, one constraint represents the upper and another the lower maximum deflection, respectively

$$\boldsymbol{h} = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} \theta - q - \varphi_{\max} \\ -\theta + q - \varphi_{\max} \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \end{pmatrix} . \tag{4.8}$$

The state inequality constraint can be taken into account by directly adjoining it to the Hamiltonian (4.7) to form an extended Hamiltonian[1]. In this work, the full optimal control formalism is omitted for the sake of brevity and readability. Instead, it is focused on the formed plots that are necessary to understand the given line of argumentation. As will be shown later, the considered problems can be divided into subproblems using physical reasoning. The solutions for these subproblems can then be determined by exploiting the solution without active state constraint and the necessary boundary control in case of an active state constraint.

Unconstrained Problem If no state constraint is active, then the costate dynamics are $\dot{\lambda}_1 = 2D\omega\lambda_1 + \lambda_2$ and $\dot{\lambda}_2 = -\omega^2\lambda_1$. Rewriting the dynamics yields $\ddot{\lambda}_1 - 2D\omega\dot{\lambda}_1 + \omega^2\lambda_1 = 0$. The resulting switching function is

$$\sigma = \frac{\partial H}{\partial u} = 2D\omega\lambda_1 + \lambda_2 = \dot{\lambda}_1 . \tag{4.9}$$

As the Hamiltonian is a linear function of the control input, the optimal control law according to the Minimum Principle of Pontryagin [205] becomes

$$u^* = \begin{cases} -u_{\max}\ \mathrm{sign}(\dot{\lambda}_1), & \dot{\lambda}_1 \neq 0 , \\ \text{singular}, & \dot{\lambda}_1 = 0 . \end{cases} \tag{4.10}$$

Singular solution arcs can be excluded, as the controllability matrix $C = [B, AB]$ has full rank [201]. Thus, u^* is of bang-bang type. Solving the costate dynamics and inserting them into (4.9) yields

$$\sigma = e^{D\omega t} \left(c_1 \cos(\omega_d t) + c_2 \cos(\omega_d t) \right) , \tag{4.11}$$

where $\omega_d = \sqrt{1 - D^2}\,\omega$ is the damped eigenfrequency and c_1 and c_2 are constants depending on the initial costate values, damping ratio, and eigenfrequency. Since the switching function is 2π-periodic, a zero crossing occurs the latest every half period π. In direct consequence, the switching time of the bang-bang controller is $t_s = \pi/\omega_d$.

Boundary Control Since (4.8) is differentiated once w.r.t. time until the input appears explicitly, the path constraint is of order one

$$\frac{d\boldsymbol{h}}{dt} = \frac{d}{dt}\begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} u - x_1 \\ -u + x_1 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \end{pmatrix} . \tag{4.12}$$

If the constraint is active, then (4.12) yields the boundary control $u_b = x_1$. When the motor and the link travel at the same velocity, the relative position $x_2 = \theta - q$ remains constant. After hitting $x_2 = \varphi_{\max}$, this control input ensures that the constraint will not be violated. Of course, the motor velocity constraint (4.2b) must be met at any time. Boundary control is therefore only possible if $|x_1| \leq u_{\max}$.

[1] A comprehensive survey on dealing with state constraints can be found in [246].

4.1.2 Influence of System Parameters on Maximum Velocity

One can identify three principal cases for the reachable states and their time-optimal trajectories, which depend on the exploitable spring energy and the link side damping. The reachable set is either bounded by

Case 1) an enclosing limit cycle, all reachable states can be hit by bang-bang control,

Case 2) the maximum elastic deflection, all reachable states can be hit by bang-bang control, or

Case 3) the maximum elastic deflection, quasi-singular solutions exist.

For each case, the maximum link velocity can be expressed as the sum of motor velocity $u_{\max}$ and a term $\Delta\dot{q}$ provided by spring energy storage and release

$$\dot{q}_{\max} = \dot{\theta}_{\max} + \Delta\dot{q}\,. \tag{4.13}$$

Instead of formulating $\dot{q}_{\max}$ as a sum of motor and spring term, one can also use a so-called speed gain

$$\epsilon = \frac{\dot{q}_{\max}}{\dot{\theta}_{\max}} \geq 1\,, \tag{4.14}$$

in order to express the benefit of the elastic mechanism on the achievable link velocity [52]. The maximum possible speed gain $\epsilon = \dot{q}_{\max}/u_{\max}$ for all cases is depicted in Fig. 4.2. It depends on the damping ratio D and $\sqrt{e_{SL}}$, a measure for the energetic capability of the joint, which can be determined as follows [52]. If the link travels with the maximum motor velocity, then its kinetic energy is given by $T_{L,\theta} = \frac{1}{2}M\dot{\theta}^2_{\max}$. The maximum storable energy in the spring is U_S, in case of a linear stiffness we get $U_S = \frac{1}{2}K_J\varphi^2_{\max}$. The ratio of both energies is

$$e_{SL} = \frac{U_S}{T_{L,\theta}} = \frac{\omega^2\varphi^2_{\max}}{\dot{\theta}^2_{\max}}\,. \tag{4.15}$$

When taking the square root of (4.15) we get $\sqrt{e_{SL}} = \frac{\omega\varphi_{\max}}{\dot{\theta}_{\max}}$, which correlates with the maximum achievable link velocity; see Sec. 4.1.4. The larger $\sqrt{e_{SL}}$, the more spring energy can be exploited to achieve high link-side velocities. For low $\sqrt{e_{SL}}$, most of the energy is provided by the motor, which means that the elastic transmission has only little benefit on the maximum achievable velocity. An alternative representation of the achievable link velocity for a 1-DOF damping-free joint is illustrated in Fig. 4.3. In the figure, the influence of the spring energy and the motor velocity on the speed gain can be analyzed separately. In Fig. 4.3 (a), the contribution of the spring to the maximum link velocity is shown, in Fig. 4.3 (b) the influence of motor and spring on the speed gain. The figures enable us to find a good balance between maximum motor velocity and elastic energy to achieve the desired speed gain. For the DLR FSJ [122], it can be observed that the contribution of the spring to the maximum achievable link velocity is $\approx 25\,\%$ if the motor travels at maximum velocity. Next, all three cases mentioned above are described in detail.

4.1.3 Case 1: Limit Cycle

If damping is present and the elastic deflection is unlimited, the authors of [248] showed that the system approaches a limit cycle when bang-bang control is continuously applied;

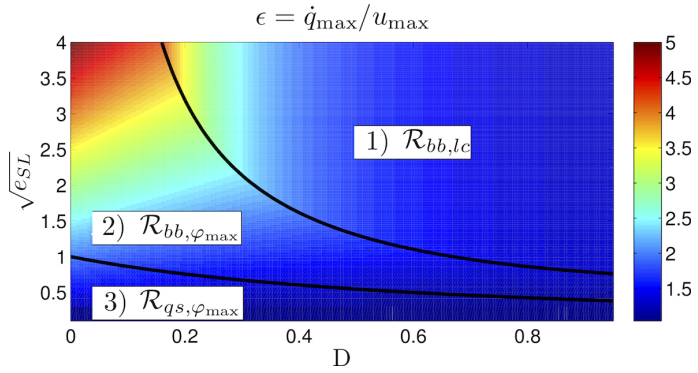


Fig. 4.2: Visco-elastic joint: Maximum link velocity increase for case 1, 2, and 3 depending on the energy ratio $\sqrt{e_{SL}}$ and damping D. © 2014 IEEE [247].

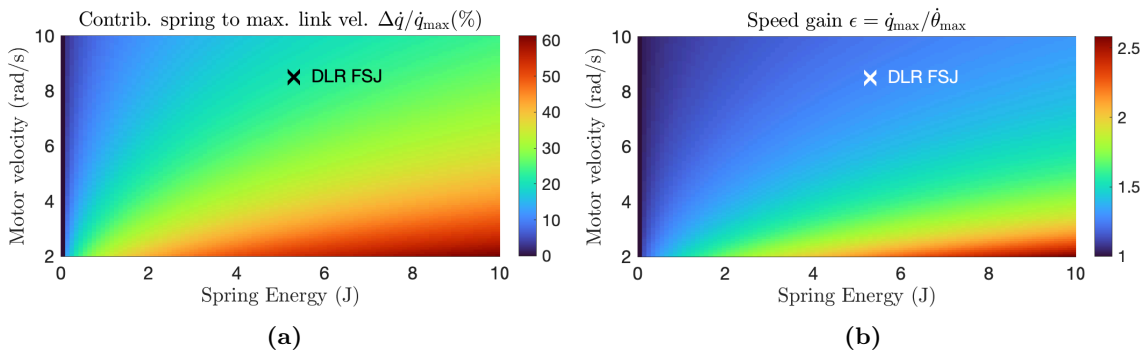


Fig. 4.3: Linear mass-spring system: Influence of motor velocity and spring energy on (a) the achievable link velocity and (b) speed gain.

see Fig. 4.4. The switching curve for time-optimal control, denoted by $\mathcal{S}_r$, can be obtained when starting at the origin $[0,0]^\mathsf{T}$ and applying $u = \pm u_{\max}$ for $t = t_s$. Starting a new spiral from every point of the resulting two curves generates the adjacent curves. Successively repeating this procedure provides the remaining spirals of $\mathcal{S}_r$; see Fig. 4.4 [201]. Above the switching curve, one must command $u = u_{\max}$, below $\mathcal{S}_r$ one must apply $u = -u_{\max}$ to hit the steady-state in minimum time. It can be observed that the velocity gain decreases with each successive motor cycle due to damping. The maximum possible velocity is

$$\dot{q}_{\max,lc} = u_{\max}\left(1 + 2\frac{F_1(D)}{1 - F_2(D)}\right), \quad \forall D > 0\,, \tag{4.16}$$

where

$$F_1(D) = e^{-\frac{D}{\sqrt{1-D^2}}\left(\pi - \operatorname{atan2}\left(2D\sqrt{1-D^2}, 1-2D^2\right)\right)}, \tag{4.17}$$

$$F_2(D) = e^{-\frac{\pi D}{\sqrt{1-D^2}}}\,. \tag{4.18}$$

This velocity can only be obtained by an infinite number of motor cycles. The reachable states, denoted by $\mathcal{R}_{bb,lc}$ and illustrated in Fig. 4.4 (green colored area), are all states that are enclosed by the limit cycle. Because no deflection constraint is considered in this case, all states can be reached by bang-bang control.

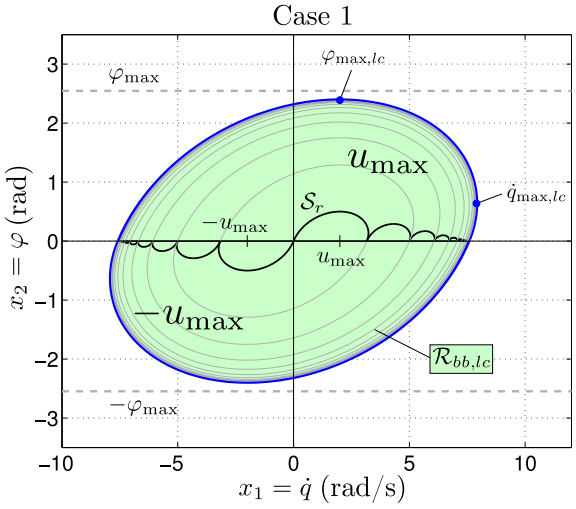


Fig. 4.4: Reachable states for the elastic deflection being larger than the maximum deflection of the limit cycle (case 1). The switching curve is denoted by $\mathcal{S}_r$. The blue line represents the limit cycle, the colored area all reachable states. © 2014 IEEE [247].

In case 1, the maximum velocity (4.16) only depends on $u_{\max}$ and D, as the deflection constraint is not active. If damping is low, the maximum velocity can reach large values. For $D = 0$ the velocity can theoretically become infinite, because there exists no limit cycle [249]. For real-world systems, however, the maximum speed is rather bounded by the deflection constraint since the elastic energy is limited. This case is described next.

4.1.4 Case 2: Bang-Bang Control

In case 2, the maximum velocity and reachable set of states are smaller than those of the previous case. First, the maximum possible velocity is derived. The corresponding time-optimal trajectory will then help to identify the set of reachable states. Afterwards, the boundary between cases 1 and 2 is derived.

Maximum Velocity

In order to optimally exploit the elastic energy, it is necessary to fully charge the spring and transform as much potential energy as possible to kinetic link energy. This charged state is $\boldsymbol{x}_{ch} = [\dot{q}_{\max}, \varphi_{\max}]^{\mathsf{T}}$; see Fig. 4.5 (upper). As shown in [52], this charged state can be reached by bang-bang control without violating the maximum deflection. Once $\boldsymbol{x}_{ch}$ is reached, one must apply the maximum motor velocity $u = u_{\max}$ (i.e., the link travels in the inertial frame of the motor) to gain maximum speed. Inserting $\boldsymbol{x}_0 = \boldsymbol{x}_{ch}$ into the system dynamics (4.4) and setting $u = u_{\max}$, one can derive the maximum velocity

$$\dot{q}_{\max,\varphi_{\max}} = u_{\max}(1 + \sqrt{e_{SL}}\ F_3(D))\,, \tag{4.19}$$

where

$$F_3(D) = e^{-\arctan\left(\frac{\sqrt{1-D^2}}{D}\right)\frac{D}{\sqrt{1-D^2}}}\,. \tag{4.20}$$

The phase plane trajectory for reaching $\dot{q}_{\max,\varphi_{\max}}$ is depicted in Fig. 4.5 (upper). The corresponding motor and link velocities are shown in the lower figure. The maximum

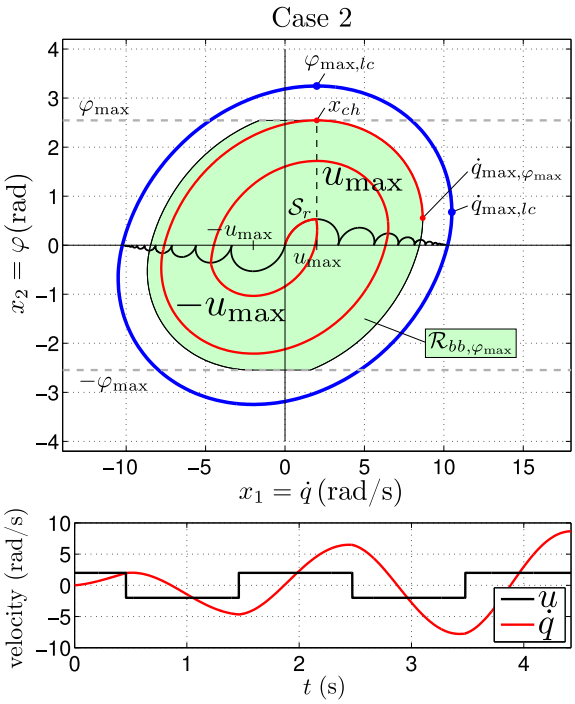


Fig. 4.5: Reachable states if the elastic deflection is lower than the maximum deflection of the limit cycle (upper). The blue line indicates the limit cycle, the colored area all reachable states. The red signal depicts the trajectory for reaching the maximum possible velocity starting from the origin, also shown in the lower figure over time. © 2014 IEEE [247].

velocity (4.19) increases linearly with $\sqrt{e_{SL}}$ and decreases exponentially with $F_3(D)$. If damping is low, which holds for most real-world joints, the energy ratio has a much stronger influence on the maximum velocity than the damping ratio; see Fig. 4.2.

Amount of Energy available for Velocity Maximization

At maximum link velocity the link acceleration becomes zero. The elastic deflection in this state can be determined by setting $\ddot{q} = 0$ in (4.5) and solving for φ, which yields

$$\varphi(\dot{q}_{\max,\varphi_{\max}}) = \frac{2D}{\omega}(\dot{q}_{\max,\varphi} - u_{\max})\,. \tag{4.21}$$

This equation shows that the elastic energy can only be utilized to a limited extent if damping is present. Only if $D = 0$ then we obtain $\varphi(\dot{q}_{\max,\varphi_{\max}}) = 0$ and one can fully transform potential energy to kinetic energy. This fact supports the general design guideline of choosing low damping for robotic systems and realizing it rather via active damping if necessary. Next, the set of reachable states is described for case 2.

Reachable Set

The trajectory with the largest distance to the origin in the phase plane defines the boundary of the reachable set. The trajectory mentioned above for obtaining the maximum possible link velocity is part of this boundary. Therefore, one may start from $\boldsymbol{x}_{ch}$ and apply $u = u_{\max}$ until $\dot{q}_{\max,\varphi_{\max}}$ is hit. Going forward in time, we keep $u = u_{\max}$ until we

reach zero deflection and do not reverse the motor velocity when passing the switching curve. Since we are interested in maximizing the distance to the origin and not deriving a time-optimal trajectory, this control leads to a larger velocity increase than switching the motor speed when $\mathcal{S}_r$ is reached. After hitting $x_1 = 0$ one must switch to $u = -u_{\max}$ until the minimum elastic deflection is obtained. Please note the described boundary is point symmetric w.r.t. the origin. To obtain the boundary trajectory on the left half of the phase plane, one must start from $[-u_{\max}, -\varphi_{\max}]^\mathsf{T}$ and apply the inverse motor input as the one described previously. Finally, the maximum and minimum elastic deflection define the upper and lower bound of the reachable set. The reachable set for case 2 is denoted by $\mathcal{R}_{bb,\varphi_{\max}}$. In Fig. 4.5 this set is represented by the colored area.

Boundary between Case 1 and Case 2

To determine whether the maximum velocity is bounded by a limited cycle (case 1) or the maximum elastic deflection (case 2), one can set $\dot{q}_{\max,\varphi_{\max}} = \dot{q}_{\max,lc}$ and solve for $\sqrt{e_{SL}}$. The boundary

$$\sqrt{e_{SL}}_{12}(D) = 2\frac{F_1(D)}{(1 - F_2(D))F_3(D)}\,, \tag{4.22}$$

is a relationship between the energy ratio and damping. In Fig. 4.2 it is represented by a black solid line between area 1) and 2). Below the boundary, the maximum velocity is bounded by the maximum deflection, while above, it is bounded by the limit cycle. The remaining area is described next.

4.1.5 Case 3: Quasi-Singular Control

In case 3, the elastic deflection is lowered such that it intersects the switching curve $\mathcal{S}_r$. Bang-bang control can no longer be applied because it would violate the deflection constraint. The maximum deflection of the switching curve is

$$\varphi_{\mathcal{S}_r,\max} = \frac{u}{\omega}F_3(D)\,. \tag{4.23}$$

In terms of energy ratio one can rewrite (4.23) and obtain

$$\sqrt{e_{SL}}_{23}(D) = F_3(D)\,. \tag{4.24}$$

Like in case 2, the maximum velocity is bounded due to the deflection constraint and can be determined by (4.19). The maximum velocity for case 3 and the boundary $\sqrt{e_{SL}}_{23}$ between 2) and 3) are depicted in Fig. 4.2. An example for case 3 is illustrated in Fig. 4.6. The reachable set of states can be obtained with the same approach as in case 2. However, the maximum velocity cannot be reached by bang-bang control, but is subject to quasi-singular solution arcs [52]. When hitting the deflection constraint $x_1 = \varphi_{\max}$, the boundary control $u_b = x_1$ must be applied; see Sec. 4.1.1. Maintaining the maximum possible spring deflection means that the maximum elastic torque is used to accelerate the link. The phase-plane trajectory travels on the constraint until the charged state $\boldsymbol{x}_{ch} = [u_{\max}, \varphi_{\max}]^\mathsf{T}$ is hit. To reach the maximum velocity, $u = u_{\max}$ must be applied; see Fig. 4.6.

Three distinct regions can be identified in the reachable set that is denoted by $\mathcal{R}_{qs,\varphi_{\max}}$. In area 3,1), bang-bang control can be applied to reach every state. The outer boundary is obtained when switching to $u = -u_{\max}$ once the maximum deflection is hit. States

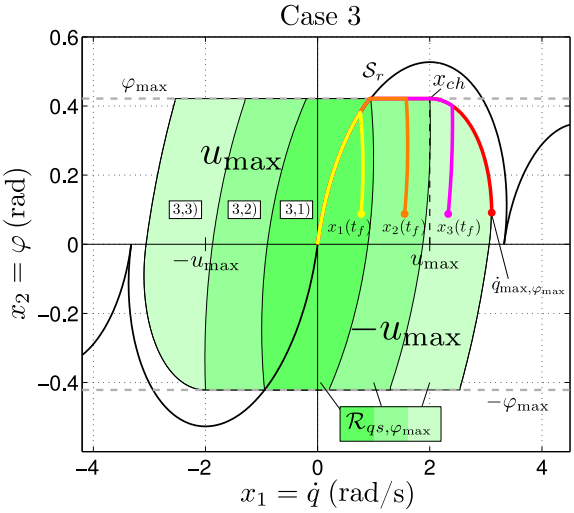


Fig. 4.6: Reachable states for case 3. States located within the subcase areas can be reached time-optimally as follows: case 3,1: bang-bang, case 3,2: bang-singular-bang, case 3,3: bang-singular-bang-bang. Four trajectories are depicted for illustrating the optimal control in each region. The corresponding motor and link velocities are depicted in Fig. 4.7. © 2014 IEEE [247].

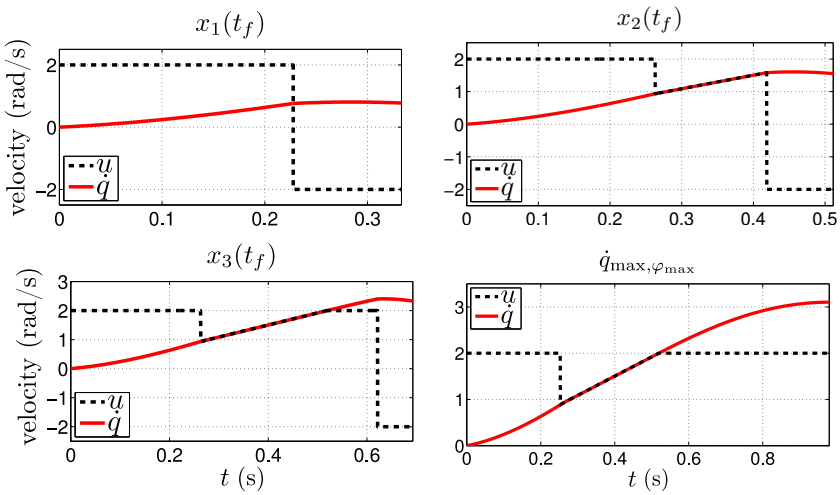


Fig. 4.7: Exemplary time-optimal trajectories according to Fig. 4.6. © 2014 IEEE [247].

located in the second area 3,2) can be hit by bang-singular-bang control; the boundary is defined by the spiral which intersects $\boldsymbol{x}_{ch}$. The third region, 3,3), is bounded by the set of reachable states. Any state belonging to this area can be reached by bang-singular-bang-bang control. In Fig. 4.6, four trajectories are illustrated to show the time-optimal solution for each area. The corresponding states are depicted in Fig. 4.7.

In this section, the speed gain of 1-DOF visco-elastic joints was analyzed. The following section extends the results to n-DOF elastic robots with intrinsic joint elasticity.

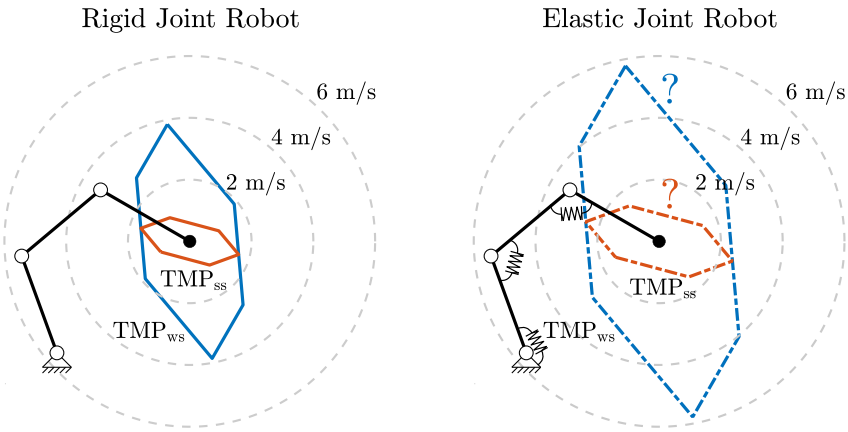


Fig. 4.8: Achievable Cartesian endpoint velocities of a planar 3R rigid (left) and elastic joint robot (right) for the illustrated configuration. For the rigid joint robot, the achievable velocities are represented by polytopes, where $\mathrm{TMP}_{\mathrm{ws}}$ denotes the weak sense (angular velocity may be non-zero) and $\mathrm{TMP}_{\mathrm{ss}}$ the strong sense (purely translational velocities) polytope [194, 237]. The goal is to derive the maximum possible endpoint velocities for elastic joint robots, which are known to be capable of achieving higher velocities than their rigid counterparts. © 2021 IEEE [236].

4.2 Speed Gain Approximation for n-DOF

In this section, the analysis of the maximum achievable velocity is extended to n-DOF serial manipulators with compliant joints. The goal is to estimate the achievable Cartesian TCP velocities with reasonable accuracy and minimal computational effort so that this estimation can be employed in motion and task planning methods and in the early design phase of robot mechanisms where quick iterations are desirable.

4.2.1 Problem Definition and Approach

Up to now, explosive motions together with the maximum peak velocity were mainly derived via optimal control [8,9,174]. In the previous section, the solution for a 1-DOF system could be solved analytically. However, for systems with more degrees of freedom, numerical techniques are usually required, which are computationally costly, time-consuming, and require accurate modeling of the complex system dynamics. Especially in the early design phase of the robot mechanism, but also in task and motion planning, it is essential that the maximum achievable TCP velocities for a particular robot configuration can be estimated in minimum time. For such applications, optimal control methods are hardly suitable due to their considerable computational effort.

In this work, a representation of the maximum possible Cartesian endpoint velocities for elastic joint robots is sought that a) is simple and computationally inexpensive to compute and b) estimates the achievable velocities with sufficient accuracy; see Fig. 4.8. Although essential for real-world implementation and understanding the energy transfer mechanisms, the derivation of the associated (time-optimal) trajectories is not considered in this thesis. Inspired by the velocity polytope for rigid joint robots, the representation of the maximum Cartesian velocities shall be determined for a specific joint configuration. It is assumed that the elastic energy can be locally transferred to kinetic energy around the desired link configuration $\boldsymbol{q}_d$ via internal spring deflection and release. In this thesis,

gravity-free elastic joint robots are considered. It is focused on the energy storage and release mechanism of the spring; the influence of gravitational potential energy on the maximum achievable velocity should be investigated in future work. The interest is in determining those velocities that can be reached from equilibrium by injecting energy via control and without external contact forces being applied. Several hypotheses are made on how and to what extent the motor velocity and the elastic energy can be exploited and converted to link kinetic energy. The hypotheses are exemplified using a 3R planar robot. For verification, the theory is compared with the optimal control solution. Here, a reasonable agreement in terms of maximum achievable TCP velocity is observed, and it is shown that the computation time of the proposed methods is orders of magnitude lower than the time required to compute the optimal control trajectories. To further confirm the validity of the presented methods, the results are compared with those from real-world throwing experiments that were previously conducted on DLR David by the authors of [8]. Finally, the approach is applied to robot safety assessment, where the safety characteristics of both the elastic and a hypothetically rigid version of DLR David are compared in terms of the Safety Map framework [208].

4.2.2 Considered Dynamics and Constraints

The reduced elastic joint dynamics (2.11) are considered, where the motors are modeled as velocity sources; cf. Chapter 2. In contrast to the previous section, the system is undamped as damping generally decreases velocity. Furthermore, the torque/deflection characteristic may be non-linear. Again the motor velocity constraint and the elastic deflection constraint

$$\dot{\boldsymbol{\theta}}_{\min} \leq \dot{\boldsymbol{\theta}} \leq \dot{\boldsymbol{\theta}}_{\max}\,, \tag{4.25a}$$

$$\boldsymbol{\varphi}_{\min} \leq \boldsymbol{\varphi} \leq \boldsymbol{\varphi}_{\max}\,, \tag{4.25b}$$

are considered, which are the most important real-world constraints besides motor torque. It is assumed that the constraints are symmetric, which holds for most systems. The overall amount of spring energy that can be actively exploited, e.g., to perform explosive motions, is denoted by $U_{S,\mathrm{dyn}} = \sum_{i=1}^{n} U_{S,\mathrm{dyn},i}$, where $U_{S,\mathrm{dyn},i}$ denotes the elastic energy in joint i. For VSA joints, $U_{S,\mathrm{dyn},i}$ is assumed to be the largest possible energy.

4.2.3 Energy Conversion Hypotheses

Two main hypotheses H1 and H2 are proposed to derive the achievable TCP velocities. Both hypotheses extend the 1-DOF results to n-DOF. The 1-DOF solution is again briefly summarized in the following. Suppose no damping is present and the elastic energy is limited. In that case, the maximum achievable link velocity of a planar, linear 1-DOF elastic joint with mass m can be expressed as the sum of the maximum motor velocity and a term $\Delta\dot{q}$ coming from spring energy and release[2]

$$\dot{q}_{\max} = \dot{\theta}_{\max} + \Delta\dot{q} = \dot{\theta}_{\max} + \sqrt{\frac{2U_{S,\mathrm{dyn}}}{m}}\,. \tag{4.26}$$

[2]Please note that not every achievable link velocity can be expressed in this convenient form.

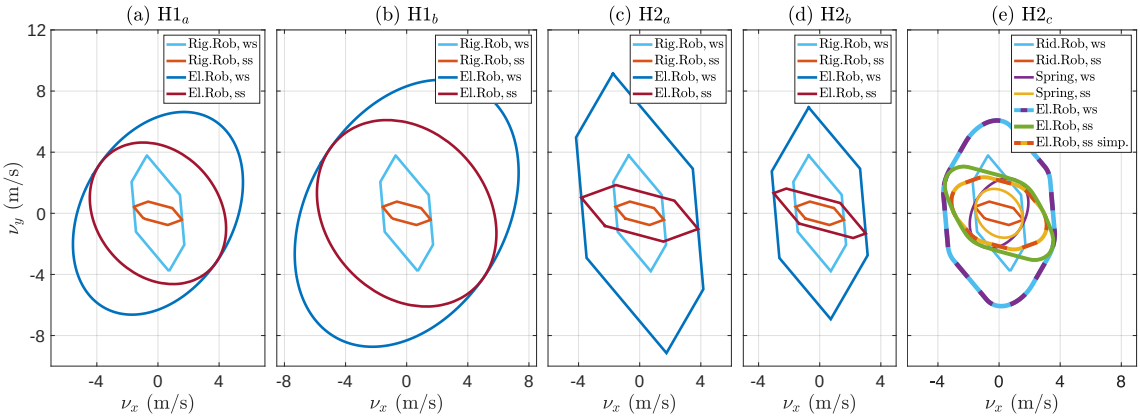


Fig. 4.9: Maximum Cartesian endpoint velocity for the 3R robot (configuration depicted in Fig. 4.8): Motor velocity polytopes (Rig.Rob) and the velocities of the elastic joint robot obtained by the hypotheses (El.Rob). Subscripts: ws: weak sense, ss: strong sense. © 2021 IEEE [236].

When the system has reached the maximum link velocity, then the total system energy is

$$V_f = \frac{1}{2} m \dot{q}_{\max}^2 \,, \tag{4.27a}$$

$$= \underbrace{\frac{1}{2} m \dot{\theta}_{\max}^2 + U_{S,\mathrm{dyn}}}_{V_{ch}} + \dot{\theta}_{\max} \sqrt{2 m U_{S,\mathrm{dyn}}} \,. \tag{4.27b}$$

The total energy consists of V_{ch}, i.e., the energy that the system has when the link travels with maximum motor speed and the spring is fully charged, and a term required to keep the maximum motor velocity while compensating for the elastic joint torque.

The first energy conversion hypothesis H1 is inspired by (4.27). It assumes that a certain amount of energy can be injected into the system, which can be converted to link kinetic energy. H1 has two sub-hypotheses H1$_a$ and H1$_b$. Hypothesis H2 is inspired by (4.26) and extends the velocity polytope approach from the rigid to the elastic joint case; cf. Sec. 3.5.5. The three sub-hypotheses of H2 are denoted by H2$_a$, H2$_b$, and H2$_c$. Starting with H1, all the hypotheses are described in the following. The theory is applied to the 3R planar robot that was introduced previously; see Fig. 4.8. Every joint is now equipped with a linear spring that can store up to 2 J potential energy. An overview of the results for all hypotheses is provided in Fig. 4.9.

Hypothesis H1

In the first hypothesis, H1, it is assumed that one can inject a certain amount of energy V into the system that can be converted (solely) to link kinetic energy. By extending (4.27) from 1-DOF to n-DOF we obtain the candidate energies

$$\mathbf{H1}_a \quad V_{ch} = \frac{1}{2} \dot{\boldsymbol{\theta}}_{\max}^{\mathsf{T}} \boldsymbol{M}(\boldsymbol{q}) \dot{\boldsymbol{\theta}}_{\max} + U_{S,\mathrm{dyn}} \,, \tag{4.28a}$$

$$\mathbf{H1}_b \quad V_f = V_{ch} + \dot{\boldsymbol{\theta}}_{\max}^{\mathsf{T}} \sqrt{2 \boldsymbol{M}(\boldsymbol{q})} \sqrt{\boldsymbol{u}_{S,\mathrm{dyn}}} \,, \tag{4.28b}$$

where $\boldsymbol{u}_{S,\mathrm{dyn}} = [U_{S,\mathrm{dyn},i}, \ldots, U_{S,\mathrm{dyn},n}]^{\mathsf{T}}$. For V_{ch}, the links have the same velocities as the motors and all springs are fully deflected. The energy V_f is the sum of V_{ch} and an additional

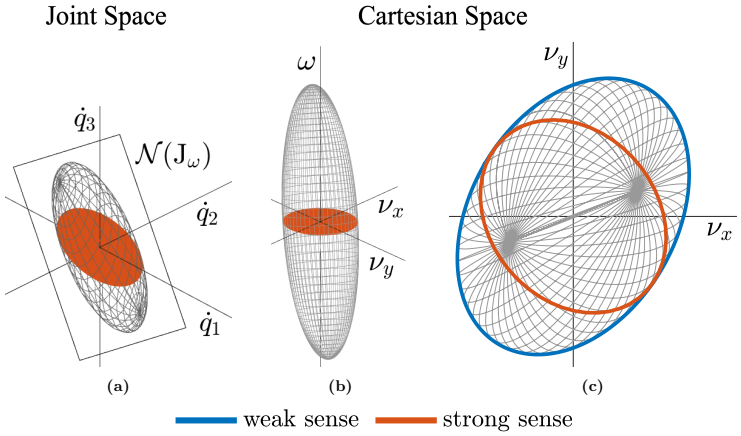


Fig. 4.10: Hypothesis H1: Strong and weak sense velocities in joint space (a) and Cartesian space (b), (c). © 2021 IEEE [236].

term analogous to (4.27). In sub-hypothesis H1$_a$ the energy $V = V_{ch}$ is assigned and in H1$_b$ $V = V_f$. By equating the total energy with the link kinetic energy $\frac{1}{2}\dot{\boldsymbol{q}}^\mathsf{T}\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} = V$ and rearranging the terms we get

$$\dot{\boldsymbol{q}}^\mathsf{T}\frac{\boldsymbol{M}(\boldsymbol{q})}{2V}\dot{\boldsymbol{q}} = 1\,, \tag{4.29}$$

which describes an ellipsoid in joint space. Equation (4.29) represents all joint velocities which are feasible given the energy V. Using the translational Jacobian matrix $\boldsymbol{J}_\nu(\boldsymbol{q})$ one can now transform the velocities (4.29) to Cartesian space. One obtains the same result when inserting $\dot{\boldsymbol{q}} = \boldsymbol{J}_\nu(\boldsymbol{q})^+\boldsymbol{\nu}$, where $\boldsymbol{J}_\nu(\boldsymbol{q})^+$ is the generalized inverse of $\boldsymbol{J}_\nu(\boldsymbol{q})$, into the previous equation, which yields

$$\boldsymbol{\nu}^\mathsf{T}\frac{(\boldsymbol{J}_\nu(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}_\nu(\boldsymbol{q})^\mathsf{T})^{-1}}{2V}\boldsymbol{\nu} \leq 1\,. \tag{4.30}$$

This ellipsoid is the weak sense representation of the achievable translational velocities. The result for the 3R robot is illustrated in Fig. 4.9 (a), (b) and Fig. 4.10. In order to calculate the strong sense velocities, one needs to determine the intersection[3] of the nullspace $\mathcal{N}(\boldsymbol{J}_\omega(\boldsymbol{q}))$ and the ellipsoid (4.29), i.e., the velocities that satisfy

$$\left\{\dot{\boldsymbol{q}} \mid \dot{\boldsymbol{q}} \in \mathcal{N}(\boldsymbol{J}_\omega(\boldsymbol{q})) \;\&\; \frac{1}{2}\dot{\boldsymbol{q}}^\mathsf{T}\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} = V\right\}\,. \tag{4.31}$$

The shape of the intersection depends on the dimension of $\mathcal{N}(\boldsymbol{J}_\omega(\boldsymbol{q}))$. For example, if $n - m_{J_\omega} = 1$, where m_{J_ω} is the number of rows of $\boldsymbol{J}_\omega(\boldsymbol{q})$, then the null space is represented by a line in $\mathbb{R}^n$, for $m_{J_\omega} = 1, n \geq 2$ it is a hyperplane with dimension $n-1$. For the 3R robot, the nullspace of $\boldsymbol{J}_\omega(\boldsymbol{q})$ is represented by a two-dimensional plane in $\mathbb{R}^3$; see Fig. 4.10 (a). Practically, one can obtain the boundary of the intersection as follows:

1. Determine the joint velocities $\dot{\boldsymbol{q}}$ that are located on the unit sphere $\dot{\boldsymbol{q}}^\mathsf{T}\dot{\boldsymbol{q}} = 1$.

[3]Please note that it is not possible to obtain the nullspace velocities via projection onto $\mathcal{N}(\boldsymbol{J}_\omega(\boldsymbol{q}))$, which could be done by replacing $\boldsymbol{J}_\nu(\boldsymbol{q})$ in (4.30) by $\boldsymbol{J}_{\nu\backslash\omega}(\boldsymbol{q}) = \boldsymbol{J}_\nu(\boldsymbol{q})(\boldsymbol{I} - \boldsymbol{J}_\omega(\boldsymbol{q})^+\boldsymbol{J}_\omega(\boldsymbol{q}))$, because the resulting velocities will not necessarily be located within the admissible ellipsoid.

2. Project these velocities onto the null space of $\boldsymbol{J}_\omega(\boldsymbol{q})$ with $\dot{\boldsymbol{q}}_{\mathcal{N}_\omega} = (\boldsymbol{I} - \boldsymbol{J}_\omega(\boldsymbol{q})^+ \boldsymbol{J}_\omega(\boldsymbol{q}))\dot{\boldsymbol{q}}$. The velocities are still located within the unit sphere.

3. Scale the null space velocities onto the boundary of the ellipsoid (4.29) via

$$\dot{\boldsymbol{q}}_{e,\mathcal{N}_\omega} = \sqrt{\frac{2V}{\dot{\boldsymbol{q}}_{\mathcal{N}_\omega}^\mathsf{T} \boldsymbol{M}(\boldsymbol{q}) \dot{\boldsymbol{q}}_{\mathcal{N}_\omega}}}\, \dot{\boldsymbol{q}}_{\mathcal{N}_\omega} \,. \tag{4.32}$$

The Cartesian velocities in the strong sense are finally given by $\boldsymbol{\nu} = \boldsymbol{J}_\nu(\boldsymbol{q})\dot{\boldsymbol{q}}_{e,\mathcal{N}_\omega}$. The result for the 3R robot is illustrated in Fig. 4.10 (c). The comparison of the weak and strong sense velocities for both the rigid (motors only) and the elastic joint 3R robot is shown in Fig. 4.9 (a) for H1$_a$ and (b) for H1$_b$. It can be observed that the maximum achievable endpoint velocity of the elastic robot is much larger than the velocity of the rigid counterpart. Because the energy V_f is larger than V_{ch} (cf. (4.28)), hypothesis H1$_b$ provides larger velocities than hypothesis H1$_a$.

Hypothesis H2

In hypothesis H1, the Cartesian velocity ellipsoid is obtained via a scalar kinetic energy that comprised the contribution of both the motors and the springs. As far as the motors are concerned (in other words, the rigid version of the robot), one can directly determine the feasible Cartesian velocity polytope according to Sec. 3.5 without the need of conversion to an (intermediate) energy. In the second hypothesis, H2, we want to start from this motor velocity polytope and add the velocities from the release of stored spring energy. As in the 1-DOF case described previously (see (4.26)), it is assumed that the maximum possible link velocity can be written in the form

$$\dot{\boldsymbol{q}}_{\max} = \dot{\boldsymbol{\theta}}_{\max} + \Delta\dot{\boldsymbol{q}} \,, \tag{4.33}$$

where $\Delta\dot{\boldsymbol{q}}$ represents the velocity gain attributed to the release of elastic energy. It is assumed that the total elastic energy can be converted to link kinetic energy. To determine $\Delta\dot{\boldsymbol{q}}$, three sub-hypotheses H2$_a$–H2$_c$ on the energy conversion mechanism are formulated.

H2$_a$ Each joint has a maximum link velocity which consists of the maximum motor velocity plus a term provided by the spring in the respective joint only. It is assumed that there is no inertial coupling between the links, i.e., n independent mass-spring systems are considered.

H2$_b$ Like H2$_a$ with the difference that the inertia about each joint is given by the configuration-dependent inertia of all successive links, which are assumed to be rigidly coupled.

H2$_c$ The available spring potential energy is converted to link kinetic energy while making no assumptions on how a particular spring contributes to the maximum possible link velocity.

In the following, each sub-hypothesis is explained and the theory is applied to the 3R robot. The resulting velocities are illustrated in Fig. 4.9 (c)–(e).

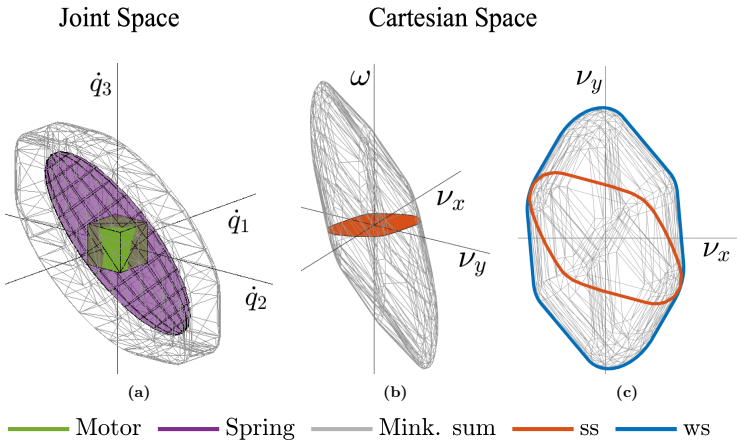


Fig. 4.11: Hypothesis H2$_c$: (a) Motor velocity bounds, spring ellipsoid (4.35), and Minkowski sum, (b) transformation of joint-space Minkowski sum to Cartesian space together with the strong sense velocities ($\boldsymbol{\omega} = \mathbf{0}$), (c) weak and strong sense velocities. © 2021 IEEE [236].

Hypothesis H2$_a$ In the same spirit as the related work on 1-DOF, H2$_a$ assumes that each elastic joint has a certain maximum link velocity. The motivation behind this is to extend the velocity polytope from the rigid joint to the elastic joint case in a very straightforward manner. Let us assume that there is no dynamic coupling between the links (in other words, we have n independent mass-spring systems) and that the total elastic energy can be converted to link kinetic energy. In each joint, there is a motor with the maximum velocity $\dot{\theta}_{\max,i}$, a spring with available energy $U_{S,\mathrm{dyn},i}$, and the link inertia ${}^{i}m_i$ about the current axis. We can again use (4.26) to determine the maximum link velocity

$$\dot{q}_{\max,i} = \dot{\theta}_{\max,i} + \sqrt{\frac{2U_{S,\mathrm{dyn},i}}{{}^{i}m_i}}, \quad i = 1, \ldots, n\,, \tag{4.34}$$

in each joint. One can now derive the weak sense velocity polytope by transforming the joint space hyperrectangle $|\dot{\boldsymbol{q}}| \leq \dot{\boldsymbol{q}}_{\max}$ to Cartesian space via $\boldsymbol{J}_\nu(\boldsymbol{q})$. This procedure is the same as for rigid joint robots, where $\dot{\boldsymbol{q}}$ is the motor velocity. The derivation of the strong sense velocity polytope also remains the same; cf. Sec. 3.5.5. The polytopes for the rigid and elastic case are illustrated in Fig. 4.9 c), which have the same shape but are different in size, meaning the elastic joint robot can reach higher velocities.

Hypothesis H2$_b$ Previously, it was assumed that the inertia about joint i is given by the decoupled link inertia ${}^{i}m_i$. A more conservative approach is to assume that all link inertias from joint i to n are rigidly coupled. The maximum joint velocity in this case is again determined by (4.34), where ${}^{i}m_i$ is now being replaced by $M_{i,i}$, the i-th element of the main diagonal of $\boldsymbol{M}(\boldsymbol{q})$. The results for the 3R robot are illustrated in Fig. 4.9 (d). Compared to the previous hypothesis H2$_a$, H2$_b$ provides lower velocities due to the larger inertia about the joints.

Hypothesis H2$_c$ In hypotheses H2$_a$ and H2$_b$, it was assumed that the potential energy of every spring is converted to kinetic energy in the particular joint. This allowed us to apply the polytope theory from rigid joint manipulators to elastic joint robots. In

hypothesis H2$_c$, it is not assumed that each joint has a certain maximum velocity but rather that the overall available elastic energy is converted to kinetic link energy. Similar to hypothesis H1 (cf. (4.29)), the joint space ellipsoid

$$\Delta\dot{\boldsymbol{q}}^{\mathsf{T}} \frac{\boldsymbol{M}(\boldsymbol{q})}{2U_{S,\mathrm{dyn}}} \Delta\dot{\boldsymbol{q}} = 1\,, \tag{4.35}$$

represents the contribution of the spring to the achievable link velocity. According to (4.33), the geometric sum, i.e., the Minkowski sum[4] of the motor and spring velocities is now determined. The result for the 3R robot is illustrated in Fig. 4.11. The joint velocities obtained by the Minkowski sum can be transformed to Cartesian space via $\boldsymbol{J}_\nu(\boldsymbol{q})$, which gives us the weak sense representation; see Fig. 4.11 (c). To derive the strong sense velocities, one needs to determine the intersection of the joint-space Minkowski sum and the nullspace of $\boldsymbol{J}_\omega(\boldsymbol{q})$, or the Cartesian velocities that satisfy $\boldsymbol{\omega} = \boldsymbol{0}$. For the 3R robot, the 3D clipping algorithm [239] was applied to compute the intersection of the Cartesian Minkowski sum with the ν_x/ν_y-plane; see Fig. 4.11 (b), (c). For systems/tasks with more DOF, it can become challenging to derive the H2$_c$ strong sense velocities. However, a subset of the feasible velocities can be determined relatively easily by first computing the strong sense representation of the motor and spring velocities independently (see Sec. 3.5.5 and (4.32)) and then calculating the Minkowski sum of both representations. The sum of the two nullspace velocities also belongs to the nullspace $\mathcal{N}(\boldsymbol{J}_\omega(\boldsymbol{q}))$. The result for this simplified approach is illustrated in Fig. 4.9 (e) for the 3R robot. In the figure, the motor polytope (red) is added to the spring ellipse (yellow), which results in the red/yellow dashed representation. It can be observed that these velocities are a subset of the achievable strong sense velocities, which are represented by a green line.

4.2.4 Verification

In order to verify the hypotheses, an optimal control problem is formulated that is solvable to determine the maximum feasible endpoint velocities and the corresponding robot trajectories. First, the problem is numerically solved for the elastic 3R robot, where the optimal control solution is compared to the velocities obtained by the hypotheses. Afterwards, this comparison is also made for the real-world throwing experiments that were previously conducted on the DLR David system [8, 250].

Problem Formulation

The motor position, link position, and velocity form the system state

$$\boldsymbol{x}_{\mathrm{opt}} = \begin{pmatrix} \boldsymbol{x}_{\mathrm{opt},1} \\ \boldsymbol{x}_{\mathrm{opt},2} \\ \boldsymbol{x}_{\mathrm{opt},3} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\theta} \\ \boldsymbol{q} \\ \dot{\boldsymbol{q}} \end{pmatrix}\,, \tag{4.36}$$

the control input is the motor velocity $\boldsymbol{w} = \dot{\boldsymbol{\theta}}$, which is bounded by $|\boldsymbol{w}| \leq \dot{\boldsymbol{\theta}}_{\max}$. The cost function for the optimal control problem has the standard form [201]

$$J_{\mathrm{opt}} = \phi(t_f, \boldsymbol{x}_{\mathrm{opt}(t_f)}) + \int_{t_0}^{t_f} L(t, \boldsymbol{x}_{\mathrm{opt}}, \boldsymbol{w})\, dt\,, \tag{4.37}$$

[4]The Minkowski sum for two sets C_1 and C_2 is defined as $C_1 \oplus C_2 = \{c_1 + c_2 | c_1 \in C_1, c_2 \in C_2\}$, i.e., the resulting set contains the sum of every element from C_1 and every element from C_2.

where $\phi(t_f, \boldsymbol{x}_{\mathrm{opt}(t_f)})$ is the Mayer term and $L(t, \boldsymbol{x}_{\mathrm{opt}}, \boldsymbol{w})$ the Lagrange term. The initial and final time are denoted by t_0 and t_f. We want to maximize the translational endpoint velocity in the desired Cartesian direction $\boldsymbol{u}_d$ and therefore select the Mayer term

$$\phi(t_f, \boldsymbol{x}_{\mathrm{opt}(t_f)}) = -\boldsymbol{u}_d^{\mathsf{T}} \boldsymbol{J}_\nu(\boldsymbol{x}_{\mathrm{opt},2}) \boldsymbol{x}_{\mathrm{opt},3}(t_f) + \epsilon t_f \,. \tag{4.38}$$

The robot should reach the maximum velocity as fast as possible; the secondary goal is thus to minimize the final time. The Mayer term thus includes t_f, which is multiplied by a small positive constant ϵ. Theoretically, one does not need to specify a Lagrange term, however, from a practical point of view, a small regularization term

$$L(t, \boldsymbol{x}_{\mathrm{opt}}, \boldsymbol{w}) = \frac{1}{2} \boldsymbol{w}^{\mathsf{T}} \boldsymbol{R} \boldsymbol{w} \,, \tag{4.39}$$

is included in order to smoothen the control input. Here, $\boldsymbol{R} \in \mathbb{R}^{n \times n}$ is a diagonal regularization matrix where the entries take relatively low values. The optimal control problem is subject to the reduced elastic joint dynamics. The maximum spring deflection yields the path constraint

$$|\boldsymbol{x}_{\mathrm{opt},1} - \boldsymbol{x}_{\mathrm{opt},2}| \leq \boldsymbol{\varphi}_{\max} \,. \tag{4.40}$$

At $t = t_f$, the robot has the desired configuration $\boldsymbol{q}_d$ and the Cartesian velocity points in the desired direction $\boldsymbol{u}_d$, which yields the terminal constraints

$$\boldsymbol{x}_{\mathrm{opt},2}(t_f) = \boldsymbol{q}_d \,, \tag{4.41a}$$

$$\frac{\boldsymbol{J}_\nu \, \boldsymbol{x}_{\mathrm{opt},3}(t_f)}{||\boldsymbol{J}_\nu \, \boldsymbol{x}_{\mathrm{opt},3}(t_f)||} = \boldsymbol{u}_d \,. \tag{4.41b}$$

For the strong sense analysis, the terminal constraint $\boldsymbol{J}_\omega \boldsymbol{x}_{\mathrm{opt},3}(t_f) = \boldsymbol{0}$ is added. In order to keep the motion as small as possible, the initial link position is selected to be the same as the desired position, i.e., $\boldsymbol{x}_{\mathrm{opt},2}(t_0) = \boldsymbol{x}_{\mathrm{opt},2}(t_f) = \boldsymbol{q}_d$.

Results for the 3R Elastic Joint Robot

For the 3R robot, three goal configurations are selected, namely $\boldsymbol{q}_{d,1}^{\mathsf{T}} = [140, -100, -100]\,^\circ$, $\boldsymbol{q}_{d,2}^{\mathsf{T}} = [110, -70, -70]\,^\circ$, and $\boldsymbol{q}_{d,3}^{\mathsf{T}} = [70, 30, 30]\,^\circ$, which reach from a folded to an outstretched configuration; see Fig. 4.12. The spring stiffness in each joint is 500 Nm/rad, the elastic energy was increased to 4.87 J ($\varphi_{\max} = 8\,^\circ$) in each joint, all other parameters remain the same. For each configuration, the optimal control problem is solved for 100 evenly-spaced Cartesian directions, i.e., points on the unit circle $\mathcal{S}^1$. The optimal trajectories were calculated with the numerical toolbox GPOPS[5] [251].

The results for the weak sense analysis are depicted in the upper row in Fig. 4.12, the results for the strong sense analysis in the lower row. The mean average percentage error (MAPE) (cumulative for all three configurations) including variance for each hypothesis compared to the optimal control solution is illustrated in the bar graph on the right. For the considered robot, the best agreement with the optimal control results is accomplished with hypothesis H2$_c$, where the mean error (with negligible variance) is 8 % for weak sense velocities and 9 % for strong sense velocities. The MAPE for the other hypotheses

[5]Following parameters were used: $\boldsymbol{R} = \mathrm{diag}\{0.1, 0.1, 0.1\}$, $\varepsilon = 0.001$, mesh tolerance: 0.001, no. mesh iterations: 3, no. nodes per interval: min: 6, max: 12.

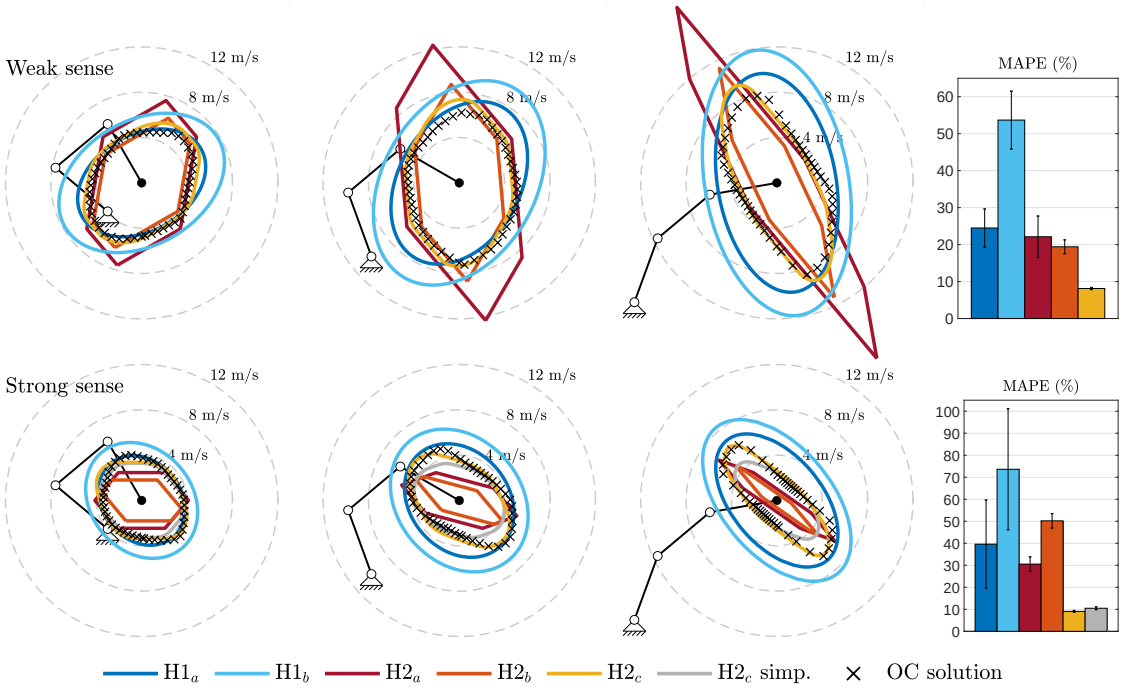


Fig. 4.12: 3R planar elastic joint robot: Comparison of the proposed methods and the optimal control solution, which serves for verification. The weak sense results for three different goal configurations are depicted in the upper row; the strong sense results in the lower row. The bar graph on the right illustrates the mean average percentage error (MAPE) including variance of each hypothesis compared to the optimal control (OC) solution. The smaller the error, the better the estimation of the maximum reachable TCP velocity. © 2021 IEEE [236].

is at least two times higher. H1$_b$ is the only hypothesis that provides a (over-)conservative approximation. Apparently, the robot's inertial coupling in $\boldsymbol{M}(\boldsymbol{q})$ influences the achievable TCP velocities. The higher the inertial coupling, the worse the estimation accuracy of hypotheses H2$_a$ and H2$_b$, which assume that the links are decoupled. Hypotheses H1$_a$ and H1$_b$ consider inertial coupling; however, the contribution of both the motors and the springs is condensed in a scalar energy V that defines the size of the ellipsoids. The best solution for this robot is obtained by H2$_c$, which extends the velocity polytope approach from the rigid to the elastic joint case by summing the contributions of the motors and the springs and taking inertial coupling into account.

In terms of computation time, it took seven hours on average on an Intel(R) Core(TM) i7-8565U CPU @ 2.0 GHz with 16 GB RAM (MATLAB 2020a) to compute the (100) optimal control solutions for each configuration and weak/strong sense representation in Fig. 4.12, i.e., approx. a full two days for all results[6]. In contrast, only 1.1 s were required in total to compute the velocities resulting from all the hypotheses shown in Fig. 4.12. An overview of the cumulated computation times for the 3R is provided in Tab. 4.1. The time required to compute H1$_a$–H2$_b$ is similar to the rigid joint case. H2$_c$ requires more computation time because the derivation of the strong sense representation

[6]Please note that for higher granularity (no. mesh iterations, mesh tolerance, etc.), even more computation time is required. Also for the throwing experiments reported in [8], more time was required to compute the desired motor trajectories.

Tab. 4.1: 3R planar elastic joint robot: Time required to compute the maximum TCP velocities illustrated in Fig. 4.12.

Rigid joint	**Elastic joint**					
	OC	$H1_a$	$H1_b$	$H2_a$	$H2_b$	$H2_c$
45 ms	≈ 2 days	53 ms	53 ms	46 ms	46 ms	860 ms

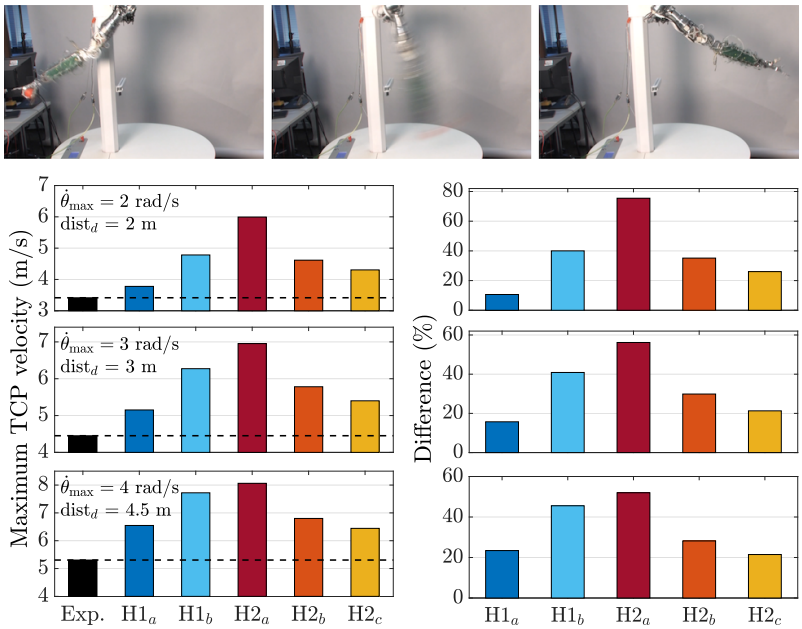
is computationally costly.

Evaluation of Previous DLR David Experiments

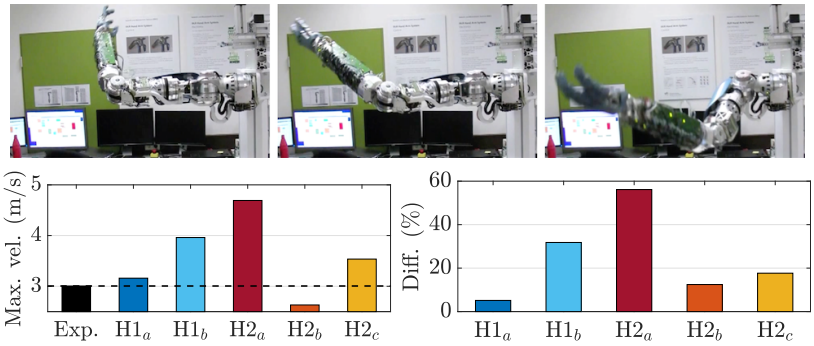
Next, the results of the presented methods are compared with those from the ball throwing experiments that were previously conducted on the DLR David system [8]; see Fig. 4.13 and the video in [250]. The goal of the optimal control problem in [8] was to throw a ball to a certain target distance for a particular final time. In the 2-DOF (planar) experiments[7] (see Fig. 4.13 (a)), joints 1 and 4 were actuated. The excitation trajectories were generated for different motor velocity limits and target distances. In Fig. 4.13 (a), the results for $\dot{\theta}_{\max} =$ 2, 3, and 4 rad/s are shown. For each considered motor velocity limit, the experiment with the farthest achieved throwing distance is selected. The maximum measured TCP velocities are illustrated in Fig. 4.13 (a). The figure compares these velocities to the TCP velocities (absolute and relative) estimated by the proposed hypotheses. In the 3-DOF (three-dimensional space) experiment, joints 2–4 (2-DOF shoulder and 1-DOF elbow) were actuated, the maximum motor velocity was limited to 2 rad/s. The results are depicted in Fig. 4.13 (b).

Please note that the problem formulation of the previous throwing experiments is slightly different from the one in this thesis. In [8] the goal was to achieve the desired throwing distance and not the maximum possible throwing distance, respectively TCP velocity. This is reflected in both the desired optimal control and the actual robot trajectories, where the maximum motor velocity and elastic deflection are typically not fully exploited. In contrast, in the 3R optimal control solution, both quantities reach their limits at a particular position along the trajectory. Therefore, it is expected that the velocities obtained by the hypotheses are larger than the velocities observed in the experiments. Nevertheless, hypotheses $H1_a$ and $H2_c$ agree reasonably well with the experimental results. Regarding the 2-DOF experiments, one obtains 11 % (2 rad/s), 16 % (3 rad/s), and 23 % (4 rad/s) difference w.r.t. the experiment for $H1_a$; for $H2_c$ the difference is 26 % (2 rad/s), 21 % (3 rad/s), and 21 % (4 rad/s). Concerning the 3-DOF experiment, the difference is 5 % for $H1_a$ and 18 % for $H1_c$. In the comparison for the 2-DOF experimental series, the difference between estimated and measured TCP velocity increases with maximum motor velocity; see Fig. 4.13 (a). In [8], the tracking accuracy of the motor position deteriorated for desired motor velocities ≥ 4 rad/s, which resulted in (slightly) lower endpoint velocities than expected. The authors of [8] explained this with unmodeled dynamics and friction. Furthermore, the assumption that the motors can be regarded as velocity sources has limitations for high motor accelerations.

[7] A soft stiffness preset was selected and the maximum deflection for the optimal control problem was limited to 12 °, which corresponds to 2.38 J exploitable potential spring energy (physical maximum: 15 °, 5.3 J).



(a) 2-DOF throwing experiments for different maximum motor velocities $\dot{\theta}_{\max}$ and desired throwing distances dist_d.



(b) 3-DOF throwing experiment with $\dot{\theta}_{\max} = 2\,\text{rad/s}$.

Fig. 4.13: Previous throwing experiments on DLR David [8,250]: Maximum measured TCP velocity and difference (absolute and relative) w.r.t. the velocities estimated by the proposed hypotheses. © 2021 IEEE [236].

4.2.5 Reduction of Computational Cost of Optimal Control

In Sec. 4.2.4, it was shown that the proposed methods could quickly estimate the achievable endpoint speed. However, compared to the proposed hypotheses, the solution of the optimal control problem is typically closer to the actual achievable speed, and it provides the input trajectory that can be commanded to the robot. As mentioned previously, the computational effort for solving optimal control problems is typically considerable though; cf. Sec. 4.2.4. The idea in the following is to employ the proposed estimation of the maximum operational velocity to reduce the computation time of the optimal control solver. For this, two possibilities are discussed and evaluated in the following.

Improvement of the Final Guess Optimal control methods usually require a guess of the initial and final values of all variables, which need to be specified by the user. To solve the optimal control problem for the 3R robot, 8 m/s was selected for the guess of the final endpoint velocity for every configuration and Cartesian direction. To speed up the calculation, one may now replace the 8 m/s by the $\mathrm{H2}_c$ velocity estimate (or the estimate of another hypothesis). However, it should be noted here that the optimal control solver generally requires the final values of the joint velocities and not the Cartesian velocities. The developed hypotheses provide only the latter. For redundant robots like the 3R, it would therefore be necessary to determine the joint velocities via the non-unique pseudoinverse of the Jacobian matrix.

Estimated Velocity as Constraint The maximum endpoint velocity and the minimum time required for the robot to reach this velocity were unknown in the optimal control formulation from in Sec. 4.2.4, making it difficult to solve the problem. The problem is better defined, i.e., there are less unknown variables, if the final velocity is set as a constraint. Then, one can formulate a minimum-time or minimum-energy optimal control problem, which can usually be solved faster [202]. However, the results illustrated in Fig. 4.12 show that the proposed approximation provides higher velocities than the system can actually reach in some cases. Setting infeasible velocities as constraints may cause significant computation time because the optimal control solver cannot find a solution. Depending on the solver and the considered system, one needs to check whether it is faster to solve a problem that contains unknown variables or defined more precisely but potentially not solvable. Concerning the latter, one could approach the search for the maximum achievable speed iteratively, e.g., start from the rigid-joint solution and increase the target velocity for the next iteration by a certain percentage, or use a golden-section search where the lower and upper bounds are defined by the rigid-joint solution and the estimate of the proposed hypotheses.

For the 3R robot, a preliminary analysis was conducted to determine whether the proposed speed gain estimate can be used to accelerate the computation time of the optimal control approach. Following variants of the optimal control problem formulated in Sec. 4.2.4 were analyzed for 20 equally distributed Cartesian directions and the robot configuration $\boldsymbol{q}^\mathsf{T} = [110, -70, -70]\,^\circ$ (see Fig. 4.12, second column):

1) **Original formulation** The maximum achievable endpoint velocity is unknown, the same guess of 8 m/s for the final velocity is chosen for all 20 Cartesian directions.

2) **Improved guess** The maximum achievable endpoint velocity is unknown, the guess

of the final velocity is given by the $\mathrm{H2}_c$ estimate.

3) Constraint The target speed (constraint) at the final time instant is given by the $\mathrm{H2}_c$ estimate.

The total computation times for the three cases on a Intel(R) Core(TM) i7-7700 @ 3.60 GHz with 16 GB RAM (MATLAB 2020b) are listed in Tab. 4.2. The time required to solve the original optimal control problem is 63:13 min. When incorporating the $\mathrm{H2}_c$ result into the guess of the final velocities (case 2)), the computation time reduces by approx. 40 % to 45:25 min. In case 3, the optimal control solver GPOPS takes considerable time to abort the computational process when the desired speed is unreachable. The optimization process had to be aborted manually, and therefore, no time is listed in Tab. 4.2. To check how fast the optimization could be performed if the target speed is reachable, the $\mathrm{H2}_c$ speed constraint was decreased by 30 %. This allowed reducing the computation time by approx. 110 % compared to the original problem formulation.

In summary, it is worth incorporating the proposed speed gain estimate as a guess of the final velocity into the optimal control problem to save computation time. Even less computation time can be achieved if the speed gain approximation is incorporated as a constraint; however, only if the solver can terminate the process within a reasonable time should the desired speed not be reachable. Unfortunately, with GPOPS, it was impossible to conduct the iterative search for the maximum achievable speed mentioned above. Future work should consider using an alternative optimal control solver.

Tab. 4.2: Optimal control computation time for 3R (20 orientations) with and without taking the proposed velocity estimation into account.

Original	**Improved Guess**	**Constraint**	
		100 % $\mathrm{H2}_c$	70 % $\mathrm{H2}_c$
63:13 min	45:25 min	-	29:51 min

4.2.6 Safety Map: Rigid vs. Elastic Joint Robot

Besides analyzing peak performance, one of the major motivations for determining the maximum Cartesian endpoint velocity of elastic joint robots is to derive their safety characteristics. The goal is to represent elastic joint robots in the Safety Map framework, allowing us to compare different robots (i.e., also rigid joint manipulators or mobile robots) to human injury data; cf. Chapter. 3. In this section, the theory is applied to derive the Safety Map representation of the DLR David system [151]. Only actuation of the first four robot joints is considered; the wrist and hand motions are ignored. The maximum velocity of each motor is 8.51 rad/s, the maximum available elastic energy in each FSJ joint is 5.3 J [122]. The reflected robot mass [181] and maximum velocity (weak sense) shall be determined in the direction of the three principal Cartesian directions X, Y, and Z for the red workspace area depicted in Fig. 4.14 (left). The considered box has a 25 cm edge length, the center is located at $[-0.2035, -0.45, -0.2]^\mathsf{T}$ m w.r.t. the robot base located in the shoulder. The Safety Map representation is determined for both the elastic and a hypothetically rigid version of the robot. The links on the rigid version of David are driven

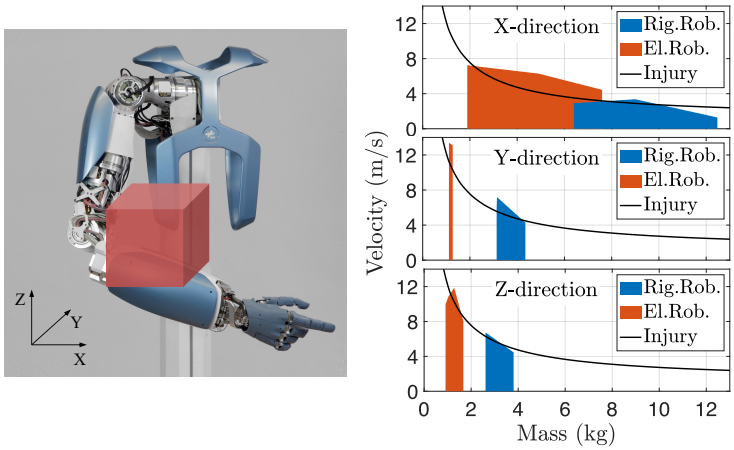


Fig. 4.14: Safety assessment of the DLR David system within the Safety Map framework [208]. The reflected mass and velocity ranges of the elastic (red) and the hypothetically rigid David (blue) are compared to the chest injury threshold (black line) in terms of the Compression Criterion (CC, threshold: 22 mm chest deflection) [252] for the illustrated workspace area (red cube). Every mass/velocity pair above the threshold corresponds to (at least) 5 % probability of serious chest injury. This threshold was derived from Lobdell's well-known chest collision model [24]. © 2021 IEEE [236].

by the motors only, and $\boldsymbol{M}(\boldsymbol{q})$ includes both the motor and the link inertia. The hypothesis $\mathrm{H2}_c$ is used to determine the maximum endpoint velocities for elastic David because it provided good and, more importantly, conservative velocity estimates for David previously. The Safety Maps are illustrated in Fig. 4.14. It can be observed that elastic David has a lower mass range, but it can reach much higher velocities than the rigid counterpart. These results agree well with expectation and previous results [17, 50, 53]. In Fig. 4.14 also the threshold for the occurrence of blunt chest injury in terms of the so-called Compression Criterion [24] is depicted. Both rigid and elastic David can harm the human when high-speed motions are performed in this workspace area. When operating an elastic robot at velocities equal to its rigid counterpart, the elastic robot is the safer option. However, as the impact velocity typically has a stronger influence on injury probability than the reflected mass [14], the elastic joint robot may pose a more significant threat to the human for specific body parts and collision scenarios than a rigid joint robot.

4.3 Summary

This section proposed and evaluated several approaches to determine the maximum achievable Cartesian endpoint velocities for gravity-free elastic joint robots. The peak TCP velocity is essential for robot design, motion, and task planning to assess and optimize collision safety and performance. For 1-DOF visco-elastic joints, an analytical solution to the achievable speed gain was derived, taking important real-world constraints on motor velocity and elastic deflection into account. For compliantly actuated n-DOF robots, numerical optimal control tools are commonly employed to determine the maximum possible velocity and the corresponding excitation trajectory. Such tools are usually computationally expensive. The methods proposed in this thesis provide a sufficiently accurate approximation of the speed gain in elastic joint robots with minimal computational requirements.

The methods were verified for a simulated 3R robot. A reasonable agreement between the real and the estimated velocities was also observed for the ball throwing experiments conducted previously on DLR David. Furthermore, it was shown for a case study that the computation of the optimal control solution could be significantly accelerated if the speed gain approximation is incorporated into the problem formulation. Finally, the developed theory was applied to the global robot safety assessment problem. The safety performance of DLR David and a hypothetically rigid version of this robot were compared within the Safety Map framework.

CHAPTER 5

Vibration Suppression in Robots with Intrinsic Joint Elasticity

The previous chapter described how the energy storage and release capabilities of elastic joint robots could be exploited to outperform their rigid counterparts in terms of peak performance. In order to achieve high link side velocities and to improve efficiency, compliant actuators are often designed such that damping and friction parallel to the spring are negligible. However, the inherent oscillatory system dynamics can deteriorate the tracking performance of link side trajectories or the time and distance required to stop the system in case of an undesired collision or an emergency situation. To benefit from the properties of the compliantly actuated robot while simultaneously achieving a desired task behavior, unwanted oscillations must be suppressed either through mechanics or control. This section regards the suppression of undesired vibrations via control, which includes stopping to an equilibrium position. The considered model and the system energies are described in Sec. 5.1. In Sec. 5.2 and Sec. 5.3, new and existing state-proportional and state-triggered control schemes are described and compared in terms of performance and passivity. In Sec. 5.4, a decoupling-based control framework is developed that allows to implement SISO state-triggered control schemes on multi-joint intrinsically elastic robots. Experiments on the elastic DLR David robot are reported in Sec. 5.5, where different controllers are implemented and validated. The experiments regard two-safety related use cases, namely a) the impact absorption after a dynamic ball impact and b) an emergency stop during task execution. Several metrics are defined for assessing the robot's impact response. In the experiments on DLR David, these metrics are used to quantitatively compare the performance of the tested control laws.

5.1 Model and System Energies

For the 1-DOF analysis on brakable states and the corresponding time-optimal trajectories, the gravity-free visco-elastic joint model (4.1) with constant stiffness is used; cf. Sec. 4.1. For n-DOF systems, the reduced model (2.11) with possibly non-linear torque/deflection characteristic is considered, which was described in Sec. 4.2. The control input of the elastic joint robot is again the motor velocity $\boldsymbol{u} = \dot{\boldsymbol{\theta}}$, which is assumed to have no underlying

dynamics. The total system energy of the undamped elastic joint robot is given by

$$V(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{q}}) = U(\boldsymbol{\theta}, \boldsymbol{q}) + \frac{1}{2}\dot{\boldsymbol{q}}^\mathsf{T}\boldsymbol{M}\dot{\boldsymbol{q}}\,. \tag{5.1}$$

Here, the total potential energy $U(\boldsymbol{\theta}, \boldsymbol{q}) = U_g(\boldsymbol{q}) + U_s(\boldsymbol{\theta}, \boldsymbol{q})$ is the sum of gravitational energy $U_g(\boldsymbol{q})$ and elastic energy $U_s(\boldsymbol{\theta}, \boldsymbol{q})$. The time derivative of $V(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{q}})$ is given by

$$\dot{V} = \frac{\mathrm{d}U_s(\boldsymbol{\theta}, \boldsymbol{q})}{\mathrm{d}t} + \frac{\mathrm{d}U_g(\boldsymbol{q})}{\mathrm{d}t} + \dot{\boldsymbol{q}}^\mathsf{T}\boldsymbol{M}\ddot{\boldsymbol{q}} + \frac{1}{2}\dot{\boldsymbol{q}}^\mathsf{T}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}}\,, \tag{5.2a}$$

$$= \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})^\mathsf{T}(\dot{\boldsymbol{\theta}} - \dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q})^\mathsf{T}\dot{\boldsymbol{q}} + \dot{\boldsymbol{q}}^\mathsf{T}\left(\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}) - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q})\right) + \frac{1}{2}\dot{\boldsymbol{q}}^\mathsf{T}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}}\,, \tag{5.2b}$$

$$= \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})^\mathsf{T}\dot{\boldsymbol{\theta}}\,. \tag{5.2c}$$

The system energy decreases when motor velocity and elastic joint torque have opposite signs. The torque/deflection curve is usually point symmetric w.r.t. the origin and strictly increasing. In this case, the system energy is reduced when [153]

$$\dot{V} \leq 0 \text{ if } \boldsymbol{\varphi}^\mathsf{T}\dot{\boldsymbol{\theta}} \leq 0\,. \tag{5.3}$$

When constantly decreasing V, an equilibrium position is typically achieved when the robot is in a vertical configuration, where the elastic deflection is minimal. To regulate the system energy, [155] used the energy-like function

$$H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}}) = U(\boldsymbol{\theta}, \boldsymbol{q}) - U(\boldsymbol{\theta}, \bar{\boldsymbol{q}}(\boldsymbol{\theta})) + \frac{1}{2}\dot{\boldsymbol{q}}^\mathsf{T}\boldsymbol{M}\dot{\boldsymbol{q}}\,. \tag{5.4}$$

In $U(\boldsymbol{\theta}, \bar{\boldsymbol{q}}(\boldsymbol{\theta}))$, $\bar{\boldsymbol{q}}(\boldsymbol{\theta})$ denotes the link equilibrium for a given motor position. This quantity was introduced in [133]; it can usually be obtained by solving a simple convex optimization problem. The collocated state $\bar{\boldsymbol{q}}(\boldsymbol{\theta})$ depends only on motor variables and can be used for regulating/tracking desired link positions, for example. If $H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}})$ becomes zero, then a static equilibrium is obtained, which can be an arbitrary joint position $\boldsymbol{q}$

$$H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}}) = 0 \Longleftrightarrow \boldsymbol{q} = \bar{\boldsymbol{q}}(\boldsymbol{\theta}), \dot{\boldsymbol{q}} = \boldsymbol{0}\,. \tag{5.5}$$

The time derivative of $H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}})$ is given by [155]

$$\dot{H} = \left(\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}) - \boldsymbol{\tau}_g(\bar{\boldsymbol{q}}(\boldsymbol{\theta}))\right)^\mathsf{T}\dot{\boldsymbol{\theta}}\,, \tag{5.6a}$$

$$= \dot{V} - \boldsymbol{\tau}_g(\bar{\boldsymbol{q}}(\boldsymbol{\theta}))^\mathsf{T}\dot{\boldsymbol{\theta}}\,. \tag{5.6b}$$

For linear stiffness we get $\dot{H} = \bar{\boldsymbol{\varphi}}(\boldsymbol{\theta}, \boldsymbol{q})^\mathsf{T}\dot{\boldsymbol{\theta}}$, where

$$\bar{\boldsymbol{\varphi}}(\boldsymbol{\theta}, \boldsymbol{q}) = \bar{\boldsymbol{q}}(\boldsymbol{\theta}) - \boldsymbol{q}\,, \tag{5.7}$$

can be considered as a deflection that becomes zero when the system reaches a static equilibrium. Also for non-linear, strictly increasing torque/deflection curves, one can replace $\boldsymbol{\tau}_J(\boldsymbol{\theta}, \bar{\boldsymbol{q}}(\boldsymbol{\theta}) - \boldsymbol{\tau}_g(\bar{\boldsymbol{q}}(\boldsymbol{\theta}))$ by $\bar{\boldsymbol{\varphi}}(\boldsymbol{\theta}, \boldsymbol{q})$, if the pseudo energy H shall be reduced via control, i.e., $\dot{H} \leq 0$ if $\bar{\boldsymbol{\varphi}}(\boldsymbol{\theta}, \boldsymbol{q})^\mathsf{T}\dot{\boldsymbol{\theta}} \leq 0$.

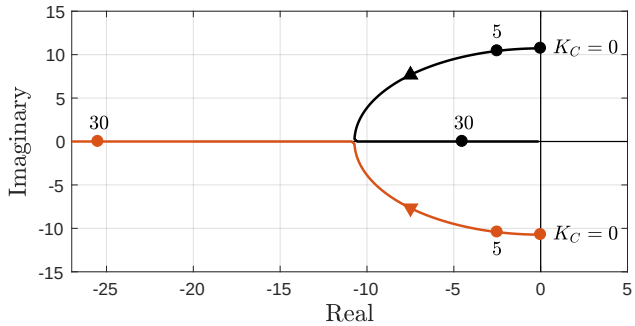


Fig. 5.1: Influence of the controller gain K_C on the closed-loop eigenvalues of a linear mass-spring system for control law (5.8).

5.2 State-Proportional Control

In this work, the problem of suppressing vibrations via control is viewed from an energetic perspective. The goal is to extract system energy V to dampen the joint velocity and achieve an energetically passive behavior of the controlled system. In the following, first state-proportional and then state-triggered control schemes are discussed. The controllers are initially proposed for the 1-DOF gravity-free case; the extension to n-DOF systems including gravity is described afterwards.

5.2.1 1-DOF

In the 1-DOF case, a very intuitive energy dissipating, state-proportional control law inspired by (5.2) is

$$u = -K_C\, \tau_J(\theta, q)\,, \tag{5.8}$$

where $K_C > 0$ is the controller gain. The time derivative of the system energy is

$$\dot{V} = -K_C\, \tau_J(\theta, q)^2\,, \tag{5.9}$$

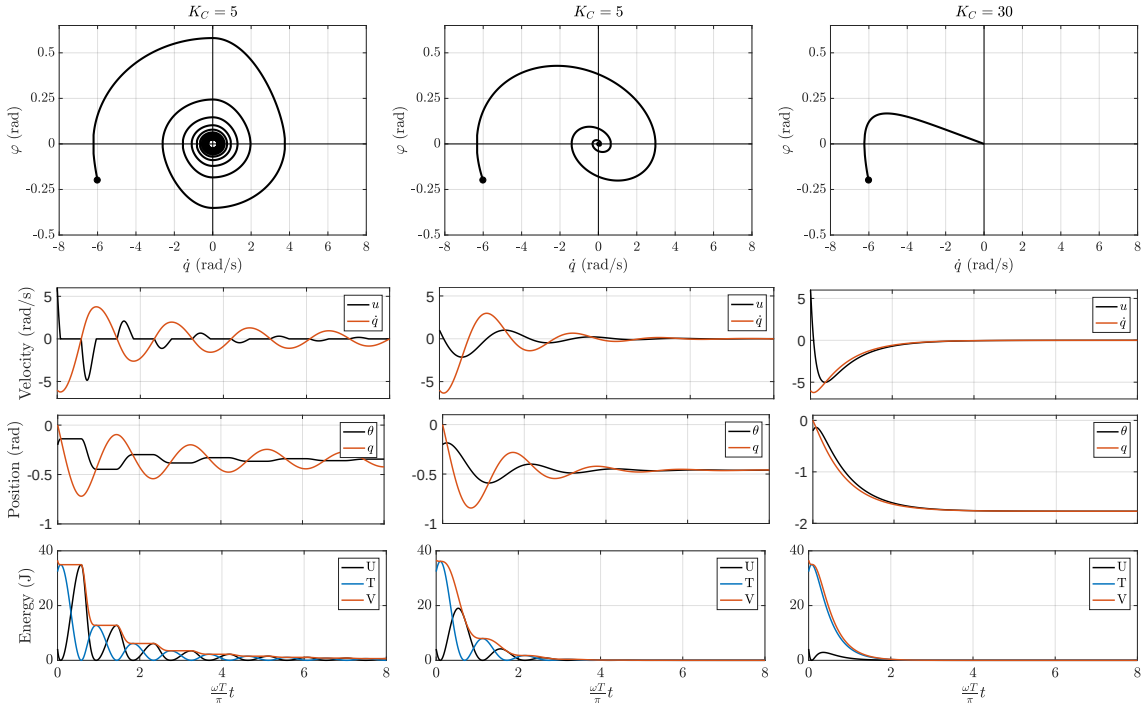
which is smaller or equal to zero. For a linear system, the two closed-loop system eigenvalues are

$$\lambda_{1,2} = -\frac{K_C}{2} \pm \frac{1}{2}\sqrt{K_C^2 - 4\omega^2}\,. \tag{5.10}$$

For $K_C < 2\omega$, the eigenvalues are complex with negative real parts, which results in a stable, oscillatory motion towards the equilibrium. For higher values of K_C, the eigenvalues are real and negative as the second term in (5.10) is smaller or equal to $1/2K_C$. This yields a stable, aperiodic system behavior. The dependency of the two closed-loop system eigenvalues on the controller gain is illustrated in Fig. 5.1. Here, a linear damping-free system is used for simulation. The system parameters are $m = 1.8$ kg, $K_J = 207$ Nm/rad, and $u_{\max} = 2$ rad/s.

The response to an initial elastic deflection and link velocity is illustrated in Fig. 5.2 for two controller gains[1]. For $K_C = 5$, it can be observed that the system oscillates about the equilibrium position. The elastic energy U is transferred to kinetic energy T and vice versa in an alternating fashion. For $K_C = 30$, the controller primarily minimizes the elastic deflection, which means the elastic energy decreases quickly. However, this also results in

[1]The controller implementation is $u = -K_C\, \varphi$ to allow comparability with control law (5.11).



(a) Controller (5.11) [153] **(b)** Controller (5.8) **(c)** Controller (5.8)

Fig. 5.2: Comparison of state-proportional control laws for a linear, gravity-free mass-spring system. In column (a), results for the controller (5.11) [153] are shown, where $K_C = 5$ is selected. In column (b) and (c), control law (5.8) is illustrated for gains $K_C = 5$ and 30.

low elastic joint torque that is available to decelerate the link. Therefore, motor and link positions converge to the equilibrium position asymptotically. In addition to the elastic deflection, it is possible to integrate the joint velocity $\dot{q}$ into the control law, e.g., in quadratic or absolute form to keep the sign of the motor velocity. In [153], the following control law was proposed that depends on both the elastic deflection and the link velocity[2]:

$$u = \begin{cases} -K_C \varphi \dot{q}, & \varphi > 0 \,\&\, \dot{q} \geq 0\,, \\ K_C \varphi \dot{q}, & \varphi < 0 \,\&\, \dot{q} \leq 0\,, \\ 0, & \text{otherwise} \end{cases} \tag{5.11}$$

Compared to (5.8), controller (5.11) typically achieves slower convergence towards the equilibrium; see Fig. 5.2. This is because a non-zero motor velocity is only commanded in two out of four quadrants in the plane spanned by the elastic deflection and link velocity. Furthermore, the low joint velocity in the vicinity of the equilibrium leads to small control action.

5.2.2 Extension to n-DOF

If no gravity is present, then $\boldsymbol{\theta} = \boldsymbol{q}$ holds whenever an equilibrium position is reached. In the presence of gravity, however, the elastic deflection in an equilibrium position is usually $\boldsymbol{\varphi} \neq \boldsymbol{0}$ due to gravity torque. As $\dot{V} = \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})^\mathsf{T} \dot{\boldsymbol{\theta}}$ holds also in the presence

[2] A strictly increasing torque/deflection characteristic is assumed.

of gravity, control law (5.8), e.g., can suppress oscillations by reducing the total system energy. However, this control law decreases energy until the elastic deflection becomes minimal, which is typically achieved if the manipulator is in a vertical configuration. To stop at a desired equilibrium position other than a vertical configuration, [155] proposed to use the energy-like function $H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}})$ described previously. When $H = 0$, then a static equilibrium $\dot{\boldsymbol{q}} = \boldsymbol{0}$ and $\boldsymbol{q} = \bar{\boldsymbol{q}}(\boldsymbol{\theta})$ is achieved. To suppress vibrations in the presence of gravity, one may now want to minimize H instead of V, or both H and V simultaneously. The state-proportional controllers described previously can be extended easily to n-DOF. Let us define

$$\Delta\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}) = \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}) - \boldsymbol{\tau}_g(\bar{\boldsymbol{q}}(\boldsymbol{\theta})) = \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}) - \boldsymbol{\tau}_J(\boldsymbol{\theta}, \bar{\boldsymbol{q}}(\boldsymbol{\theta}))\,. \tag{5.12}$$

Control law (5.8) can now be reformulated as

$$\boldsymbol{u} = -\boldsymbol{K}_C \Delta\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})\,, \tag{5.13}$$

where $\boldsymbol{K}_C = \mathrm{diag}\{K_{C,1}, \ldots, K_{C,n}\}$ is the diagonal, positive definite controller gain matrix. This control law minimizes H as can be seen by inserting (5.13) into (5.6)

$$\dot{H} = -(\boldsymbol{K}_C \Delta\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}))^{\mathsf{T}} \Delta\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})\,. \tag{5.14}$$

To reduce both H and V simultaneously, the control law can be modified as follows

$$\boldsymbol{u} = \begin{cases} -\boldsymbol{K}_C \Delta\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}) & \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})^{\mathsf{T}} \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}) \geq \boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})^{\mathsf{T}} \boldsymbol{\tau}_J(\bar{\boldsymbol{q}}(\boldsymbol{\theta}))\,, \\ \boldsymbol{0}, & \text{otherwise}\,. \end{cases} \tag{5.15}$$

If the torque/deflection characteristic is strictly increasing, then

$$\boldsymbol{u} = -\boldsymbol{K}_C \bar{\boldsymbol{\varphi}}\,, \tag{5.16}$$

can be implemented to achieve $\dot{H} \leq 0$ and

$$\boldsymbol{u} = \begin{cases} -\boldsymbol{K}_C \bar{\boldsymbol{\varphi}} & \bar{\boldsymbol{\varphi}}^{\mathsf{T}} \boldsymbol{\varphi} > 0\,, \\ \boldsymbol{0}, & \text{otherwise}\,. \end{cases} \tag{5.17}$$

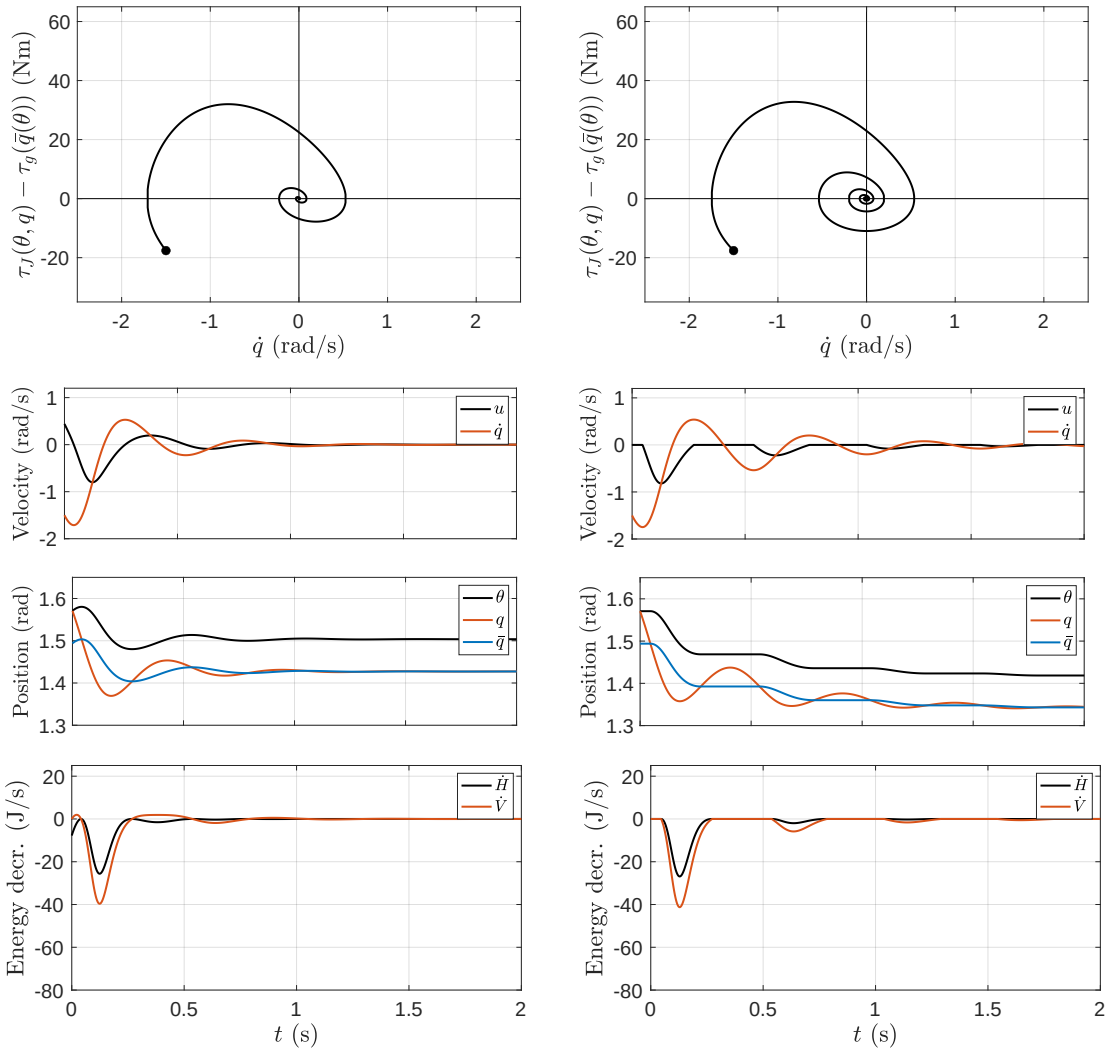
to reduce both $\dot{H}$ and $\dot{V}$.

Remark The control law proposed in [153] and analyzed in Fig. 5.2 can be implemented joint wise

$$u_i = \begin{cases} -K_{C,i} \bar{\varphi}_i \dot{q}_i, & \bar{\varphi}_i > 0 \,\&\, \dot{q}_i \geq 0\,, \\ K_{C,i} \bar{\varphi}_i \dot{q}_i, & \bar{\varphi}_i < 0 \,\&\, \dot{q}_i \leq 0\,, \\ 0, & \text{otherwise} \end{cases} \tag{5.18}$$

for joints $i = 1, \ldots, n$.

In Fig. 5.3, simulation results for a 1-DOF rotational joint with DLR FSJ elastic mechanism and gravity are provided. In the figure, the positions, velocities, and energies achieved with control law (5.13) and (5.15) are illustrated. A stable oscillatory motion towards equilibrium is observed for both controllers. Controller (5.18) is omitted, because like in the gravity-free case, the performance is comparatively poor. This will be also shown in Sec. 5.5, where the controllers (5.16) and (5.18) are implemented on DLR David and validated in practice.



(a) Controller (5.13), $\dot{H} \leq 0$

(b) Controller (5.15), $\dot{H} \leq 0, \dot{V} \leq 0$

Fig. 5.3: Comparison of state-proportional control laws for a rotational FSJ joint with non-linear joint torque/deflection characteristic and gravity. In the left column, results for controller (5.13) are illustrated, results for control law (5.15) in the right column.

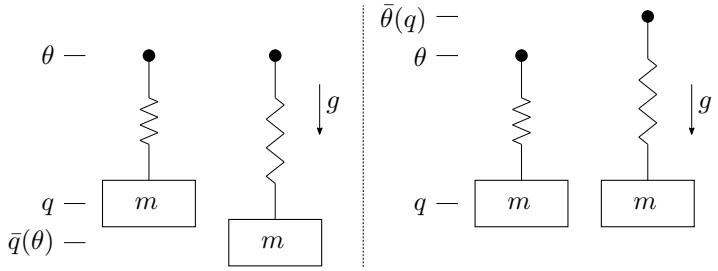


Fig. 5.4: Link equilibrium position $\bar{q}(\theta)$ for a given motor position θ (left) and motor equilibrium position $\bar{\theta}(q)$ for a given link position q (right).

5.2.3 Link vs. Motor Equilibrium Position

Instead of using the link equilibrium position $\bar{\boldsymbol{q}}(\boldsymbol{\theta})$ for vibration suppression control, one can also use the motor equilibrium position $\bar{\boldsymbol{\theta}}(\boldsymbol{q})$ for damping control. $\bar{\boldsymbol{\theta}}(\boldsymbol{q})$ is the motor position that corresponds to a static equilibrium for a given link position $\boldsymbol{q}$, which can be determined as follows. Assuming that the elastic joint torque $\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})$ is a bijective function of $\boldsymbol{\varphi} = \boldsymbol{\theta} - \boldsymbol{q}$, one needs to determine the inverse relationship function $\boldsymbol{\varphi}(\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q}))$. For a given $\boldsymbol{q}$, one can then compute the motor equilibrium position via

$$\bar{\boldsymbol{\theta}}(\boldsymbol{q}) = \boldsymbol{\varphi}(\boldsymbol{\tau}_g(\boldsymbol{q})) + \boldsymbol{q}\,. \tag{5.19}$$

In contrast to $\bar{\boldsymbol{q}}(\boldsymbol{\theta})$, $\bar{\boldsymbol{\theta}}(\boldsymbol{q})$ can be determined directly via the deflection/torque curve without the need of solving an optimization problem. However, in terms of stability it is desirable to use collocated quantities. The difference between $\bar{\boldsymbol{q}}(\boldsymbol{\theta})$ and $\bar{\boldsymbol{\theta}}(\boldsymbol{q})$ is illustrated in Fig. 5.4.

5.3 State-Triggered Control

5.3.1 Time-Optimal Braking for Visco-Elastic Joints

In Sec. 4.1, the maximum achievable velocity and the set of reachable states with their corresponding time-optimal trajectories were determined for a 1-DOF joint. This section continues the analysis. For a linear visco-elastic joint with limited elastic deflection, all states, i.e., velocities and deflections, shall be determined that can be brought to equilibrium. Zero deflection and velocity shall be reached without violating the motor velocity and deflection constraints $|u| \leq u_{\max}$ and $|\varphi| \leq \varphi_{\max}$. The set of brakable states is denoted by $\mathcal{B}$, the initial and final conditions for this problem are

$$\dot{q}(0) = \dot{q}_0, \quad \dot{q}(t_f) = 0\,, \tag{5.20a}$$

$$\varphi(0) = \varphi_0, \quad \varphi(t_f) = 0\,,\,, \tag{5.20b}$$

where $\boldsymbol{x}_0 = [\dot{q}_0, \varphi_0]^\mathsf{T} \in \mathcal{B}$ is the initial state. Regarding the time-optimal trajectory and resulting set $\mathcal{B}$ two cases are identified:

Case 1) All states can be stopped via bang-bang control

Case 2) Quasi-singular solutions may occur

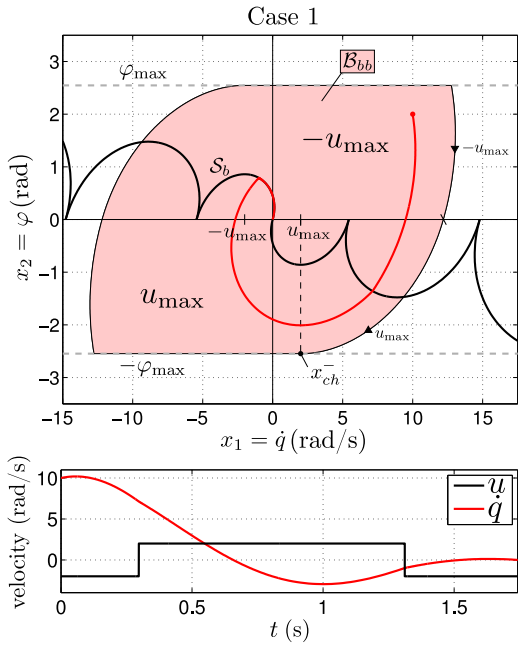


Fig. 5.5: The colored area in the upper figure represents all brakable states (case 1)). The red solid line is an exemplary trajectory that reaches the origin in minimum time. The corresponding evolution of motor and link speed is depicted in the lower figure. © 2014 IEEE [247].

For both cases, the switching curve for time-optimal control, denoted by $\mathcal{S}_b$, can be obtained in a similar fashion as the switching curve for velocity maximization; see Fig. 5.5 (upper). We get the first curves of $\mathcal{S}_b$ when starting from $[0, 0]^\mathsf{T}$ and going backwards in time with $u = \pm u_{\max}$ for $t = t_s$, where $t_s = \pi/\omega_d$. Above the switching curve, one must apply $u = -u_{\max}$, below $\mathcal{S}_b$, $u = u_{\max}$ applies to hit the steady state in minimum time. Next, the boundary of the brakable set is determined. The solution is the same for cases 1) and 2).

Brakable States

In the previous problem of reaching desired states, one important state located on the boundary of the reachable set was $\boldsymbol{x}_{ch} = [u_{\max}, \varphi_{\max}]^\mathsf{T}$. This maximum energetic state must be hit asymptotically in order to reach the maximum velocity. Also for this problem, the trajectory with the largest distance to the origin defines the boundary of $\mathcal{B}$ and reaches the marginal deflection asymptotically. Since the system trajectory is moving clockwise in the phase plane, this state is denoted by $\boldsymbol{x}_{ch}^- = [u_{\max}, -\varphi_{\max}]^\mathsf{T}$. The boundary can be obtained as follows. In a reversed time evolution starting from $\boldsymbol{x}_{ch}^-$, one applies $u = u_{\max}$ until zero deflection is reached; see Fig. 5.5 (upper). Then, one switches to $u = -u_{\max}$ until the maximum deflection is hit. The boundary is point symmetric w.r.t. the origin; the inverse procedure gives the boundary on the left-hand side of the phase plane. The upper and lower boundary of the brakable set are finally defined by the maximum and minimum elastic deflection, respectively.

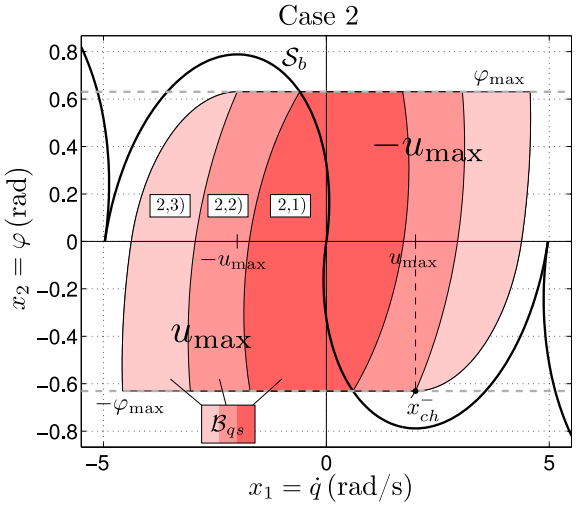


Fig. 5.6: Brakable states for case 2). States located within the subcase areas can time-optimally be stopped as follows: 2,1) bang-bang, 2,2) bang-singular-bang, 2,3) bang-bang-singular-bang. © 2014 IEEE [247].

Case 1

Bang-bang control is possible for all brakable states if the first curve of the switching curve can be hit without violating the motor velocity or deflection constraint. This is the case if the maximum elastic deflection is larger than the maximum deflection of the first switching curve. The boundary between cases 1) and 2) in terms of energy and damping ratio is

$$\sqrt{e_{SL}}_{12}(D) = e^{\frac{D}{\sqrt{1-D^2}}\left(\pi - \arctan\left(\frac{\sqrt{1-D^2}}{D}\right)\right)} . \tag{5.21}$$

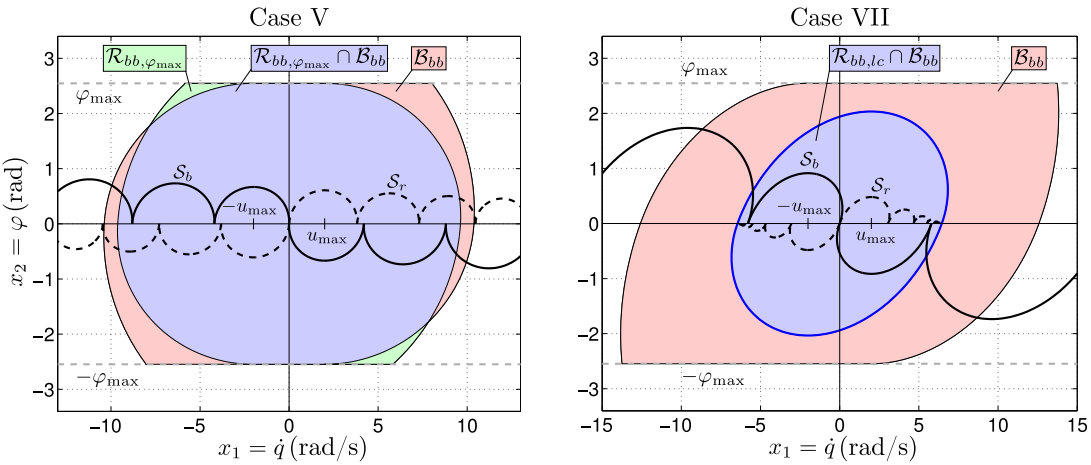
An example where all states within the brakable set can be stopped by bang-bang control is depicted in Fig. 5.5. The colored area represents the brakable set denoted by $\mathcal{R}_{bb}$.

Case 2

If the elastic deflection intersects the first curve of $\mathcal{S}_b$, quasi-singular solution pieces may occur because bang-bang control could violate the constraint. An example for case 2) is illustrated in Fig. 5.6. One can identify three subcases 2,1), 2,2), and 2,3). The brakable set $\mathcal{B}_{qs}$ and the corresponding time-optimal control can be derived similarly to case 3) of the velocity maximization problem described in Sec. 4.1. For the sake of brevity, a thorough elaboration is omitted.

Reachable vs. Brakable States

In Fig. 5.7 (a) the reachable and brakable set are depicted for a given joint design. Bang-bang control is possible for both sets. The blue area represents states which are reachable and brakable; however, there also exist states that can be reached but not be brought to equilibrium (green area) and vice versa (red area) without violating the deflection constraint. If damping is increased as shown in Fig. 5.7 (b), then the reachable set becomes a subset of the brakable set ($\mathcal{R} \subset \mathcal{B}$). The condition for this case is as follows. The boundary



(a) $\sqrt{e_{SL}} = 4, D = 0.03$ **(b)** $\sqrt{e_{SL}} = 4, D = 0.2$

Fig. 5.7: Two exemplary combinations of reachable and brakable sets. © 2014 IEEE [247].

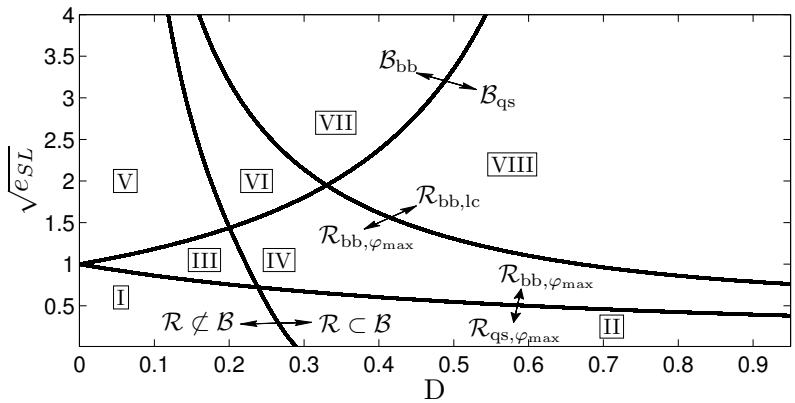


Fig. 5.8: Cases for the reachable and brakable set, depending on the energy ratio and damping. © 2014 IEEE [247].

of the brakable set always hits $[u_{\max}, -\varphi_{\max}]^{\mathsf{T}}$ in the fourth quadrant as described previously. The reachable set is a subset of the brakable set if the boundary of the reachable set intersects the minimum elastic deflection at $\dot{q}(-\varphi_{\max}) \geq u_{\max}$. The analytic formulation of the boundary in terms of $\sqrt{e_{SL}}$ and D is omitted for the sake of brevity.

Figure 5.8 summarizes the influence of the system parameters on the type of control input required to excite or dampen the joint time-optimally. Boundaries (4.22) and (4.24) are shown for all reaching cases, the boundary (5.21) for the braking cases, and finally the boundary whether $\mathcal{R}$ is a subset of $\mathcal{B}$ or not. In total, one can identify eight distinct cases depending on the energy ratio $\sqrt{e_{SL}}$ and damping factor D. With Fig. 5.8, one can analyze which concrete design parameters should be selected in order to achieve a desired joint behavior. Regions I and II are less desirable because quasi-singular solutions may exist for velocity maximization and braking. Bang-bang control can be applied to reach desired states if the joint design is located in region III-VIII and for braking in region V-VII. Thus, reaching desired states and braking to equilibrium is always possible with bang-bang excitation if the joint design is located in region V, VI, or VII. For maximum

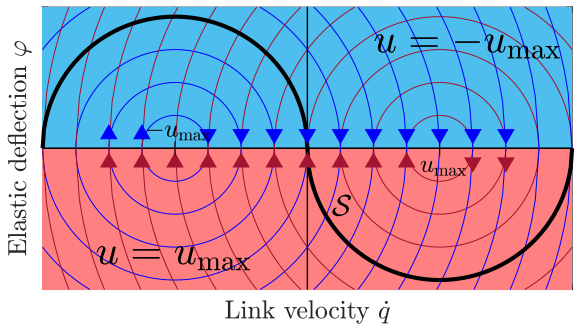


Fig. 5.9: Behavior of a linear mass-spring system when applying control law (5.22). The system state travels clockwise on the blue ellipses when minimum motor speed is applied, and on the red ellipses when the maximum velocity is commanded. © 2015 IEEE [253].

velocity increase, V is the most desirable region because the joint has good energy storage and release capabilities and low damping; see Fig. 4.2 and Fig. 5.7 a).

5.3.2 1-DOF: Controller Comparison

In the following, existing and new/modified damping controllers are reviewed and compared for a 1-DOF linear oscillator in terms of energetic passivity and performance. The control laws are illustrated in Fig. 5.10. To extract system energy V via state-triggered control, one can formulate the control law

$$u = -\text{sign}(\varphi) u_{\max} \,, \tag{5.22}$$

which shall minimize $\dot{V}$ based on (5.3). The closed-loop system energy derivative is $\dot{V} = \varphi u = -|\varphi| u_{\max} \leq 0$. This control law reduces the system energy locally; however, it is possible that the system does not converge to the equilibrium, but the system state remains at a point on the line segment $\dot{q} \in [-u_{\max}, u_{\max}]$, $\varphi = 0$. This is because the ellipses centered around $\pm u_{\max}$ meet on this line segment with opposite direction of travel; see Fig. 5.9. Because $\varphi = 0$, no elastic torque is available to decelerate the link.

Time-Optimal Control

The time-optimal control law for the linear system is depicted in Fig. 5.10 (upper row), where the red area represents maximum velocity input and the blue area minimum velocity input. In the phase plane, the solid black line represents an exemplary braking trajectory. The corresponding motor and link velocities are shown in the middle column. The spring potential energy U, link kinetic energy T, and total system energy $V = U + T$ are depicted in the right column. The time-optimal controller reduces energy as fast as possible. However, the energy does not decrease monotonically; cf. Fig. 5.10 (upper right) at approx. 0.2 and 1.2 half cycles. In the first and third quadrant in the phase plane, energy decrease, i.e., $\dot{V} \leq 0$ according to (5.2), is always ensured. In the second and fourth quadrant, however, the system energy increases when the current state is located within the area enclosed by the switching curve and the x-axis. This is because motor velocity and elastic deflection have equal signs, which results in a non-passive behavior. Next, a modification of the time-optimal control law is proposed, which leads to suboptimality in

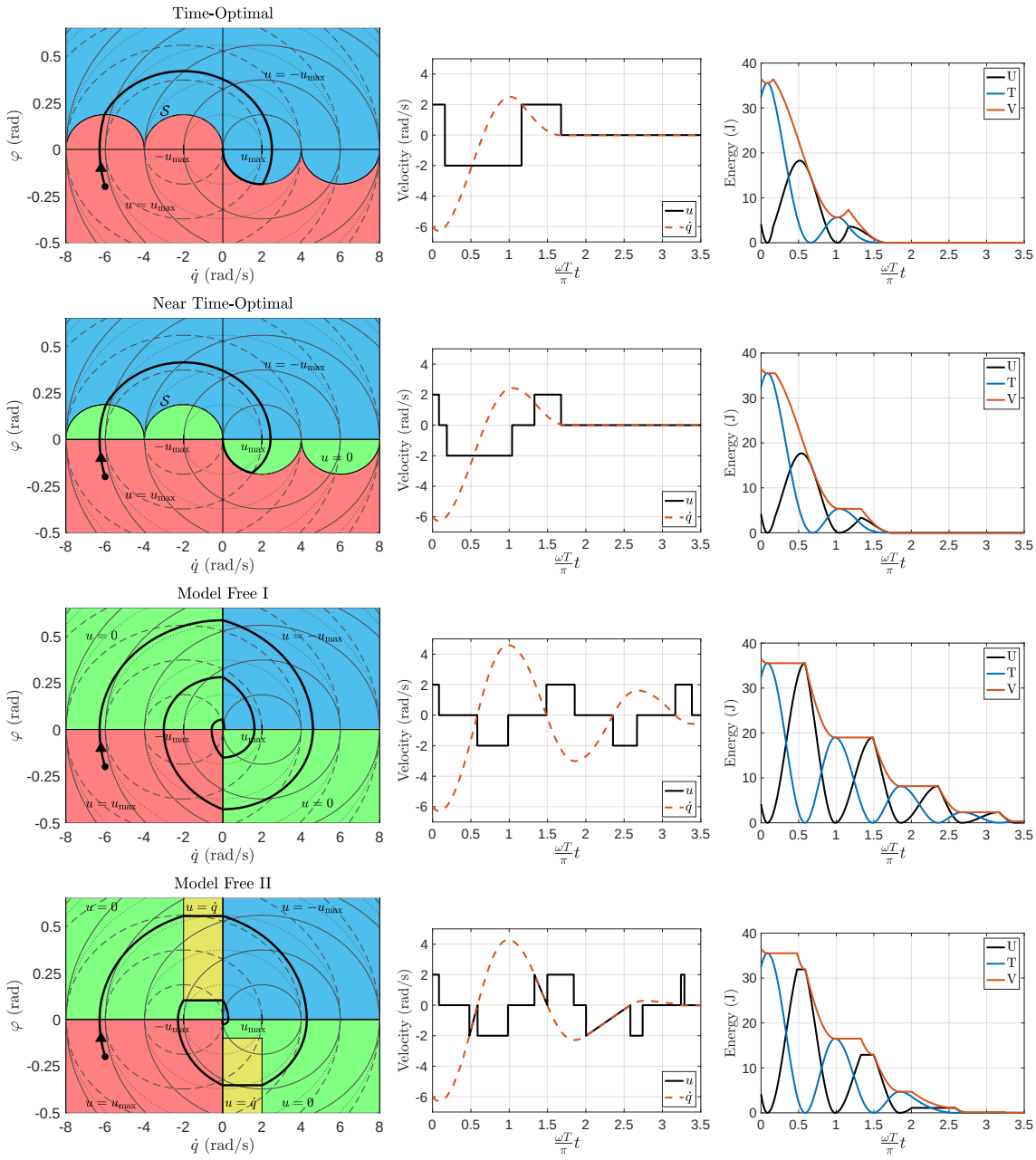


Fig. 5.10: Results for state-triggered control for two model-based and three model-free control laws for a linear, gravity-free mass-spring system.

terms of braking time but monotonic energy decrease, i.e., the controller shows passive behavior.

Near Time-Optimal, Passive Control

The time-optimal control law injects energy into the system in the area enclosed by the x-axis and the switching curve. Due to the problems mentioned above, this area should be avoided. Instead, one can select $u = 0$, which keeps the current system energy constant

and only slightly modifies the controller in order to achieve passivity. When applying zero velocity, the system state travels on an ellipse centered around the origin in the phase plane until the switching curve $\mathcal{S}$ is hit; see Fig. 5.10 (second row). The maximum motor velocity can then be commanded because torque/deflection and motor velocity have opposite signs, resulting in a total energy decrease.

The controller's phase plane representation together with an exemplary trajectory is depicted in Fig. 5.10 (second row), where the green areas represent zero commanded motor velocity and the blue and red area minimum and maximum motor velocity, respectively. From the evolution of the total system energy depicted in Fig. 5.10 it can be observed that passivity and only slightly larger braking time in comparison to time-optimal control are achieved. The previous two control laws require complete knowledge of the dynamics, particularly the torque/deflection curve. Next, three model-free control laws are described.

Model-Free Control I

The switching curves for time-optimal and near time-optimal control depend on the torque/deflection characteristic and can be difficult to derive, especially for joints with nonlinear stiffness. In [153], the model-free state-triggered control law

$$u_{MFI} = \begin{cases} -\mathrm{sign}(\varphi) u_{\max}, & \varphi \dot{q} \geq 0\,, \\ 0, & \text{otherwise}\,. \end{cases} \tag{5.23}$$

was proposed, which is the pendant of the state-proportional controller (5.11). In phase plane quadrants one and three, the minimum/maximum motor velocity is commanded. This also applies to the previous two model-based schemes, as there are no switching curves in these quadrants. In quadrants two and four, the desired motor velocity is zero, i.e., the energy remains constant. This leads to an overall energetically passive behavior of the controlled oscillator. The controller's phase plane representation and an exemplary trajectory are illustrated in Fig. 5.10 (third row). Please note that the origin can only be reached asymptotically with this control law. In other words, the joint cannot be stopped in finite time without frictional or damping effects in case the system trajectory does not travel on the first switching curve described previously. In the controller presented next, this model-free approach is modified to achieve a faster energy decrease.

Model-Free Control II

In addition to commanding maximum, minimum, or zero velocity, one can apply $u = \dot{q}$ if $|\dot{q}| \leq u_{\max}$. With this choice, the motor travels with the same velocity as the link. The elastic deflection, i.e., the potential energy remains the same while the kinetic energy is being reduced until $u = \dot{q} = 0$. In quadrant two and four, one can use $u = \dot{q}$ to reduce energy where the previous control scheme kept constant energy whenever $u = 0$; see Fig. 5.10 (fourth row). The control law is given by

$$u_{MFII} = \begin{cases} -\mathrm{sign}(\varphi) u_{\max}, & \varphi \dot{q} \geq 0\,, \\ \dot{q}, & \varphi \dot{q} < 0 \,\&\, |\dot{q}| \leq u_{\max} \,\&\, |\varphi| \geq \varphi_{\min}\,, \\ 0, & \text{otherwise}\,. \end{cases} \tag{5.24}$$

When commanding $u = \dot{q}$, the time to reach zero link velocity depends on the available elastic torque. If the elastic torque is small, then energy decreases only very slowly. In

this situation, one may achieve a faster energy decrease when keeping system energy with $u = 0$ and applying minimum/maximum motor velocity in the next quadrant (I or III). The problem can be avoided by applying $u = \dot{q}$ only when the elastic deflection is above a threshold $\varphi_{\min}$; see Fig. 5.10 (fourth row). The selection of $\varphi_{\min}$ is subject to future work.

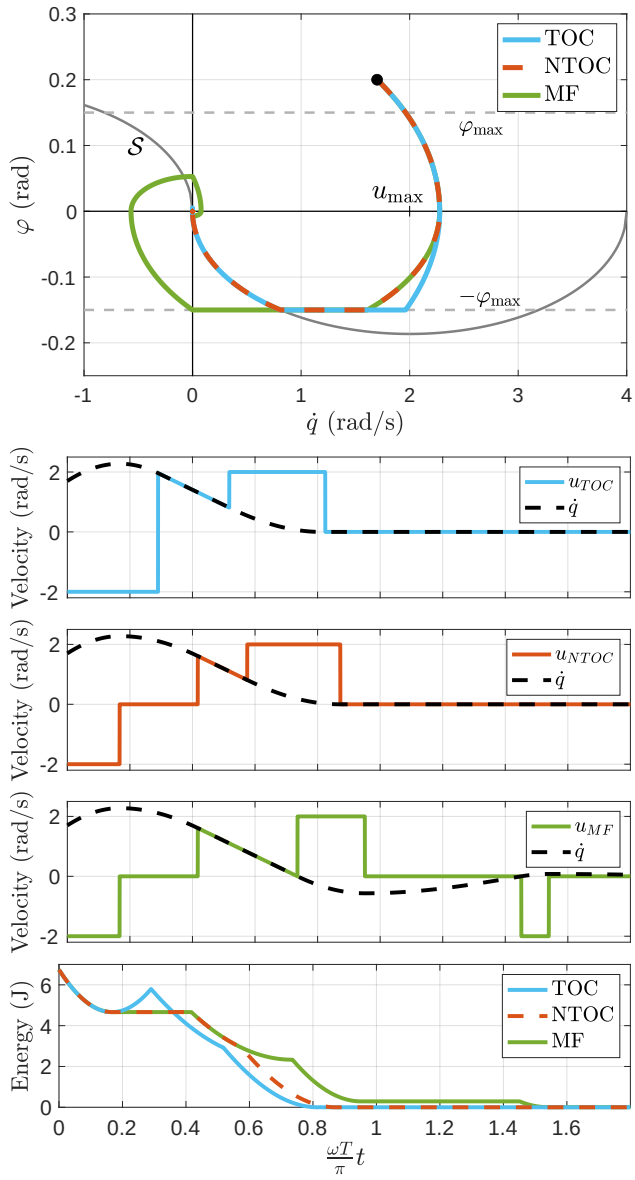


Fig. 5.11: Boundary control in case the maximum joint deflection $\varphi_{\max}$ intersects the switching curve $\mathcal{S}$. In the upper figure, exemplary braking trajectories are depicted in the phase plane for time-optimal (TOC), near time-optimal (NTOC), and model-free (MF) control. The corresponding motor and link velocities are depicted in figure two to four, the energies in the bottom figure. © 2015 IEEE [253].

Boundary Control

As described previously, the deflection constraint may be violated when commanding maximum or minimum motor velocity. Strategies to avoid constraint violation are illustrated in Fig. 5.11 (upper). When hitting the deflection constraint, one must apply $u = \dot{q}$ and keep the maximum deflection until the switching curve of the time-optimal control law is reached. After hitting the switching curve, maximum motor velocity must be commanded to reach equilibrium. Boundary control is only possible if the system reaches the constraint for link velocities $|\dot{q}| \leq u_{\max}$, otherwise the system cannot be brought to equilibrium without violating the deflection constraint. In contrast to the time-optimal solution, which always commands either minimum or maximum motor speed, the near time-optimal scheme commands zero motor velocity in the area enclosed by the first switching curve. This results in a passive behavior but in longer braking time; see Fig. 5.11 (lower). For the model-free controllers, one can command $u = \dot{q}$ along the constraint as long as $0 \leq |\dot{q}| \leq u_{\max}$.

5.3.3 Extension to n-DOF

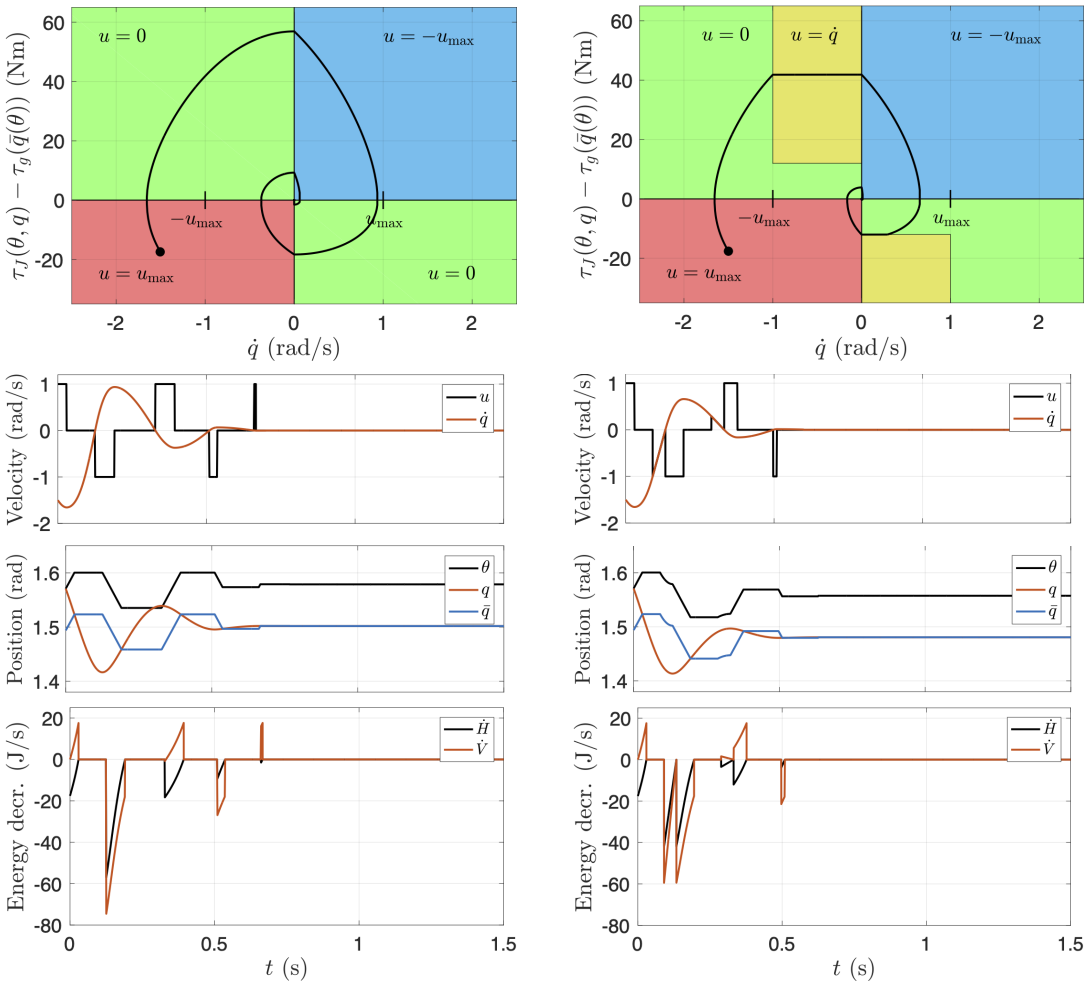
In case the spring stiffness is linear and gravity constant, all gravity-free controllers described in the previous section are applicable in the presence of gravity by replacing $\boldsymbol{\varphi}$ by $\bar{\boldsymbol{\varphi}}$, thereby minimizing $H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}})$ instead of $V(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{q}})$. However, constant gravity force/torque is usually present in academic systems like a vertical mass-spring system. For non-linear stiffness and constant gravity, braking control laws from the optimal control literature can be applied that take the respective torque/deflection relationship into account [254]. However, in general, both the torque/deflection relationship and the gravity torque are non-linear, making it difficult to derive such (time-)optimal control laws. Please note that in case of a planar, gravity-free joint, both the state-proportional and model-free bang-bang controllers can be implemented without knowing the system model. If gravity is present, then the spring torque/deflection curve and the gravity torque vector are required to compute $\bar{\boldsymbol{q}}(\boldsymbol{\theta})$; in other words, the controllers are no longer model-free. For n-DOF elastic joint robots, the considered model-free schemes can be implemented joint wise Control law (5.23) becomes

$$u_i = \begin{cases} -\operatorname{sign}(\Delta\tau_{J,i}(\theta_i, q_i))u_{\max,i}, & \Delta\tau_{J,i}(\theta_i, q_i)\dot{q}_i \geq 0\,, \\ 0, & \text{otherwise}\,, \end{cases} \tag{5.25}$$

and the second model-free controller (5.24) can be formulated as

$$u_i = \begin{cases} -\operatorname{sign}(\Delta\tau_{J,i}(\theta_i, q_i))u_{\max,i}, & \Delta\tau_{J,i}(\theta_i, q_i)\dot{q}_i \geq 0\,, \\ \dot{q}_i, & \Delta\tau_{J,i}(\theta_i, q_i)\dot{q}_i < 0 \,\&\, |\dot{q}_i| \leq u_{\max,i} \\ & \&\, |\Delta\tau_{J,i}(\theta_i, q_i)| \geq \Delta\tau_{J,\min,i}\,, \\ 0, & \text{otherwise}\,, \end{cases} \tag{5.26}$$

for joints $i = 1, \ldots, n$. As mentioned previously, one can replace $\Delta\boldsymbol{\tau}_J(\boldsymbol{\theta}, \boldsymbol{q})$ by $\bar{\boldsymbol{\varphi}}$ if the torque/deflection curve is strictly increasing. In Fig. 5.12, controller (5.25) and (5.26) are applied to a 1-DOF rotational joint with an DLR FSJ elastic mechanism and gravity. It can be observed that controller (5.26) reduces energy faster than control law (5.25). To avoid chattering in the vicinity of the equilibrium position, a switch to the state-proportional



(a) Controller (5.25) **(b)** Controller (5.26)

Fig. 5.12: Comparison of state-triggered control laws for a rotational joint with non-linear joint torque/deflection characteristic and gravity. In the left column, results for control law (5.25) are illustrated, results for (5.26) in the right column.

controller (5.13) occurs when the system energy becomes lower than a threshold. Real-world experiments are reported in Sec. 5.5, where the controllers are implemented and validated on DLR David.

Decoupling-Based Approach for Model-Based Controllers

Unfortunately, it is not possible to implement the time-optimal and near-time optimal control laws in joint space. The switching curve of these schemes requires knowledge about the maximum motor velocity, the torque/deflection curve, and the link inertia of the respective joint. Furthermore, it is impossible to assign a scalar mass to each joint because the robot mass matrix is coupled. A solution to this problem is implementing the control law in modal space, where the system is decoupled and SISO control can be applied. For elastic joint robots, such a decoupling-based approach was used in [135], e.g., where a state feedback damping controller was proposed.

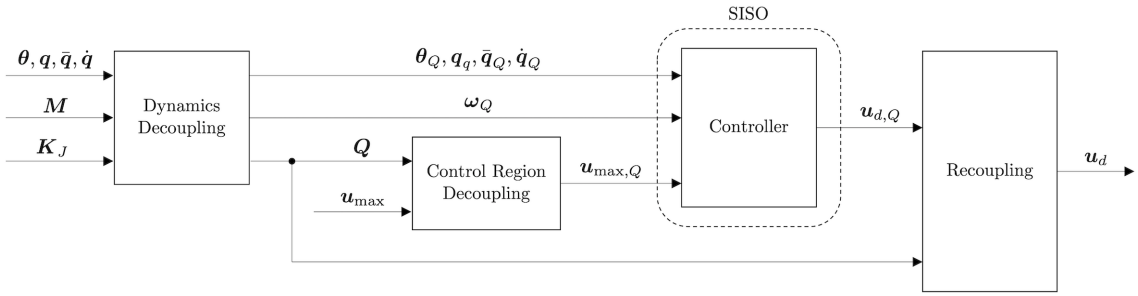


Fig. 5.13: Decoupling the system dynamics and input constraints to perform SISO control in modal coordinates.

Modal space control design usually involves the transformation of the dynamics from joint space to modal space, the controller application in the new modal coordinates, and the retransformation of the desired control variables to the original space. The proposed controller structure is depicted in Fig. 5.13. Here, $\boldsymbol{M} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{K}_J \in \mathbb{R}^{n\times n}$ denote the linearized mass and stiffness matrix, the decoupling transformation matrix is denoted by $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$, and modal space quantities are indicated by the subscript Q. The desired motor velocity is denoted by $\boldsymbol{u}_{d,Q} \in \mathbb{R}^n$ in modal space and $\boldsymbol{u}_d \in \mathbb{R}^n$ after retransformation to joint space. In order to apply the considered control laws in decoupled coordinates, we need to transform not only the system dynamics but also the input constraints (maximum motor velocities) to modal space. Unfortunately, not the same decoupling transformation $\boldsymbol{Q}$ can be applied for both the dynamics and input constraints. Loosely speaking, in joint space, the dynamics are coupled while the input constraints are decoupled, while in modal space, the dynamics are decoupled and the input constraints are coupled.

In the next section, this problem is discussed in detail. Several solutions for the missing control region decoupling and implementation of the control framework depicted in Fig. 5.13 are proposed.

5.4 Maximal Input Limits for Modal Space Control

In this section, modal decoupling is first briefly reviewed for more general second-order dynamic systems than the considered elastic joint robot. Then, the problem of deriving modal space input limits that satisfy the actuator constraints is addressed.

5.4.1 Problem Definition

Modal decoupling is a well-known tool commonly employed to analyze and control linear or linearized mechanical systems. It allows diagonalizing the system dynamics, which has several advantages. Firstly, one can gain insight into the natural oscillations of a mechanical system [255]. Secondly, the control complexity of large-order systems can be reduced significantly as the control design may take place in the simplified modal space instead of the original, coupled space [256, 257]. Many applications for such decoupled control exist. A standard problem is the damping of undesired vibrations in elastic systems [135, 247, 258, 259]. In robotics, modal decoupling was used for hybrid force/position control [260] and excitation of limit cycles [158], for example. One aspect that is often not considered is compliance with system constraints. In general, it is desirable/required to

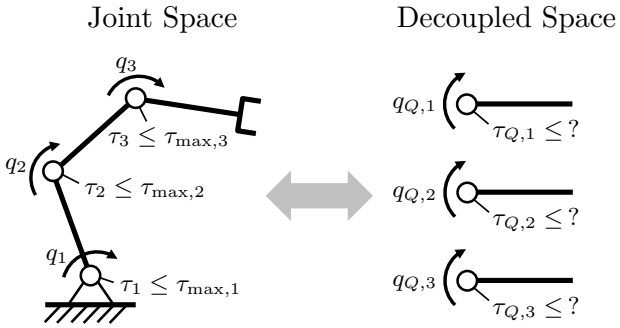


Fig. 5.14: Decoupling of a (linearized) 3-DOF robot with joint positions q_i and limited motor torques τ_i, $i = 1, \ldots, 3$ (only positive limits shown in figure). The modal positions $q_{Q,i}$ can be obtained by coordinate transformation, however, it is not clear how to determine the maximum torques in modal space. © 2016 IEEE [263].

meet them, and in particular, compliance with actuator saturations is a vivid research area in control theory. In Model Predictive Control (MPC) methods, for example, constraints are usually incorporated into the optimization procedure [261]. In linear control theory, there exists a large number of anti-windup techniques to respect saturations. These methods allow for an unconstrained controller design and add a compensator that becomes active as soon as a control variable is saturated [262].

In this work, the problem is considered of how input constraints can be transformed from original to modal space such that independent single-input single-output (SISO) control is made possible and the desired control variables meet the actuator constraints in original space. The problem is illustrated in Fig. 5.14.

5.4.2 Modal Decoupling

In this section, linear(ized), n-dimensional mass-spring-damper systems are considered. Three relevant systems and their corresponding original and modal space equations are depicted in Fig. 5.15. Their properties and the dynamics diagonalization are described in the following. System 1) is the classical dynamic system used in modal analysis. The control input is the force/torque $\boldsymbol{u} = \boldsymbol{f} \in \mathbb{R}^n$. Following input constraints are considered

$$u_{\min,i} \leq u_i \leq u_{\max,i}, \quad i = 1, \ldots, n\,, \tag{5.27}$$

where $u_{\min} < 0$ and $u_{\max,i} > 0$. These limits are not necessarily symmetric. This holds, e.g., for the available motor torque in robots after compensating for gravity. The symmetric, positive definite mass matrix is denoted by $\boldsymbol{M} \in \mathbb{R}^{n\times n}$, the damping matrix by $\boldsymbol{D} \in \mathbb{R}^{n\times n}$, and the stiffness matrix by $\boldsymbol{K} \in \mathbb{R}^{n\times n}$. It is assumed that the system is classically damped, i.e., the damping matrix is a linear combination of the mass and stiffness matrix

$$\boldsymbol{D} = \alpha_1 \boldsymbol{M} + \beta_1 \boldsymbol{K}\,, \tag{5.28}$$

where $\alpha_1 \in \mathbb{R}$ and $\beta_1 \in \mathbb{R}$. However, this is not the general case [264]. When solving the generalized eigenvalue problem, a nonsingular matrix $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$ can be obtained that is

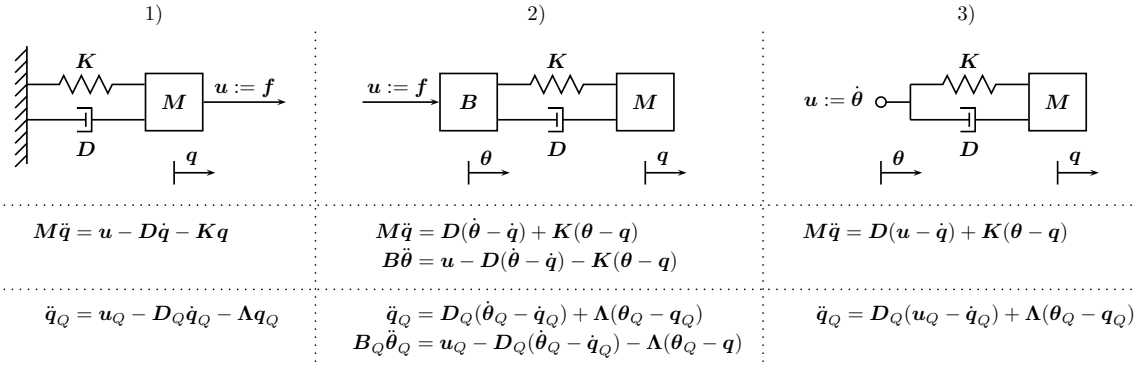


Fig. 5.15: Mass-spring-damper systems with their dynamic equations in original (middle row) and modal space (lower row). © 2016 IEEE [263].

orthogonal w.r.t. the mass, stiffness, and damping matrix, which gives us

$$\boldsymbol{Q}^\top \boldsymbol{M}\boldsymbol{Q} = \boldsymbol{I}\,, \tag{5.29a}$$

$$\boldsymbol{Q}^\top \boldsymbol{K}\boldsymbol{Q} = \boldsymbol{\Lambda}\,, \tag{5.29b}$$

$$\boldsymbol{Q}^\top \boldsymbol{D}\boldsymbol{Q} = \boldsymbol{D}_Q\,, \tag{5.29c}$$

where $\boldsymbol{I}$ is the $n \times n$ identity matrix, $\boldsymbol{D}_Q = \alpha_1 \boldsymbol{I} + \beta_1 \boldsymbol{\Lambda}$ a diagonal $n \times n$ matrix, and $\boldsymbol{\Lambda} = \mathrm{diag}\{\lambda_i\}, i = 1, \ldots, n$ a matrix containing the system's eigenvalues. If the product $\boldsymbol{K}^{-1}\boldsymbol{M}$ is symmetric, then $\boldsymbol{Q}$ is mutually orthogonal ($\boldsymbol{Q}^\top \boldsymbol{Q} = \boldsymbol{I}$). Substituting the coordinate transformation $\boldsymbol{q} = \boldsymbol{Q}\boldsymbol{q}_Q$ in the original space dynamics in Fig. 5.15 (middle left) and premultiplying by $\boldsymbol{Q}^\top$ yields the modal space dynamics (lower left). The transformed control input is

$$\boldsymbol{u}_Q = \boldsymbol{Q}^\top \boldsymbol{u}\,. \tag{5.30}$$

Since $\boldsymbol{D}_Q$ and $\boldsymbol{\Lambda}$ are diagonal, one obtains n decoupled equations with $\boldsymbol{u}_Q$ being the modal control input. System 2) represents systems with elastic transmission, e.g., flexible joint robots such as the LWR [49]. The motor position and mass are denoted by $\boldsymbol{\theta} \in \mathbb{R}^n$ and $\boldsymbol{B} \in \mathbb{R}^{n \times n}$, the link side quantities are the same as in system 1). The control input is the motor force, the motor and link side dynamics are coupled via the elastic joint force $\boldsymbol{f}_J = \boldsymbol{D}(\boldsymbol{u} - \dot{\boldsymbol{q}}) + \boldsymbol{K}(\boldsymbol{\theta} - \boldsymbol{q})$. To decouple the dynamics, the motor inertia matrix must fulfill (in analogy to the damping matrix)

$$\boldsymbol{B} = \alpha_2 \boldsymbol{M} + \beta_2 \boldsymbol{K}\,, \tag{5.31}$$

with $\alpha_2 \in \mathbb{R}$ and $\beta_2 \in \mathbb{R}$ to obtain a $n \times n$ diagonal matrix $\boldsymbol{B}_Q = \alpha_2 \boldsymbol{I} + \beta_2 \boldsymbol{\Lambda}$. Of course, $\boldsymbol{B}$ does not necessarily have the form (5.31). In [135] it was proposed to replace $\boldsymbol{B}$ by a matrix $\boldsymbol{B}'$ that is described as a linear combination of $\boldsymbol{M}$ and $\boldsymbol{K}$. The parameters α_2 and β_2 were optimized such that deviation of $\boldsymbol{B}$ from $\boldsymbol{B}'$ is as small as possible. For system 2) the coordinate transformation for the motor position is $\boldsymbol{\theta} = \boldsymbol{Q}\boldsymbol{\theta}_Q$, otherwise the decoupling procedure is the same as in the previous case. System 3) represents the reduced elastic joint robot dynamics (cf. Sec. 2.1), where the motors are modeled as velocity sources. While the input to system 1) and 2) is a force, it is a velocity in system 3). Here, the transformed control input is given by

$$\boldsymbol{u}_Q = \boldsymbol{Q}^{-1} \boldsymbol{u}\,. \tag{5.32}$$

The transformations (5.30) and (5.32) are identical if $\boldsymbol{Q}$ is orthonormal.

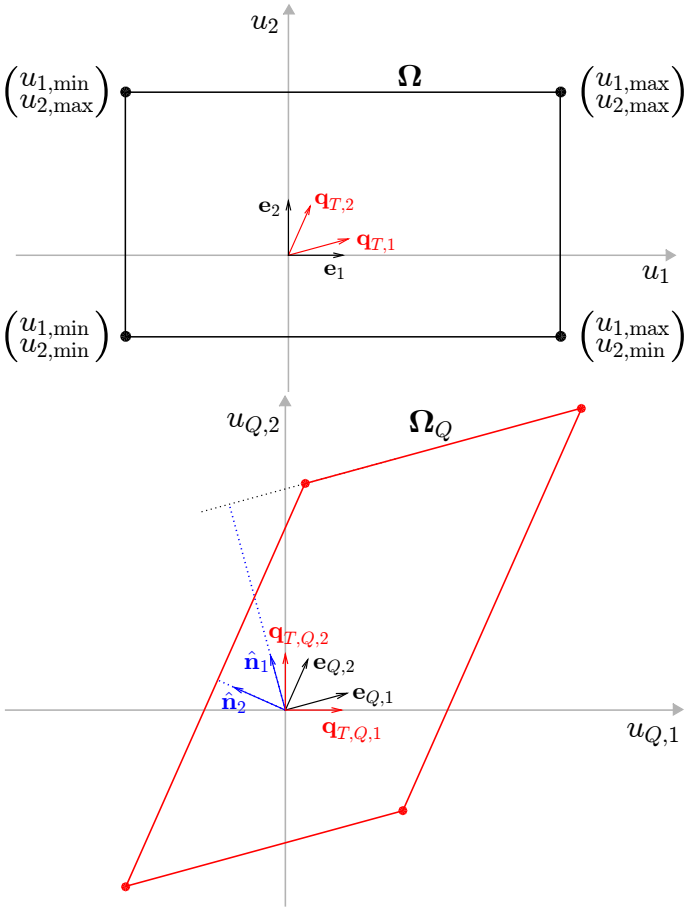


Fig. 5.16: Original control region $\mathbf{\Omega}$ and basis vectors $\boldsymbol{e}_i$ (original space) and $\boldsymbol{q}_{T,i}$ (modal space), represented in original coordinates (upper). In the lower figure, the basis vectors are represented in the decoupled space. The hyperplane normals are denoted by $\hat{\boldsymbol{n}}_i$. © 2016 IEEE [263].

5.4.3 Control Region Coupling

Using a coordinate transformation, one can bring the system dynamics 1) – 3) into a diagonal form. However, one property of this approach is that the control inputs $\boldsymbol{u}$ and thus also the actuator constraints (5.27) are coupled via the transformation (5.30), respectively (5.32). In the following, the original control region defined by the actuator limits and the transformed control region are described in detail. Then, the problem of determining suitable input values in decoupled space and its dependence on the control law is described. Finally, a simple dynamic system is used to exemplify the proposed methods.

Original and Transformed Control Region

Define the control region in original space as the hyperrectangle

$$\mathbf{\Omega} = [u_{\min,1}, u_{\max,1}] \times \cdots \times [u_{\min,n}, u_{\max,n}] \,, \tag{5.33}$$

which has $2n$ faces and 2^n vertices. Its basis vectors are denoted by $\boldsymbol{e}_i \in \mathbb{R}^n, i = 1, \ldots, n$. For each vector $\boldsymbol{e}_i$, the ith component is 1 while the other components are 0. The matrix

that contains all basis vectors is

$$\boldsymbol{E} = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n] = \boldsymbol{I}\,. \tag{5.34}$$

Assuming (5.30) (respectively (5.32)) is used to transform the control input, the basis vectors $\boldsymbol{q}_{T,i}, i = 1, \ldots, n$ of the decoupled space are the columns of the transformation matrix $\boldsymbol{Q}^\mathsf{T}$. An example for a system with two actuators (e.g., a double pendulum) is depicted in Fig. 5.16. In the upper figure, $\boldsymbol{\Omega}$ and $\boldsymbol{e}_i$ are represented in original space, in the lower figure in modal space. The transformed control region is defined as

$$\boldsymbol{\Omega}_Q = \boldsymbol{Q}^\mathsf{T}(\boldsymbol{\Omega})\,, \tag{5.35}$$

meaning every vertex of $\boldsymbol{\Omega}$ is transformed via $\boldsymbol{Q}^\mathsf{T}$. The transformation is linear, i.e., the hyperrectangle may be scaled, rotated, reflected, and/or sheared. The transformed control region $\boldsymbol{\Omega}_Q$ is thus a parallelotope. The basis vectors $\boldsymbol{e}_i$, represented in modal coordinates, are parallel to the parallelotope faces. When transforming all basis vectors to modal space, we get

$$\boldsymbol{E}_Q = \boldsymbol{Q}^\mathsf{T}\boldsymbol{E} = \boldsymbol{Q}^\mathsf{T}\,, \tag{5.36}$$

with $\boldsymbol{E}_Q = [\boldsymbol{e}_{Q,1}, \ldots, \boldsymbol{e}_{Q,n}]$. The vectors parallel to the parallelotope faces are thus simply given by the columns of $\boldsymbol{Q}^\mathsf{T}$. The parallelotope faces can be represented by n-dimensional hyperplanes. To formally describe these planes, n normals and $2n$ distances (positive and negative direction) from the origin to the planes are required. The Hesse normal form of the two parallel planes that have the same normal $\hat{\boldsymbol{n}}_i \in \mathbb{R}^n$ is

$$\hat{\boldsymbol{n}}_i^\mathsf{T}\boldsymbol{u}_Q = d_{\max,i}\,, \tag{5.37a}$$

$$\hat{\boldsymbol{n}}_i^\mathsf{T}\boldsymbol{u}_Q = d_{\min,i}\,, \tag{5.37b}$$

where $d_{\max,i} \in \mathbb{R}$ and $d_{\min,i} \in \mathbb{R}$ are the distances from the origin to the planes in positive/negative direction along $\hat{\boldsymbol{n}}_i$. The plane normal $\hat{\boldsymbol{n}}_i$ is orthogonal to the $n-1$ vectors $\boldsymbol{e}_{Q,j} = \boldsymbol{q}_{T,j}, j \in \{1, \ldots, n\}\backslash i$. To obtain $\hat{\boldsymbol{n}}_i$, the columns of $\boldsymbol{Q}^\mathsf{T}$ are normalized first

$$\hat{\boldsymbol{Q}}^\mathsf{T} = \left[\hat{\boldsymbol{q}}_{\mathsf{T},1}, \ldots, \hat{\boldsymbol{q}}_{\mathsf{T},n}\right] = \left[\frac{\boldsymbol{q}_{\mathsf{T},1}}{||\boldsymbol{q}_{\mathsf{T},1}||}, \ldots, \frac{\boldsymbol{q}_{\mathsf{T},n}}{||\boldsymbol{q}_{\mathsf{T},n}||}\right]\,. \tag{5.38}$$

Then, $\hat{\boldsymbol{Q}}^\mathsf{T}$ is transposed and the ith row is omitted to obtain the $(n-1) \times n$ matrix $\hat{\boldsymbol{Q}}_{\backslash i}$. The kernel (null space) of this matrix is $\hat{\boldsymbol{n}}_i$, since

$$\ker(\hat{\boldsymbol{Q}}_{\backslash i}) = \{\hat{\boldsymbol{n}}_i \mid \hat{\boldsymbol{Q}}_{\backslash i}\,\hat{\boldsymbol{n}}_i = \boldsymbol{0}\}\,. \tag{5.39}$$

Having determined the plane normals, one obtains the distances $\boldsymbol{d}_{\max} = [d_{\max,1}, \ldots, d_{\max,n}]^\mathsf{T}$ and $\boldsymbol{d}_{\min} = [d_{\min,1}, \ldots, d_{\min,n}]^\mathsf{T}$ from the origin to the planes by

$$\boldsymbol{d}_{\max} = \hat{\boldsymbol{N}}^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{u}_{\max}\,, \tag{5.40a}$$

$$\boldsymbol{d}_{\min} = \hat{\boldsymbol{N}}^\mathsf{T}\boldsymbol{Q}^\mathsf{T}\boldsymbol{u}_{\min}\,, \tag{5.40b}$$

where $\boldsymbol{u}_{\max} = [u_{\max,1} \ldots, u_{\max,n}]^\mathsf{T}$ and $\boldsymbol{u}_{\min} = [u_{\min,1}, \ldots, u_{\min,n}]^\mathsf{T}$. The $n \times n$ matrix assembled by the hyperplane normals (5.39) is denoted by $\hat{\boldsymbol{N}} = [\hat{\boldsymbol{n}}_1, \ldots, \hat{\boldsymbol{n}}_n]$. The formal definition of the parallelotope bounds will help us later to derive suitable modal input constraints.

Dependence of Modal Input Limits on Controller Type

For some control laws that command the maximum input value, both the current system state and the input limits must be known to determine the control variable. Other controllers, however, only require information about the current system state. Control laws thus have different requirements to determine the control variable. As shown in the next section, these requirements influence the optimization of the modal input limits. Three principal cases can be identified. Without prior knowledge of the modal control limits

Case 1) the sign and absolute value of the modal control variable are unknown,

Case 2) its sign is known, but not its absolute value, or

Case 3) both its sign and absolute value are known to the controller.

In this work, modal input limits for cases 1 and 2 shall be determined that a) are as large as possible and b) do not violate the real actuator limits after retransformation to original space. In case 3, no limit values are required to determine the desired control input. Instead, the modal input constraints can limit the desired control variable to a realizable value if necessary.

Exemplary 2-DOF System

To motivate and describe the requirements of all three cases, a velocity-controlled, 2-DOF elastic system is used, e.g., a linearized double pendulum. The joint space dynamics are

$$\boldsymbol{M}\ddot{\boldsymbol{q}} = \boldsymbol{K}(\boldsymbol{\theta} - \boldsymbol{q})\,, \tag{5.41a}$$

$$\boldsymbol{\theta} = \int \dot{\boldsymbol{\theta}}\; dt + \boldsymbol{\theta}_0\,, \tag{5.41b}$$

where $\boldsymbol{M} \in \mathbb{R}^{2\times 2}$, $\boldsymbol{K} \in \mathbb{R}^{2\times 2}$, and $\boldsymbol{u} = \dot{\boldsymbol{\theta}} \in \mathbb{R}^2$, c.f. Fig. 5.15 (upper right). The same original actuator constraints are selected and the same decoupling transformation as in the example depicted in Fig. 5.16 are assumed. After transforming to modal space, one obtains two harmonic oscillators. Each oscillator has the second-order SISO dynamics

$$\ddot{q}_Q = w_Q^2(\theta_Q - q_Q)\,, \tag{5.42a}$$

$$u_Q = \dot{\theta}_Q\,, \tag{5.42b}$$

where the quadratic eigenfrequency w_Q^2 is an entry of the diagonal matrix $\boldsymbol{\Lambda}$ that contains the eigenvalues of $\boldsymbol{M}^{-1}\boldsymbol{K}$. Here, the index $i \in \{1, 2\}$ is omitted for the sake of brevity. It is possible to describe the first-order dynamics by selecting the state $\boldsymbol{x}_Q = [\dot{q}_Q, \varphi_Q]^\mathsf{T}$, where the elastic deflection is $\varphi_Q = \theta_Q - q_Q$. The control aim in the following section is to bring the system from an arbitrary initial state $\boldsymbol{x}_Q(0)$ to the equilibrium $\boldsymbol{x}_Q = \boldsymbol{0}$. In each of the three cases 1 – 3, an exemplary control law is provided and applied to both modal oscillators independently. Please note that all algorithms are developed for n-DOF while the example is only 2-DOF. Examples with more degrees of freedom are provided in Sec. 5.4.7, experiments on DLR David in Sec. 5.5.

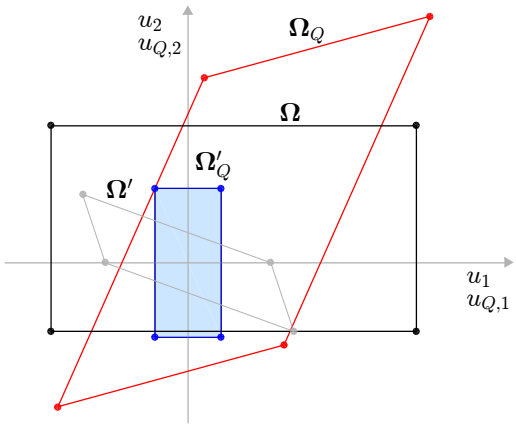


Fig. 5.17: The blue control region $\mathbf{\Omega}'_Q$ in decoupled space has the largest possible volume that ensures symmetric maximum and minimum bounds for each modal control input. The retransformed region is denoted by $\mathbf{\Omega}'$. The modal and original inputs share the same axes in the figure. © 2016 IEEE [263].

5.4.4 Case 1: Sign and Absolute Value of Control Input Unknown

Some control laws depend on both the system state $\boldsymbol{x}_Q$ and the maximum control input $\boldsymbol{u}_{\max,Q}$, e.g., the time-optimal control law described in Sec. 5.3.2. Assuming $u_{\min,Q} = -u_{\max,Q}$, one obtains the time-optimal input u_Q by evaluating whether the current state $x_{Q,2}$ is below or above the switching curve $\mathcal{S}$; see Fig. 5.10. The switching curve depends on both the eigenfrequency and the absolute value of $u_{\max,Q}$. The sign of u_Q can only be determined if $u_{\max,Q}$ is known in advance. In order to implement such a controller, symmetric and independent limits are required in modal space, i.e., $u_{\min,Q,i} = -u_{\max,Q,i}, i = 1, \ldots, n$. The modal input constraints $\boldsymbol{u}_{\min,Q}$ for this case are determined in the following.

Optimization of Symmetric Hyperrectangle Volume

Geometrically speaking, one wants to increase the size of a hyperrectangle, which has symmetric limits w.r.t. the origin, until the boundary of $\mathbf{\Omega}_Q$ is hit. To solve this problem, a static, constrained, nonlinear optimization problem is formulated. The goal is to maximize the volume of the modal control region; the cost function is

$$J = -\prod_{i=1}^{n} |u_{Q,i}| \,. \tag{5.43}$$

Here, all modal inputs are weighted equally. If certain coordinates are more important than others, e.g., if undesired high/low-frequency vibrations shall be damped, then inequality or equality constraints can be introduced such as $u_{Q,1} \geq \alpha u_{Q,2}$, where α is a scalar constant. If one or more modal inputs shall be zero, then the respective inputs $u_{Q,i}$ are excluded from the cost function (5.43). For a uniform scaling, i.e., the modal control region has the same aspect ratio as the original control region, scaling can solve the problem instead of an optimization problem. This problem is addressed later in this section. The cost function (5.43) is subject to the bounds of the transformed control region $\mathbf{\Omega}_Q$. The hyperplane definitions described in Sec. 5.4.3 constitute the $2n$ inequality constraints on the control

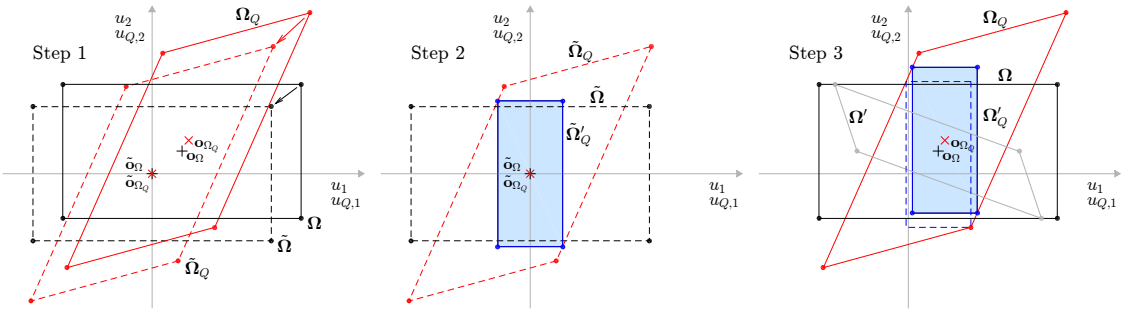


Fig. 5.18: Determining the hyperrectangle with maximum volume in modal space. In step 1, the original and transformed control region $\boldsymbol{\Omega}$ and $\boldsymbol{\Omega}_Q$ are translated to the origin to obtain $\tilde{\boldsymbol{\Omega}}$ and $\tilde{\boldsymbol{\Omega}}_Q$. In step 2, the optimization is carried out to compute $\tilde{\boldsymbol{\Omega}}_Q^{'}$. In the last step, the optimization result is translated again; $\boldsymbol{\Omega}_Q'$ is the desired control region in modal space. Here, the size of $\boldsymbol{\Omega}_Q'$ is unique but not its position. The dashed blue line represents an alternative solution. © 2016 IEEE [263].

inputs

$$|\hat{\boldsymbol{N}}|\boldsymbol{u}_Q \leq |\boldsymbol{d}_{\max}|\,, \tag{5.44a}$$

$$|\hat{\boldsymbol{N}}|\boldsymbol{u}_Q \leq |\boldsymbol{d}_{\min}|\,. \tag{5.44b}$$

Since the desired control region is symmetric, one can use the absolute values of $\hat{\boldsymbol{N}}$, $\boldsymbol{d}_{\max}$, and $\boldsymbol{d}_{\min}$ to restrict the search for an optimal solution to the first quadrant, where $\boldsymbol{u}_Q > \boldsymbol{0}$.

The result for the control bounds and decoupling transformation shown in Fig. 5.16 is illustrated in Fig. 5.17. Please note that the original and modal quantities share the same coordinate axes in the figure. The blue area, denoted by $\boldsymbol{\Omega}_Q'$, represents all feasible inputs for this problem. Blue dots represent the limit values. Only one constraint is active in this example. After retransforming $\boldsymbol{\Omega}_Q'$ to original space (grey rectangle $\boldsymbol{\Omega}'$), it can be observed that no actuator constraints are violated. The solution is thus feasible.

Maximum Possible Hyperrectangle Volume

The previous solution $\boldsymbol{\Omega}'$ has a relatively small volume compared to the original control region $\boldsymbol{\Omega}$; see Fig. 5.17. Only one out of four possible vertices in decoupled space reaches an actuator limit. Obviously, it is possible to find a larger hyperrectangle inside $\boldsymbol{\Omega}_Q$ if the original bounds $u_{\max,i}$ and $u_{\min,i}, i = 1, \ldots, n$ are not symmetric w.r.t. the origin. To determine the largest possible hyperrectangle, the requirement $u_{\min,Q,i} = -u_{\max,Q,i}, i = 1, \ldots, n$ is now dropped. By applying some modifications, one can solve a similar optimization problem[3]. The procedure consists of three steps which are depicted in Fig. 5.18 and described in the following.

Step 1 First, $\boldsymbol{\Omega}$ is translated such that its center $\boldsymbol{o}_\Omega$ coincides with the origin. The center is located at $\boldsymbol{o}_\Omega = [o_1, \ldots, o_n]^\mathsf{T}$, where $o_i = (u_{i,\max} + u_{i,\min})/2, i = 1, \ldots, n$. This gives us new, symmetric control limits

$$u_{\text{symm},i} = (u_{\max,i} - u_{\min,i})/2, \quad i = 1, \ldots, n\,. \tag{5.45}$$

[3]Remark: Computing the maximum axis-aligned (hyper-)rectangle among obstacles is a common problem in computational geometry and is often referred to as the *maximum empty rectangle problem* [265,266].

By replacing the maximal input limits $\boldsymbol{u}_{\max}$ in (5.40a) by $\boldsymbol{u}_{\text{symm}} = [u_{\text{symm},1}, \ldots, u_{\text{symm},n}]^{\mathsf{T}}$ one obtains the modified distances to the hyperplanes

$$\boldsymbol{d}_{\max} = \hat{\boldsymbol{N}}^{\mathsf{T}} \boldsymbol{Q}^{\mathsf{T}} \boldsymbol{u}_{\text{symm}} \,. \tag{5.46}$$

The plane normals $\hat{\boldsymbol{N}}$ remain the same. The new inequality constraints are

$$|\hat{\boldsymbol{N}}| \boldsymbol{u}_Q \leq |\boldsymbol{d}_{\max}| \,. \tag{5.47}$$

Since the hyperplanes are point symmetric w.r.t. the origin, one only requires n inequality constraints instead of $2n$ constraints as in the previous problem.

Step 2 Maximize the hyperrectangle volume by solving the nonlinear optimization problem (5.43).

Step 3 Translate the optimization result back to the origin of $\boldsymbol{\Omega}_Q$, which is

$$\boldsymbol{o}_{\Omega_Q} = \boldsymbol{Q} \boldsymbol{o}_{\Omega} \,, \tag{5.48}$$

see Fig 5.18 (right). The feasible control region in modal space has the largest possible volume; however, it may occur that the origin is not part of $\boldsymbol{\Omega}'_Q$; see Fig 5.18. This means that it may not be possible to command both positive and negative inputs in each coordinate, which is crucial for most controllers. In Fig. 5.18 (right) the first modal input only takes positive values. However, the position of $\boldsymbol{\Omega}'_Q$ inside $\boldsymbol{\Omega}_Q$ is not unique if not all constraints are active. In Fig. 5.18 (right) it can be seen that $\boldsymbol{\Omega}'_Q$ can be translated while still satisfying the constraints. By translating the control region, it is possible in this example to include the origin inside $\boldsymbol{\Omega}'_Q$. The alternative solution is indicated by a dashed blue line in Fig. 5.18 (right). The optimization problem can therefore be extended by minimizing the distance from the center of $\boldsymbol{\Omega}'_Q$ to the origin. For the sake of brevity, this problem is left to future work.

5.4.5 Case 2: Sign of Control Input Known, Absolute Value Unknown

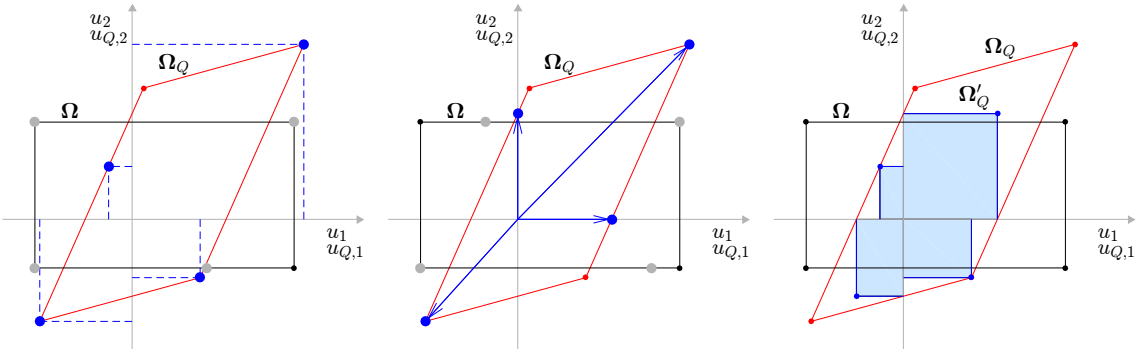
In the second case, controllers provide information about the sign of the control variable given the current system state only. The magnitude of the control variable still needs to be determined. An exemplary control for this case is (5.23)

$$u_Q = \begin{cases} u_{\min,Q} & \text{if } x_{Q,1} \geq 0 \,\&\, x_{Q,2} > 0 \,, \\ u_{\max,Q} & \text{if } x_{Q,1} \leq 0 \,\&\, x_{Q,2} < 0 \,, \\ 0 & \text{otherwise,} \end{cases} \tag{5.49}$$

which reduces the kinetic and potential energy of the harmonic oscillator (5.42). Here, the input limit values are exploited for fast energy decrease. Given the sign of $u_{Q,i}$ provided $\boldsymbol{x}_{Q,i}, i = 1, \ldots, n$, the modal control limits shall be maximized in the respective quadrant. For this, a constrained optimization problem is again formulated. To search for an optimal solution in a specific quadrant, one has to take the sign of the desired $\boldsymbol{u}_Q$ into account. The $2n$ inequality constraints for this case are formed by the parallelotope faces

$$\hat{\tilde{\boldsymbol{N}}}_{\max} \boldsymbol{u}_Q \leq |\boldsymbol{d}_{\max}| \,, \tag{5.50a}$$

$$\hat{\tilde{\boldsymbol{N}}}_{\min} \boldsymbol{u}_Q \leq |\boldsymbol{d}_{\min}| \,, \tag{5.50b}$$



(a) Maximum volume **(b)** Maximum vector norm **(c)** Conservative volume

Fig. 5.19: Maximization of (a) the volume and (b) the quadratic vector norm of $\boldsymbol{u}_Q$ when its sign (quadrant) is known in advance. A separate optimization was carried out for each quadrant. The blue dots indicate the optimal solution in modal space, the grey dots are the retransformed input values. In (c), the region $\mathbf{\Omega}'_Q$ represents the maximum possible volume in every quadrant of the modal space, where independent $u_{Q,i}, i = 1, \ldots, n$ can be commanded. © 2016 IEEE [263].

where

$$\hat{\tilde{\boldsymbol{N}}}_{\max} = \operatorname{diag}\{\operatorname{sign}(\boldsymbol{u}_Q)\}\, \hat{\boldsymbol{N}}\, \operatorname{diag}\{\operatorname{sign}(\boldsymbol{d}_{\max})\}\,, \tag{5.51a}$$

$$\hat{\tilde{\boldsymbol{N}}}_{\min} = \operatorname{diag}\{\operatorname{sign}(\boldsymbol{u}_Q)\}\, \hat{\boldsymbol{N}}\, \operatorname{diag}\{\operatorname{sign}(\boldsymbol{d}_{\min})\}\,. \tag{5.51b}$$

The solutions for all four quadrants are depicted in Fig. 5.19 (left). In quadrants I, III, and IV, the optimization result coincides with a vertex of $\mathbf{\Omega}_Q$. In the second quadrant, however, $\mathbf{\Omega}_Q$ has no vertex. Therefore, one of the parallelotope faces is the active constraint.

Remark: Alternative Cost Functions

To determine modal input limits that are as large as possible, the volume spanned by the inputs is selected as a cost function. The solution for every quadrant is a point in $\mathbb{R}^n$. The volume is, however, only one of several possible cost functions. An alternative is the quadratic vector norm

$$J = -\sum_{i=1}^{n} w_i u_{Q,i}^2\,, \tag{5.52}$$

where $w_i, i = 1, \ldots, n$ are coordinate weights. The optimization outcome for identical weights is illustrated in Fig. 5.19 (b). In quadrants I and III, the solution is identical to volume maximization. In quadrant II and IV, however, the maximum norm is given by the vector, which points from the origin to an intersection of the parallelotope with the $u_{Q,1}/u_{Q,2}$-axis. In quadrant II, the maximum norm vector is aligned with the $u_{Q,2}$-axis and hits the intersection of $\mathbf{\Omega}_Q$ and this axis. While $u_{Q,2}$ is maximized, $u_{Q,1}$ is zero. This is disadvantageous for most controllers, as no control action is possible for the first decoupled system.

Conservative Volume

Instead of solving the optimization problem, we can find a computationally cheaper yet more conservative solution, which is illustrated in Fig. 5.19 (right) and can be determined

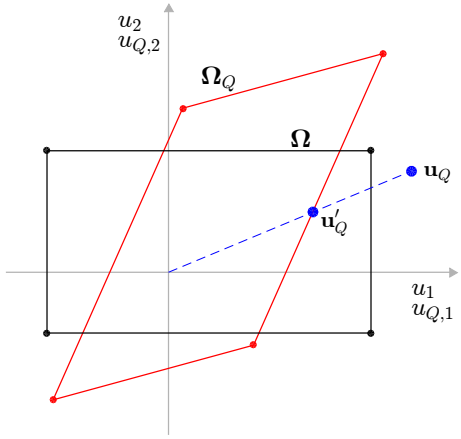


Fig. 5.20: Limitation of a control input that is located outside of the permissible control region $\mathbf{\Omega}_Q$. The new input $\boldsymbol{u}'_Q$ is given by the intersection of $\mathbf{\Omega}_Q$ and the (blue dashed) line between the origin and $\boldsymbol{u}_Q$. © 2016 IEEE [263].

as follows. First, the intersection of the hyperplanes and the coordinate axes is determined. The intersection with the minimum distance to the origin along the coordinate axis then defines the maximum permissible value for this coordinate. By comparing the result to Fig. 5.17, one can see that the conservative volume is significantly larger if the sign of $\boldsymbol{u}_Q$ is known.

5.4.6 Case 3: Sign and Absolute Value of Control Input Known

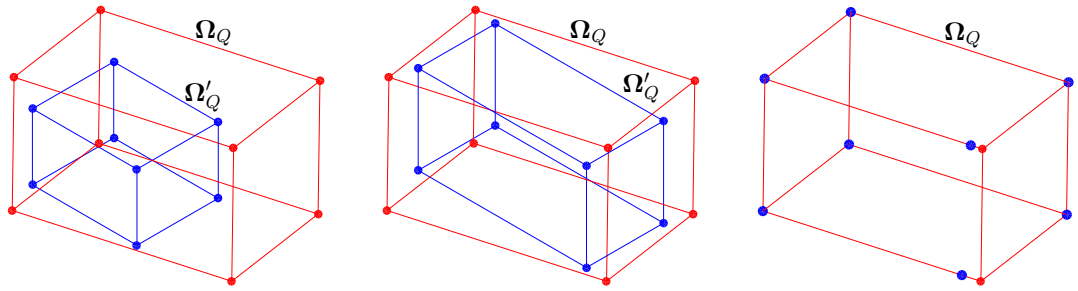
In case 3, controllers are considered that do not require information about actuator limit values in order to determine the control input $\boldsymbol{u}_Q$. For the harmonic oscillator, this could be controller (5.8)

$$u_Q = -K_C x_{Q,2}\,, \tag{5.53}$$

Since u_Q is proportional to the system state, one does not need to compute feasible limits $u_{\max,Q}$ and $u_{\min,Q}$ via optimization. Given $\boldsymbol{u}_Q$ for all SISO controllers, one can transform the control variable back to the original space via $\boldsymbol{u} = \boldsymbol{Q}\boldsymbol{u}_Q$ and check whether the original actuator constraints (5.27) are violated or not. If the desired control input is located within $\mathbf{\Omega}$, it can be applied to the actuators. If one or more limit values are exceeded, then a possible solution is to scale $\boldsymbol{u}_Q$ uniformly. This method is illustrated in Fig. 5.20. The intersection of $\mathbf{\Omega}_Q$ and the line from the origin to $\boldsymbol{u}_Q$ provides a feasible modal control input that is located in $\mathbf{\Omega}$ after retransformation to original space. This method is, however, only one of several possible solutions.

5.4.7 Application to Higher-Order Systems

Up to this point, a 2-DOF example was used to illustrate the optimization procedures. In this section, simulation examples for 3− and 7-DOF systems are provided. Experiments with DLR David are described in Sec. 5.5.



(a) Maximum volume, symmetric limits **(b)** Maximum volume **(c)** Goal quadrant known

Fig. 5.21: 3-DOF example of modal control limit maximization. In (a), the volume is maximized while ensuring symmetric limits w.r.t. the origin (case 1). In (b), the volume is maximized when dropping the requirement $u_{\min,Q,i} = -u_{\max,Q,i}, i = 1, \ldots, n$ (case 1). In (c), the volume is maximized in each quadrant separately (case 2). © 2016 IEEE [263].

Static 3-DOF Example

In Fig. 5.21, the optimization methods of cases 1 and 2 are applied to a linear academic system with three degrees of freedom. In (a), the volume spanned by the modal control limits is maximized such that the bounds are symmetric w.r.t. the origin. In (b), the maximum possible volume is depicted. As in the 2-DOF case, it can be observed that the maximum possible volume is significantly larger than the one obtained when demanding symmetry w.r.t. the origin. In (c), the sign of the desired control variable is taken into account (case 2). The maximum volume is determined for each quadrant, which better exploits the available input constraints than the previous two methods. Next, the problem of stopping a 7-DOF rigid joint robot is addressed.

DLR LWR

A hypothetically rigid-joint 7-DOF LWR with dynamics

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}\,, \tag{5.54}$$

is considered. The motor torques are bounded by[4] $|\tau_i| \leq \tau_{\max,i}, i = 1, \ldots, 7$. Part of the available motor torque is spent for gravity compensation. The configuration dependent bounds are

$$\tau'_{\min,i}(\boldsymbol{q}) = -\left(\tau_{\max,i} + g_i(\boldsymbol{q})\right) < 0\,, \tag{5.55a}$$

$$\tau'_{\max,i}(\boldsymbol{q}) = \tau_{\max,i} - g_i(\boldsymbol{q}) > 0\,, \tag{5.55b}$$

These bounds are typically asymmetric. The control law

$$\tau'_i = \left\{ \begin{array}{ll} \tau'_{\min,i}(\boldsymbol{q}) & \text{if } \dot{q}_i \geq \varepsilon_i \\ \tau'_{\min,i}(\boldsymbol{q})\dot{q}_i/\varepsilon_i & \text{if } \varepsilon_i > \dot{q}_i \geq 0 \\ -\tau'_{\max,i}(\boldsymbol{q})\dot{q}_i/\varepsilon_i & \text{if } -\varepsilon_i < \dot{q}_i \leq 0 \\ \tau'_{\max,i}(\boldsymbol{q}) & \text{if } \dot{q}_i \leq -\varepsilon_i \end{array} \right\}, \tag{5.56}$$

$$\boldsymbol{\tau} = \boldsymbol{\tau}' + \boldsymbol{g}(\boldsymbol{q})\,, \tag{5.57}$$

[4] Constraints on the velocities and positions are omitted.

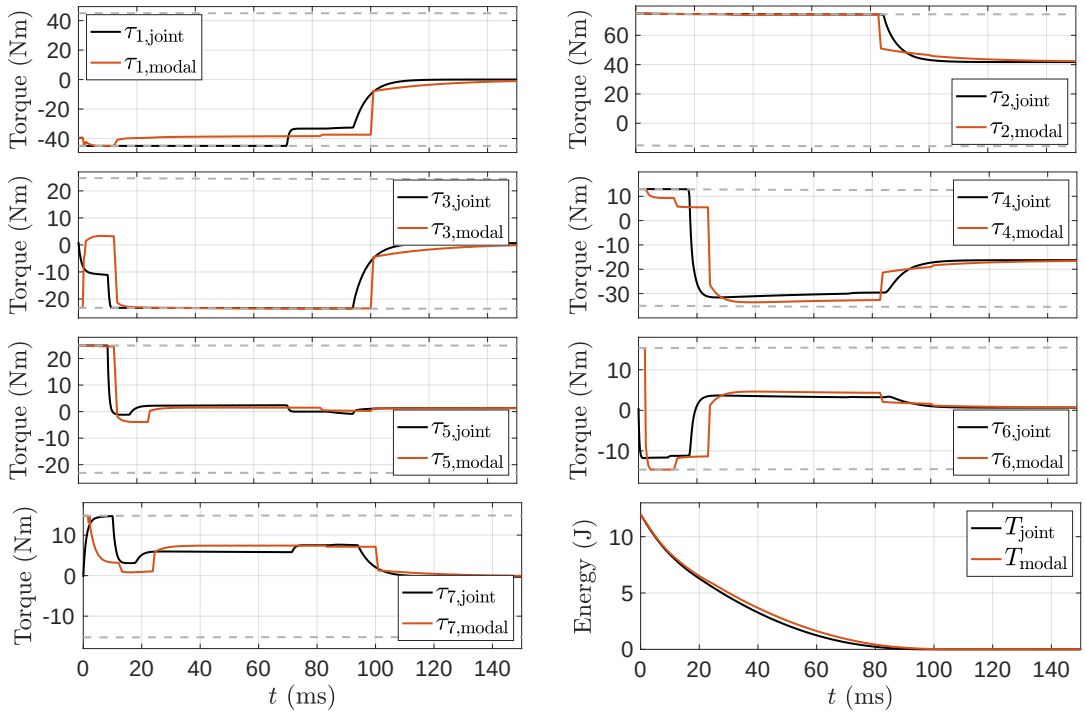


Fig. 5.22: Recorded motor torques and kinetic energy from simulated braking trajectories (joint and modal space control) of a LWR. Dashed red lines represent the time-varying motor torque bounds. © 2016 IEEE [263].

is implemented with $2\varepsilon_i > 0$ being a region in which the equilibrium $\dot{\boldsymbol{q}} = \boldsymbol{0}, \ddot{\boldsymbol{q}} = \boldsymbol{0}$ is reached smoothly [104]. This control law is formulated in joint space. To implement the controller in decoupled space, the system dynamics are linearized along the braking trajectory. In modal space, the velocity $\dot{q}_i$ is replaced by $\dot{q}_{Q,i}$ and the bounds $\tau'_{\text{min},Q,i}$ and $\tau'_{\text{max},Q,i}$ are determined via optimization (case 2). Having determined the modal control variable $\boldsymbol{\tau}'_Q$, one obtains the motor torques via $\boldsymbol{\tau} = \boldsymbol{Q}\boldsymbol{\tau}'_Q + \boldsymbol{g}(\boldsymbol{q})$. In simulation, the above control law is implemented in both joint and modal space. One would like to know how the joint-space and modal-space implementation compare in terms of performance. Furthermore, it shall be checked whether feasible modal input limits for dynamic, asymmetric motor torque limits are obtained. The symmetric motor torque constraints are $\boldsymbol{\tau}^{\mathsf{T}}_{\text{max}} = [150, 150, 80, 80, 80, 50, 50]$ Nm. After compensating for gravity, conservative torque bounds are used, namely 30 % of $\boldsymbol{\tau}'_{\text{max}}$. The initial position and velocity are $\boldsymbol{q}_0^{\mathsf{T}} = [0, 45, 0, -75, 0, 75, 0]$ ° and $\dot{\boldsymbol{q}}_0^{\mathsf{T}} = [100, -100, 0, -100, -100, 0, 0]$ °/s. The motor torques and the robot kinetic energy $T = \frac{1}{2}\dot{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}$ are depicted in Fig. 5.22. It can be observed that the desired motor torques comply with the actuator constraints. As the constraints depend on $\boldsymbol{q}$, they are not constant during braking (see $\tau_{\text{min},4}$ and $\tau_{\text{max},4}$, for example), but in this example, the variation is negligible because braking is a local behavior. Only one constraint of the available modal control region may be active when computing the modal input limits. In this situation also only one motor takes the maximum possible torque. Because the control region is not fully exploited, the modal space controller can have worse performance than the joint space implementation in terms of energy decrease. However, in this example, the required stopping time for the decoupling-based control

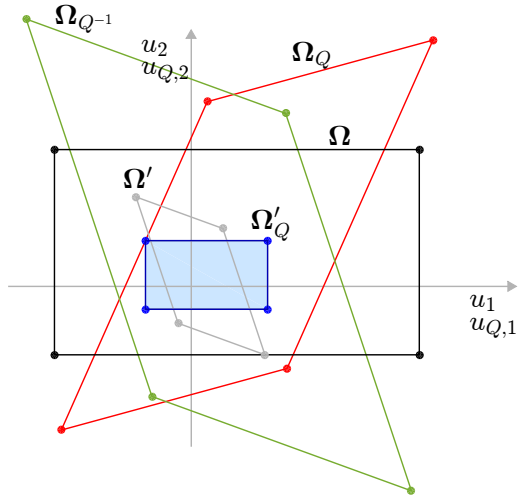


Fig. 5.23: Symmetric hyperrectangle volume with specified aspect ratio obtained via scaling. The joint space control region is denoted by $\boldsymbol{\Omega}$ and the forward and backward transformed control region by $\boldsymbol{\Omega}_Q$ and $\boldsymbol{\Omega}_{Q^{-1}}$, respectively. Scaling $\boldsymbol{\Omega}_{Q^{-1}}$ uniformly gives $\boldsymbol{\Omega}'$, which can be transformed via $\boldsymbol{Q}$ to obtain the feasible modal control region $\boldsymbol{\Omega}'_Q$.

law is only marginally larger than the joint space implementation. This means that the application of the decoupling transformation does not result in much performance loss. At the same time, the simple SISO structure in decoupled space enables enhanced analyses. For example, the braking controller can be extended by braking distance and braking time estimation to improve safety as proposed in [253, 267].

5.4.8 Real-Time capable Scaling with fixed Aspect Ratio

If the aspect ratio of the actuator limits can/shall remain the same in both joint and modal space and the decoupling matrix is orthonormal, then the maximum possible input limits in decoupled space can be determined via scaling instead of optimization. This allows computing the feasible modal control region faster. If $\boldsymbol{Q}$ is orthonormal, e.g., when the stiffness matrix is diagonal and all joints have the same stiffness, then the decoupling transformation is a pure rotation or deflection. This means that the aspect ratio of the control region remains the same in both the original and the decoupled space. The sought control region in modal space that takes the real actuator constraints into account can be obtained as follows; see Fig. 5.23.

1. Apply the inverse decoupling transformation $\boldsymbol{Q}$ to the original control region $\boldsymbol{\Omega}$, which yields $\boldsymbol{\Omega}_{Q^{-1}}$.

2. Determine the vertex of $\boldsymbol{\Omega}_{Q^{-1}}$ that has the largest distance to $\boldsymbol{\Omega}$.

3. Derive the scaling factor for this vertex and scale the entire control region $\boldsymbol{\Omega}_{Q^{-1}}$, which gives us $\boldsymbol{\Omega}'$.

4. Apply transformation $\boldsymbol{Q}$ to $\boldsymbol{\Omega}'$, which results in the admissible control region $\boldsymbol{\Omega}'_Q$ in modal space.

Algorithm 1 Calculate input limits $\boldsymbol{u}'_{Q,\max}$ in decoupled space via scaling

for $i \leftarrow 1$ to 2^n **do**
  $\boldsymbol{v}_{i,\tilde{Q}} \leftarrow Q\,\boldsymbol{v}_i$
  $\boldsymbol{u}_{i,\tilde{Q}} \leftarrow \mathrm{diag}\{\mathrm{sign}(\boldsymbol{v}_{i,\tilde{Q}})\}\boldsymbol{u}_{\mathrm{symm}} + \boldsymbol{o}_\Omega$
  $\boldsymbol{d}_i \leftarrow |\boldsymbol{v}_{i,\tilde{Q}}| - |\boldsymbol{u}_{i,\tilde{Q}}|$
  $j \leftarrow \mathrm{argmax}\,\boldsymbol{d}_i(j)$
  if $i = 1$ **then**
    $k \leftarrow \frac{\boldsymbol{u}_{i,\tilde{Q}}(j)}{|\boldsymbol{v}_{i,\tilde{Q}}(j)|}$
  else
    $k \leftarrow \min\left\{k, \frac{\boldsymbol{u}_{i,\tilde{Q}}(j)}{|\boldsymbol{v}_{i,\tilde{Q}}(j)|}\right\}$
  end if
end for
$\boldsymbol{u}'_{Q,\max} \leftarrow k\,\boldsymbol{u}_{\max}$

The algorithm is provided in Alg. 1; its complexity is $\mathcal{O}(2^n)$. For collaborative robots with six or seven joints, the eigenvalue decomposition (complexity typically $\mathcal{O}(3n^3)$) required for modal decoupling is usually computationally more demanding than the control region decoupling. This was verified on a LWR and DLR David.

5.4.9 Implementation of the Decoupling-Based Control Framework

With the dynamics decoupling transformation (5.29) and the control region decoupling obtained via optimization or scaling, we now have all components to implement the control framework illustrated in Fig. 5.13. After the dynamics and control region decoupling, one can implement the time-optimal or near time-optimal control law (or another controller) for each decoupled coordinate and transform the desired control variable back to the original space, where the motors are controlled. On the tested LWR and DLR David, the framework is real-time capable. The scaling-based solution is exemplified in the following section, where experiments on the DLR David system are reported.

5.5 Experiments

Two experiments were conducted to validate the considered control laws, namely, a) a ball impact experiment and b) an emergency stop experiment. First, the implemented control laws are described, then the experimental procedure and the results.

5.5.1 Considered Control Laws

Currently, the default safety control law for DLR David is a motor PD controller with online gravity compensation

$$\boldsymbol{\tau} = \boldsymbol{K}_P(\boldsymbol{\theta}_d - \boldsymbol{\theta}) - \boldsymbol{K}_D\dot{\boldsymbol{\theta}} + \boldsymbol{g}(\bar{\boldsymbol{q}}(\boldsymbol{\theta}))\,, \tag{5.58}$$

where $\boldsymbol{\theta}_d$ is the desired motor position and $\boldsymbol{K}_P$ and $\boldsymbol{K}_D$ are the diagonal proportional and damping gain matrices. This controller and the following other controllers were implemented for comparison.

I Motor PD
$\boldsymbol{K}_P = \text{diag}\{8000, 8000, 8000, 8000\}$, $\boldsymbol{K}_D = \text{diag}\{200, 200, 200, 200\}$

II Energy dissipation controller [153] (state-proportional)
$K_{H,i} = 50, i = 1, \ldots, 4$

III Energy dissipation controller [153] (bang-bang)
$\dot{\theta}_{\text{max},i} = 1$ rad/s, $i = 1, \ldots, 4$

IV Energy dissipation controller [247] (bang-bang, modal space)
$\dot{\theta}_{\text{max},i} = 1$ rad/s, $i = 1, \ldots, 4$

V Energy dissipation controller [253] (state-proportional)
$\boldsymbol{K}_H = \text{diag}\{15, 15, 15, 15\}$

VI Energy regulation controller [155]
$K_H = 20$, $H_d = 0$, $\boldsymbol{K}_P = \text{diag}\{1000, 1000, 1000, 1000\}$,
$\boldsymbol{K}_D = \text{diag}\{100, 100, 100, 100\}$

VII ESP link-side damping controller [150]
$\boldsymbol{K}_P = \text{diag}\{1000, 900, 750, 750\}$, $\xi_{D,i} = 0.5, \xi_{K_D,i} = 0.7, i = 1, \ldots, 4$,
where $\xi_{D,i}$ and $\xi_{K_D,i}$ are damping parameters.

VIII ESP+ link-side damping controller [150]
$\boldsymbol{K}_P = \text{diag}\{1000, 900, 750, 750\}$, $\xi_{D,i} = 0.5, \xi_{K_D,i} = 0.7, i = 1, \ldots, 4$

In addition to controllers I – VIII, the variable stiffness controller proposed in [157] was implemented and tested on David. Because no stable vibration suppression was achieved with this control scheme, it is not included in the above controller list. However, simulation and real-world results for the variable stiffness scheme are reported in Sec. 5.5.3.

5.5.2 Implementation

In this thesis, the motors of elastic joint robots are modeled as velocity sources because the motor side dynamics are typically much faster than the link side dynamics. In order to apply the proposed control laws to torque-controlled DLR David, a cascaded controller structure is used; see Fig. 5.24. The desired motor velocity coming from the damping control law is integrated, then the desired motor position and velocity are forwarded to a motor PD controller with online gravity compensation, which computes the desired motor torque. Regarding the state-triggered control laws, it is well known that bang-bang control can lead to chattering in the vicinity of an equilibrium position. This can cause unwanted

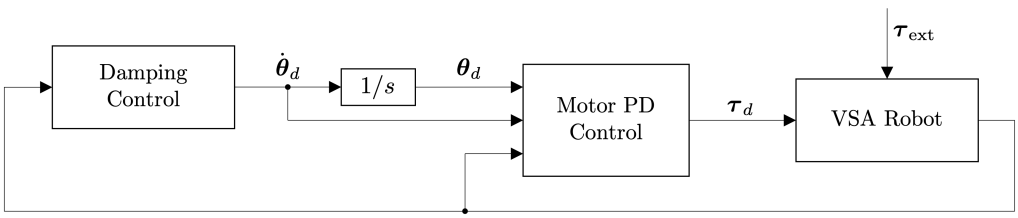


Fig. 5.24: From desired motor velocity of the damping controller to commanded motor torque.

effects such as high wear of moving mechanical parts, heat losses in electrical circuits, etc. Many solutions exist to prevent chattering, i.e., filtering the control input [268]. In this work, chattering is prevented by switching to the state-proportional controller IV once the total pseudo energy $|H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}})|$ of the robot becomes lower than a certain threshold.

5.5.3 Ball Impact Experiment

In the ball impact experiment, a 3 kg ball hits the right forearm of DLR David, which is initially at rest; see Fig. 5.25 (top left). Only the first four FSJ joints are actuated, the positions of the wrist and hand are kept constant using stiff position control. A medium joint stiffness setup is selected ($\boldsymbol{\sigma}^\mathsf{T} = [5, 5, 5, 5]\ ^\circ$). The damping controllers are activated prior to impact. It would also be possible to employ a collision detection scheme to trigger the control law, usually done during nominal operation. Here, a possible delay caused by the collision detection scheme shall be avoided, as only the controller response shall be investigated. The robot configuration was selected such that the impact induces significant oscillations in joints 1–4. The initial pendulum deflection (which corresponds to impact speed) was successively increased until the maximum elastic deflection of one or more joints was reached when using motor PD control[5]. This drop height was then selected for all following impact experiments.

The experimental results are reported in Fig. 5.25. In the first row (controller I), one can see that the impact induces significant oscillations; it takes approx. 4 s until the system reaches the static equilibrium. The elastic deflection of the first joint almost reaches the limit of 10 ° after impact. For control scheme II (second row), one observes faster convergence towards equilibrium. Furthermore, the distance in elastic deflection towards the limits becomes larger, which improves safety. However, the commanded motor velocity oscillates significantly (see row two, column four). As the equilibrium position is not defined, but the control law only minimizes the pseudo energy $H(\boldsymbol{\theta}, \boldsymbol{q}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{q}})$, the equilibrium position is different from the initial position (see the evolution of joint position in the third column). For the motor PD controller, both the initial and the final position were identical. The bang-bang controllers III and IV in the third and fourth row successfully suppress the induced vibrations; however, strong oscillations in motor velocity can be observed. The response of the state-proportional controller V is illustrated in row five. In the previous 1-DOF analysis, control law V performed better than the similar state-proportional controller II. This result is confirmed in the David experiments. The equilibrium is reached faster, also with fewer oscillations both on the motor and the link side. This is because control law II commands a non-zero control action in only two out of four quadrants in the $\varphi/\dot{q}$-plane, while controller V exploits all four quadrants. In row six, the torque-based energy regulation controller VI is represented. After suppressing the vibrations, the initial motor position prior to impact is reached again, which was set as the desired equilibrium position. In the seventh and the last row, the responses of the ESP and ESP+ controllers are depicted. Both controllers achieve good link side damping and the desired link position is reached quickly[6].

[5] For this controller, the largest oscillations are expected during impact absorption.

[6] In this experiment, motor side friction was not compensated for. It is expected that the results obtained with the ESP/ESP+ (and maybe also with the other controllers) will improve when including friction compensation [269].

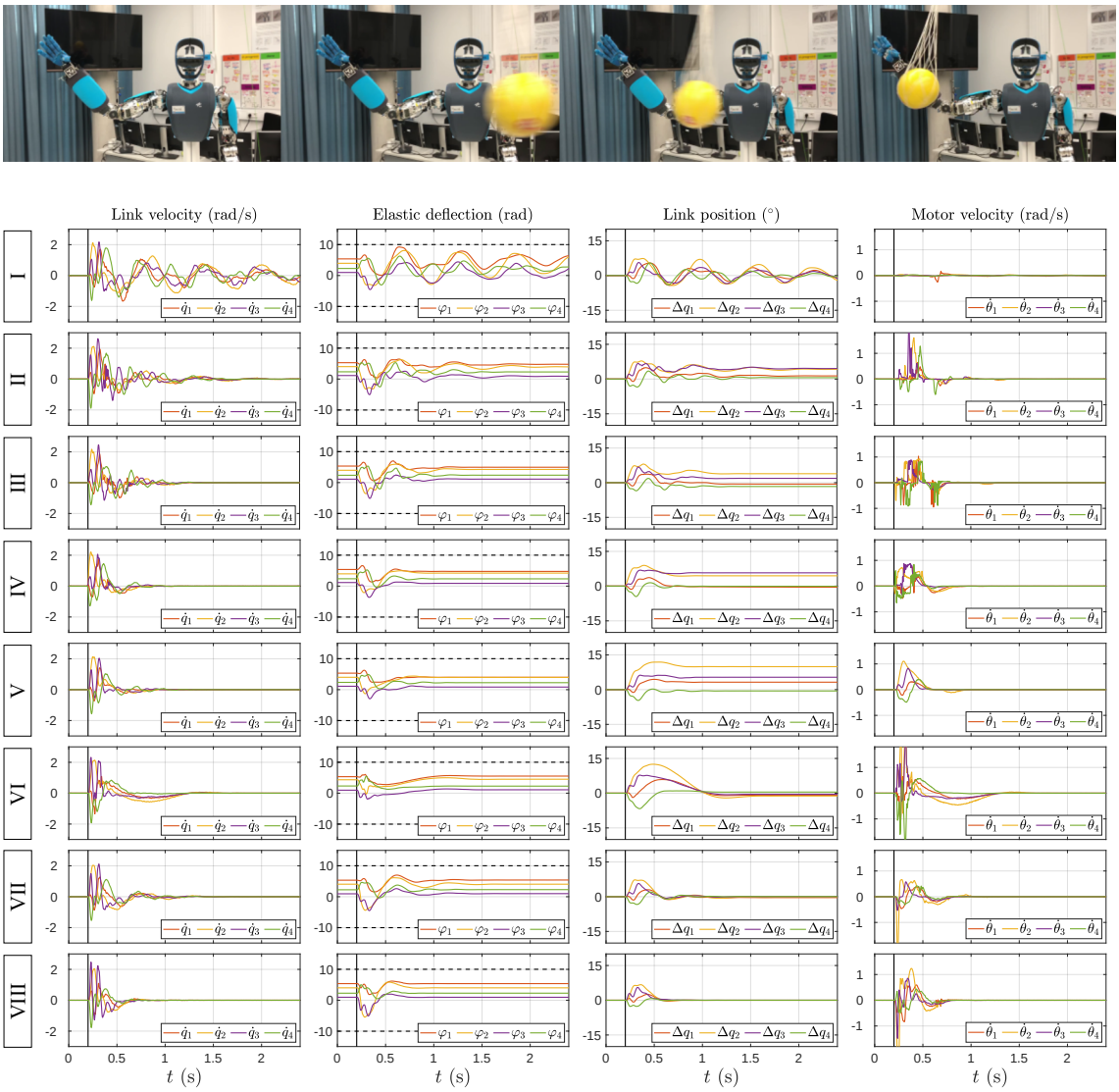


Fig. 5.25: Ball impact experiment. Each row illustrates the response of a certain controller I – VIII. In the first column, the link velocity is shown, in the second column, the elastic deflection (bounds represented by dashed lines), in the third column, the difference $\Delta q_i = q_i - q_{i,0}, i = 1, \ldots, 4$ of actual and initial joint position before impact, and in the fourth column, the motor velocities. A vertical black line represents the instant of impact.

Results for Variable Stiffness Control

Simulation results for the variable stiffness control scheme [157] are depicted in Fig. 5.26. The conditions are very similar to the real ball impact scenario. The same initial robot configuration was selected as in Fig. 5.25. A 100 N force with 0.1 s duration was applied to the robot's forearm. The minimum and maximum stiffness setup values were selected as $\sigma_{\min} = 0\,^\circ$ and $\sigma_{\max} = 5\,^\circ$. No stiffness actuator dynamics were considered; the desired stiffness could be reached instantaneously. The simulation results are illustrated in Fig. 5.26 (left column). It can be observed that the stiffness change is of bang-bang type and that minimum and maximum stiffness alternate with high frequency. The equilibrium position

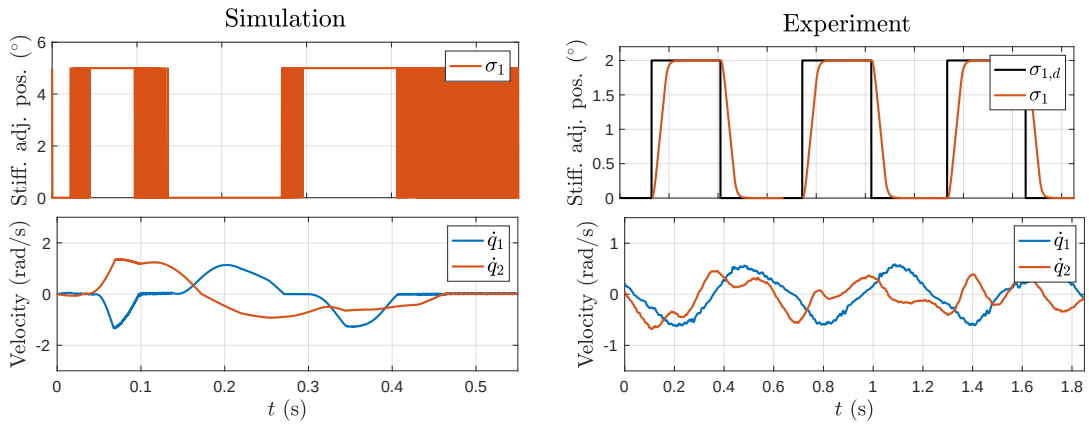


Fig. 5.26: Performance of the variable stiffness controller proposed in [157] in simulation (left column) and experiment (right column).

is reached after approx. 0.45 s.

On DLR David, the controller was applied to joints 1 and 2, as the stiffness adjustment mechanism of joints 3 and 4 was not working properly when the experiment was conducted. The positions of joints 3 and 4 were thus kept constant. The minimum and maximum stiffness setup values were selected as $\sigma_{\min} = 0\,°$ and $\sigma_{\max} = 2\,°$. In the experiment, the robot was initially at rest; then, a slight external force was exerted manually. This force induced unstable oscillations, which could not be damped with the controller. The results are depicted in Fig. 5.26 (right column). It can be observed that the settling time of the stiffness adjustment mechanism is approx. 70 ms. Apparently, this relatively small time delay is sufficient to cause instability. The authors of [157] noted that a limited settling time of the stiffness adjuster could deteriorate the practical controller performance. For the VSA Cube [270], they found a heuristic to anticipate the stiffness switchings. For DLR David, such a heuristic is still to be found (if possible).

Quantitative Evaluation of Impact Behavior

In the following, the robot's impact response shall be assessed quantitatively. For this, several performance metrics are defined and evaluated. These metrics regard the time and traveled distance to reach equilibrium, the compliance with constraints, and the number of oscillations induced by the collision. The metrics are used to compare the control laws used in the ball impact experiments. A summary of the results is provided in Fig. 5.27 and explained in the following. Please note that the controllers were not tuned with regard to one or more metrics, which would generally be possible.

Stopping Time The time required to bring the system to equilibrium can be defined in terms of the kinetic energy T, pseudo energy H, or the norm of joint velocities, for example. The latter has the advantage that no model but only sensor data is required. In this work, the stopping time t_f is defined as the time when the norm of joint velocities becomes smaller than a threshold ε, i.e. $||\dot{\boldsymbol{q}}(t)|| < \varepsilon$, $t \in T$, before reaching the static equilibrium. The total impact duration is defined as $T = [t_0, t_f]$, where t_0 is the initial instant of collision. For the experiments $\varepsilon = 0.01$ m/s is selected. A comparison of the stopping times for the ball impact experiment is given in Fig. 5.27. Controller VIII is the

fastest scheme (0.68 s), followed by controller IV (0.837 s), and V (0.857 s).

Compliance with Constraints Besides damage to the surface and the structure (links, electronics, etc.), the main mechanical threats from impacts regard the drive train. Following (among other) quantities usually need to be kept within certain bounds:

- Motor (incl. gearbox) torque, position, and velocity
- Link position
- Elastic deflection
- Stiffness adjuster torque, position, and velocity

For DLR David, the elastic deflection is typically the most critical quantity as the elastic energy and maximum deflection are limited. One is interested in how close the deflection in each joint gets to the constraint while the impact is absorbed. To assess the compliance w.r.t. to the deflection limits, the maximum relative deflection

$$r_{\varphi,\max} = \max_{t\in T, j\in\mathcal{J}} \frac{|\varphi_j(t)|}{\varphi_{\max,j}(\sigma_j(t))} \leq 1 \tag{5.59}$$

is evaluated for $t \in T$ and over all joints $j \in \mathcal{J}$, where $\mathcal{J}$ is the set of joints. Here, it is assumed that the deflection constraint is symmetric. In the experiment $r_{\varphi,\max} = 92$ % is observed for control law I, the remaining controllers improve impact absorption by reducing the maximum relative deflection to 60 % $\leq r_{\varphi,\max} \leq$ 70 %; see Fig. 5.27.

Oscillations In order to quantify how much the system oscillates after an impact, the number of zero-crossings of every joint velocity $\dot{q}_i(t), i \in 1, \ldots, n$ is counted in the period $t \in T$. Then, the average number of all joints is determined. To quantify motor side oscillations (especially those of the bang-bang controllers), one could also formulate a metric in terms of motor velocity. For the controllers I, II, and III, where significant oscillations were observed, we count 24, 15, and 13 zeros crossings in joint velocity, respectively. For the remaining controllers, typically less than six zero-crossings occurred.

Traveled Distance w.r.t. Initial Position In addition to evaluating the number of oscillations, it is of interest how much the joint or Cartesian position deviates from the initial position while the impact is being absorbed. The first proposed metric considers the maximum Cartesian distance of the end-effector w.r.t. to the initial position in the time span $t \in T$

$$d_{\max} = \max_{t\in T} ||\boldsymbol{x}_{EE}(t_0) - \boldsymbol{x}_{EE}(t)||_2 \,. \tag{5.60}$$

Controllers I, VI, VII, and VIII have a defined equilibrium position, while control laws II, III, IV, and V do not. The metric

$$d_{\text{eq}} = ||\boldsymbol{x}_{EE}(t_0) - \boldsymbol{x}_{EE}(t_f)||_2 \,, \tag{5.61}$$

quantifies the Cartesian deviation of the end-effector's equilibrium from the initial position. The controller with the largest maximum traveled end-effector distance in the ball impact experiment is controller VI (17.7 cm), followed by controller V (16.4 cm), the control laws

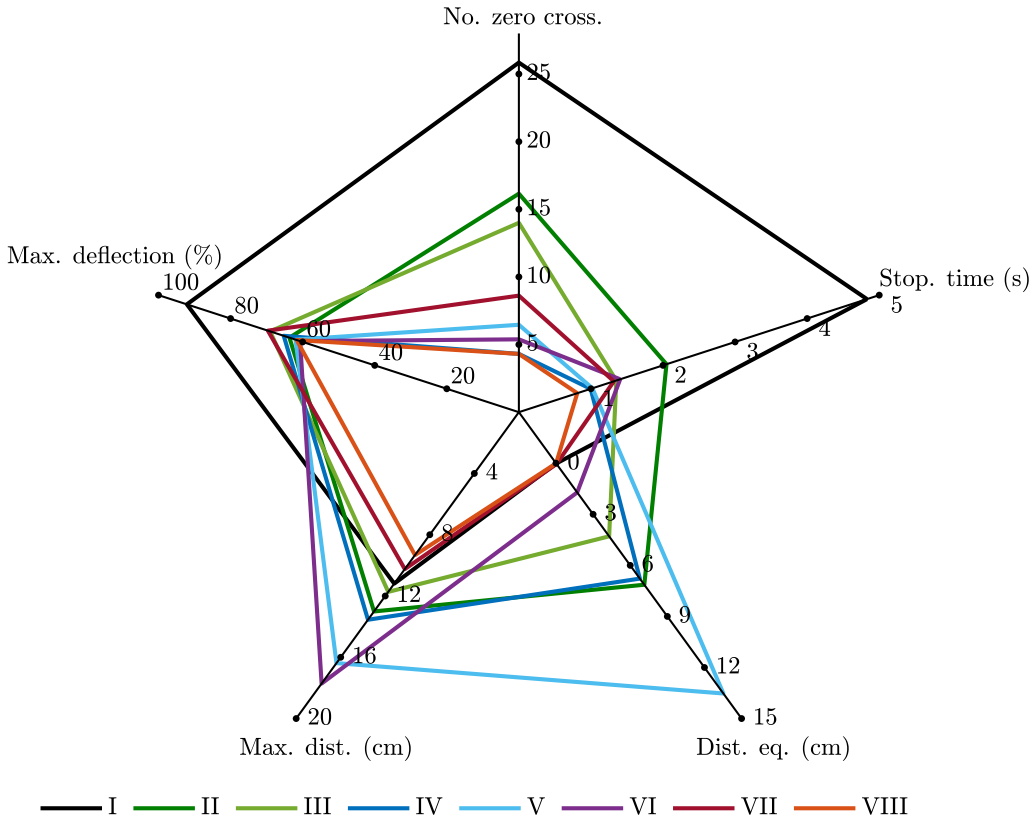


Fig. 5.27: Ball impact experiment: Controller comparison in terms of the proposed performance metrics.

with unspecified equilibrium position, the motor PD controller (11.2 cm), and the ESP controllers ($\leq$ 10 cm). The largest distance between the initial and equilibrium position is observed for controller V (13.53 cm), II (7.1 cm), IV (6.8 cm), and III (4.3 cm). In contrast, the equilibrium position for controllers I, VI, VII, and VIII is (almost) identical to the configuration prior to impact.

To sum up, for the considered experimental conditions, controller parameters, and evaluation metrics, controller VIII (ESP+) performs best, with controller VII (ESP) having similar performance. For each metric illustrated in Fig. 5.27, the ESP+ has the smallest distance to the origin. However, please note that some metrics can be more relevant than others, depending on the application and safety strategy. For example, it makes more sense for the robot to move away from the contact location in clamping situations instead of keeping the position, which could induce further collisions or undesired clamping forces. In this situation, a larger stopping distance is tolerable, respectively desirable. Regarding safety from the robot's point of view, controllers II-VIII significantly reduce the maximum elastic deflection. In terms of stopping time, which also correlates with the number of link-side oscillations for most controllers, controllers IV, V, and VI perform well besides control laws VII and VIII. As expected, the largest distance between the initial and equilibrium position is achieved by controllers II, III, IV, and V, which do not regulate the goal configuration. For these controllers and controller VI, the maximum distance w.r.t. the initial configuration is also larger than the one observed for the other controllers.

5.5.4 Emergency Stop Experiment

In the second experiment, an emergency stop is performed. The robot shall follow a desired joint trajectory using an (arbitrary) tracking controller. At a predefined position along the robot trajectory, an immediate switch to one of the damping controllers takes place to perform the emergency stop. This position is set as the goal configuration for the controllers that regulate the target position. The desired trajectory is illustrated in Fig. 5.28 (first row). The robot moves from an upper to a lower position. In the middle of the trajectory (see red line in Fig. 5.28, upper left), the respective damping controller is activated based on the robot's forward kinematics. To test the controllers' damping performance, the velocity of the robot trajectory should be as large as possible. The velocity was successively increased until one or more joints reached the maximum deflection when using motor PD control for the emergency stop. This velocity was then selected to test all control laws.

The results are illustrated in Fig. 5.28. The evaluation of the measured signals in terms of the selected metrics is provided in Fig. 5.29. The interpretation of the results is very similar to one of the previous experiment. For controllers I and II, strong oscillations in joint velocity can be observed; the remaining control laws suppress vibrations effectively. Controllers III-VIII also have a very similar stopping time in this experiment. The best compliance with the deflection constraint is achieved with controllers V and VI. However, this comes along with a larger max. stopping distance. Because the target position is not regulated, controllers II-V have the largest distance between equilibrium and starting position. Like in the previous experiment, controller VII (ESP) and VIII (ESP+) perform best for most metrics.

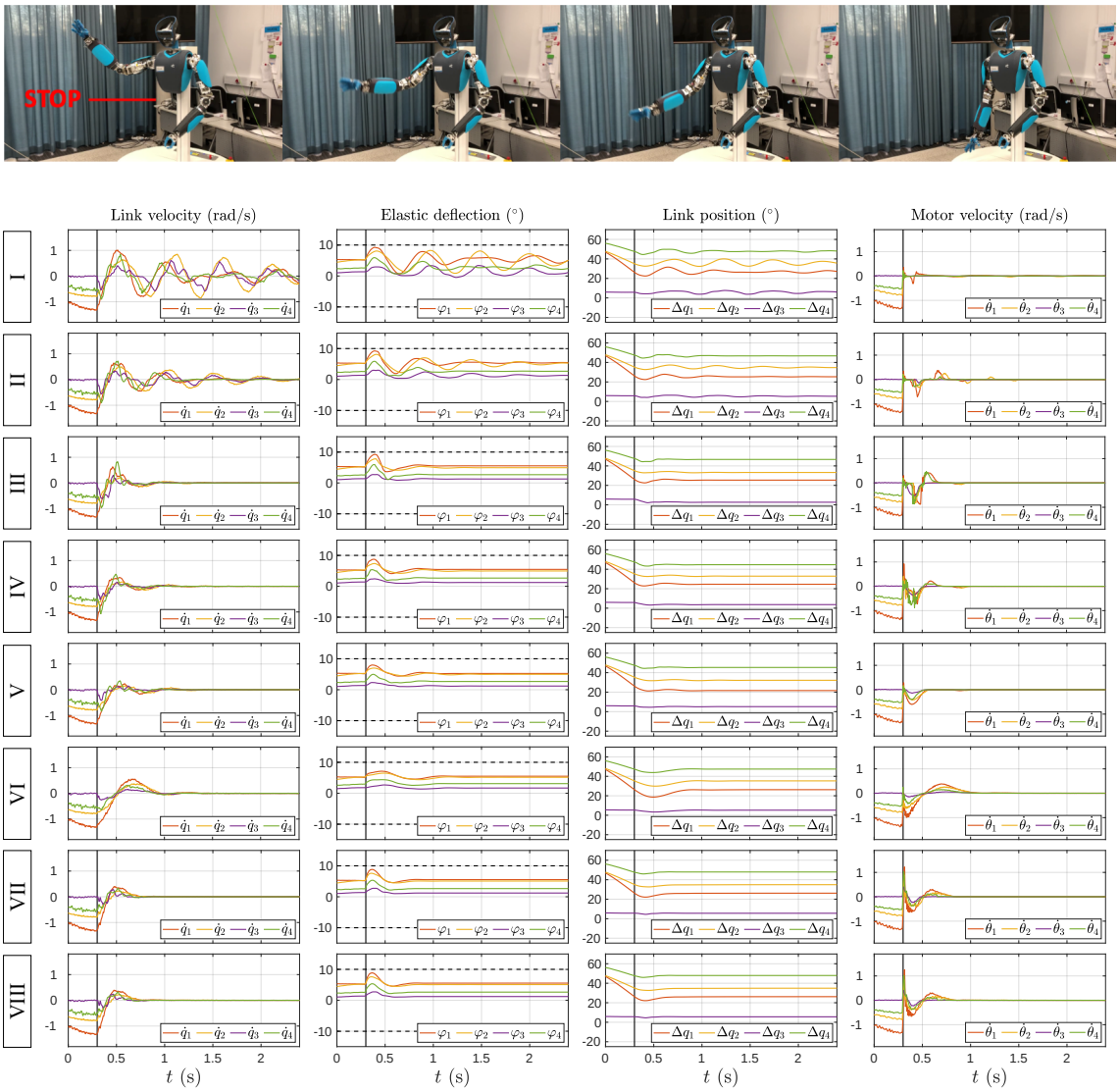


Fig. 5.28: Emergency stop experiment. The trajectory is shown on the top; a braking controller is activated when the robot hand reaches the solid red line. Each row illustrates the response of a certain controller, which is mentioned on the left. In the first column, the link velocity is shown, in the second column, the elastic deflection (bounds represented by dashed lines), in the third column, the joint position, and in the fourth column, the motor velocities. A vertical black solid line represents the instant of the emergency stop.

Possible Velocity Increase

To compare the damping control schemes, the same conditions were selected for every experiment. The speed of the nominal trajectory was comparatively low, as motor PD control would lead to system damage if higher speeds were chosen. For proper damping control schemes, the system should be able to brake to equilibrium safely also from higher speeds. In an additional experiment, the velocity of the trajectory was increased by 123 %. In Fig. 5.30 the Cartesian velocity of the nominal trajectory with motor PD control and the high-velocity trajectory for controller V is illustrated. The maximum Cartesian velocity of

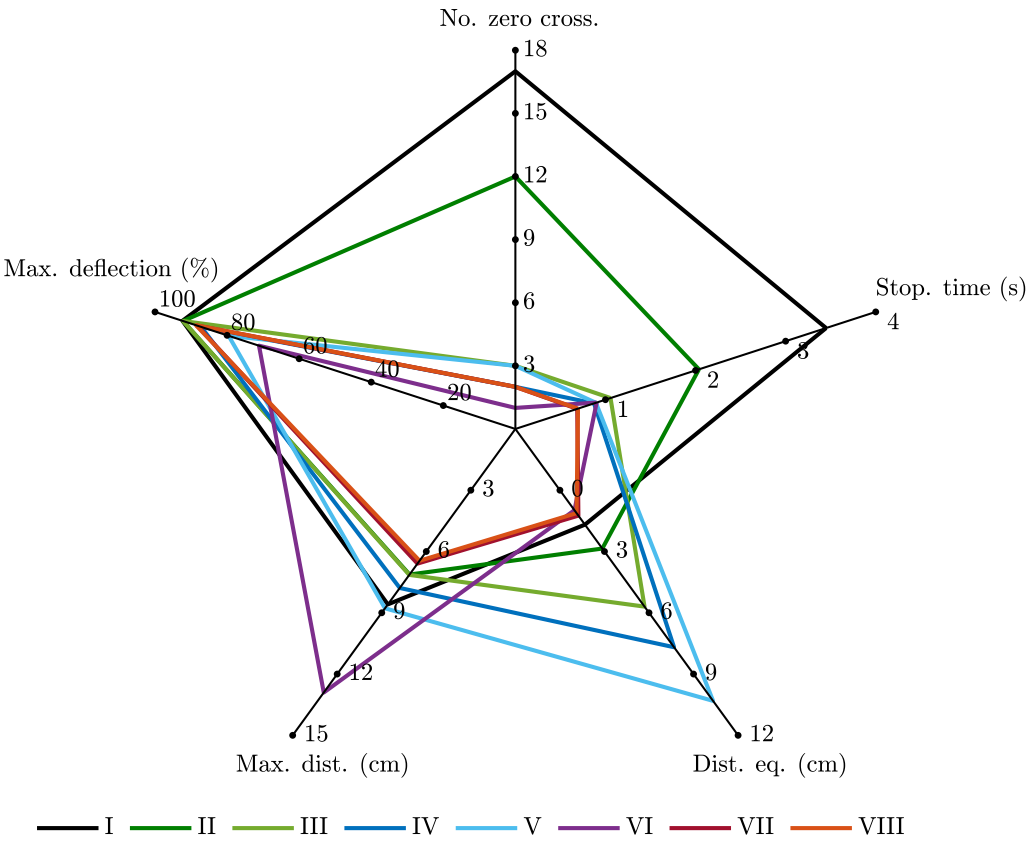


Fig. 5.29: Emergency stop experiment: Controller comparison in terms of the proposed metrics.

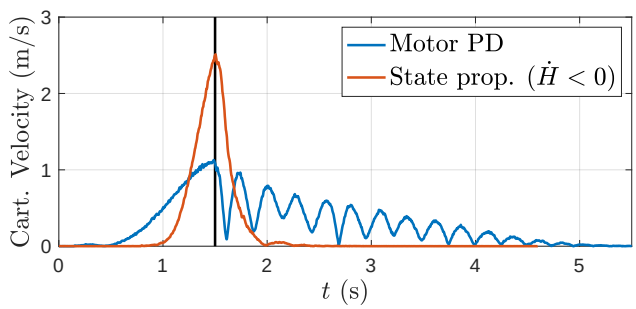


Fig. 5.30: DLR David: Cartesian end-effector speed in the emergency stop experiment. Blue: Nominal trajectory (cf. Fig. 5.28) with motor PD controller used as braking scheme, red: Trajectory with 123 % higher velocity and braking scheme V. The black line indicates the moment of activation of the respective damping controller.

the wrist is 1.13 m/s in the nominal trajectory and 2.51 m/s in the modified trajectory. It can be shown that control law V ($K_C = 30$) brings the system safely to equilibrium while allowing for a significant velocity increase.

5.6 Summary

In this section, the problem of suppressing undesired oscillations in elastic joint robots via control was addressed. In the 1-DOF analysis, an analytical solution to stopping a visco-elastic joint in minimum time was derived. Newly proposed and existing state-proportional and state-triggered control laws were compared in terms of passivity and performance. For control of bang-bang type, a modal decoupling-based framework was proposed that decouples both the dynamics and the control inputs, thereby allowing to apply simple SISO control laws to more complex (linearized) n-DOF systems. A selection of eight different damping control laws was implemented on DLR David. Two safety-related experiments were performed, namely, a) a dynamic ball impact against the robot structure and b) an emergency stop during task execution. Prior to this work, some of the tested controllers had not been validated in practice yet, or so far only on 1-DOF systems. Finally, impact response metrics were proposed to quantitatively compare the performance of the tested controllers and to analyze the robot's impact absorption behavior.

CHAPTER 6

Null Space Control for Safe and Efficient pHRI

Many of today's collaborative and tactile robots have more degrees of freedom than necessary to accomplish the desired task. This property enhances the robot's dexterity and allows it to fulfill secondary tasks which do not interfere with the primary task. Classical and widely used secondary tasks are the avoidance of singularities [271–273], joint limits [274–276], and obstacles [81, 94, 277, 278]. This thesis investigates how the robot's redundant degrees of freedom can be exploited to realize safe and efficient motions. In particular, three control schemes are proposed that a) improve the performance of safe robot motions, b) the performance of auxiliary tasks, and c) enable an intuitive and interactive reconfiguration of the robot. In Sec. 6.1, a controller is elaborated that locally minimizes the robot's reflected mass in the direction of motion via self motions. The controller is combined with the Safe Motion Unit [12]. A reduced reflected mass enables the robot to travel at higher operational velocities while still respecting the safety constraint. This improves the efficiency in HRI tasks, particularly in industrial applications, where short cycle times are essential. The proposed scheme is validated on a seven- and eight-DOF LWR. In Sec. 6.2, it is investigated how the performance of auxiliary nullspace tasks can be improved by temporarily reducing the speed of the primary task. Hierarchical task control schemes are usually defined instantaneously and do not take the temporal dimension into account [196, 278]. It is thus possible that the primary task hinders the auxiliary tasks from achieving their objectives when they depend on the entire joint configuration. The goal is to keep the same primary and null space controller but to modify the timing law of the desired task online. Such relaxation leaves the principal behavior of the system unchanged and improves task fulfillment at the cost of some extra cycle time. Several time scaling schemes are proposed and validated on an LWR and a 4R robot. In practice, it is often necessary to kinematically reconfigure the robot, e.g., when the robot is in an unfavorable configuration or obstructs the human. In Sec. 6.3, simple yet intuitive and interactive methods are proposed to reconfigure a redundant robot while keeping the end-effector pose. The methods are validated on eight- and ten-DOF systems in simulation and experiment.

6.1 Performance Improvement of Safe Velocity Control

In addition to classical secondary tasks such as singularity or joint limit avoidance, several safety-related schemes have been proposed. These works include, e.g., collision detection and reaction [83], the reduction of the impact force [79, 84, 279, 280], or the amount of dissipated energy in blunt inelastic impacts [76]. In [84, 279], contact models were used to determine the relationship between robot parameters and the resulting collision force. It was stated that a reduction of the reflected mass results in lower force. In robotics injury analysis, however, it was shown that a reduction in reflected mass does not always reduce injury probability [42, 43]. Furthermore, the consistency of injury prediction obtained by contact models was shown to be often insufficient w.r.t. the medically observed injury [12]; cf. Chapter 1. In this section, the aim is to improve the performance of biomechanically safe velocity control, i.e., to avoid a possible velocity reduction by the Safe Motion Unit [12]. For this, a redundancy resolution scheme is elaborated that minimizes the reflected mass of a given point of interest (POI) in the direction of motion. Based on recent results in robotics injury analysis, it is first analyzed when such a null space strategy would improve safety. While previous works on reflected mass minimization considered velocity control, a torque-based null space controller is developed that systematically takes joint position limits into account. The null space strategy is then combined with the SMU, and experiments on the LWR show the effectiveness of the approach. Finally, the idea is extended to an eight-DOF system consisting of an LWR and a linear axis.

6.1.1 Potential Benefit of Reflected Mass Reduction on Collision Safety

Let us recall the robot reflected mass $m_u(\boldsymbol{q})$ perceived at the end-effector in the unit direction $\boldsymbol{u} \in \mathbb{R}^3$; cf Sec. 2.1.4. The reflected mass depends on robot's inertial properties and the joint configuration $\boldsymbol{q}$. Possible ways to reduce the reflected mass are a) modifying the robot's inertial parameters by design or b) altering the joint configuration. In redundant robots, one can use reconfiguration, i.e., self-motions, to minimize the reflected mass for the desired pose and in a specific Cartesian direction.

An important question is how much benefit a reduction in effective mass has on human injury probability, and in which situations collision safety can be improved via self motions. The influence of mass and velocity on human injury was thoroughly studied in [42, 43]. For

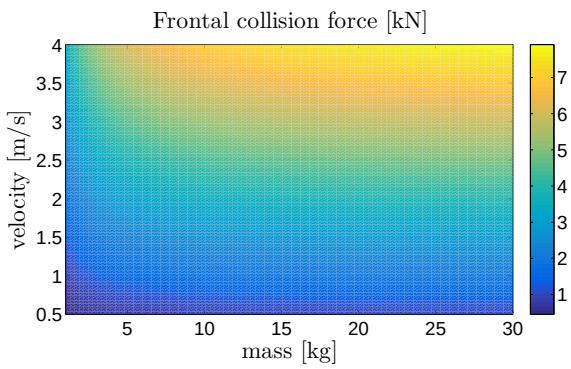


Fig. 6.1: Frontal collision force for an unconstrained impact against the human head obtained by simulation. © 2017 IEEE [281].

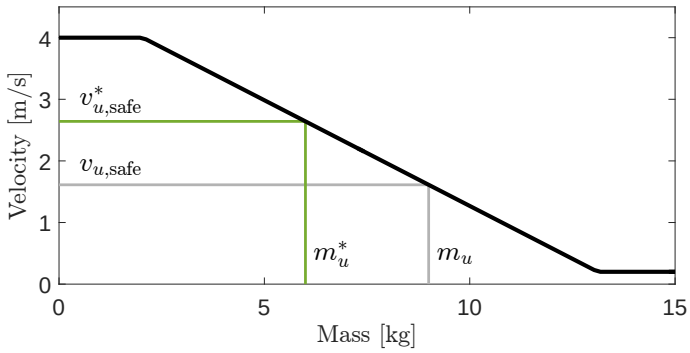


Fig. 6.2: Effect of mass minimization on biomechanically safe velocity for a spherical impactor with 12.5 mm radius. For the reflected mass m_u, the safety curve outputs the velocity $v_{u,\text{safe}}$. When reducing the reflected robot mass to m^*_u, a larger velocity $v^*_{u,\text{safe}}$ can be commanded. © 2017 IEEE [281].

blunt, unconstrained impacts it was shown that a saturation effect in reflected mass takes place when a certain impactor mass is reached. Assume that a mass-spring-mass system models the collision between robot and human forehead. The maximal force during the collision can be determined with (3.1), where $K_h = 1000$ N/mm contact stiffness and $m_h = 4.5$ kg head mass are selected [209]. The maximal contact force is depicted in Fig. 6.1. It can be observed that the saturation effect occurs already at $m_r \approx 15$ kg. If the robot has a considerable weight, then a small reduction in reflected mass will negligible affect collision force. In contrast, the impact velocity always has a significant influence on injury severity. For lower reflected inertias which are inherent to typical pHRI robots, a change in reflected mass *does* affect contact force. However, the injury severity is typically low for the considered mass and velocity range during blunt impacts [14]. A reduction of the reflected mass is therefore rather relevant for sharp and edgy contact as shown in [12], where systematic drop-test experiments were conducted with geometric contact primitives such as edges, wedges, and spheres.

6.1.2 Approach

Especially in industrial applications, it is desired that the robot moves as fast as possible while ensuring human safety at the same time. A possible velocity reduction imposed by the SMU guarantees safety but may deteriorate performance in terms of cycle time. As described in Chapter 3, a possibility to regain performance is to modify the end-effector surface geometry to allow higher safe velocities given the same reflected mass. Alternatively, the reflected mass in redundant robots can be reduced by suitable exploitation of the redundant degrees of freedom. In this work, the SMU scheme is extended by a null space controller that minimizes the reflected robot inertia to further exploit potential performance (speed) increase without affecting the end-effector task. When reducing the reflected mass, the maximum permissible velocity may become higher than the desired task speed; see Fig. 6.2. In this situation, one can either travel with the nominal velocity or speed up the robot until the maximum safe velocity is reached. In this thesis, only the problem of avoiding a velocity reduction by the SMU is considered. If the SMU does not limit the robot speed, i.e., the mass/velocity pairs are below the safety curve, then mass minimization has no benefit on safety because the motion is regarded as safe already. The

mass minimization is, therefore, only advantageous if the SMU would reduce the desired velocity. Practically, the seven-DOF LWR III is mainly considered. In Section 6.3.3 also results on the extension to an eight-DOF system consisting of an LWR IV and a linear axis are provided. It is assumed that the points of interest are located at the end-effector. The robot's links are assumed to be blunt, and collisions with these surfaces are regarded as safe given the light mass properties and the limited maximum velocity of the robot. The reflected mass minimization shall be added as a redundancy resolution scheme to joint torque control, e.g., the impedance control framework [4]. It can also be integrated into a hierarchical redundancy resolution scheme [196, 278], which is, however, not the scope of this work. The null space strategy and the SMU shall be usable in a modular fashion. They may either be used independently or in combination with each other.

6.1.3 Minimization of the Robot Reflected Mass via Self-Motions

This section describes local, real-time capable optimization methods for minimizing the reflected robot inertia.

Gradient-Based Minimization

To obtain the null space joint torque that minimizes the reflected inertia, an intuitive idea is to project the gradient of the reflected mass onto the null space of the Jacobian matrix, i.e.,

$$\boldsymbol{\tau}_d = \boldsymbol{\tau}_{\text{prim}} - K_H \left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^\mathsf{T} \left(\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{M}+} \right)^\mathsf{T} \right) \nabla m_u(\boldsymbol{q}) \,, \tag{6.1}$$

where K_H is a scaling factor, $\boldsymbol{I} \in \mathbb{R}^{n \times n}$ the identity matrix, and $\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{M}+}$ the mass-weighted generalized inverse of the Jacobian matrix, which ensures static and dynamic consistency [181]; cf. Chapter 2. The output of the primary control law is denoted by $\boldsymbol{\tau}_{\text{prim}}$. The gravity compensation is included in this torque. The gradient of the reflected mass is given by

$$\nabla m_u(\boldsymbol{q}) = \frac{\partial m_u(\boldsymbol{q})}{\partial \boldsymbol{q}} = \frac{\partial (\boldsymbol{u}^\mathsf{T} \boldsymbol{\Lambda}_v^{-1}(\boldsymbol{q}) \boldsymbol{u})^{-1}}{\partial \boldsymbol{q}} \,. \tag{6.2}$$

With this approach, which has been used by other authors for velocity control previously, the reflected mass is minimized locally. Close to extrema, the gradient is typically very low, and so is the commanded torque. When applying the joint torque controller (6.1) to the LWR in a joint configuration that corresponds to a (local) maximum in reflected mass, the robot does not move due to link side friction, even for large K_H. To systematically overcome such friction effects, an alternative method is proposed in the following.

Minimization via Attractive Potential

It is proposed to use the gradient of an attractive potential with its maximum at the configuration of minimal reflected mass. It is defined as

$$\boldsymbol{U}(\boldsymbol{q}) = \frac{1}{2} (\boldsymbol{q}^*_{m_u} - \boldsymbol{q})^\mathsf{T} \boldsymbol{K} (\boldsymbol{q}^*_{m_u} - \boldsymbol{q}) \,, \tag{6.3}$$

where $\boldsymbol{q}^*_{m_u}$ is the joint position that corresponds to a (local) minimum in reflected mass. The positive definite, symmetric controller gain matrix is denoted by $\boldsymbol{K} =$

diag$\{k_1, \ldots, k_n\}$. By differentiating (6.3) w.r.t. the joint position one obtains the desired control torque

$$\boldsymbol{\tau}^*_{m_u} = -\frac{\partial \boldsymbol{U}(\boldsymbol{q})}{\partial \boldsymbol{q}} = \boldsymbol{K}(\boldsymbol{q}^*_{m_u} - \boldsymbol{q})\,. \tag{6.4}$$

The torque is projected onto the null space of the Jacobian matrix using the mass-weighted pseudoinverse and then added to the output of the primary controller, which yields

$$\boldsymbol{\tau}_d = \boldsymbol{\tau}_{\text{prim}} + \left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}} \left(\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{M}+}\right)^{\mathsf{T}}\right) \boldsymbol{\tau}^*_{m_u}\,. \tag{6.5}$$

Next, the problem of computing the desired position $\boldsymbol{q}^*_{m_u}$ is addressed. First, it is described how the achievable null space positions can be determined for the LWR. Then an algorithm is proposed that determines the desired position and takes the joint position limits into account.

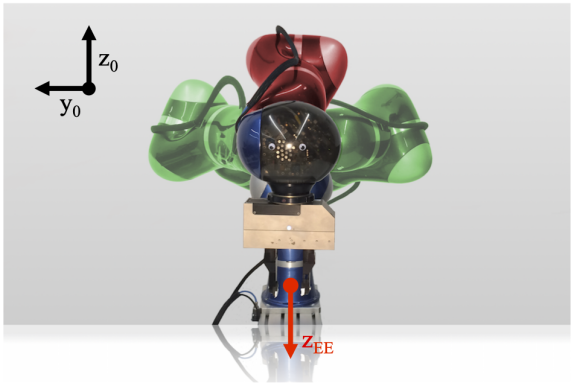
Achievable Null Space Joint Positions Since the considered task trajectory has six DOF and the LWR seven DOF, the null space of the Jacobian matrix is of dimension one if $\boldsymbol{J}(\boldsymbol{q})$ is non-singular. The vector $\boldsymbol{w}(\boldsymbol{q})$, where

$$\boldsymbol{w} : \boldsymbol{q} \mapsto \boldsymbol{\nu} \in \ker(\boldsymbol{J}(\boldsymbol{q})) \subset \mathbb{R}^{n\times 1}\,, \tag{6.6}$$

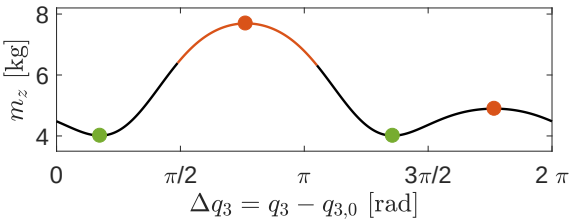
can be regarded as a joint velocity that results in zero operational speed since $\boldsymbol{J}(\boldsymbol{q})\boldsymbol{w}(\boldsymbol{q}) = \boldsymbol{0}$. The analytical form of the kernel was determined in [189]. By successively integrating $\boldsymbol{w}(\boldsymbol{q})$ starting from the initial position $\boldsymbol{q}_{\text{ns}}(0) = \boldsymbol{q}_0$ one obtains all joint positions $\boldsymbol{q}_{\text{ns}}$ that correspond to self-motions (see Fig. 6.3 (lower)):

$$\boldsymbol{q}_{\text{ns}}(t) = \int_{t_0}^{t} \pm\boldsymbol{w}(\boldsymbol{q}_{\text{ns}}(\tilde{t})) \,+\, \boldsymbol{C}_x \boldsymbol{J}(\boldsymbol{q}_{\text{ns}}(\tilde{t}))^{\dagger} \boldsymbol{e}_x(\boldsymbol{q}_{\text{ns}}(\tilde{t}))\,\mathrm{d}\tilde{t}\,. \tag{6.7}$$

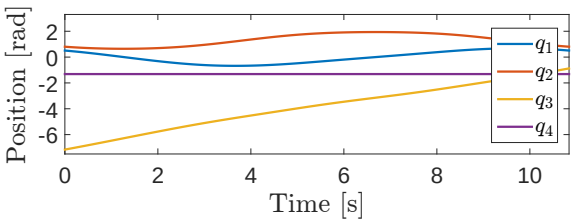
The integration includes a correction term that is based on the end-effector position and orientation error $\boldsymbol{e}_x(\boldsymbol{q}) = \boldsymbol{x}_d(\boldsymbol{q}) - \boldsymbol{x}(\boldsymbol{q})$ between the goal pose and the current pose. This term compensates for integration drift from a practical point of view [282]. The weighting matrix for the correction term is denoted by $\boldsymbol{C}_x$, the Moore-Penrose pseudoinverse of the Jacobian matrix is denoted by $\boldsymbol{J}(\boldsymbol{q})^{\dagger}$. It is assumed that the third entry of $\boldsymbol{w}(\boldsymbol{q})$ is greater than zero during integration. The resulting null space motion, i.e., the rotation of the LWR's so-called elbow, is 2π-periodic. The reflected mass is now evaluated for all positions $\boldsymbol{q}_{\text{ns}}(t)$ in a particular Cartesian direction $\boldsymbol{u}$. In Fig. 6.3 (lower) it can be observed that q_3 strictly increases over the self-motion. One can therefore use it as a coordinate that represents the self-motion. The resulting reflected mass over q_3 plot for a typical pose of the LWR is illustrated in Fig. 6.3 (b). In the figure, it can be observed that the LWR has two minima and two maxima for this end-effector pose. For $\boldsymbol{u} = \boldsymbol{z}_{EE} = -\boldsymbol{z}_0$ the minima are the "elbow left" and "elbow right" position, the two maxima are the "elbow up" and "elbow down" configuration, respectively. As the LWR has (symmetric) joint position limits $|\boldsymbol{q}| \leq \boldsymbol{q}_{\max}$, it can occur that an extremum is not reachable. This can be observed for $\Delta q_3 \approx 3/4\pi$ rad in Fig. 6.3 (b), for example. Due to the joint position limitations (solid red line), the "elbow down" position is not reachable.



(a) Maximum (red) and minima (green) in reflected mass in $\boldsymbol{z}_{EE}$-direction.



(b) Reflected mass in $\boldsymbol{z}_{EE}$-direction. The x-axis represents the difference of q_3 w.r.t. its initial value $q_{3,0}$.



(c) Positions of the first four joints.

Fig. 6.3: LWR: Reflected robot mass perceived at the end-effector in $\boldsymbol{z}_{EE}$-direction. The reflected mass over one full elbow rotation is depicted in (b). The solid red line indicates joint positions which are not reachable due to joint position limits. Red and green dots represent maxima and minima in reflected mass. In (a), the positions associated with the extrema are visualized. In the lower figure, the 2π-periodic joint positions are depicted (joints 5-7 are omitted for better readability). © 2017 IEEE [281].

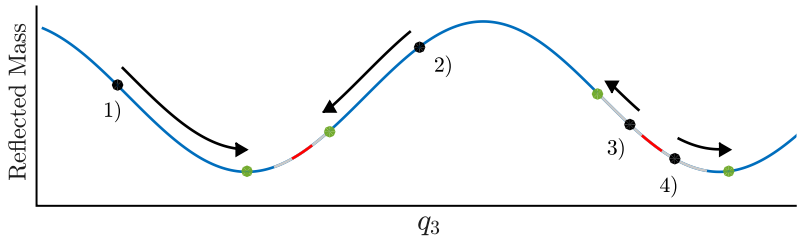


Fig. 6.4: Determining the desired reflected mass and the associated joint configuration. In the figure, four cases are illustrated that show the behavior of the algorithm. In each case, the black dot indicates the initial reflected mass and the green dot the goal. The conservative position limits are represented by a gray line, the real hardware limits by a red line. All masses/positions on the blue line are reachable without violating the constraints.

Determining the Goal Position The target position $\boldsymbol{q}^*_{m_u}$ is the next local/global minimum in reflected mass in the direction of the gradient. The gradient of the instantaneous reflected mass w.r.t. q_3 is obtained by numerical differentiation

$$\left.\frac{\partial m_u(\boldsymbol{q})}{\partial q_3}\right|_{\boldsymbol{q}_{\mathrm{ns}}(t_i)} \approx \frac{m_u(\boldsymbol{q}_{\mathrm{ns}}(t_{i+1})) - m_u(\boldsymbol{q}_{\mathrm{ns}}(t_{i-1}))}{q_3(t_{i+1}) - q_3(t_{i-1})}\,. \tag{6.8}$$

To determine the next position in the direction of the gradient descent, an Euler forward integration method is selected that uses the same correction term as in (6.7). The integration step is

$$\boldsymbol{q}_{\mathrm{ns}}(t_{i+1}) = \boldsymbol{q}_{\mathrm{ns}}(t_i) + \mathrm{sign}\left(\left.\frac{\partial m_u(\boldsymbol{q})}{\partial q_3}\right|_{\boldsymbol{q}_{\mathrm{ns}}(t_i)}\right)\frac{\boldsymbol{w}(\boldsymbol{q}_{\mathrm{ns}}(t_i))}{||\boldsymbol{w}(\boldsymbol{q}_{\mathrm{ns}}(t_i))||_2}\Delta t + \boldsymbol{C}_x\boldsymbol{J}(\boldsymbol{q}_{\mathrm{ns}}(t_i))^{\dagger}\boldsymbol{e}_x(\boldsymbol{q}_{\mathrm{ns}}(t_i))\,. \tag{6.9}$$

Here, the gradient of the reflected mass determines the direction of the null space motion, i.e., the velocity $\boldsymbol{w}(\boldsymbol{q})$ given by the kernel of the Jacobian matrix, which is normalized in (6.9). The step size is denoted by Δt. The third term in (6.9) corrects the deviation from the desired end-effector pose. If the joint position limits do not obstruct the robot motion, one can simply follow the gradient until the next local or global minimum in reflected mass is reached. However, in the vicinity of position bounds, it is necessary to stop or even reverse the motion direction. In the algorithm, conservative bounds are used, i.e., $|\boldsymbol{q}| \leq \boldsymbol{q}_{\mathrm{consv}} < \boldsymbol{q}_{\mathrm{max}}$, to avoid for hardware limits. The behavior of the optimization procedure in the presence of position limits is illustrated in Fig. 6.4. There, four cases are depicted that are explained in the following.

1) The current position does not violate bounds; the gradient is followed until the unconstrained local minimum is reached.

2) If the minimum is not reachable, the border of the conservative joint position limit is selected as the goal position.

3) If the current position violates the conservative joint limits and following the gradient would further violate the constraint, then the direction is reversed and followed until the boundary of the joint limits is reached. This ensures that a reasonable distance to the hardware limits is kept.

4) The current position violates the joint limits, but the robot moves away from the constraint in the direction of the gradient. The gradient is therefore followed until the constraint is not exceeded anymore.

On the experimental LWR setup, the goal position could be computed in real-time, i.e., at 1 kHz control frequency. In order to avoid large step responses in the commanded joint torque, the number of iterations was limited. For large Δt, chattering may occur as the minimum is not computed exactly. The integration step size must therefore be kept reasonably small. Furthermore, a damping term may be added to (6.4). The experiment is described in the next section.

6.1.4 Experiment

A pick and place experiment was carried out to evaluate the performance of the mass minimization scheme in combination with the SMU. Even though the experiment is relatively simple, the results allow analyzing the benefits and limitations of the approach, as will be discussed in this section.

The desired Cartesian trajectory is depicted in Fig. 6.5. The motion sequence is $1 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 1$. The same trajectory was commanded for three control schemes. Firstly, the Cartesian impedance controller was used without SMU or redundancy resolution (see black line in Fig. 6.6). Secondly, the SMU was activated (blue), and finally, the combination of SMU and local mass minimization (green). In the SMU, only one point of interest (POI), the end-effector tip, is considered for the sake of clarity. The shape of the closed gripper brackets is very similar to the spherical impactor with a 12.5 mm radius, which was analyzed in [12]. Therefore the safety curve of this geometry is assigned to this POI; see Fig. 6.2. The mass/velocity pairs of the LWR almost always remain below the safety curve. Therefore, the offset of the curve is conservatively shifted such that the effect of the SMU becomes visible. The SMU may reduce the robot speed if the $\boldsymbol{z}_{EE}$-axis of the end-effector (see Fig. 6.3) is (partly) moving in the direction of travel. As mentioned previously, mass minimization is only beneficial when the safety curve reduces the velocity. However, to evaluate the performance of the mass minimization, the null space scheme is activated during the entire trajectory. The reflected mass is minimized in the direction of the end-effector movement. For the motions $1 \leftrightarrow 3$ the mass is minimized in $\boldsymbol{y}_{EE} = \boldsymbol{y}_0$-direction, for $1 \leftrightarrow 2$ and $3 \leftrightarrow 4$ in $\boldsymbol{z}_{EE} = -\boldsymbol{z}_0$ direction.

The robot motion with and without redundancy, the recorded signals, and a classification of the results in terms of safety and performance are depicted in Fig. 6.6. The results can

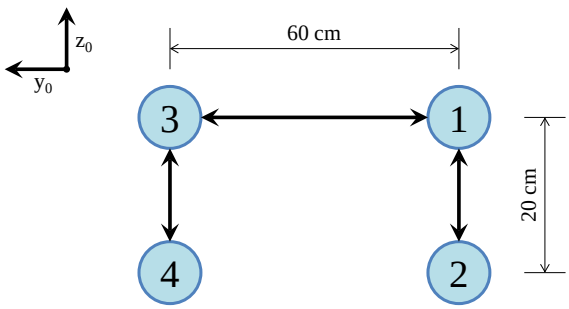


Fig. 6.5: Schematic of the pick and place task trajectory in the y_0/z_0 plane. © 2017 IEEE [281].

be interpreted as follows.

Motion $1 \rightarrow 2$

In the first movement, the robot is traveling in negative $\boldsymbol{z}_0$-direction. The end-effector $\boldsymbol{z}_{EE}$-axis points in the direction of movement, which is the potentially dangerous direction. The SMU reduces the velocity to a biomechanically safe value (see fourth row in Fig. 6.6). The motion with activated SMU is slower than the nominal trajectory; however, the reflected mass is not altered. When activating the reflected mass minimization (enabled before starting the trajectory), one can observe that the mass becomes lower than the one in the original trajectory. Due to lower reflected mass, the safety curve outputs a higher safe velocity. Therefore, the mass minimization allows the robot to travel at the same speed as the nominal trajectory while ensuring collision safety. As a result, the motion including mass minimization reaches the desired position earlier than the trajectory with SMU only, namely at $\approx 1.2\,\mathrm{s}$ instead of $\approx 2.6\,\mathrm{s}$. The robot could now initiate the motion to the next goal position. In Fig. 6.6, however, this motion is delayed. For a better comparison of the results, the motion segments start at the same time for all controllers.

Motion $2 \rightarrow 1$

In the second motion segment, the robot moves in positive $\boldsymbol{z}_0$-direction. The end-effector $\boldsymbol{z}_{EE}$-axis points in the opposite direction of travel, which means that the SMU does not reduce the operational velocity. The velocity is the same for all controllers, while the third controller additionally reduces the reflected mass.

Motion $1 \rightarrow 3$

The motion from position 1 to 3 is along the $\boldsymbol{y}_0/\boldsymbol{y}_{EE}$-axis only. As in the last movement, the end-effector is not pointing in the direction of travel, and therefore, the SMU does not influence the robot speed. Please note that the jump in reflected mass from the second to the third movement is due to the change from $\boldsymbol{u} = \boldsymbol{z}_{EE}$ to $\boldsymbol{u} = \boldsymbol{y}_{EE}$. The reflected mass with activated minimization was always lower than the mass in the other two experiments for the first two movements. During this movement, it can be observed that between $4.3\,\mathrm{s}$ and $4.9\,\mathrm{s}$, the minimization scheme outputs a larger reflected mass than the nominal trajectory. This is because the initial position in this motion segment is different for both methods. For the trajectory including mass minimization and SMU, the elbow moves from the right to the upper and finally to the left position. For the nominal trajectory, the elbow is always in the upper position. The remaining three movements $3 \rightarrow 4$, $4 \rightarrow 3$, and $3 \rightarrow 1$ are not described in detail because the analysis is identical to the first three motion segments.

Classification of Safety and Performance

In the lower plot in Fig. 6.6, the safety and performance of the three controllers are classified. In segments $1 \rightarrow 2$ and $3 \rightarrow 4$, the nominal trajectory partially exceeds the biomechanically safe velocity because no safety constraint is taken into account (red). By activating the SMU, safety is always ensured, but the performance is decreased, i.e., the robot requires more time to reach the target pose (green/yellow hatched). Finally,

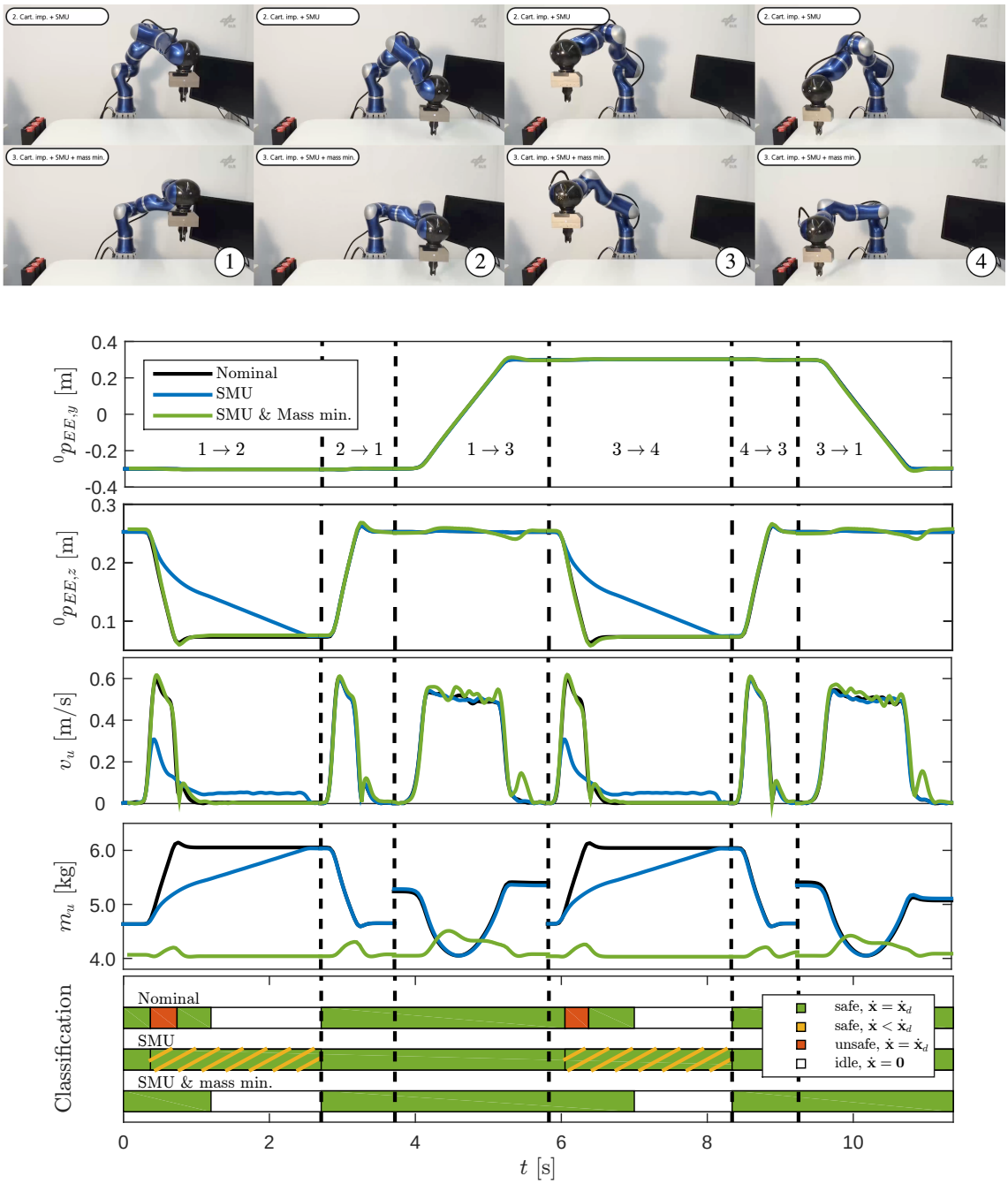


Fig. 6.6: Performance of the SMU and local mass minimization in a pick and place task. The top image illustrates the task execution without (upper row) and with (lower row) active redundancy resolution scheme. In the results depicted in rows 2 – 6, the nominal trajectory without SMU or mass minimization is represented by a black line, the result with activated SMU by a blue line, and the trajectory with SMU and mass minimization by a green line. The Cartesian position of the end-effector, denoted by ${}^0p_{EE,y}$ and ${}^0p_{EE,z}$, is depicted in rows two and three, the absolute Cartesian velocity of the end-effector in row four, and the reflected mass in current $\boldsymbol{u}$-direction row five. In the bottom row, a classification of the results in terms of safety and performance is provided. Green segments indicate a safe motion with nominal speed, green/yellow hatched segments a safe motion where the velocity is below the nominal speed, red segments an unsafe motion with nominal speed, and white segments an idle position. © 2017 IEEE [281].

the combination of SMU and mass minimization keeps the performance of the original trajectory and ensures safety (green).

Discussion

For the considered task, it was shown that the proposed method improves the performance of safe velocity control, i.e., the cycle time is reduced. However, the method has limitations due to its local nature. In the timely evolution of the reflected mass in Fig. 6.5, one can see at ≈ 4.5 s that the optimized reflected mass is higher than the one of the original trajectory. Firstly, this is because the trajectory consists of different motion segments and the elbow position in the nominal trajectory and the one including minimization scheme is different at the initial position of each segment. Secondly, the robot dynamics are limited, so the minimum in reflected mass cannot be reached instantaneously. If another POI was present at the end-effector that points in Cartesian $\boldsymbol{y}_0$-direction, then the increase in reflected mass *could* result in a reduction of the operational velocity by the SMU.

Suppose only the minimization of the reflected mass was considered without the usage of the SMU. In that case, one could use a time scaling approach similar to [105] to synchronize the null space strategy with the end-effector trajectory. The difference between the current reflected mass and the next local minimum could serve as a measure to reduce the velocity of the end-effector. The higher the difference in reflected mass, the lower the velocity. This may result in a lower speed, but one would ensure that the minimum in reflected mass is always achieved. This problem is investigated in Sec. 6.2.

In the experiment, the changes in $\boldsymbol{u}$ were abrupt, which led to significant reconfiguration of the LWR's elbow position. In practice, one could include transitions such that $\boldsymbol{u}$ changes smoothly and the minimization scheme has more time to reconfigure the elbow. Furthermore, one could use instants where the end-effector manipulates an object or is steady to reconfigure the elbow kinematically. A minimum in reflected mass is then already achieved when the robot starts the next dynamic movement. Lastly, if the trajectory was available offline, one could formulate an optimal control problem that minimizes the final time while taking the safety curves as inequality constraints into account. A preliminary analysis is provided in Sec. 3.8. However, this solution also has limitations because it is not sensor-based.

6.1.5 Extension to LWR Mounted on Linear Axis

In this section, first results on the extension of the method to an eight-DOF robot are provided. It consists of an LWR that is mounted on a linear axis; see Chapter 2. The generalized coordinates are $\boldsymbol{q}^\mathsf{T} = [q_x, q_1, q_2, q_3, q_4, q_5, q_6, q_7]$, where q_x is the position of the linear axis and $q_i, i = 1, \ldots, 7$ are the LWR positions. The robot has two redundant degrees of freedom for a six-DOF task when the Jacobian matrix is non-singular. Again, a joint configuration is sought that corresponds to a minimum in reflected mass and does not alter the end-effector task. Having determined the goal configuration, one can apply the control law (6.5).

In Section 6.1.3, it was shown that the positions of the LWR's third joint could represent the null space elbow rotation because it (locally) strictly increases/decreases over the 2π-periodic self-motion. Also for this robot, coordinates are sought that represent the self-motion and can be understood intuitively. These coordinates shall be the position of the linear axis and the third LWR joint. From singular value decomposition (SVD) of

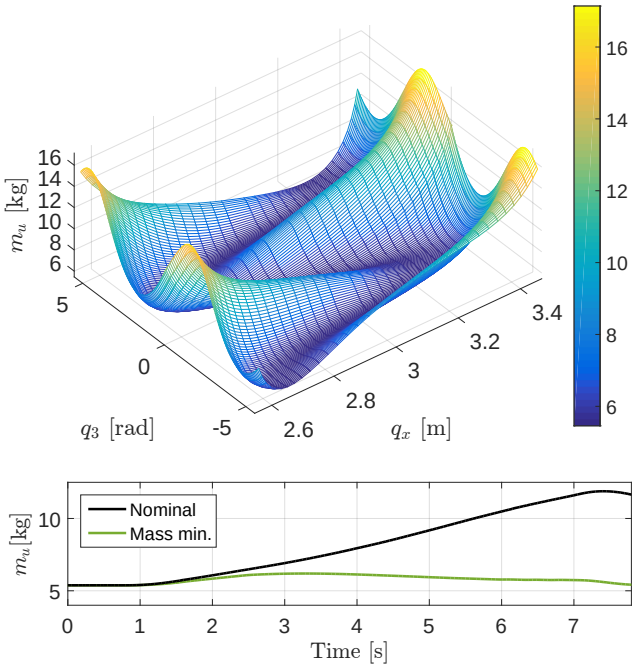


Fig. 6.7: Reflected mass in the Cartesian y_0-direction of the eight-DOF system over the position of the linear axis and the third LWR joint (upper). Here, a configuration similar to the one depicted in Fig. 6.3 a) (red configuration) was selected. The timely evolution of reflected mass for a movement in Cartesian y_0-direction is depicted in the lower figure. The black and green lines represent the reflected mass without and with an enabled minimization scheme, respectively. © 2017 IEEE [281].

$\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{6\times 8}$ one obtains the orthogonal matrix $\boldsymbol{V}(\boldsymbol{q}) = [\boldsymbol{V}_1(\boldsymbol{q}) \,|\, \boldsymbol{V}_2(\boldsymbol{q})]$, where $\boldsymbol{V}_1(\boldsymbol{q})$ and $\boldsymbol{V}_2(\boldsymbol{q})$ constitute the orthonormal bases for the range space of $\boldsymbol{J}(\boldsymbol{q})^\mathsf{T}$ and the null space of $\boldsymbol{J}(\boldsymbol{q})$, respectively. The matrix $\boldsymbol{V}_2(\boldsymbol{q}) = [\boldsymbol{v}_{21}(\boldsymbol{q}), \boldsymbol{v}_{22}(\boldsymbol{q})]$ contains two null space vectors that are orthogonal w.r.t. each other. One now wants to decouple the motion of the manipulator and the linear axis. Firstly, a null space vector $\boldsymbol{w}(\boldsymbol{q}) = [0, *, \ldots, *] \in \ker(\boldsymbol{J}(\boldsymbol{q}))$ is derived, in which the first entry is zero and the other entries take arbitrary values. By projecting any joint velocity onto this part of the null space, zero velocity in the linear axis is obtained, i.e., only the LWR moves. The vector $\boldsymbol{w}(\boldsymbol{q})$ can be expressed as a linear combination of $\boldsymbol{v}_{21}(\boldsymbol{q})$ and $\boldsymbol{v}_{22}(\boldsymbol{q})$, namely

$$\boldsymbol{w}(\boldsymbol{q}) = \boldsymbol{v}_{21}(\boldsymbol{q}) - \frac{v_{21,1}(\boldsymbol{q})}{v_{22,1}(\boldsymbol{q})}\boldsymbol{v}_{22}(\boldsymbol{q})\,, \tag{6.10}$$

where $v_{21,1}(\boldsymbol{q})$ and $v_{22,1}(\boldsymbol{q})$ are the first elements of $\boldsymbol{v}_{21}(\boldsymbol{q})$ and $\boldsymbol{v}_{22}(\boldsymbol{q})$, respectively. If $v_{22,1}(\boldsymbol{q}) = 0$, then one can select $\boldsymbol{w}(\boldsymbol{q}) = \boldsymbol{v}_{22}(\boldsymbol{q})$. The second null space vector $\boldsymbol{w}_\perp(\boldsymbol{q})$ shall be independent from $\boldsymbol{w}(\boldsymbol{q})$, i.e., the vectors are orthogonal w.r.t. each other. To obtain $\boldsymbol{w}_\perp(\boldsymbol{q})$, the Gram-Schmidt process is used to orthogonalize $\boldsymbol{w}(\boldsymbol{q})$ and either $\boldsymbol{v}_{21}(\boldsymbol{q})$ or $\boldsymbol{v}_{22}(\boldsymbol{q})$. While a velocity that is projected onto $\boldsymbol{w}(\boldsymbol{q})$ halts the linear axis, the projection onto $\boldsymbol{w}_\perp(\boldsymbol{q})$ maximizes its velocity. To sum up, two vectors are obtained which are orthogonal w.r.t. each other and span the entire null space of the Jacobian matrix like $\boldsymbol{v}_{21}(\boldsymbol{q})$ and $\boldsymbol{v}_{22}(\boldsymbol{q})$.

One can now systematically integrate $\boldsymbol{w}(\boldsymbol{q})$ and $\boldsymbol{w}_\perp(\boldsymbol{q})$ and use q_x and q_3 to represent the self-motions. In Fig. 6.7 the reflected robot mass in Cartesian y_0-direction is illustrated over

q_x and q_3. All points illustrated in the q_x/q_3 plane correspond to configurations that do not alter the end-effector position. Using this grid, one can calculate the gradient descent of the reflected mass and determine the next minimum. This procedure was applied in iterative form to minimize the reflected mass for a dynamic trajectory. The initial configuration was similar to the one depicted in Fig. 6.3 (a) (red configuration). While keeping the same orientation and Cartesian x_0- and z_0-position, the end-effector moves 1.5 m in Cartesian y_0-direction. The timely evolution of the reflected mass with and without reflected mass minimization is depicted in Fig. 6.7. The results show that the reflected mass is minimized effectively also for this system.

6.2 Performance Improvement of Auxiliary Tasks

Like in the previous section, the objective of a null space task is often formulated in terms of a performance or safety criterion that shall be minimized, e.g., the manipulability measure in case of singularity avoidance [194], the distance to the joint position limits [274], or the estimated contact force during a collision with the robot [76]. If the optimization criterion is defined globally, i.e., it depends on all joint positions, then both the primary task and the null space task influence this criterion during task execution. This means that the main task can either support or hinder the null space scheme from achieving its objective. This is also because typical hierarchical task control schemes are defined instantaneously and do not take the temporal dimension of task fulfillment into account [196, 278]. In the experiment reported in the previous section, e.g., the nullspace scheme did not have enough time, respectively actuation torque to achieve a lower reflected mass than the one observed in the nominal trajectory without redundancy resolution.

One obvious solution is to generate trajectories via motion planning schemes that respect all constraints and optimization criteria. However, motion planning is typically done offline and computationally very costly if done online. In terms of real-time control, a possible solution is to relax the main task through a suitable task scaling [283, 284] or priority switching scheme [89, 285–287]. However, such approaches sacrifice the nominal execution of the primary task to satisfy the auxiliary tasks. They have been used to avoid obstacles [89, 285], joint limits [283, 284, 286], or kinematic singularities [286, 287], for example.

The solution developed in this thesis is based on slowing down the primary task temporarily while preserving its geometric description. The null space task is given more time to optimize for the performance criterion by reducing the execution speed. The idea was – to the best of the author's knowledge – first proposed in [289]. There, the operational speed was reduced whenever the ratio of available and desired null space velocity/torque for minimizing a particular optimization function became smaller than a threshold. In this work, the concept of temporal scaling of the main task for improving the performance of auxiliary tasks is further investigated. The objective is to keep the same primary and null space controller but to modify the timing law of the desired task trajectory online; see Fig. 6.8. Such relaxation leaves the principal behavior of the system unchanged. In this work, several time scaling schemes are proposed and validated experimentally on an LWR with one redundant DOF. Furthermore, the concept is extended to multiple prioritized tasks, and simulation results are provided for a planar 4R robot.

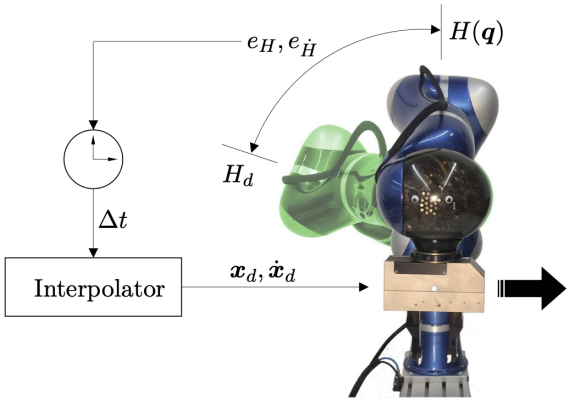


Fig. 6.8: Basic time scaling concept. The null space performance criterion at the current configuration is denoted by $H(\boldsymbol{q})$, the desired value by H_d. The difference $e_H = H(\boldsymbol{q}) - H_d$ and a suitable synchronization error $e_{\dot{H}}$ are used to modify the timing law of the interpolator, which provides the desired joint or Cartesian space trajectory (here: $\boldsymbol{x}_d, \dot{\boldsymbol{x}}_d$) to the robot. The speed reduction gives the null space scheme more time to minimize the performance criterion. © 2019 IEEE [288].

6.2.1 Problem Definition

The rigid joint dynamics (5.54) are considered; cf. Sec. 2.1.2. For the sake of clarity, let us assume that one secondary task shall be fulfilled in addition to the main task in order to resolve the robot's redundancy (cf. Sec. 6.2.4 for analysis on multiple, hierarchical tasks). The goal of the secondary task shall be the local optimization of a certain performance criterion $H(\boldsymbol{q})$ that is differentiable w.r.t $\boldsymbol{q}$. The joint velocity resulting from the control of the main and the secondary task is given by

$$\dot{\boldsymbol{q}} = \dot{\boldsymbol{q}}_x + \dot{\boldsymbol{q}}_{\text{ns}} \,, \tag{6.11}$$

where $\dot{\boldsymbol{q}}_x$ can be associated to the main task and $\dot{\boldsymbol{q}}_{\text{ns}} \in \ker(\boldsymbol{J}(\boldsymbol{q}))$ is the null space velocity, which can be obtained, e.g., by projecting the gradient $\nabla H(\boldsymbol{q}) = \frac{\partial H(\boldsymbol{q})}{\partial \boldsymbol{q}}$ of $H(\boldsymbol{q})$ onto the null space of the Jacobian matrix. The time derivative of the objective is given by

$$\dot{H} = \nabla H \dot{\boldsymbol{q}} \,, \tag{6.12a}$$

$$= \nabla H \dot{\boldsymbol{q}}_x + \nabla H \dot{\boldsymbol{q}}_{\text{ns}} \,, \tag{6.12b}$$

$$= \dot{H}_x + \dot{H}_{\text{ns}} \,. \tag{6.12c}$$

Since the null space controller locally minimizes $H(\boldsymbol{q})$, one can assume that $\dot{H}_{\text{ns}}(\boldsymbol{q})$ is negative, or zero if a local minimum has been reached. The rate $\dot{H}_x(\boldsymbol{q})$, however, can be either positive or negative, meaning the main task can support or hinder the null space scheme from minimizing $H(\boldsymbol{q})$. Let us consider the following two cases:

1) $\dot{H}_x(\boldsymbol{q}) < 0$, $\dot{H}_{\text{ns}}(\boldsymbol{q}) \leq 0$: Both the main and auxiliary task minimize H.

2) $\dot{H}_x(\boldsymbol{q}) > 0$, $\dot{H}_{\text{ns}}(\boldsymbol{q}) \leq 0$: The main task deteriorates the null space performance. If $|\dot{H}_x(\boldsymbol{q})| \leq |\dot{H}_{\text{ns}}(\boldsymbol{q})|$, then H is being minimized (slowly), otherwise $H(\boldsymbol{q})$ even increases.

While the first case is certainly desirable, the second case should be avoided, most significantly if $\dot{H}_x(\boldsymbol{q}) > 0$ and $|\dot{H}_x(\boldsymbol{q})| > |\dot{H}_{\text{ns}}(\boldsymbol{q})|$. Clearly, the null space dynamics play an

important role here. If they are (deliberately) slower than the dynamics of the main task, then $H(\boldsymbol{q})$ may increase during task execution. The goal is now to ensure $\dot{H}(\boldsymbol{q}) \leq 0$ by limiting the magnitude of $\dot{H}_x(\boldsymbol{q})$. This shall be accomplished by temporarily slowing down the main task, which gives the null space task more time to minimize the performance or safety criterion. In addition to keeping $\dot{H}(\boldsymbol{q})$ negative or equal to zero, one may want to keep the magnitude of $H(\boldsymbol{q})$ as small as possible, respectively the difference of $H(\boldsymbol{q})$ and a (preferably feasible) desired value H_d. For this, the execution speed needs to be reduced even further to fulfill $\dot{H}(\boldsymbol{q}) \leq 0$. In the following, several time scaling methods are proposed to solve the problem. First, the concept of time scaling and related work on the topic is briefly summarized.

6.2.2 Time Scaling-Based Relaxation of the Primary Task

The desired joint space trajectory $\boldsymbol{q}_d(t) \in \mathbb{R}^n$ or Cartesian space trajectory $\boldsymbol{x}_d(t) \in \mathbb{R}^6$ are typically provided by an interpolator and are parameterized w.r.t. time. In a discrete implementation of the interpolator, the current time is given by $t_i = t_{i-1} + \Delta t$, where t_{i-1} is the last time instant and Δt is the increment, usually the sampling time. By multiplying the time increment by a factor α

$$t_i = t_{i-1} + \alpha \Delta t \,, \tag{6.13}$$

one can scale the trajectory in time, i.e., slow down ($\alpha \in [0, 1)$), speed up ($\alpha > 1$), or even go backwards along the trajectory ($\alpha < 0$). In the robotics literature, time scaling has been used to solve various control problems. In [290, 291] it was employed to satisfy dynamic and kinematic constraints during task execution. The authors of [292] aimed at improving the tracking performance of pre-planned trajectories, and in [115], time scaling was used for collision reaction, where α was defined as a function of the estimated external torque. Depending on the amount and the direction of the external torque, the user could push the robot back and forth along the desired path. In this work, the concept of time scaling is applied to the synchronization of the main task and the (prioritized) auxiliary null space tasks. The aim is to define α as a function of a suitable null space performance error.

General Scheme

In the previous section, two objectives were defined for the time scaling scheme, namely 1) $\dot{H}(\boldsymbol{q})$ shall be less or equal to zero, and 2) $H(\boldsymbol{q})$ shall be kept as small as possible. The general time scaling scheme for achieving these goals is defined as

$$\alpha = 1 - (K_H e_H + K_{\dot{H}} e_{\dot{H}}) \,, \tag{6.14}$$

where e_H represents the error in performance criterion and $e_{\dot{H}}$ is the so-called synchronization error. The corresponding scalar gains for these errors are denoted by K_H and $K_{\dot{H}}$, respectively. This definition is also inspired by the literature on motion coordination of two or more dynamical systems [293, 294]. In the following, $e_{\dot{H}}$ is defined and possible definitions of e_H are proposed. Prior to this, the considered range and dynamics of α are described.

Considered Range and Dynamics of the Time Scaling Factor

The range of the time scaling factor shall be $\alpha \in [0, 1]$, i.e., one wants to decelerate the robot but not go back along the planned path ($\alpha < 0$) or reach higher speeds than the nominal velocity ($\alpha > 1$). When commanding $\alpha = 0$, the robot will stop its motion, which means that the null space scheme will reach its goal (if reachable) after a certain period. However, in many applications, the cycle time shall not exceed a critical value. Therefore, it is possible to define the allowable range $\alpha \in [\alpha_{\min}, 1]$, where $\alpha_{\min} \in (0, 1)$. This ensures that the robot always executes the task at a velocity greater than a certain threshold. Another possibility is to supervise the evolution of α during task execution and the time lost due to time scaling to decide when the trajectory should resume to nominal speed to meet the cycle time requirement. In this work, however, only a constant lower bound $\alpha_{\min}$ on α is considered. Practically, one may submit α to critically damped second-order dynamics with a relatively low time constant to ensure that the scaling factor is continuous considering that the errors e_H and $e_{\dot{H}}$ can be discontinuous.

Definition of the Synchronization Error

The synchronization error is defined as

$$e_{\dot{H}} = \dot{H}_x(\boldsymbol{q}) - \dot{H}_{\mathrm{ns}}(\boldsymbol{q})\,. \tag{6.15}$$

Since this error is only relevant when $\dot{H}_x(\boldsymbol{q})$ is larger than $\dot{H}_{\mathrm{ns}}(\boldsymbol{q})$ and both rates have opposite sign, $e_{\dot{H}}$ is reformulated as

$$e_{\dot{H}} = \begin{cases} \dot{H}_x(\boldsymbol{q}) - \dot{H}_{\mathrm{ns}}(\boldsymbol{q}), & \text{if } \operatorname{sign}(\dot{H}_x(\boldsymbol{q})) \neq \operatorname{sign}(\dot{H}_{\mathrm{ns}}(\boldsymbol{q})) \\ & \wedge\, |\dot{H}_x(\boldsymbol{q})| > |\dot{H}_{\mathrm{ns}}(\boldsymbol{q})|\,, \\ 0, & \text{otherwise}\,. \end{cases} \tag{6.16}$$

To determine $\dot{H}_x(\boldsymbol{q})$ and $\dot{H}_{\mathrm{ns}}(\boldsymbol{q})$ based on the measured Cartesian and joint velocity, the kinematically decoupled decomposition of the joint space velocities

$$\dot{\boldsymbol{q}} = \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\dot{\boldsymbol{x}} + \boldsymbol{Z}(\boldsymbol{q})^{\mathsf{T}}\boldsymbol{v}_n(\boldsymbol{q})\,, \tag{6.17}$$

proposed in [295] is used where $\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}$ is the weighted pseudoinverse [181] and $\boldsymbol{v}_n(\boldsymbol{q}) = (\boldsymbol{Z}(\boldsymbol{q})\boldsymbol{W}\boldsymbol{Z}(\boldsymbol{q})^{\mathsf{T}})^{-1}\boldsymbol{Z}(\boldsymbol{q})\boldsymbol{W}\dot{\boldsymbol{q}}$ is a minimal parameterization of the self motion velocity [296]. The weighting matrix is denoted by $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ and is typically selected as $\boldsymbol{W} = \boldsymbol{M}(\boldsymbol{q})$. The full row rank[1] matrix $\boldsymbol{Z}(\boldsymbol{q}) \in \mathbb{R}^{(n-m)\times n}$ spans the null space of the Jacobian matrix and satisfies $\boldsymbol{J}(\boldsymbol{q})\boldsymbol{Z}(\boldsymbol{q})^{\mathsf{T}} = \boldsymbol{0}$. The $\boldsymbol{Z}(\boldsymbol{q})$ matrix can be obtained numerically by Singular Value Decomposition (SVD) of $\boldsymbol{J}(\boldsymbol{q})$ or analytically as described in [297], for example. The rates of $H(\boldsymbol{q})$ due to the velocity of the main and auxiliary task can now be determined as follows

$$\dot{H}_x(\boldsymbol{q}) = \nabla H(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\dot{\boldsymbol{x}}\,, \tag{6.18a}$$

$$\dot{H}_{\mathrm{ns}}(\boldsymbol{q}) = \nabla H(\boldsymbol{q})\boldsymbol{Z}(\boldsymbol{q})^{\mathsf{T}}\boldsymbol{v}_n(\boldsymbol{q})\,. \tag{6.18b}$$

[1] It is assumed that the Jacobian matrix is non-singular.

Definitions of the Performance Criterion Error

Gradient Because the gradient $\nabla H(\boldsymbol{q})$ is usually available for null space control, one can use this quantity also for time scaling. When the gradient is large, the velocity shall be reduced while the robot shall resume to nominal speed when the gradient is close to zero (extremum). A similar approach was used in [289]. However, $\nabla H(\boldsymbol{q})$ does not reflect the actual null space capabilities since only locally constrained minimization of $H(\boldsymbol{q})$ is possible. Instead, the weighted gradient is considered [298]

$$\boldsymbol{h}_n(\boldsymbol{q}) = (\boldsymbol{Z}(\boldsymbol{q})\boldsymbol{W}\boldsymbol{Z}(\boldsymbol{q})^\mathsf{T})^{-1}\boldsymbol{Z}(\boldsymbol{q})\,(\nabla H(\boldsymbol{q}))^\mathsf{T}\,, \tag{6.19}$$

where $\boldsymbol{h}_n(\boldsymbol{q}) \in \mathbb{R}^{(n-m)}$ can be regarded as the gradient of $H(\boldsymbol{q})$ projected into minimal null space coordinates. In the sequel, it will be referred to as the minimal gradient. The error in performance criterion can now be defined as the magnitude of the minimal gradient

$$e_H = |\boldsymbol{h}_n(\boldsymbol{q})|\,. \tag{6.20}$$

Gradient and Hessian A problem of $|\boldsymbol{h}_n(\boldsymbol{q})|$ is that its magnitude can be equally low near both a constrained local maximum and a minimum. Accordingly, the minimal gradient on its own is a relatively insufficient definition of the performance error unless augmented by information about the kind of extremum (minimum/maximum), i.e., the second-order derivative. Numerically, the derivative of the gradient can be obtained by

$$h_n(\boldsymbol{q})' = \frac{H(\boldsymbol{q}_-) - 2H(\boldsymbol{q}) + H(\boldsymbol{q}_+)}{|\boldsymbol{q}_+ - \boldsymbol{q}_-|^2}\,. \tag{6.21}$$

Here, the configurations $\boldsymbol{q}_-$ and $\boldsymbol{q}_+$ in the direction of the negative and positive gradient in the vicinity of the current robot configuration $\boldsymbol{q}$ can be obtained by an Euler integration step

$$\boldsymbol{q}_- = \boldsymbol{q} - \boldsymbol{Z}(\boldsymbol{q})^\mathsf{T}\boldsymbol{h}_n(\boldsymbol{q})\Delta t\,, \tag{6.22}$$

$$\boldsymbol{q}_+ = \boldsymbol{q} + \boldsymbol{Z}(\boldsymbol{q})^\mathsf{T}\boldsymbol{h}_n(\boldsymbol{q})\Delta t\,, \tag{6.23}$$

where Δt is a small step size. Having determined $h_n(\boldsymbol{q})'$, α is redesigned as

$$\alpha = \begin{cases} \alpha_{\min} & \text{if } h_n(\boldsymbol{q})' < 0, \\ 1 - (K_H e_H + K_{\dot{H}} e_{\dot{H}}) & \text{otherwise}\,, \end{cases} \tag{6.24}$$

which means that the robot speed is limited to the minimum possible velocity if the current robot configuration is near a constrained local maximum in terms of the objective.

Difference Current and Desired Value of the Performance Criterion Since the range of the minimal gradient can be large, it may be challenging to tune the gain K_H in (6.14). Furthermore, the gradient provides no clear information about the error in $H(\boldsymbol{q})$ itself. The difference

$$e_H = H(\boldsymbol{q}) - H_d\,, \tag{6.25}$$

thus gives a more intuitive formulation of the null space performance error of the current value $H(\boldsymbol{q})$ and a desired value H_d. When selecting H_d, the kinematic self-motion

capabilities of the robot should be taken into account. If H_d is selected arbitrarily, then it can occur that the goal is not reachable. If $\alpha = 0$ is allowed, the robot may stop its motion entirely because e_H does not converge to zero. In this situation, task scaling or task transition schemes can relax constraints on the main task and resume operation. If it is required that the objective fulfills $H(\boldsymbol{q}) \leq H_d$, then it should be considered to reformulate the task as a constraint [299].

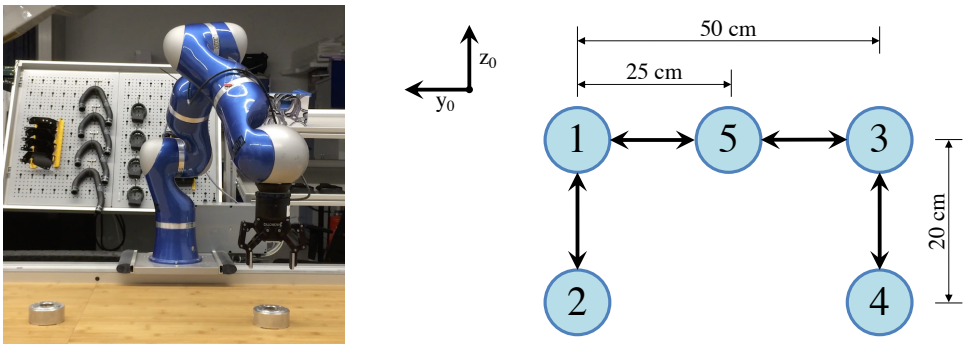
To ensure that H_d is feasible, it should be located on a reachable part of the self-motion manifold. For local null space optimization schemes, H_d can be defined as the next local minimum in the direction of the negative gradient, which can be determined by iteratively integrating the null space velocity $\dot{\boldsymbol{q}}_n(\boldsymbol{q}) = -\boldsymbol{Z}(\boldsymbol{q})^\mathsf{T}\boldsymbol{h}_n(\boldsymbol{q})$. In the optimization literature, many schemes exist to find such a minimum, e.g., via gradient descent. For the LWR, this was done in the previous section, for example. Having selected a feasible H_d, (6.25) represents a meaningful error in terms of the performance criterion. Accordingly, the time scaling parameterization should be more intuitive than one of the previous schemes.

6.2.3 Experiment

In order to validate the performance of the proposed time scaling schemes, an experiment with a 7-DOF LWR IV is conducted. The robot shall perform a pick and place task; the desired Cartesian end-effector trajectory is illustrated in Fig. 6.9, which is very similar to the one in the previous section. The motion sequence is $1 \to 2 \to 1 \to 3 \to 4 \to 3 \to 5$, the desired robot speed is 0.8 m/s. A Cartesian $SE(3)$ task is considered; thus, the robot has one redundant degree of freedom. The primary robot controller is a Cartesian impedance controller [4], the null space controller shall minimize the reflected robot mass $H(\boldsymbol{q}) = m_u(\boldsymbol{q})$ in the direction of travel. The controller was implemented according to Sec. 6.1 and shall improve the robot's collision safety. First, a nominal motion without a null space scheme or time scaling is performed. Then, the null space controller is activated, again without time scaling. Finally, the null space scheme is used in combination with all proposed time scaling methods. For all schemes, $\alpha_{\min} = 0.3$ is selected in order to allow for a significant velocity decrease but to avoid a (temporary) complete stop of the system.

The experimental results are illustrated in Fig. 6.10. For the sake of brevity, only the results for three motion segments are depicted, namely $1 \to 2$ (left column), $1 \to 3$ (middle column), and $3 \to 4$ (right column). The results for the other motion segments are similar. The reflected mass is illustrated over the Cartesian end-effector position in the top row, represented by the path parameter $s \in [0, 1]$. For motions $1 \leftrightarrow 2$ and $3 \leftrightarrow 4$, s corresponds to 25 cm distance in z_0-direction, for $1 \leftrightarrow 3$ to 50 cm in y_0-direction; see Fig. 6.9. The time scaling factor α over time is depicted in the middle row, and in the bottom row, the path parameter over time. This representation allows determining when and to what extent the robot speed is reduced and how much extra time is required to accomplish the task compared to the nominal motion. The nominal trajectory without null space scheme or time scaling is represented by a black line, null space control without time scaling by a gray line, null space control with time scaling based on gradient-based scheme, gradient and Hessian, and known local minimum $H_d(\boldsymbol{q})$ by a blue, red, and yellow line. The next local minimum in the direction of the negative gradient of $H(\boldsymbol{q})$ is represented by a green line.

Regarding the nominal motion (top row in Fig. 6.10) one can observe that the reflected



(a) **(b)**

Fig. 6.9: Time scaling experiment: Robot setup (a) and schematic of the pick and place task trajectory in the y_0/z_0 plane (b). © 2019 IEEE [288].

mass is generally higher[2] when compared to the other schemes that employ the null space controller. However, during motion $1 \rightarrow 3$, the nominal trajectory reaches a local minimum in reflected mass (at $s \approx 0.4$ s) by coincidence. When the null space controller is used without any time scaling scheme, the reflected mass increases in all motion segments for the largest portion of the motion. This is due to the competing dynamics between the primary and the null space task. Given the relatively high end-effector velocity, the magnitude of $\dot{H}_x(\boldsymbol{q})$ is typically larger than $\dot{H}_{\text{ns}}(\boldsymbol{q})$. Due to the limited null space dynamics, the minimization of the reflected mass mainly takes place at the end of a motion segment, where the velocity of the end-effector becomes zero.

The introduction of a time scaling scheme clearly improves the performance of the null space scheme as the robot can reach a lower reflected mass throughout the trajectory. For the considered problem and trajectory, the time scaling law based on (6.25) has a relatively better ability to maintain the reflected mass closer to the minimum than the other schemes. The performance of the gradient-based and gradient and Hessian schemes are identical for all motion segments except for segment $3 \rightarrow 4$[3]. The initial configuration in this segment is close to a local maximum in reflected mass, and therefore, the gradient itself does not give a proper indication about the error in reflected mass.

For motion $1 \rightarrow 2$, the reflected mass is very close to the desired value (note the range of the y-axis in the top left figure) for all considered control schemes. This means that time scaling is only active for a short period; otherwise, the motion and final time remain unchanged; see the timely evolution of α and s in the first column. For the motion segments $1 \rightarrow 3$ and $3 \rightarrow 4$, the velocity is being reduced for a larger period, which leads to a significant decrease in reflected mass, but in turn also to an increase in cycle time.

Discussion

The experimental results confirm that time scaling is a simple, yet effective method to improve the performance of auxiliary tasks. The influence of the time scaling scheme on the operational velocity depends on several factors. If the dynamics of the main task are

[2]For motion $1 \rightarrow 2$ the reflected mass is not illustrated because it is above the considered range.

[3]In the top right plot in Fig. 6.10 (motion segment $3 \rightarrow 4$), s is below zero at the beginning of the motion. Here, the robot moves ≈ 2 mm in the opposite direction due to imperfect null space projection and/or controller overshoot.

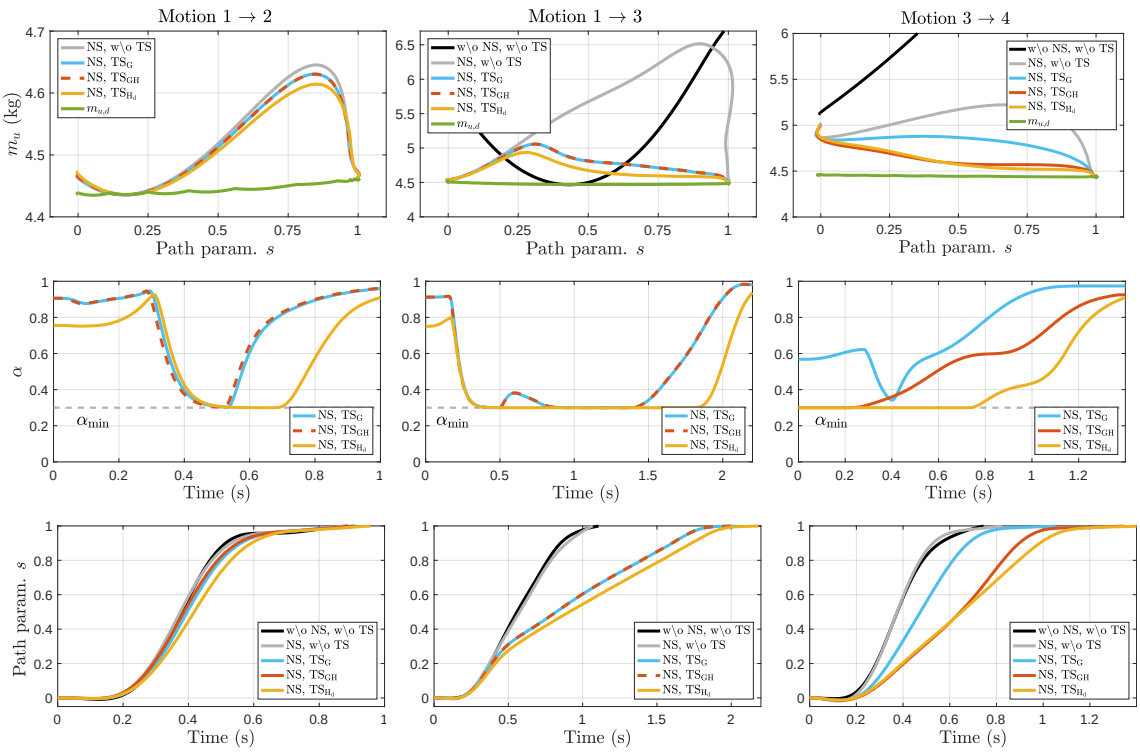


Fig. 6.10: Experimental results for a pick and place task on a LWR IV. The left column represents the motion $1 \rightarrow 2$, the middle column the motion $1 \rightarrow 3$, and the right column the motion $3 \rightarrow 4$; see Fig. 6.9. In the upper row, the reflected mass over the path parameter s is depicted, in the middle row the time scaling factor α over time, and in the lower row the path parameter over time. Legend: w\o NS, w\o TS: nominal trajectory without null space control or time scaling (black); NS, w\o TS: null space control without time scaling (gray); NS, $\{\mathrm{TS_G}, \mathrm{TS_{GH}}, \mathrm{TS_{H_d}}\}$: Null space control with gradient-based scheme, gradient and Hessian, and known local minimum H_d, respectively (blue, red, yellow); $m_{u,d}$: desired local minimum in reflected mass (green). © 2019 IEEE [288].

slower than those of the auxiliary task, then the time scaling scheme will likely be inactive most of the time. If the main task is much faster than the auxiliary task, then the error in the performance index may lead to a significant velocity reduction for a large part of the trajectory. However, being able to speed up means that the method is generally superior to only scaling the robot velocity by a constant factor for the entire trajectory. The value of the minimum allowable scaling factor $\alpha_{\min}$ can influence the system behavior significantly. Setting $\alpha_{\min}$ to very low values may result in a "stop-and-go" like motion. If $\alpha_{\min}$ is large, then the velocity decrease of the primary task can have a negligible benefit on null space task fulfillment.

6.2.4 Extension to Multiple Prioritized Tasks

In this section, the concept is extended to an arbitrary number of hierarchical auxiliary tasks. Given a task hierarchy with r priority levels, each task $\boldsymbol{x}_i(\boldsymbol{q}) \in \mathbb{R}^{m_i \times n}$ is defined by the mapping $\boldsymbol{x}_i = \boldsymbol{f}_{k,i}(\boldsymbol{q})$ on a kinematic level and the mapping $\dot{\boldsymbol{x}}_i = \boldsymbol{J}_i(\boldsymbol{q})\dot{\boldsymbol{q}}$ on a differential level with $\boldsymbol{J}_i(\boldsymbol{q}) = \frac{\partial \boldsymbol{f}_{k,i}(\boldsymbol{q})}{\partial \boldsymbol{q}}$. Furthermore, each task consists of minimizing the performance criterion $H_i(\boldsymbol{x}_i)$ locally[4]. As before, the primary task is an end-effector positioning task, where the desired trajectory provided an interpolator is parameterized w.r.t. time. The problem is to design the time scaling factor such that the primary and the auxiliary tasks are synchronized while taking the task priority and performance of all $r-1$ null space tasks into account.

Scaling Factor Design

The time scaling factor α is determined in two steps. First, $\alpha_i, i = 2, \ldots, r$ is designed for each task independently from the other tasks. Then, the scaling factors of each level are combined to an overall time scaling factor. For each task, the time scaling law

$$\alpha_i = 1 - (K_{H,i} e_{H,i} + K_{\dot{H},i} e_{\dot{H},i}) \,, \tag{6.26}$$

is used, where $K_{H,i}$ and $K_{\dot{H},i}$ are positive scalars. The problem is now how to define $e_{H,i}$ and $e_{\dot{H},i}$ in a hierarchy consistent way. The original task space velocities $\dot{\boldsymbol{x}}_i$ comprise couplings between the hierarchy levels and can thus not be used. Therefore, local null space velocities $\boldsymbol{v}_i \in \mathbb{R}^{m_i}$ (introduced in [296, 300]) are considered, which are given by

$$\underbrace{\begin{pmatrix} \boldsymbol{v}_1 \\ \vdots \\ \boldsymbol{v}_r \end{pmatrix}}_{\boldsymbol{v}} = \underbrace{\begin{pmatrix} \bar{\boldsymbol{J}}_1(\boldsymbol{q}) \\ \vdots \\ \bar{\boldsymbol{J}}_r(\boldsymbol{q}) \end{pmatrix}}_{\bar{\boldsymbol{J}}} \dot{\boldsymbol{q}} \,. \tag{6.27}$$

Here, $\bar{\boldsymbol{J}}(\boldsymbol{q})$ is the hierarchy consistent Jacobian matrix [300]. The time derivative of $H_i(\boldsymbol{q})$ due to the influence of the primary task and the control action on level i are given by

$$\dot{H}_{i,1}(\boldsymbol{q}) = \frac{\partial H_i(\boldsymbol{q})}{\partial \boldsymbol{x}_i} \boldsymbol{J}_i(\boldsymbol{q}) \bar{\boldsymbol{J}}_1(\boldsymbol{q})^{\boldsymbol{W}+} \dot{\boldsymbol{x}}_1 \,, \tag{6.28a}$$

$$\dot{H}_{i,i}(\boldsymbol{q}) = \frac{\partial H_i(\boldsymbol{q})}{\partial \boldsymbol{x}_i} \boldsymbol{J}_i(\boldsymbol{q}) \bar{\boldsymbol{J}}_i(\boldsymbol{q})^{\boldsymbol{W}+} \boldsymbol{v}_i \,. \tag{6.28b}$$

[4] In the sequel, the dependency on $\boldsymbol{q}$ and $\boldsymbol{x}$ is omitted.

One can now systematically define the synchronization error on each level as

$$e_{\dot{H},i} = \begin{cases} \dot{H}_{i,1}(\boldsymbol{q}) - \dot{H}_{i,i}(\boldsymbol{q}), & \text{if } \operatorname{sign}(\dot{H}_{i,1}(\boldsymbol{q})) \neq \operatorname{sign}(\dot{H}_{i,i}(\boldsymbol{q})) \\ & \wedge |\dot{H}_{i,1}(\boldsymbol{q})| > |\dot{H}_{i,i}(\boldsymbol{q})|\,, \\ 0, & \text{otherwise}\,, \end{cases} \tag{6.29}$$

which means that $e_{\dot{H},i}$ is only relevant when the primary task has a negative influence on the null space task on level i. The error $e_{H,i}$ in performance criterion can be formulated in terms of any of the schemes proposed in Sec. 6.2.2. Here, the error is defined as $e_{H,i} = H_i(\boldsymbol{q}) - H_{i,d}$ with $H_{i,d}$ being a constrained local minimum for task $\boldsymbol{x}_i$. To determine $H_{i,d}$, one can define the minimal gradient on each level as

$$\boldsymbol{h}_{n,i}(\boldsymbol{q}) = \bar{\boldsymbol{J}}_i(\boldsymbol{q})\boldsymbol{W}^{-1}\boldsymbol{J}_i(\boldsymbol{q})^{\mathsf{T}}\left(\frac{\partial H_i(\boldsymbol{x}_i)}{\partial \boldsymbol{x}_i}\right)^{\mathsf{T}}, \tag{6.30}$$

where the gradient of the objective (possibly defined in task space) is first mapped to configuration space using $\boldsymbol{J}_i(\boldsymbol{q})^{\mathsf{T}}$; then $\boldsymbol{W}^{-1}$ compensates for the rotation of the gradient included in $\bar{\boldsymbol{J}}_i(\boldsymbol{q})$ [301]. Finally, the gradient is mapped to local null space coordinates via $\bar{\boldsymbol{J}}_i(\boldsymbol{q})$. The null space velocity in the direction of the gradient can be obtained by $\dot{\boldsymbol{q}}_{n,i} = \bar{\boldsymbol{J}}_i(\boldsymbol{q})^{\boldsymbol{W}+}\boldsymbol{h}_{n,i}(\boldsymbol{q})$. Similar to the case of one degree of redundancy, one can now repeatedly integrate $\dot{\boldsymbol{q}}_{n,i}$ on all levels independently until the next constrained local minimum $H_{i,d}$ is determined.

Having determined α_i on all levels, the goal is now to combine them to the overall time scaling factor. One possible solution is to select the most conservative α_i of all the hierarchy levels, i.e.,

$$\alpha = \min(\alpha_i), \quad i = 2, \ldots, r\,. \tag{6.31}$$

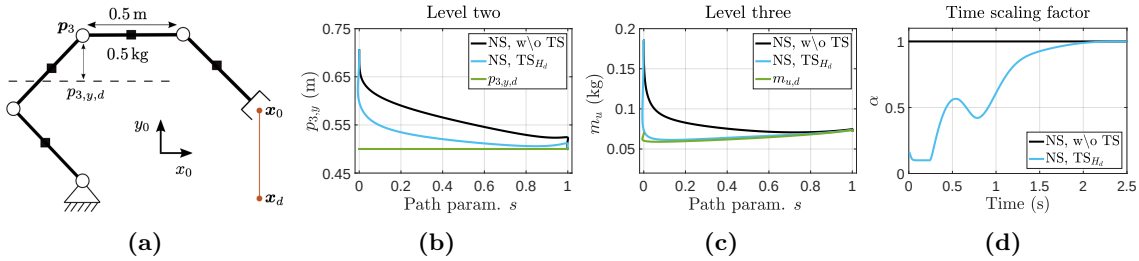
Alternatively, each α_i can be weighted and combined as

$$\alpha = \sum_{i=2}^{r} w_i \alpha_i\,. \tag{6.32}$$

Here, w_i are non-negative weights which are selected such that $\sum_{i=2}^{r} w_i = 1$ and $w_i > w_j, \forall i < j$. This means that a high priority task influences the robot speed more than a low priority task.

Simulation Results

In order to validate the hierarchical time scaling approach, simulations were carried out on the 4R DOF planar robot shown in Fig. 6.11 (left). Each link has 0.5 m length, a 0.5 kg point mass located at the link center. The initial configuration is $\boldsymbol{q}_0^{\mathsf{T}} = [135, -90, -45, -45]^\circ$. The primary task is an x_0/y_0 linear Cartesian end-effector motion of 40 cm distance in negative y_0-direction. For the considered task, the robot has two redundant DOF. The secondary task is to minimize the vertical distance $p_{3,y,}$ of the third joint to a horizontal plane located at $y_0 = 0.5$ m. The tertiary task minimizes the reflected mass perceived at the end-effector like in the previous experiment. The simulation results are illustrated in Fig. 6.11. Both auxiliary tasks converge to their local minimum ($y_{3,d}$ on level two and $m_{u,d}$ on level three), which shows that the definition of $e_{H,i}$ is



(a) **(b)** **(c)** **(d)**

Fig. 6.11: Time scaling in combination with hierarchical null space control: Simulation results for a planar 4R robot. The robot is illustrated in (a), where the vertical position of the third joint is $\boldsymbol{p}_3^\mathsf{T} = [p_{3,x}, p_{3,y}]$; the initial and desired Cartesian position are $\boldsymbol{x}_0$ and $\boldsymbol{x}_d$, respectively. In (b) and (c), the results for level two (vertical position of the third joint) and level three (minimization of reflected mass) are depicted. The evolution of the time scaling factor is shown in (d). Legend: NS, wno TS: nominal motion with null space control but without time scaling; NS, TS$_{\mathrm{H_d}}$: null space control with time scaling; $p_{3,y,d}, m_{u,d}$: Desired vertical position of second joint and reflected mass. © 2019 IEEE [288].

consistent in terms of the task prioritization framework. This ensures that time scaling is only enabled when possible to improve the performance index on level i. When using time scaling, all auxiliary tasks achieve better performance and converge faster to the local minimum than without using time scaling.

6.3 Intuitively Interpretable Reconfiguration via Interactive Control

In practice, it often occurs that the joint configuration associated with an end-effector pose is unfavorable, which means that the (redundant) robot should be reconfigured kinematically. Four exemplary problems the author considers to be important are:

a) **"Get out of my way!"**

The robot obstructs movements of the human coworker, e.g., the human intends to grasp an object which is not reachable without reconfiguring the robot; see Fig. 6.12 a).

b) **"Untwist yourself!"**

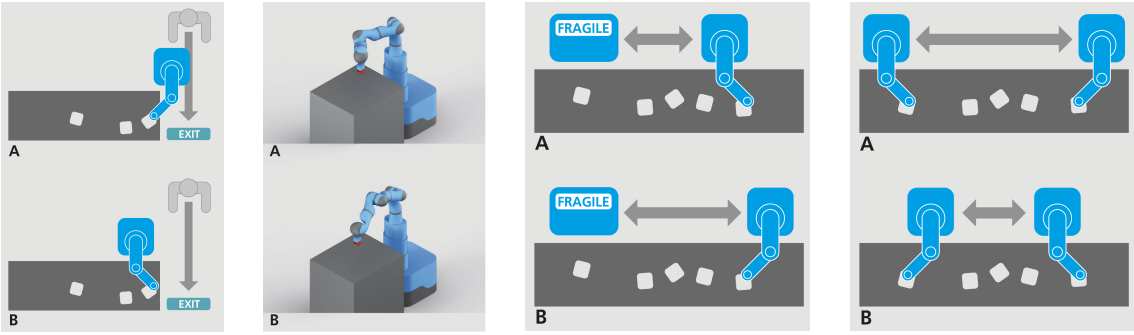
The robot has an unfavorable configuration to execute a certain task. A different configuration would lead to better manipulability or reachability; see Fig. 6.12 b).

c) **"Don't touch the stuff!"**

The robot is in the vicinity of persons or objects, which can lead to undesired, potentially dangerous collisions; see Fig. 6.12 c). By reconfiguring the robot, one can increase the distance to the persons/obstacles and reduce the hazard potential.

d) **"Glue motion segments together!"**

A task that consists of several motions shall be optimized, e.g., in terms of traveled distance; see Fig. 6.12 d). This can be achieved by modifying configurations in different motion segments.



(a) (b) (c) (d)

Fig. 6.12: Four situations in which an internal reconfiguration of the robot improves collaboration, safety, or performance. In each figure, A illustrates the robot configuration before and B after reconfiguration. © 2017 IEEE [302].

It is either necessary or beneficial to kinematically reconfigure the robot in these cases while the end-effector position and orientation remain unchanged. Ideally, one would like to use verbal commands expressed in natural language (such as "Get out of my way!" in problem a)) to initiate the desired robot motion. Such intuitive actions need to be interpreted before the robot can carry out the desired motion [68, 69]. Alternatively, the robot may be guided manually using remote or haptic interfaces. For the latter problem, [303, 304] introduced a method to teach null space motions through a six-DOF teaching device or by directly touching and moving the links.

This thesis contributes to manual, interactive null space control for redundant robots, where the user can guide the robot via remote or haptic interfaces. Single-arm manipulators, in particular the LWR, are considered which may be mounted on a mobile platform and have (including platform) more DOF than necessary to fulfill a particular end-effector task. The dynamics of the fully actuated, holonomic robot can be expressed by the standard rigid body dynamics (5.54); cf. Chapter 2. Users are provided with an intuitive tool to kinematically reconfigure the robot, which can be integrated with only little effort. Typically, it is difficult to anticipate a null space motion, even when the movement is commanded on joint level. This work aims to generate predictable self-motions in the sense that the user knows in advance *how* the robot will move during reconfiguration. This turns the motion into an easily interpretable and safe robotic behavior. The approach employs joint-space constraints based on a selection of suitable coordinates to obtain null space projections that can be used with teaching devices or direct physical interaction. Practical examples for a seven-DOF LWR mounted on a one-DOF linear axis and a three-DOF mobile platform are given.

6.3.1 Related Work

There exist several ways to induce self-motions that solve the four practical problems mentioned above. Reconfiguration can be tackled on different levels of complexity and abstraction, ranging from pure manual control to verbal commands. The latter can be expressed in natural language and require no a-priori knowledge about robot kinematics or dynamics but must be translated and grounded accordingly. In the following, the considered robot dynamics are described first. Then, a brief overview of different ways to

command self-motions is given. Problem a) where the robot obstructs the human coworker is taken as an example. The goal is that the robots clears the path of the human by performing a null space motion. Afterwards, the related research of this work is reviewed.

Metrics-Based Redundancy Resolution

The classical approach to redundancy resolution in robotics is to formulate mathematical metrics which are to be optimized. One can regard the intuitive action "Get out of my way!" as a collision avoidance problem, i.e., a possible metric would be the minimum Cartesian distance between the robot structure and the human. Appropriate proprioceptive/exteroceptive sensing is required to evaluate this metric. The optimization then provides velocities or joint torques, which are projected onto the null space of the Jacobian matrix and subsequently added to the main task controller.

Generating Robot Motions from Symbolic Action Specifications

Instead of treating reconfiguration solely from a control point of view, the authors of [70, 71] aim at bridging the gap between the symbolic and the control layer. The idea is to define constraint-based movement specifications, which have representations in both the symbolic and the control domain. This enables the robot to reason about its actions, and it provides an interface that high-level planning algorithms can access. The approach proposed in [70, 71] is briefly summarized as well as applied to problem a) in the following. The human posture, the environment's geometry, and the robot posture are represented by so-called Geometric Features (GF) like points, lines, or planes, that are semantically annotated. The GF are linked by Feature Functions (FF), which can be the height or the distance between GF, for example. Using the FF, the spatial relation between the robot and objects can be expressed; two examples: "The robot is close to the human" or "The mobile platform is aligned with box A". The sum of these relations is denoted Qualitative Robot Posture (QRP). It can be accessed by higher planning layers and used to command movements. The desired QRP can be translated into action by formulating so-called Movement Constraints, which relate a pair of GF and a FF to a desired evaluation value. These constraints enable a controller to execute robot motions. Concerning the obstacle avoidance problem, one can assign GF to robot links and the human and relevant objects in the environment. Distance linkages between these GF are then defined in terms of FF. The logic-based representation of the command "Get out of my way!" comprises the distances of the robot to the obstacles and the objective to maximize this distance. With this representation, the constraint set can be generated and forwarded to the controller, which executes the self-motion.

Interactive Null Space Control

In terms of interactive guidance, a method to intuitively teach redundant robots was proposed in [303, 304]. The idea is to map desired Cartesian velocities to a reference frame attached to a certain joint through a six-DOF teaching device such as a SpaceMouse, which is now part of many industrial programming devices. The kinematic robot chain is virtually cut off at joint i. The relationship of the desired velocity $\dot{\boldsymbol{x}}_{d,i} \in \mathbb{R}^6$ of the reference frame and the respective joint velocity can be described by

$$\dot{\boldsymbol{q}}_{d,i} = \boldsymbol{J}_i(\boldsymbol{q})^{\boldsymbol{W}+} \dot{\boldsymbol{x}}_{d,i} \,, \tag{6.33}$$

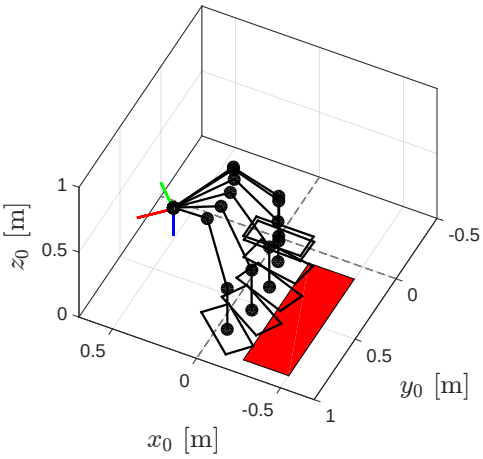


Fig. 6.13: Reconfiguration of a mobile robot consisting of a LWR and a three-DOF platform using a Moore-Penrose pseudoinverse null space projection. The red rectangle represents an obstacle located in the robot's workspace. © 2017 IEEE [302].

where $\dot{\boldsymbol{q}}_{d,i} \in \mathbb{R}^i$ is the desired joint velocity and $\boldsymbol{J}_i(\boldsymbol{q})^{\boldsymbol{W}+} \in \mathbb{R}^{i \times m}$ is the weighted inverse of the body Jacobian matrix associated to joint i. This velocity is projected onto the null space of the Jacobian matrix and added to the main task velocity, which yields

$$\dot{\boldsymbol{q}}_d = \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\dot{\boldsymbol{x}}_d + \left(\boldsymbol{I} - \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\boldsymbol{J}(\boldsymbol{q})\right)\begin{pmatrix}\dot{\boldsymbol{q}}_{d,i} \\ \boldsymbol{0}_{(n-i)\times 1}\end{pmatrix} . \tag{6.34}$$

To project the desired $\dot{\boldsymbol{q}}_{d,i}$ onto the null space of $\boldsymbol{J}(\boldsymbol{q})$, the classical Moore-Penrose pseudoinverse projector ($\boldsymbol{W} = \boldsymbol{I}$) was used in [303, 304]. In addition to using a six-DOF teaching device, the authors proposed using the estimated external torque of each joint as input for null space control. Then, touching the robot's links would perform an evasive motion to minimize the external joint torques. In the following example, the resulting self-motion and the limitations of the approach are described.

Consider a seven-DOF LWR, which is mounted on an omnidirectional platform; see Fig. 6.13. The initial position of the mobile platform is $x_0 = y_0 = 0$ m, the robot platform shall move in positive Cartesian y_0-direction while keeping the same end-effector pose. When attaching the reference frame to the first LWR joint and commanding $\dot{\boldsymbol{x}}_{d,i}^{\mathsf{T}} = [0, \dot{y}_d, 0, 0, 0, 0]$ using (6.33), (6.34) with $\dot{\boldsymbol{x}}_d = \boldsymbol{0}$, one obtains a self-motion that kinematically reconfigures the robot. However, it is generally not possible to move exactly in direction $\dot{\boldsymbol{x}}_{d,i}$ (here: y_0-direction) with the selected pseudoinverse null space projector. This is because the projector minimizes the joint velocities but does not take any other constraints into account. This behavior is illustrated in Fig. 6.13, where it is shown that the robot also moves to some extent in the Cartesian x_0-direction. During the self-motion, the robot collides against the red obstacle located in the workspace.

Concerning this example, the goal of this work is (loosely speaking) to move the robot platform in y_0-direction only and not in x_0-direction. This avoids a collision with the obstacle and makes the motion predictable. Furthermore, the interaction interface shall be intuitive and the implementation straightforward. The approach to solving this problem is presented in the next section.

6.3.2 Approach

The problem of generating intuitive and predictable internal motions is approached by selecting coordinates that allow interpreting the resulting self-motion when they are actuated or constrained. Based on this selection, null space projections are developed that incorporate joint space constraints that ensure the chosen coordinates always move in a coordinated fashion. For the sake of clarity, only static end-effector poses are considered, i.e., the task velocity is zero. The approach consists of seven steps:

Step 1 Identify and select interpretable joint coordinates

Step 2 Formulate joint-space constraints

Step 3 Incorporate constraints into the extended Jacobian matrix

Step 4 Derive the null space projection

Step 5 Integrate the null space projection into the control law

Step 6 Select the teaching interface

Step 7 Assign the null space projection(s) and actuated joints to the user interface

In the following, these steps are described in detail.

Interpretable Subset of Joint Coordinates (Step 1 and 2)

Firstly, the user needs to determine those joint coordinates $\boldsymbol{q}_c \subset \boldsymbol{q}$, respectively a combination of these, that allow the user to anticipate the resulting null space motion when actuated and/or constrained. The selected coordinates should be suitable to represent the null space motion, i.e., they should (locally) strictly increase/decrease over the self-motion. For (holonomic) mobile robots, consisting of a platform and a manipulator with at least six DOF, all platform coordinates can be part of $\boldsymbol{q}_c$. Concerning the manipulator, only the LWR is considered in this work. The LWR self-motion is governed by a rotation of the elbow when the Jacobian matrix is non-singular; cf. Sec. 6.1. The position of joints 3 and 5 strictly increases over the elbow rotation. The generalization to other kinematics is subject to future research.

Incorporation of Constraints via the Extended Jacobian Matrix (Step 3)

Having selected $\boldsymbol{q}_c$, the next step is to formulate constraints for the auxiliary tasks. A constraint may comprise several joints or a single joint only. The latter case infers that the motion of the respective joint is blocked. The well-known extended Jacobian matrix $\boldsymbol{J}_\mathrm{e}(\boldsymbol{q})$ is used to incorporate the joint-space constraints, where one can add up to $r \leq n - m$ independent constraints to the original end-effector task [198, 199]; cf. Sec. 2.3. Typically, $n - m$ constraints are considered such that $\boldsymbol{J}_\mathrm{e}(\boldsymbol{q})$ becomes invertible [198]. However, in this situation, no null space can be exploited to generate self-motions. Since we want to perform null space motions, we can consider at most $r \leq n - m - 1$ joint-space constraints. The kernel of the extended Jacobian matrix then fulfills

$$\boldsymbol{J}_\mathrm{e}(\boldsymbol{q}) \ker(\boldsymbol{J}_\mathrm{e}(\boldsymbol{q})) = \boldsymbol{0}\,. \tag{6.35}$$

When rewriting the previous equation as

$$\begin{pmatrix} \boldsymbol{J}(\boldsymbol{q}) \ker(\boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})) \\ \dfrac{\partial \boldsymbol{h}(\boldsymbol{q})}{\partial \boldsymbol{q}} \ker(\boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})) \end{pmatrix} = \boldsymbol{0}\,, \tag{6.36}$$

one can see that the kernel of the extended Jacobian matrix is a subspace of the task Jacobian matrix null space, i.e., $\ker(\boldsymbol{J}_e(\boldsymbol{q})) \subset \ker(\boldsymbol{J}(\boldsymbol{q}))$.

Null Space Projection and Control (Step 4 and 5)

To project joint velocities onto the null space of the Jacobian matrix while fulfilling the constraint $\boldsymbol{h}(\boldsymbol{q})$, one can use the projector

$$\boldsymbol{N}_{\mathrm{vel}}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})^{\boldsymbol{W}+}\boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})\,, \tag{6.37}$$

for velocity control and

$$\boldsymbol{N}_{\mathrm{tor}}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})^{\mathsf{T}} \left(\boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})^{\boldsymbol{W}+}\right)^{\mathsf{T}}\,, \tag{6.38}$$

for joint torque control. Here, $\boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})^{\boldsymbol{W}+}$ is the weighted, generalized inverse of the extended Jacobian matrix. Projectors (6.37) and (6.38) are identical if $\boldsymbol{W} = \boldsymbol{I}$. In the sequel, the projector (6.38) is mainly considered (without loss of generality). In [305], it was shown that the projection can also be calculated using the kernel of the (extended) Jacobian matrix. From singular value decomposition (SVD) of $\boldsymbol{J}(\boldsymbol{q})$ one obtains the orthonormal matrix $\boldsymbol{V}(\boldsymbol{q}) = [\boldsymbol{V}_1(\boldsymbol{q}) \,|\, \boldsymbol{V}_2(\boldsymbol{q})]$, where $\boldsymbol{V}_1(\boldsymbol{q})$ and $\boldsymbol{V}_2(\boldsymbol{q})$ constitute the orthonormal bases of the range space of $\boldsymbol{J}(\boldsymbol{q})^{\mathsf{T}}$ and the null space of $\boldsymbol{J}(\boldsymbol{q})$, respectively. The projection matrix can be expressed as

$$\boldsymbol{N}_{\mathrm{tor}}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})^{\mathsf{T}} \left(\boldsymbol{J}_{\mathrm{e}}(\boldsymbol{q})^{\boldsymbol{W}+}\right)^{\mathsf{T}} \tag{6.39a}$$

$$= \boldsymbol{W}^{\mathsf{T}}\boldsymbol{V}_2(\boldsymbol{q})^{\mathsf{T}} \left(\boldsymbol{V}_2(\boldsymbol{q})\boldsymbol{W}^{\mathsf{T}}\boldsymbol{V}_2(\boldsymbol{q})^{\mathsf{T}}\right)^{-1} \boldsymbol{V}_2(\boldsymbol{q})\,. \tag{6.39b}$$

To execute a desired motion, the CLIK control law

$$\dot{\boldsymbol{q}} = \boldsymbol{J}(\boldsymbol{q})^{\boldsymbol{W}+}\boldsymbol{K}\left(\boldsymbol{x}_d - \boldsymbol{x}\right) + \boldsymbol{N}_{\mathrm{vel}}(\boldsymbol{q})\dot{\boldsymbol{q}}_d\,, \tag{6.40}$$

can be used when differential kinematics are considered [282]. Here, $\boldsymbol{x}_d$ is the desired Cartesian end-effector pose, and $\boldsymbol{K}$ is the proportional gain matrix in the correction term. The vector $\dot{\boldsymbol{q}}_d$ includes the desired velocities of coordinates $\boldsymbol{q}_c$ while the other entries are zero. In joint torque control, one can add a null space torque to the control variable of the primary controller, i.e.,

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\mathrm{prim}} + \boldsymbol{N}_{\mathrm{tor}}(\boldsymbol{q})\boldsymbol{\tau}_{\mathrm{ns},d}\,. \tag{6.41}$$

Here, $\boldsymbol{\tau}_{\mathrm{prim}}$ is the torque of the primary control law, e.g., an impedance controller that achieves a desired stiffness and damping in either joint or Cartesian space [4]. Gravity compensation is included in this term. The desired null space torque denoted by $\boldsymbol{\tau}_{\mathrm{ns},d}$ is projected onto the null space of the Jacobian matrix.

Remark: Blocking Joint Movements If the movement of a joint q_j shall be blocked, one can modify the columns of the Jacobian matrix instead of adding the extra row

$$\boldsymbol{h}_i(\boldsymbol{q}) = [0, \cdots, q_j, \cdots, 0]\,. \tag{6.42}$$

Let $\boldsymbol{z}$ be a vector that contains the indices of the joints that shall be blocked. Now the columns of the Jacobian matrix with all indices contained in $\boldsymbol{z}$ are replaced by zero column vectors. This yields the modified Jacobian matrix $\boldsymbol{J}_{\setminus \boldsymbol{z}}(\boldsymbol{q})$; the null space projector is given by

$$\boldsymbol{N}_{\text{tor}}(\boldsymbol{q}) = \boldsymbol{I}_{\setminus \boldsymbol{z}} - \boldsymbol{J}_{\setminus \boldsymbol{z}}(\boldsymbol{q})^{\mathsf{T}} \left(\boldsymbol{J}_{\setminus \boldsymbol{z}}(\boldsymbol{q})^{\boldsymbol{W}+}\right)^{\mathsf{T}}\,, \tag{6.43}$$

where $\boldsymbol{I}_{\setminus \boldsymbol{z}}$ is formed by replacing the entries I_{z_i,z_i} of the identity matrix by zeros.

Teaching Interfaces (Step 6 and 7)

To set the joint constraints and command desired joint velocities, positions, or torques, one can use a remote teaching device or a haptic interface, typically the robot itself. In [303], a six-DOF SpaceMouse was used to command the Cartesian velocity (translational and angular) of a reference frame that is associated with a particular joint. The resulting joint velocity was then projected onto the null space of the Jacobian matrix. This approach can be extended by introducing joint-space constraints to the null space projection (6.34); the interaction would otherwise remain the same. Instead of commanding the desired Cartesian velocity of a reference frame, the teaching device can also be used for joint-space control. Specific (combinations of) joints and/or constraints can be assigned to the controller axes. When commanding positive/negative values in the respective direction, the robot would move back and forth along the constraint. For example, the three-DOF platform described in Section 6.3.3 has two translational coordinates q_x and q_y, which enable the robot to move in x_0- and y_0-direction, respectively. The input from a gampad's analog stick could define the desired angle between both coordinates, for example.

A popular alternative to remote-programming devices is the physical interaction with the robot itself, as collaborative robots like the LWR have rich sensing capabilities. Also here, one can distinguish between joint-space and Cartesian space interaction. For example, in [304], the links were touched directly to initiate evasive self-motions of the robot. For this, the robot must be capable of estimating external joint torques. Especially for mobile platforms, which usually have a velocity interface and not a torque interface, this interaction is often impossible. Furthermore, even if external joint torques can be measured or estimated, it can occur that some links are not in the reach of the human, which means that the joint cannot be moved manually.

A well-established method to interact with torque-controlled robots is to use haptic gestures at the end-effector, i.e., to push the robot in Cartesian x_0-, y_0-, and z_0-direction and interpret the signals [102, 306–308]. The end-effector can either be equipped with a force-torque sensor or the external forces $\boldsymbol{\tau}_{\text{ext}}$ are estimated through a collision detection scheme [104]. With

$${}^0\hat{\mathcal{F}}_{\text{ext}} = \begin{pmatrix} {}^{x_0}\hat{f}_{\text{ext}} \\ {}^{y_0}\hat{f}_{\text{ext}} \\ {}^{z_0}\hat{f}_{\text{ext}} \\ {}^{0}\hat{\mathcal{N}}_{\text{ext}} \end{pmatrix} = {}^0\boldsymbol{J}(\boldsymbol{q})_{EE}^{\#,\mathsf{T}}\,\hat{\boldsymbol{\tau}}_{\text{ext}}\,, \tag{6.44}$$

one obtains the estimated external forces at the end-effector, where ${}^{0}\hat{\mathcal{F}}_{\text{ext}}$ is the estimated external wrench in world coordinates; ${}^{x_0}\hat{f}_{\text{ext}}$, ${}^{y_0}\hat{f}_{\text{ext}}$, and ${}^{z_0}\hat{f}_{\text{ext}}$ the external force in Cartesian x_0-, y_0-, and z_0-direction, respectively. The external torque is ${}^{0}\hat{\mathcal{N}}_{\text{ext}}$, ${}^{0}\boldsymbol{J}(\boldsymbol{q})^{\#}_{EE}$ is an arbitrary pseudoinverse of the Jacobian matrix associated with the end-effector, and $\hat{\boldsymbol{\tau}}_{\text{ext}}$ are the estimated external joint torques. Similar to the control with a SpaceMouse, a specific constraint can be assigned to each Cartesian direction. Alternatively, it is possible to implement an interactive menu to switch between constraints.

6.3.3 Applications

In this section, two practical examples are provided that demonstrate the approach.

Lightweight Robot Mounted on Linear Axis

The first system consists of a linear axis and a seven-DOF LWR. The generalized coordinates are $\boldsymbol{q}^{\mathsf{T}} = [q_x, q_1, q_2, q_3, q_4, q_5, q_6, q_7]$, where q_x is the position of the linear axis and q_i, $i = 1, \ldots, 7$ are the LWR positions. The robot is depicted in Fig. 6.14. It has eight DOF, of which two can be used for null space motions when the end-effector position and orientation (six-DOF task) remain constant. Cartesian impedance control is used, where the position-controlled linear axis has an admittance interface [4,190]. For interactive null space control, $\boldsymbol{q}_c^{\mathsf{T}} = [q_x, q_3]$ is selected, i.e., the position of the platform and the 3rd rotational LWR joint, which represents the elbow configuration. The goal is to decouple the null space motions of the linear axis and the LWR. For this, two null space projectors are required. The first projector shall block the motion of the linear axis while the manipulator is reconfigured. The second projector shall maximize the velocity of the linear axis while the manipulator moves passively. The first projection matrix can be obtained by extending the task Jacobian matrix by the constraint $\partial h(\boldsymbol{q})/\partial \boldsymbol{q} = [1, 0, 0, 0, 0, 0, 0, 0]$. Using the inertia-weighted pseudoinverse $\boldsymbol{J}_{\text{e}}(\boldsymbol{q})^{\boldsymbol{M}+}$, we get the projection matrix

$$\boldsymbol{N}_{\text{tor},1}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}_{\text{e}}(\boldsymbol{q})^{\mathsf{T}} \left(\boldsymbol{J}_{\text{e}}(\boldsymbol{q})^{\boldsymbol{M}+}\right)^{\mathsf{T}} . \tag{6.45}$$

The null space vector associated to this projection has the form

$$\boldsymbol{w}(\boldsymbol{q}) = [0 * \cdots *]^{\mathsf{T}} \in \ker(\boldsymbol{J}(\boldsymbol{q})) . \tag{6.46}$$

With the first entry being zero, it is ensured that the linear axis does not move. According to (6.39), the first projection matrix can also be calculated using $\boldsymbol{w}(\boldsymbol{q})$, namely

$$\boldsymbol{N}_{\text{tor},1}(\boldsymbol{q}) = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{w}(\boldsymbol{q}) \left(\boldsymbol{w}(\boldsymbol{q})^{\mathsf{T}}\boldsymbol{M}(\boldsymbol{q})\boldsymbol{w}(\boldsymbol{q})\right)^{-1} \boldsymbol{w}(\boldsymbol{q})^{\mathsf{T}} . \tag{6.47}$$

In order to determine the second projection matrix, we can make use of the results in Sec. 6.1.5. It was shown that $\boldsymbol{w}(\boldsymbol{q})$ and the vector $\boldsymbol{w}_{\perp}(\boldsymbol{q})$ that is orthogonal to $\boldsymbol{w}(\boldsymbol{q})$ span the null space of the (non-singular) Jacobian matrix. The perpendicular vector is given by (cf. Sec. 6.1.5)

$$\boldsymbol{w}_{\perp}(\boldsymbol{q}) = \boldsymbol{v}_{21}(\boldsymbol{q}) - \frac{\langle \boldsymbol{v}_{21}(\boldsymbol{q}), \boldsymbol{w}(\boldsymbol{q})\rangle}{\langle \boldsymbol{w}(\boldsymbol{q}), \boldsymbol{w}(\boldsymbol{q})\rangle}\boldsymbol{w}(\boldsymbol{q}) , \tag{6.48}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product. The second projector is finally given by

$$\boldsymbol{N}_{\text{tor},2}(\boldsymbol{q}) = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{w}_{\perp}(\boldsymbol{q}) \left(\boldsymbol{w}_{\perp}(\boldsymbol{q})^{\mathsf{T}}\boldsymbol{M}(\boldsymbol{q})\boldsymbol{w}_{\perp}(\boldsymbol{q})\right)^{-1} \boldsymbol{w}_{\perp}(\boldsymbol{q})^{\mathsf{T}} . \tag{6.49}$$

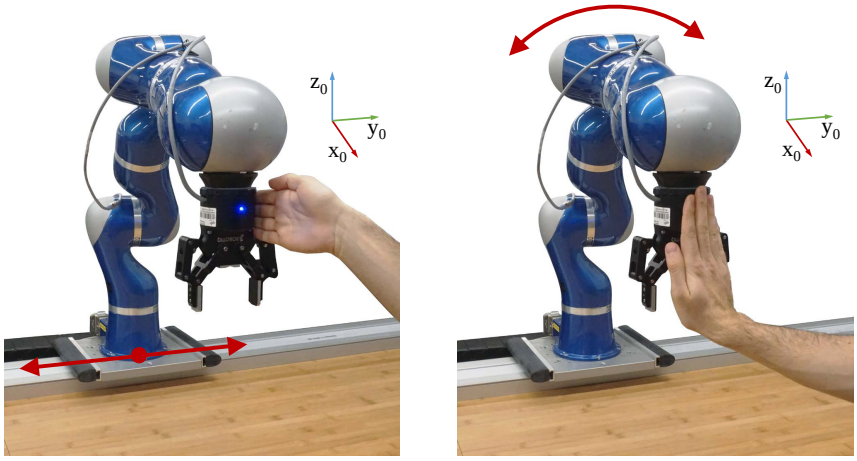


Fig. 6.14: Interactive null space control. Pushing the robot's end-effector in Cartesian y_0-direction results in a null space motion with moving linear axis (left), pushing in x_0-direction results in a rotation of the LWR elbow while the position of the linear axis remains unchanged (right). © 2017 IEEE [302].

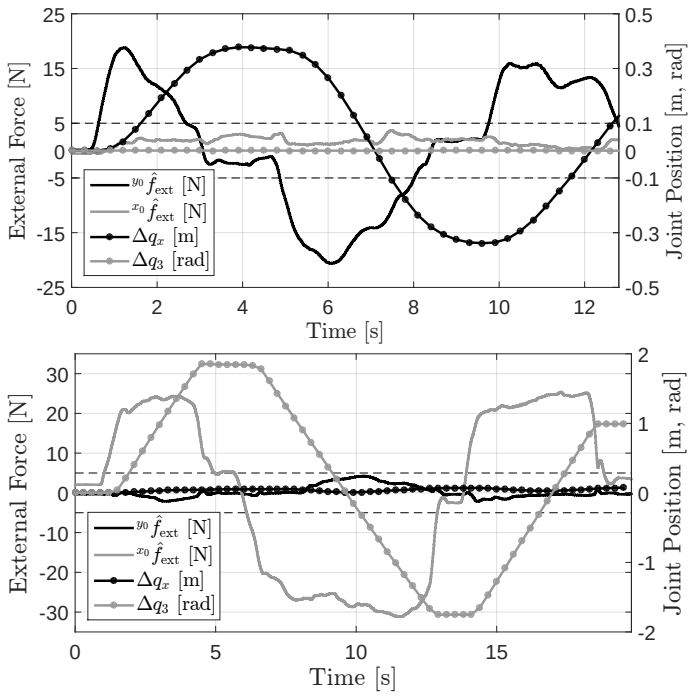


Fig. 6.15: Experimental results of the haptic reconfiguration of the eight-DOF system. In both figures, the estimated Cartesian forces in y_0- and x_0-direction are depicted as well as the position of the linear axis and the 3rd manipulator joint. The joint positions are illustrated in terms of their difference to the initial positions $q_{x,0}$ and $q_{3,0}$ before interaction, i.e., $\Delta q_x = q_x - q_{x,0}$ and $\Delta q_3 = q_3 - q_{3,0}$. The black dashed lines indicate the minimum force required to induce the null space motion (here $\epsilon_{x_0} = \epsilon_{y_0} = 5$ N). © 2017 IEEE [302].

Reconfiguration Through Haptic Gestures In the following, haptic gestures are used to intuitively reconfigure the robot via self-motions. One should be able to move the

linear axis and elbow of the LWR by simply pushing the robot at the end-effector. The translation of the linear axis shall be proportional to a force in Cartesian y_0-direction, the rotation of the elbow proportional to a force in x_0-direction; see Fig. 6.14. The null space motion shall not affect the execution of the main task, i.e., the end-effector pose and impedance. A torque-based null space strategy shall be added to the impedance control framework to ensure a compliant robot behavior at any time. A torque-based control law that incorporates the projectors (6.45) and (6.49) is given by

$$\boldsymbol{\tau}_{\mathrm{ns},d} = \boldsymbol{N}_{\mathrm{tor},1}(\boldsymbol{q}) \begin{pmatrix} 0 \\ 0 \\ 0 \\ k_3(\dot{q}_{3,\mathrm{d}} - \dot{q}_3) \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \boldsymbol{N}_{\mathrm{tor},2}(\boldsymbol{q}) \begin{pmatrix} k_x(\dot{q}_{x,\mathrm{d}} - \dot{q}_x) \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{6.50}$$

where k_3 and k_x are gains for the 3rd robot joint and the linear axis, and $\boldsymbol{\tau}_{\mathrm{ns},d}$ is the null space joint torque that is added to the Cartesian impedance control action. The first term in the previous equation achieves a rotation of the LWR's elbow while the position of the linear axis remains constant. The second term maximizes the velocity of the linear axis while the lightweight arm moves passively. To sense external Cartesian forces, the estimated external joint torques are employed that are described in the previous section, c.f. (6.44). The desired velocities in (6.50) are replaced by

$$\dot{q}_{x,\mathrm{d}} = \begin{cases} 0, & |{}^{y_0}\hat{f}_{\mathrm{ext}}| < \epsilon_{y_0} \\ k_{f,x}\left({}^{y_0}\hat{f}_{\mathrm{ext}} - \mathrm{sign}({}^{y_0}\hat{f}_{\mathrm{ext}})\epsilon_{y_0}\right), & \text{otherwise} \end{cases}, \tag{6.51}$$

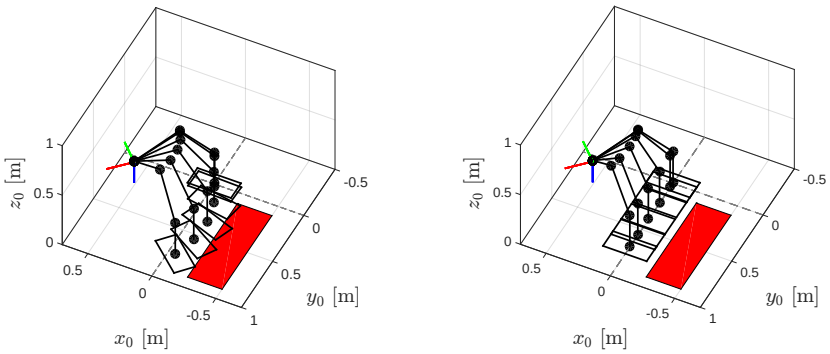
$$\dot{q}_{3,\mathrm{d}} = \begin{cases} 0, & |{}^{x_0}\hat{f}_{\mathrm{ext}}| < \epsilon_{x_0} \\ k_{f,3}\left({}^{x_0}\hat{f}_{\mathrm{ext}} - \mathrm{sign}({}^{x_0}\hat{f}_{\mathrm{ext}})\epsilon_{x_0}\right), & \text{otherwise} \end{cases}, \tag{6.52}$$

where ϵ_{y_0} and ϵ_{x_0} are force thresholds in y_0- and x_0-direction, respectively. The gains for the linear axis and the 3rd LWR joint are denoted by $k_{f,x}$ and $k_{f,3}$.

Experimental Results Control law (6.50) in combination with (6.51) and (6.52) was implemented on the real eight-DOF system. The experimental results are depicted in Fig. 6.15. In the upper figure, a force in y_0-direction induces a linear axis motion while the elbow, represented by the position of the 3rd LWR joint, moves passively. By coincidence, q_3 almost remains constant in this experiment. In the lower figure, the elbow moves when an external force in x_0-direction is exerted, while the linear axis is (almost) at rest. The recorded data shows that the reconfiguration can be performed effectively and intuitively. It is possible to switch from interactive null space control to normal operation seamlessly.

Lightweight Robot mounted on Omnidirectional Platform

Next, a ten-DOF robotic system consisting of a LWR and a three-DOF omnidirectional platform is simulated. The generalized coordinates are $\boldsymbol{q}^\mathsf{T} = [q_x, q_y, q_\alpha, q_1, q_2, q_3, q_4, q_5, q_6, q_7]$, where q_x, q_y are the two translational coordinates of the platform and q_α represents the angle about the vertical axis. The variables q_i,



(a) Pseudoinverse approach **(b)** Proposed approach

Fig. 6.16: Reconfiguration of a mobile platform in y_0-direction with fixed platform rotation: Comparison between Moore-Penrose pseudoinverse projection (a) and the projection proposed in this work (b). © 2017 IEEE [302].

$i = 1, \ldots, 7$ are the positions of LWR. In practice, it is beneficial to change the position and orientation of the mobile platform and the angle of the manipulator's elbow while keeping the end-effector pose. Relevant cases for reconfiguration are

1) translational motion of the platform in Cartesian x_0/y_0-direction with fixed/free platform rotation,

2) rotation of the platform with fixed x_0/y_0-position, and

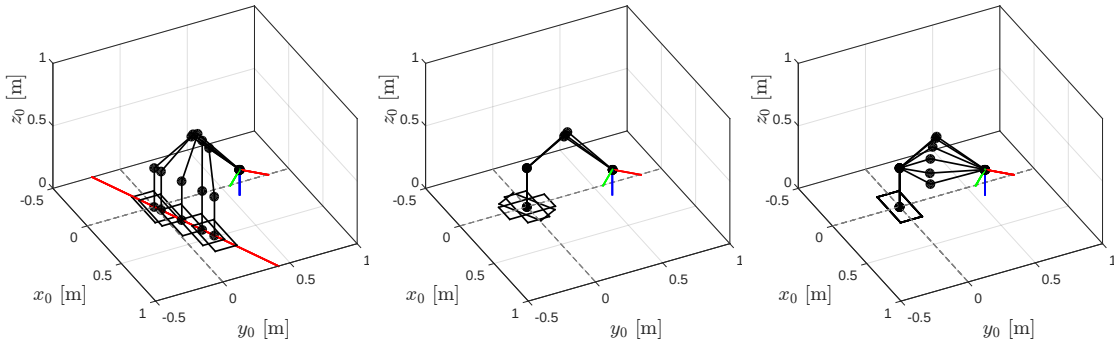
3) rotation of the manipulator elbow with fixed platform position and orientation.

For interactive control, $\boldsymbol{q}_c^\mathsf{T} = [q_x, q_y, q_\alpha, q_3]$ is selected, i.e., the third manipulator joint (robot elbow) and all platform coordinates. The latter can be controlled independently when the target frame is located in the robot's workspace because the seven-DOF manipulator can compensate for the platform motion. For translational platform movements in x_0/y_0-direction, which correspond to motions of q_x/q_y, the angle β between both coordinates is defined. The constraint for the extended Jacobian matrix is

$$\frac{\partial h(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{pmatrix} \sin\beta & -\cos\beta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} . \tag{6.53}$$

For motions in x_0-direction only, $\beta = 0°$ is selected, for movements in y_0-direction $\beta = 90°$. For each case with the corresponding constraints and extended Jacobian matrix, a separate null space projection is defined.

Simulation results are provided in Fig. 6.16 and Fig. 6.17, where cases 1–3 are exemplified. In the first figure, the resulting trajectory using the proposed projection is compared to the Moore-Penrose pseudoinverse solution (6.34). The proposed method respects the constraint, which makes the movement interpretable to the user. Furthermore, a collision with the obstacle located in the robot workspace is avoided; the trajectory obtained by the Moore-Penrose pseudoinverse solution collides with the object, c. f. Sec. 6.3.1. To parameterize the constraints and command desired velocities, both haptic and remote teaching interfaces can be used as described in Sec. 6.3.2. The implementation on a real system like the KUKA omniRob and the extension to non-holonomic systems (cf. Chapter 2) is subject to future work.



(a) Platform translation **(b)** Platform rotation **(c)** Elbow rotation

Fig. 6.17: Further applications of interactive null space control: Motion of the platform in $\beta = 25\,^\circ$ direction with fixed base rotation (left), rotation of the platform only (middle), and rotation of the manipulator's elbow only (right). © 2017 IEEE [302].

6.4 Summary

In this work, the performance and safety in HRI were improved by combining the well-established Safe Motion Unit with a reactive redundancy resolution scheme that simultaneously minimizes the robot reflected mass in the direction of movement. A reduction in reflected mass allows the robot to travel at higher speeds while simultaneously ensuring human biomechanical safety constraints. For the LWR, minima and maxima in reflected mass were determined for static end-effector poses. Then, a real-time capable local minimization scheme was proposed that provides a desired joint torque based on an attractive field spanned between the current joint position and the position associated with a (local) minimum in reflected mass. The combination of the nullspace scheme and SMU was validated experimentally, and the advantages and limitations of the approach were discussed. Furthermore, it was investigated how the trajectory of the robot's primary task can be relaxed via time scaling such that the performance of one or multiple auxiliary null space task(s) in minimizing a performance/safety criterion can be improved. By temporarily limiting the velocity of the main task based on a suitable error in the performance index, the null space tasks are given more time to achieve their objectives. The proposed schemes can be implemented with minimal effort and leave the existing primary and null space controller unchanged. The methods were validated experimentally on an LWR. The concept was extended to the general case of multiple prioritized task control and validated in simulation using a 4R planar robot. It was shown that time scaling is a simple yet effective method to improve the task achievement of the auxiliary tasks at the cost of some extra cycle time. The developed method is an efficient alternative to existing task transition or priority switching schemes, as the geometric description of the main task remains unchanged. However, such schemes may also be combined with time scaling. Finally, null space interaction patterns were developed that are intuitively interpretable to the robot user and straightforward to implement. The well-known extended Jacobian matrix was used to formulate null space projections that incorporate joint-space constraints for interactive guidance of the robot. It was shown how these could be integrated into remote or haptic teaching interfaces, and two practical examples validated the approach.

CHAPTER 7

Conclusion

7.1 Summary

In this thesis, the global safety characteristics of rigid and intrinsically elastic manipulators as well as mobile robots were investigated, which are essential for understanding and minimizing the hazard potential in human-robot interaction. Controllers for realizing safe and efficient motions were developed, validated, and compared in simulations and experiments. The main contributions to state of the art are described in the following, organized by chapter. Table 7.1 provides an overview how the research questions posed in Sec. 1.1 map to the chapters.

Chapter 3 The Safety Map framework was introduced, which serves as a unified representation of global robot safety performance and human injury data. It provides the designer with clear and quantitative information about the robot's safety and can guide the hardware development process. Furthermore, it can be utilized as a cost map for path planning and interaction control. Together with the Safe Motion Unit (SMU) proposed in [12], which locally ensures biomechanically safe velocities via control, a hierarchical tool stack was assembled that

- provides a biomechanics-based safety assessment for robot design, planning, and control,
- can use all available simulation and experimental injury data,
- is data-driven, model-independent, and interpretation-free,
- applicable to arbitrary robots, and
- compatible with existing standards.

The Safety Map was applied to rigid and elastic joint manipulators (KUKA LWR, Unimate PUMA, Franka Emika Panda, and DLR David) and mobile manipulators consisting of a platform and a Franka Emika Panda. This allowed quantitatively comparing the safety

Tab. 7.1: Evaluation of the research questions posed in Sec. 1.1.

	Research question	Chapter	Answered?
Q1	How can entire robot designs be related to available human injury data and safety thresholds to assess the robot's safety characteristics on a global or task-dependent scale?	3	
Q2	How can safe and efficient trajectories be generated for collaborative applications provided the global characterization of a robot's impact dynamics, the shared workspace, and safety thresholds?	3	
Q3	How much operational speed can robots with multiple intrinsically elastic joints achieve compared to rigid joint robots, and how can the speed gain be computed efficiently?	4	
Q4	What are the global safety properties of robots with intrinsically elastic joints, and how do elastic and rigid joint robots compare in terms of safety?	4	
Q5	How can undesired vibrations in robots with intrinsic joint elasticity be suppressed via simple but effective control schemes?	5	
Q6	How can the kinematics of redundant robots be exploited to maximize the task velocity while ensuring safety simultaneously?	6	
Q7	How can the user kinematically reconfigure a redundant robot intuitively and interactively?	6	

performance of different robot types in the same framework and also linking collision safety to other performance metrics, particularly the robot's reachability. The robot representations were related to human biomechanics injury data, which was classified, validated, and processed during a thorough biomechanics literature survey. Examples demonstrated how the Safety Map can be used for safety assessment in particular tasks, how risk reduction measures could be taken, and how safe, time-optimal trajectories could be generated. Furthermore, within the scope of a collaboration with an industrial partner, it was shown how injury data could be generated targeted to the risk assessment and reduction of a real-world application. Several results and insights gained in this thesis were communicated to and acknowledged by robot safety standardization committees. To the best of the author's knowledge, the presented framework is the first global dynamic and exact safety analysis tool for robot manipulators. It may drive the development of safety algorithms and standardization forward and lead to significant changes in how human-friendly robots are designed in the future.

Chapter 4 In order to assess both the safety properties and the maximum performance of a robot, it is necessary to determine the achievable Cartesian velocities. It is well known that elastic joint robots can outperform their rigid counterparts in peak velocity and energy efficiency by systematically exploiting the spring's energy storage and release mechanisms. In Chapter 4, the achievable speed gain was analyzed for gravity-free elastic joint robots while taking the motor velocity and elastic energy bounds into account, the two most

important real-world constraints besides limited motor torque. In the 1-DOF case, the influence of system parameters on maximum speed and the time-optimal trajectory were derived. For multiple-joint systems, a computationally inexpensive but reasonably accurate estimation of the maximum achievable Cartesian endpoint velocities was developed and verified with the optimal control solution and previous ball throwing experiments conducted in [8]. The developed methods give the robot designer valuable insight into how the springs and motors should be dimensioned to achieve the desired dynamic performance, particularly for realizing explosive motions such as throwing.

Chapter 5 One problem concerning intrinsically elastic robots is to excite targeted cyclic and high-speed motions; the other is to dampen these systems if undesired vibrations occur. Elastic joint robots are inherently oscillatory because the physical damping is typically very low. In Chapter 5, existing and new vibration suppression control schemes were described and compared in terms of performance and passivity. The controllers were implemented on DLR David and tested - many of them for the very first time - in two benchmark tests, namely an emergency stop and a ball impact experiment. Impact response metrics were proposed to quantitatively compare the controller performance and the robot's ability to absorb impacts. The proposed methods and results enable researchers and practitioners to select suitable system design parameters and control schemes to effectively suppress undesired vibrations in their robots.

Chapter 6 Three redundancy resolution schemes were developed that improve the system performance. First, a control scheme was proposed that locally minimizes the effective mass in the direction of travel via self-motions. Reducing the reflected mass showed that a velocity reduction imposed by the SMU used as the core safety unit could be avoided. This resulted in fast yet safe trajectories. Second, a time scaling-based controller was proposed that temporally relaxes the main task to improve the task achievement of the auxiliary tasks at the cost of some extra cycle time. It was demonstrated that the developed method is an efficient alternative to existing task transition or priority switching schemes, as the geometric description of the main task and the controller structure remain unchanged. Third, interaction patterns were developed that enable reconfiguring the robot via self-motions. These patterns are intuitively interpretable to the robot user and straightforward to implement. It was shown how these could be integrated into remote and haptic teaching interfaces, and practical examples demonstrated the proposed methods.

7.2 Future Work

As summarized in Tab. 7.1, most of the posed research questions could be answered adequately. Further research is required especially for safe, time-optimal trajectory planning (Q2), since the concept proposed in this thesis was only validated in a basic simulation. Possible future work in this direction and other interesting directions of research are described next.

Safety in pHRI In the context of safety, this work elaborated how the robot kinematic and dynamic properties and the human injury data can be processed towards the Safety Map. Basic examples were provided that showed how safety could be opti-

mized via mechanical design changes and trajectory planning. The next step would be to integrate the Safety Map into the (possibly automatic) robot design process, in which the mechanism and electronics are optimized w.r.t. safety and other performance criteria; cf. [60–62]. Considering the robot deployment in real pHRI applications, future work could address the optimal placement of the robot w.r.t. task requirements, dexterity, performance, and safety criteria by combining the Safety Map, reachability map, and possibly further tools [233, 309]. This thesis showed for a basic robot how the Safety Map could be utilized to generate safe, time-optimal motions. The concept should be pursued in future work and extended to collaborative and tactile robots with six or more joints. The tool would be of great value to the industry because it allows planning safe trajectories based on the safety thresholds given in the standards and provides information about the shortest possible cycle time. The Safety Map is based on the well-established reflected mass model proposed in [181] and the SMU introduced in [12]. In practice, the effective mass is usually lower due to structural elasticities in the robot joints and links. The calculated reflected mass is thus more conservative than the actual effective mass. Future work could consider the replacement of the theoretical effective mass with the one measured in a suitable collision experiment. First results in this direction are reported in [244, 310, 311]. In [12] and this thesis, it was explained that data-driven safety schemes have several conceptual advantages over model-based safety ratings, which cannot fully capture the complex human injury biomechanics. However, model-based approaches technically require no collision data and can be conveniently employed in control schemes, making them appealing to many engineers. Also the current standardization efforts in the ISO/DIS 10218-2 rely on a simplified collision model instead of a more reliable and realistic data-driven approach. In order to exploit the full potential of the proposed data-driven tool stack, a substantial amount of collision data is still to be collected, which requires several years of fundamental biomechanics research. In the author's opinion, this should not be regarded as a weakness of the data-driven approach over model-based approaches but as a basic necessity to making robots truly aware of human safety.

Intrinsically Elastic Joint Robots: Achievable Speed Gain and Generation of Explosive Motions Previous work and this thesis have shown that elastic joint robots can achieve much higher operational speeds than their rigid counterparts. However, it has been challenging to generate suitable excitation trajectories for coordinated explosive motions. The computation is typically done with optimal control methods, which are time-consuming, difficult to parameterize, and particularly for multiple-joint robots, prone to local minima. One approach to generating explosive motions that are accessible in real-time is to use the estimation of the maximum achievable velocity proposed in Chapter 4 either as a constraint or as an initial guess for the optimal control solver to speed up the computation of the trajectories; cf. Sec. 4.2. Then, one could generalize the trajectories with the approach proposed in [175] that encodes sample trajectories into a Dynamic Movement Primitive (DMP) system. The DMP system can reconstruct learned trajectories and extrapolate to other tasks with near-optimal performance. In this endeavor, the speed gain approximation in elastic robots requires verification also for systems with more than three DOF. The second approach would be gaining further insight into the excitation of nonlinear normal modes in elastic joint robots and deducing motion generators/controllers from this knowledge [161,163,164]. Finally, the methods proposed in this work assume that the motors in elastic joint robots can reach desired velocities instantaneously. This means

it is assumed that enough motor torque is always available to compensate the elastic joint torque and accelerate the motor shaft. The optimal control trajectories for the real-world throwing experiments conducted in [8] were generated based on the same assumption and yielded good results; however, an intermediate PD motor position controller was required. Future work could consider the extension of the methods from velocity to torque input, which would resemble the system dynamics more realistically, but also imply higher model complexity.

Intrinsically Elastic Joint Robots: Vibration Suppression and Impact Absorption A decoupling-based control framework for suppressing undesired oscillations was proposed in Chapter 5. By assuming braking is a rather local behavior, the controller linearizes and then decouples the system dynamics and control region such that SISO control is made possible in modal space. The eigenvalue decomposition and the optimization-based control region decoupling are computationally rather costly. A problem to be tackled in future research is to reduce the computational complexity of the optimization algorithm used for the simultaneous decoupling of system dynamics and control region. Similar problems are addressed in computational geometry, e.g., the maximum empty rectangle problem [265, 266]. Another future research direction would be the excitation of the inherently nonlinear oscillation modes instead of linearizing the system along the trajectory [162, 163]. In Chapter 5, impact absorption metrics were proposed and evaluated in two experiments to compare different control schemes in terms of their ability to suppress undesired vibrations. In future research, one could extend the metrics proposed for elastic joint robots and apply them to other robot types. This includes identifying relevant impact scenarios and defining benchmark experiments similar to those proposed in this work. Such an analysis allows to quantitatively assess and optimize the impact shock absorption performance of robot systems. For humanoid robots, which are often prone to damage in case of a fall, some research in this direction has been done already [312, 313].

Null Space Control for Safe and Efficient pHRI The reflected mass minimization scheme proposed in Chapter 6 was so far only tested for one point of interest on the robot structure. Future work should consider multiple ones, possibly located on different links. Furthermore, it should be considered to activate the nullspace scheme only if the SMU would reduce velocity rather than always keeping the controller activated, which can have disadvantages due to the local nature of the method. Typically, the SMU is activated when the human enters the shared workspace. Use is made of the conservative assumption that a collision could occur at any time. While ensuring safety using the SMU as the core unit, performance could be further increased by taking more information about the human and the robot into account, e.g., the current velocity and direction of motion, the human's intention, and the human and robot dynamic properties. For this, additional sensors and models are required [88, 95, 96, 96, 98]. Finally, the SMU is still to be verified in practice by performing collision experiments with a robot and human/animal soft tissue. The time scaling scheme for improving the performance of auxiliary tasks was validated for one redundant DOF experimentally and for two DOF in simulation. Future research could address the stability proof of the method. Work in this direction was recently done in [314]. Furthermore, one could monitor the time lost due to time scaling during task execution. With this information, one can decide when the trajectory should resume to nominal speed to meet the cycle time requirement.

List of Figures

List of Tables

Bibliography

[1] A. Dietrich, J. Vogel, A. Hagengruber, D. Leidner, P. Schmaus, N. Mansfeld, J. Klodmann, T. Bahls, M. Panzirsch, L. Suchenwirth, and A. Albu-Schäffer, "Feinfühlige, interaktive Roboter in Krankenhaus und Pflege: Wo stehen wir und wohin geht die Reise?" *Zeitschrift für medizinische Ethik*, vol. 64, no. 4, pp. 307–325, 2018.

[2] S. Haddadin, L. Johannsmeier, J. Schmid, T. Ende, S. Parusel, S. Haddadin, M. Schappler, T. Lilge, and M. Becker, "Roboterfabrik: A pilot to link and unify german robotics education to match industrial and societal demands," in *International Conference on Robotics and Education*, 2018, pp. 3–17.

[3] N. Hogan, "Impedance control: An approach to manipulation: Part I - theory, Part II - implementation, Part III - applications," *Journal of Dynamic Systems, Measurement and Control*, vol. 107, pp. 1–24, 1985.

[4] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *International Journal of Robotics Research*, vol. 26, no. 1, pp. 23–39, 2007.

[5] B. Vanderborght, A. Albu-Schäffer, A. Bicchi, E. Burdet, D. G. Caldwell, R. Carloni, M. G. Catalano, O. Eiberger, W. Friedl, G. Ganesh, M. Garabini, M. Grebenstein, G. Grioli, S. Haddadin, H. Höppner, A. Jafari, M. Laffranchi, D. Lefeber, F. Petit, S. Stramigioli, N. G. Tsagarakis, M. V. Damme, R. V. Ham, L. C. Visser, and S. Wolf, "Variable impedance actuators: A review," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1601–1614, 2013.

[6] S. Wolf, G. Grioli, O. Eiberger, W. Friedl, M. Grebenstein, H. Höppner, E. Burdet, D. G. Caldwell, R. Carloni, M. G. Catalano, *et al.*, "Variable stiffness actuators: Review on design and components," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2418–2430, 2015.

[7] G. Grioli, S. Wolf, M. Garabini, M. Catalano, E. Burdet, D. Caldwell, R. Carloni, W. Friedl, M. Grebenstein, M. Laffranchi, D. Lefeber, S. Stramigioli, N. Tsagarakis, M. van Damme, B. Vanderborght, A. Albu-Schäffer, and A. Bicchi, "Variable stiffness actuators: The user's point of view," *International Journal of Robotics Research*, vol. 34, no. 6, pp. 727–743, 2015.

[8] S. Haddadin, F. Huber, and A. Albu-Schäffer, "Optimal control for exploiting the natural dynamics of variable stiffness robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 3347–3354.

[9] D. J. Braun, F. Petit, F. Huber, S. Haddadin, P. van der Smagt, A. Albu-Schäffer, and S. Vijayakumar, "Robots driven by compliant actuators: optimal control under actuation constraints," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1085–1101, 2013.

[10] D. Lakatos, F. Petit, and A. Albu-Schäffer, "Nonlinear oscillations for cyclic movements in human and robotic arms," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 865–879, 2014.

[11] C. Della Santina, D. Lakatos, A. Bicchi, and A. Albu-Schaeffer, "Using nonlinear normal modes for execution of efficient cyclic motions in soft robots," *arXiv preprint arXiv:1806.08389*, 2018.

[12] S. Haddadin, S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer, "On making robots understand safety: Embedding injury knowledge into control," *International Journal of Robotics Research*, vol. 31, no. 13, pp. 1578–1602, 2012.

[13] R. Behrens and N. Elkmann, "Study on meaningful and verified thresholds for minimizing the consequences of human-robot collisions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 3378–3383.

[14] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Requirements for safe robots: Measurements, analysis & new insights," *International Journal of Robotics Research*, vol. 28, no. 11-12, pp. 1507–1527, 2009.

[15] S. Haddadin, A. Albu-Schäffer, M. Frommberger, J. Rossmann, and G. Hirzinger, "The "DLR Crash Report": Towards a standard crash-testing protocol for robot safety - part I: Results," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 272–279.

[16] ——, "The "DLR Crash Report": Towards a standard crash-testing protocol for robot safety - part II: Discussions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 280–287.

[17] S. Haddadin, T. Laue, U. Frese, S. Wolf, A. Albu-Schäffer, and G. Hirzinger, "Kick it with elasticity: Safety and performance in human–robot soccer," *Robotics and Autonomous Systems*, vol. 57, no. 8, pp. 761–775, 2009.

[18] S. Haddadin, A. Albu-Schäffer, O. Eiberger, and G. Hirzinger, "New insights concerning intrinsic joint elasticity for safety," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 2181–2187.

[19] A. Nahum, J. Gatts, C. Gadd, and J. Danforth, "Impact tolerance of the skull and face," Society of Automotive Engineers, Tech. Rep., 1968.

[20] A. Nahum and R. Smith, "An experimental model for closed head impact injury," *SAE Paper no. 760825*, 1976.

[21] V. Hodgson, "Tolerance of the facial bones to impact," *American Journal of Anatomy*, vol. 120, no. 1, pp. 113–122, 1967.

[22] V. Hodgson, J. Brinn, L. Thomas, and S. Greenberg, "Fracture behavior of the skull frontal bone against cylindrical surfaces," in *Proceedings of the 14th Stapp Car Crash Conference*, 1970, pp. 341–355.

[23] C. Kroell, D. Schneider, and A. Nahum, "Impact tolerance and response of the human thorax," *SAE Paper no. 710851, Proceedings of the 15th Stapp Car Crash Conference*, pp. 84–134, 1971.

[24] T. Lobdell, C. Kroell, D. Scheider, and W. Hering, "Impact response of the human thorax," *Symposium on Human Impact Response*, pp. 201–245, 1972.

[25] R. L. Stalnaker, J. H. McElhaney, V. L. Roberts, and M. L. Trollope, *Human Torso Response to Blunt Trauma.* Springer, 1973, pp. 181–199.

[26] J. Melvin, R. Snyder, L. Travis, and N. Olson, "Response of human larynx to blunt loading," Tech. Rep., 1973.

[27] J. Melvin, "Human tolerance to impact conditions as related to motor vehicle design," *Society of Automotive Engineers*, 1980.

[28] I. Lau and D. Viano, "Role of impact velocity and chest compression in thoraic injury," *Aviation, space, and environmental medicine*, vol. 54, no. 1, pp. 16–21, 1983.

[29] ——, "The viscous criterion - basis and applications of an injury severity index for soft tissues," *SAE Technical Paper No. 861882, Proceedings of 30th Stapp Car Crash Conference*, pp. 123–142, 1986.

[30] G. W. Nyquist, J. M. Cavanaugh, S. J. Goldberg, and A. I. King, "Facial impact tolerance and response," in *Proceedings of the 30th Stapp Car Crash Conference*, 1986, pp. 733–754.

[31] D. Allsop, T. Perl, and C. Warner, "Force/deflection and fracture characteristics of the temporo-parietal region of the human head," *Society of Automotive Engineers*, pp. 2009–2018, 1991.

[32] S. Backaitis and H. Mertz, *Hybrid III: The first human-like crash test dummy.* Society of Automotive Engineers, 1993.

[33] N. Yoganandan, F. Pintar, A. Sances, G. Harris, K. Chintapalli, J. Myklebust, D. Schmaltz, J. Reinartz, J. Kalbfleisch, and S. Larson, "Steering wheel induced facial trauma," SAE Technical Paper, Tech. Rep., 1988.

[34] K.-U. Schmitt, P. Niederer, M. Muser, and F. Walz, *Trauma biomechanics.* Springer, 2010.

[35] Y. Yamada, Y. Hirasawa, S. Huand, and Y. Umetani, "Fail-safe human/robot contact in the safety space," *IEEE International Workshop on Robot and Human Communication*, pp. 59–64, 1996.

[36] M. Melia, M. Schmidt, B. Geissler, J. König, U. Krahn, H. J. Ottersbach, S. Letzel, and A. Muttray, "Measuring mechanical pain: the refinement and standardization of pressure pain threshold measurements," *Behavior research methods*, vol. 47, no. 1, pp. 216–227, 2015.

[37] S. Haddadin and E. Croft, "Physical human-robot interaction," in *Handbook of Robotics*, 2nd ed., B. Siciliano and O. Khatib, Eds. Springer, 2016, pp. 1835–1874.

[38] AAAM, "The Abbreviated Injury Scale," 1980.

[39] ——, *The Abbreviated Injury Scale (1990) Revision Update 1998*, 1998.

[40] J. Versace, "A review of the severity index," *SAE Paper No.710881, Proceedings of the 15th Stapp Car Crash Conference*, pp. 771–796, 1971.

[41] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Approaching asimov's 1st law: The impact of the robot's weight class," *Robotics: Science and Systems Conference Workshop: Robot Manipulation: Sensing and adapting the real world*, 2007.

[42] ——, "The role of the robot mass and velocity in physical human-robot interaction - part I: Unconstrained blunt impacts," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1331–1338.

[43] ——, "The role of the robot mass and velocity in physical human-robot interaction - part II: Constrained blunt impacts," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1339–1345.

[44] ——, "An experimental safety study for stab/puncture and incised wounds," *IEEE Robotics and Automation Magazine*, vol. 18, pp. 20–34, 2011.

[45] T. Rüedi, T. Ruedi, R. Buckley, and C. Moran, *AO principles of fracture management*. George Thieme Verlag, 2007.

[46] R. Behrens, *Biomechanische Grenzwerte für die sichere Mensch-Roboter Kollaboration*. Springer, 2019.

[47] M. Y. Park, D. Han, J. H. Lim, M. K. Shin, Y. R. Han, D. H. Kim, S. Rhim, and K. S. Kim, "Assessment of pressure pain thresholds in collisions with collaborative robots," *PloS one*, vol. 14, no. 5, 2019.

[48] M. Melia, B. Geissler, J. König, H. J. Ottersbach, M. Umbreit, S. Letzel, and A. Muttray, "Pressure pain thresholds: Subject factors and the meaning of peak pressures," *European Journal of Pain*, vol. 23, no. 1, pp. 167–182, 2019.

[49] A. Albu-Schäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger, "The DLR lightweight robot - lightweight design and soft robotics control concepts for robots in human environments," *Industrial Robot Journal*, vol. 34, no. 5, pp. 376–385, 2007.

[50] A. Bicchi and G. Tonietti, "Fast and soft arm tactics: Dealing with the safety-performance trade-off in robot arms design and control," *IEEE Robotics and Automation Magazine*, vol. 11, no. 2, pp. 22–33, 2004.

[51] J.-J. Park, H.-S. Kim, and J.-B. Song, "Safe robot arm with safe joint mechanism using nonlinear spring system for collision safety," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3371–3376, 2009.

[52] S. Haddadin, K. Krieger, N. Mansfeld, and A. Albu-Schäffer, "On impact decoupling properties of elastic robots and time optimal velocity maximization on joint level," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 5089–5096.

[53] S. Haddadin, A. Albu-Schäffer, O. Eiberger, and G. Hirzinger, "New insights concerning intrinsic joint elasticity for safety," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 2181–2187.

[54] M. Van Damme, P. Beyl, B. Vanderborght, R. Versluys, R. Van Ham, I. Vanderniepen, F. Daerden, and D. Lefeber, "The safety of a robot actuated by pneumatic muscles – a case study," *International Journal of Social Robotics*, vol. 2, no. 3, pp. 289–303, 2010.

[55] T. Lens and O. Von Stryk, "Investigation of safety in human-robot-interaction for a series elastic, tendon-driven robot arm," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 4309–4314.

[56] K. Suita, Y. Yamada, N. Tsuchida, K. Imai, H. Ikeda, and N. Sugimoto, "A failure-to-safety "kyozon" system with simple contact detection and stop capabilities for safe human-autonomous robot coexistence," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1995, pp. 3089–3096.

[57] M. Zinn, O. Khatib, B. Roth, and J. K. Salisbury, "Playing it safe," *IEEE Robotics and Automation Magazine*, vol. 11, no. 2, pp. 12–21, 2004.

[58] J. Park, S. Haddadin, J. Song, and A. Albu-Schäffer, "Designing optimally safe robot surface properties for minimizing the stress characteristics of human-robot collisions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 5413–5420.

[59] R. Weitschat, J. Vogel, S. Lantermann, and H. Höppner, "End-effector airbags to accelerate human-robot collaboration," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 2279–2284.

[60] K. T. Ulrich, T. Tuttle, J. P. Donoghue, and W. T. Townsend, "Intrinsically safer robots," *Barrett Technology, Inc., Project Report*, 1995.

[61] K. Ikuta, H. Ishii, and M. Nokata, "Safety evaluation method of design and control for human-care robots," *International Journal of Robotics Research*, vol. 22, no. 5, pp. 281–298, 2003.

[62] S.-D. Lee, B.-S. Kim, and J.-B. Song, "Human–robot collision model with effective mass and manipulability for design of a spatial manipulator," *Advanced Robotics*, vol. 27, no. 3, pp. 189–198, 2013.

[63] P. Langley and D. Choi, "A unified cognitive architecture for physical agents," in *Proceedings of the National Conference on Artificial Intelligence*, vol. 21, no. 2, 2006, p. 1469.

[64] D. Vernon, G. Metta, and G. Sandini, "A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 2, pp. 151–180, 2007.

[65] M. Beetz, L. Mösenlechner, and M. Tenorth, "CRAM - a cognitive robot abstract machine for everyday manipulation in human environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 1012–1017.

[66] M. Beetz, D. Jain, L. Mosenlechner, M. Tenorth, L. Kunze, N. Blodow, and D. Pangercic, "Cognition-enabled autonomous robot control for the realization of home chore task intelligence," *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2454–2471, 2012.

[67] M. Beetz, G. Bartels, A. Albu-Schäffer, F. Bálint-Benczédi, R. Belder, D. Beßler, S. Haddadin, A. Maldonado, N. Mansfeld, T. Wiedemeyer, R. Weitschat, and J.-H. Worch, "Robotic agents capable of natural and safe physical interaction with human co-workers," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 6528–6535.

[68] D. K. Misra, J. Sung, K. Lee, and A. Saxena, "Tell me Dave: Context-sensitive grounding of natural language to manipulation instructions," *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 281–300, 2016.

[69] S. Guadarrama, L. Riano, D. Golland, D. Go, Y. Jia, D. Klein, P. Abbeel, T. Darrell, *et al.*, "Grounding spatial relations for human-robot interaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 1640–1647.

[70] G. Bartels, I. Kresse, and M. Beetz, "Constraint-based movement representation grounded in geometric features," in *IEEE/RAS International Conference on Humanoid Robots*, 2013, pp. 547–554.

[71] I. Kresse and M. Beetz, "Movement-aware action control - integrating symbolic and control-theoretic action execution," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 3245–3251.

[72] J. Heinzmann and A. Zelinsky, "Quantitative safety guarantees for physical human-robot interaction," *International Journal of Robotics Research*, vol. 22, no. 7-8, pp. 479–504, 2003.

[73] D. Kulic and E. Croft, "Pre-collision strategies for human robot interaction," *Autonomous Robots*, vol. 22, no. 2, pp. 149–164, 2007.

[74] B. Lacevic, P. Rocco, and A. M. Zanchettin, "Safety assessment and control of robotic manipulators using danger field," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1257–1270, 2013.

[75] T. S. Tadele, T. J. de Vries, and S. Stramigioli, "Combining energy and power based safety metrics in controller design for domestic robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 1209–1214.

[76] R. Rossi, M. P. Polverini, A. M. Zanchettin, and P. Rocco, "A pre-collision control strategy for human-robot interaction based on dissipated energy in potential inelastic impacts," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 26–31.

[77] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 882–893, 2015.

[78] M. Ragaglia, A. M. Zanchettin, and P. Rocco, "Trajectory generation algorithm for safe human-robot collaboration based on multiple depth sensor measurements," *Mechatronics*, vol. 55, pp. 267–281, 2018.

[79] I. D. Walker, "Impact configurations and measures for kinematically redundant and multiple armed robot systems," *IEEE Transactions Robotics and Automation*, vol. 10, no. 5, pp. 670–683, 1994.

[80] M. Laffranchi, N. G. Tsagarakis, and D. G. Caldwell, "Safe human robot interaction via energy regulation control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009, pp. 35–41.

[81] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *International Journal of Robotics Research*, vol. 4, no. 3, pp. 109–117, 1985.

[82] A. Dietrich, T. Wimböck, A. Albu-Schäffer, and G. Hirzinger, "Integration of reactive, torque-based self-collision avoidance into a task hierarchy," *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1278–1293, 2012.

[83] A. De Luca and L. Ferrajoli, "Exploiting robot redundancy in collision detection and reaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008, pp. 3299–3305.

[84] I. D. Walker, "The use of kinematic redundancy in reducing impact and contact effects in manipulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1990, pp. 434–439.

[85] E. A. Sisbot, L. F. Marin-Urias, R. Alami, and T. Simeon, "A human aware mobile robot motion planner," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 874–883, 2007.

[86] E. A. Sisbot and R. Alami, "A human-aware manipulation planner," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1045–1057, 2012.

[87] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[88] F. Flacco, T. Kröger, A. De Luca, and O. Khatib, "A depth space approach to human-robot collision avoidance," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 338–345.

[89] O. Brock, O. Khatib, and S. Viji, "Task-consistent obstacle avoidance and motion behavior for mobile manipulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2002, pp. 388–393.

[90] O. Brock and O. Khatib, "Elastic strips: A framework for motion generation in human environments," *International Journal of Robotics Research*, vol. 21, no. 12, pp. 1031–1052, 2002.

[91] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.

[92] A. Dietrich, T. Wimbock, A. Albu-Schaffer, and G. Hirzinger, "Integration of reactive, torque-based self-collision avoidance into a task hierarchy," *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1278–1293, 2012.

[93] M. Becker, T. Lilge, M. A. Müller, and S. Haddadin, "Circular fields and predictive multi-agents for online global trajectory planning," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 2618–2625, 2021.

[94] M. P. Polverini, A. M. Zanchettin, and P. Rocco, "Real-time collision avoidance in human-robot interaction based on kinetostatic safety field," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014, pp. 4136–4141.

[95] A. Pereira and M. Althoff, "Overapproximative human arm occupancy prediction for collision avoidance," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 2, pp. 818–831, 2017.

[96] R. Weitschat, J. Ehrensperger, M. Maier, and H. Aschemann, "Safe and efficient human-robot collaboration part I: estimation of human arm motions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1993–1999.

[97] R. Weitschat and H. Aschemann, "Safe and efficient human–robot collaboration part II: Optimal generalized human-in-the-loop real-time motion generation," *IEEE Robotics and Automation Letters (RA-L)*, vol. 3, no. 4, pp. 3781–3788, 2018.

[98] A. Palleschi, M. Hamad, S. Abdolshah, M. Garabini, S. Haddadin, and L. Pallottino, "Fast and safe trajectory planning: Solving the cobot performance/safety trade-off in human-robot shared environments," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 3, pp. 5445–5452, 2021.

[99] C. Schindlbeck and S. Haddadin, "Unified passivity-based cartesian force/impedance control for rigid and flexible joint robots via task-energy tanks," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 440–447.

[100] E. Magrini, F. Flacco, and A. De Luca, "Control of generalized contact motion and force in physical human-robot interaction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2298–2304.

[101] E. Magrini and A. De Luca, "Hybrid force/velocity control for physical human-robot collaboration tasks," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 857–863.

[102] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[103] A. De Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1011–1016, 2005.

[104] A. De Luca, A. Albu-Schäffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight manipulator arm," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006, pp. 1623–1630.

[105] S. Haddadin, A. Albu-Schäffer, A. D. Luca, and G. Hirzinger, "Collision detection & reaction: A contribution to safe physical human-robot interaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008, pp. 3356–3363.

[106] J. Vorndamme, M. Schappler, and S. Haddadin, "Collision detection, isolation and identification for humanoids," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4754–4761.

[107] T. Tomić, C. Ott, and S. Haddadin, "External wrench estimation, collision detection, and reflex reaction for flying robots," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1467–1482, 2017.

[108] Y. J. Heo, D. Kim, W. Lee, H. Kim, J. Park, and W. K. Chung, "Collision detection for industrial collaborative robots: A deep learning approach," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 2, pp. 740–746, 2019.

[109] G. Garofalo, N. Mansfeld, J. Jankowski, and C. Ott, "Sliding mode momentum observers for estimation of external torques and joint acceleration," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6117–6123.

[110] V. J. Lumelsky and E. Cheung, "Real-time collision avoidance in teleoperated whole-sensitive robot arm manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 1, pp. 194–203, 1993.

[111] M. Strohmayr, "Artificial skin in robotics," Ph.D. dissertation, 2012.

[112] R. S. Dahiya, P. Mittendorfer, M. Valle, G. Cheng, and V. J. Lumelsky, "Directions toward effective utilization of tactile skin: A review," *IEEE Sensors Journal*, vol. 13, no. 11, pp. 4121–4138, 2013.

[113] S. A. B. Birjandi and S. Haddadin, "Model-adaptive high-speed collision detection for serial-chain robot manipulators," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 4, pp. 6544–6551, 2020.

[114] L. Manuelli and R. Tedrake, "Localizing external contact using proprioceptive sensors: The contact particle filter," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 5062–5069.

[115] S. Haddadin, A. Albu-Schäffer, A. De Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical human-robot interaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008, pp. 3356–3363.

[116] S. Golz, C. Osendorfer, and S. Haddadin, "Using tactile sensation for learning contact knowledge: Discriminate collision from physical interaction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3788–3794.

[117] A. D. Luca and L. Ferrajoli, "Exploiting robot redundancy in collision detection and reaction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008, pp. 3299–3305.

[118] G. A. Pratt and M. M. Williamson, "Series elastic actuators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1995, pp. 399–406.

[119] S. A. Migliore, E. A. Brown, and S. P. DeWeerth, "Biologically inspired joint stiffness control," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2005, pp. 4508–4513.

[120] F. Petit, W. Friedl, H. Höppner, and M. Grebenstein, "Analysis and synthesis of the bidirectional antagonistic variable stiffness mechanism," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 2, pp. 684–695, 2014.

[121] R. Van Ham, B. Vanderborght, M. Van Damme, B. Verrelst, and D. Lefeber, "Maccepa, the mechanically adjustable compliance and controllable equilibrium position actuator: Design and implementation in a biped robot," *Robotics and Autonomous Systems*, vol. 55, no. 10, pp. 761–768, 2007.

[122] S. Wolf and G. Hirzinger, "A new variable stiffness design: Matching requirements of the next robot generation," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1741–1746, 2008.

[123] S. Wolf, O. Eiberger, and G. Hirzinger, "The DLR FSJ: Energy based design of a variable stiffness joint," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 5082–5089.

[124] A. Jafari, N. G. Tsagarakis, B. Vanderborght, and D. G. Caldwell, "A novel actuator with adjustable stiffness (AwAS)," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 4201–4206.

[125] S. S. Groothuis, G. Rusticelli, A. Zucchelli, S. Stramigioli, and R. Carloni, "The variable stiffness actuator vsaUT-II: Mechanical design, modeling, and identification," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 2, pp. 589–597, 2013.

[126] S. Sugano, S. Tsuto, and I. Kato, "Force control of the robot finger joint equipped with mechanical compliance adjuster," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1992, pp. 2005–2013.

[127] S. Kawamura, T. Yamamoto, D. Ishida, T. Ogata, Y. Nakayama, O. Tabata, and S. Sugiyama, "Development of passive elements with variable mechanical impedance for wearable robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2002, pp. 248–253.

[128] J. M. Schimmels and D. R. Garces, "The Arched Flexure VSA: A compact variable stiffness actuator with large stiffness range," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 220–225.

[129] P. Tomei, "A simple PD controller for robots with elastic joints," *IEEE Transactions on Automatic Control*, vol. 36, no. 10, pp. 1208–1213, 1991.

[130] R. Ortega, R. Kelly, and A. Loría, "A class of output feedback globally stabilizing controllers for flexible joints robots," *IEEE Transactions Robotics and Automation*, vol. 11, no. 5, pp. 766–770, 1995.

[131] L. Zollo, A. De Luca, and B. Siciliano, "Regulation with on-line gravity compensation for robots with elastic joints," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2004, pp. 2687–2692.

[132] C. Ott, A. Albu-Schaffer, A. Kugi, S. Stamigioli, and G. Hirzinger, "A passivity based cartesian impedance controller for flexible joint robots - part I: Torque feedback and gravity compensation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2004, pp. 2659–2665.

[133] A. Albu-Schäffer, F. Petit, and C. Ott, "Energy shaping control for a class of underactuated euler-lagrange systems," *World Congress of the International Federation of Automatic Control*, vol. 45, no. 22, pp. 567–575, 2012.

[134] A. Albu-Schäffer and G. Hirzinger, "A globally stable state feedback controller for flexible joint robots," *Advanced Robotics*, vol. 15, no. 8, pp. 799–814, 2001.

[135] F. Petit and A. Albu-Schäffer, "State feedback damping control for a multi dof variable stiffness robot arm," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 5561–5567.

[136] I. Sardellitti, G. A. Medrano-Cerda, N. Tsagarakis, A. Jafari, and D. G. Caldwell, "Gain scheduling control for a class of variable stiffness actuators based on lever mechanisms," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 791–798, 2013.

[137] B. Brogliato, R. Ortega, and R. Lozano, "Global tracking controllers for flexible-joint manipulators: a comparative study," *Automatica*, vol. 31, no. 7, pp. 941–956, 1995.

[138] S. Ozgoli and H. Taghirad, "A survey on the control of flexible joint robots," *Asian Journal of Control*, vol. 8, no. 4, pp. 332–344, 2006.

[139] M. W. Spong, "Modeling and control of elastic joint robots," *ASME Journal of Dynamic Systems, Measurements, and Control*, vol. 109, no. 4, pp. 310–319, 1987.

[140] S. Nicosia and P. Tomei, "Design of global tracking controllers for flexible-joint robots," *Journal of Robotic Systems*, vol. 10, no. 6, pp. 835–846, 1993.

[141] A. Loria and R. Ortega, "On tracking control of rigid and flexible joints robots," *International Journal of Applied Mathematics and Computer Science*, vol. 5, no. 2, pp. 101–113, 1995.

[142] J.-J. Slotine and L. Weiping, "Adaptive manipulator control: A case study," *IEEE Transactions on Automatic Control*, vol. 33, no. 11, pp. 995–1003, 1988.

[143] F. Ghorbel, J. Y. Hung, and M. W. Spong, "Adaptive control of flexible-joint manipulators," *IEEE Control Systems Magazine*, vol. 9, no. 7, pp. 9–13, 1989.

[144] A. De Luca and P. Lucibello, "A general algorithm for dynamic feedback linearization of robots with elastic joints," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1998, pp. 504–510.

[145] G. Palli, C. Melchiorri, and A. De Luca, "On the feedback linearization of robots with variable joint stiffness," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1753–1759.

[146] J. H. Oh and J. S. Lee, "Control of flexible joint robot system by backstepping design approach," *Intelligent Automation and Soft Computing*, vol. 5, no. 4, pp. 267–278, 1999.

[147] C. Ott, A. Albu-Schäffer, A. Kugi, and G. Hirzinger, "Decoupling based cartesian impedance control of flexible joint robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2003, pp. 3101–3107.

[148] M. Keppler, D. Lakatos, C. Ott, and A. Albu-Schäffer, "A passivity-based controller for motion tracking and damping assignment for compliantly actuated robots," in *IEEE Conference on Decision and Control (CDC)*, 2016, pp. 1521–1528.

[149] ——, "A passivity-based approach for trajectory tracking and link-side damping of compliantly actuated robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 1079–1086.

[150] ——, "Elastic structure preserving (ESP) control for compliantly actuated robots," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 317–335, 2018.

[151] M. Grebenstein, A. Albu-Schäffer, T. Bahls, M. Chalon, O. Eiberger, W. Friedl, R. Gruber, S. Haddadin, U. Hagn, R. Haslinger, H. Höppner, S. Jörg, M. Nickl, A. Nothelfer, F. Petit, J. Reill, N. Seitz, T. Wimböck, S. Wolf, T. Wüsthoff, and G. Hirzinger, "The DLR hand arm system," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 3175–3182.

[152] M. Keppler, D. Lakatos, C. Ott, and A. Albu-Schäffer, "Elastic structure preserving impedance (ESπ) control for compliantly actuated robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5861–5868.

[153] F. Petit, C. Ott, and A. Albu-Schäffer, "A model-free approach to achieve damping via vibration suppression for intrinsically elastic robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 2176–2182.

[154] M. C. Özparpucu, "Optimal control strategies for maximizing the link velocity of elastic joints with nonlinear impedance," Ph.D. dissertation, ETH Zürich, 2018.

[155] G. Garofalo, J. Englsberger, and C. Ott, "On the regulation of the energy of elastic joint robots: excitation and damping of oscillations," in *American Control Conference (ACC)*, 2015, pp. 4825–4831.

[156] B. Andresen, K. H. Hoffmann, J. Nulton, A. Tsirlin, and P. Salamon, "Optimal control of the parametric oscillator," *European Journal of Physics*, vol. 32, no. 3, p. 827, 2011.

[157] G. Gasparri, M. Garabini, L. Pallottino, L. Malagia, M. Catalano, G. Grioli, and A. Bicchi, "Variable stiffness control for oscillation damping," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 6543–6550.

[158] D. Lakatos, G. Garofalo, F. Petit, C. Ott, and A. Albu-Schäffer, "Modal limit cycle control for variable stiffness actuated robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013, pp. 4934–4941.

[159] G. Garofalo and C. Ott, "Energy based limit cycle control of elastically actuated robots," *IEEE Transactions on Automatic Control*, vol. 62, no. 5, pp. 2490–2497, 2016.

[160] D. Lakatos, W. Friedl, and A. Albu-Schäffer, "Eigenmodes of nonlinear dynamics: Definition, existence, and embodiment into legged robots with elastic elements," *IEEE Robotics and Automation Letters (RA-L)*, vol. 2, no. 2, pp. 1062–1069, 2017.

[161] L. B. Bonacchi, M. A. Roa, A. Sesselmann, F. Loeffl, A. Albu-Schäffer, and C. Della Santina, "Efficient and goal-directed oscillations in articulated soft robots: The point-to-point case," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 2555–2562, 2021.

[162] C. Della Santina and A. Albu-Schäffer, "Exciting efficient oscillations in nonlinear mechanical systems through eigenmanifold stabilization," *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 1916–1921, 2020.

[163] A. Albu-Schäffer and C. Della Santina, "A review on nonlinear modes in conservative mechanical systems," *Annual Reviews in Control*, 2020.

[164] C. Della Santina, D. Calzolari, A. M. Giordano, and A. Albu-Schäffer, "Actuating eigenmanifolds of conservative mechanical systems via bounded or impulsive control actions," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 2783–2790, 2021.

[165] A. Albu-Schäffer, D. Lakatos, and S. Stramigioli, "Strict nonlinear normal modes of systems characterized by scalar functions on riemannian manifolds," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 1910–1917, 2021.

[166] S. Haddadin, M. Weis, S. Wolf, and A. Albu-Schäffer, "Optimal control for maximizing link velocity of robotic variable stiffness joints," *World Congress of the International Federation of Automatic Control*, vol. 44, no. 1, pp. 6863–6871, 2011.

[167] M. Garabini, A. Passaglia, F. Belo, P. Salaris, and A. Bicchi, "Optimality principles in variable stiffness control: The VSA hammer," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 3770–3775.

[168] ——, "Optimality principles in stiffness control: The VSA kick," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 3341–3346.

[169] M. C. Özparpucu and S. Haddadin, "Optimal control for maximizing link velocity of visco-elastic joints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 3035–3042.

[170] M. C. Özparpucu and A. Albu-Schäffer, "Optimal control strategies for maximizing the performance of variable stiffness joints with nonlinear springs," in *IEEE Conference on Decision and Control (CDC)*, 2014, pp. 1409–1416.

[171] M. C. Özparpucu, S. Haddadin, and A. Albu-Schäffer, "Optimal control of variable stiffness actuators with nonlinear springs," *World Congress of the International Federation of Automatic Control*, vol. 47, no. 3, pp. 8487–8495, 2014.

[172] M. C. Özparpucu and S. Haddadin, "Optimal control of elastic joints with variable damping," in *European Control Conference (ECC)*, 2014, pp. 2526–2533.

[173] M. Keppler and A. De Luca, "On time-optimal control of elastic joints with bounded input," in *IEEE Conference on Decision and Control (CDC)*, 2020.

[174] D. Braun, M. Howard, and S. Vijayakumar, "Optimal variable stiffness control: formulation and application to explosive movement tasks," *Autonomous Robots*, vol. 33, no. 3, pp. 237–253, 2012.

[175] R. Weitschat, S. Haddadin, F. Huber, and A. Albu-Schäffer, "Dynamic optimality in real-time: A learning framework for near-optimal robot motions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 5636–5643.

[176] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert, "Learning movement primitives," in *International Symposium of Robotic Research (ISSR)*, 2005, pp. 561–572.

[177] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation.* CRC press, 1994.

[178] R. Featherstone, *Rigid body dynamics algorithms.* Springer, 2014.

[179] M. Grebenstein, M. Chalon, G. Hirzinger, and R. Siegwart, "Antagonistically driven finger design for the anthropomorphic DLR hand arm system," in *IEEE/RAS International Conference on Humanoid Robots*, 2010, pp. 609–616.

[180] A. De Luca and W. Book, "Robots with flexible elements," in *Springer Handbook of Robotics*, 2008, pp. 287–319.

[181] O. Khatib, "Inertial properties in robotic manipulation: an object-level framework," *Int. Journal of Robotics Research*, vol. 14, no. 1, pp. 19–36, 1995.

[182] A. Albu-Schäffer, M. Fischer, G. Schreiber, F. Schoeppe, and G. Hirzinger, "Soft robotics: What cartesian stiffness can we obtain with passively compliant, uncoupled joints?" *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3295–3301, 2004.

[183] Y. Yamamoto and X. Yun, "Coordinating locomotion and manipulation of a mobile manipulator," in *IEEE Conference on Decision and Control (CDC)*, 1992, pp. 2643–2648.

[184] J. Barraquand and J.-C. Latombe, "Nonholonomic multibody mobile robots: Controllability and motion planning in the presence of obstacles," *Algorithmica*, vol. 10, no. 2, pp. 121–155, 1993.

[185] D. Lee and M. W. Spong, "Passive bilateral teleoperation with constant time delay," *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 269–281, 2006.

[186] C. Ott, A. Albu-Schäffer, and G. Hirzinger, "A passivity based cartesian impedance controller for flexible joint robots - Part I: Torque feedback and gravity compensation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2004, pp. 2659–2665.

[187] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A passivity based cartesian impedance controller for flexible joint robots - Part II: Full state feedback, impedance design and experiments," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2004, pp. 2666–2673.

[188] A. Dietrich, *Whole-body impedance control of wheeled humanoid robots.* Springer, 2016.

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

[190] A. Dietrich, K. Bussmann, F. Petit, P. Kotyczka, C. Ott, B. Lohmann, and A. Albu-Schäffer, "Whole-body impedance control of wheeled mobile manipulators," *Autonomous Robots*, vol. 40, no. 3, pp. 505–517, 2016.

[191] E. Colgate and N. Hogan, "An analysis of contact instability in terms of passive physical equivalents," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1989, pp. 404–409.

[192] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control.* Springer Science & Business Media, 2010.

[193] A. Dietrich, C. Ott, and A. Albu-Schäffer, "An overview of null space projections for redundant, torque-controlled robots," *International Journal of Robotics Research*, vol. 34, no. 11, pp. 1385–1400, 2015.

[194] T. Yoshikawa, "Manipulability of robotic mechanisms," *International Journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985.

[195] G. Antonelli, "Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems," *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 985–994, 2009.

[196] B. Siciliano and J.-J. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *International Conference on Robots in Unstructured Environments (ICAR)*, 1991, pp. 1211–1216.

[197] L. Sentis and O. Khatib, "Synthesis of whole-body behaviors through hierarchical control of behavioral primitives," *International Journal of Humanoid Robotics*, vol. 2, no. 4, pp. 505–518, 2005.

[198] J. Baillieul, "Kinematic programming alternatives for redundant manipulators," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1985, pp. 722–728.

[199] P. Chang, "A closed-form solution for inverse kinematics of robot manipulators with redundancy," *IEEE Journal Robotics and Automation*, vol. 3, no. 5, pp. 393–403, 1987.

[200] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions Robotics and Automation*, vol. 13, no. 3, pp. 398–410, 1997.

[201] D. Kirk, *Optimal control theory.* Prentice-Hall, 1970.

[202] A. Bryson and Y. Ho, *Applied optimal control.* Hemisphere Publishing Corporation, 1975.

[203] J. Nocedal and S. Wright, *Numerical optimization.* Springer Science & Business Media, 2006.

[204] D. Bell and D. Jacobson, *Singular optimal control problems.* Elsevier Science, 1975, vol. 117.

[205] L. Pontrjagin, *Mathematische Theorie optimaler Prozesse.* Oldenbourg, 1967.

[206] W. Powers, "On the order of singular optimal control problems," *Journal of Optimization Theory and Applications*, vol. 32, no. 4, pp. 479–489, 1980.

[207] F. Petit, M. Chalon, W. Friedl, M. Grebenstein, A. Albu-Schäffer, and G. Hirzinger, "Bidirectional antagonistic variable stiffness actuation: Analysis, design & implementation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 4189–4196.

[208] N. Mansfeld, M. Hamad, M. Becker, A. G. Marin, and S. Haddadin, "Safety map: A unified representation for biomechanics impact data and robot instantaneous dynamic properties," *IEEE Robotics and Automation Letters (RA-L)*, vol. 3, no. 3, pp. 1880–1887, 2018.

[209] D. Allsop, C. Warner, M. Wille, D. Schneider, and A. Nahum, "Facial impact response – a comparison of the Hybrid III dummy and human cadaver," *SAE Paper No.881719, Proceedings of the 32th Stapp Car Crash Conference*, pp. 781–797, 1988.

[210] Y. Yamada, Y. Hirasawa, S. Huang, Y. Umetani, and K. Suita, "Human-robot contact in the safeguarding space," *IEEE/ASME Transactions on Mechatronics*, vol. 2, no. 4, pp. 230–236, 1997.

[211] F. H. Netter, *Atlas of human anatomy.* Elsevier Health Sciences, 2014.

[212] R. Chandler, C. Clauser, J. McConville, H. Reynolds, and J. Young, "Investigation of inertial properties of the human body," Aerospace Medical Research Laboratory, Tech. Rep. DOT HS-801 430, 1975.

[213] M. Mungiole and P. E. Martin, "Estimating segment inertial properties: comparison of magnetic resonance imaging with existing methods," *Journal of Biomechanics*, vol. 23, no. 10, pp. 1039–1046, 1990.

[214] C. E. Clauser, J. T. McConville, and J. W. Young, "Weight, volume, and center of mass of segments of the human body," DTIC Document, Tech. Rep., 1969.

[215] D. Schneider and A. Nahum, "Impact studies of facial bones and skull," *SAE Paper No.720965, Proceedings of the 16th Stapp Car Crash Conference*, pp. 186–204, 1972.

[216] H. Delye, P. Verschueren, B. Depreitere, I. Verpoest, D. Berckmans, J. Vander Sloten, G. Van Der Perre, and J. Goffin, "Biomechanics of frontal skull fracture," *Journal of Neurotrauma*, vol. 24, no. 10, pp. 1576–1586, 2007.

[217] J. Swearingen and C. A. R. I. (US), *Tolerances of the human face to crash impact.* Civil Aeromedical Research Institute, 1965.

[218] G. W. Nyquist, J. M. Cavanaugh, S. J. Goldberg, and A. I. King, "Facial impact tolerance and response," *SAE Paper No.861896, Proceedings of the 30th Stapp Car Crash Conference*, pp. 733–754, 1986.

[219] V. Hodgson and L. Thomas, "Comparison of head acceleration injury indices in cadaver skull fracture," *SAE Paper No710854, Proceedings of the 15th Stapp Car Crash Conference*, pp. 299–307, 1971.

[220] K. Ono, A. Kikuchi, M. Nakamura, H. Kobayashi, and N. Nakamura, "Human head tolerance to sagittal impact reliable estimation deduced from experimental head injury using subhuman primates and human cadaver skulls," in *Proceedings of the 24th Stapp Car Crash Conference*, 1980.

[221] V. Hodgson and L. Thomas, "Breaking strength of the human skull versus impact surface curvature," Wayne State University, Detroit, Final Report DOT HS-801 002, 1973.

[222] E. Gurdjian, J. E. Webster, and H. R. Lissner, "Studies on skull fracture with particular reference to engineering factors," *The American Journal of Surgery*, vol. 78, no. 5, pp. 736 – 742, 1949.

[223] J. Cormier, S. Manoogian, J. Bisplinghoff, S. Rowson, A. Santago, C. McNally, S. Duma, and J. Bolte Iv, "The tolerance of the nasal bone to blunt impact," in *Annals of Advances in Automotive Medicine/Annual Scientific Conference*, vol. 54, 2010, p. 3.

[224] L. Patrick, C. Kroell, and H. Mertz, "Forces on the human body in simulated crashes," *Proceedings of the 9th Stapp Car Crash Conference*, pp. 237–260, 1965.

[225] L. Patrick, H. Mertz, and C. Kroell, "Cadaver knee, chest, and head impact loads," *SAE Paper no. 670913, Proceedings of the 9th Stapp Car Crash Conference*, pp. 168–182, 1967.

[226] A. M. Nahum, C. W. Gadd, D. C. Schneider, and C. Kroell, "Deflection of the human thorax under sternal impact," in *International Automobile Safety Conference*, 1970.

[227] C. Kroell, D. Scheider, and A. Nahum, "Impact tolerance and response of the human thorax II," *SAE Paper No.741187, Proceedings of the 18th Stapp Car Crash Conference*, pp. 383–457, 1974.

[228] D. Viano and C. Warner, "Thoracic impact response of live porcine subjects," in *Proceedings of the 20th Stapp Car Crash Conference*, 1976, pp. 441–474.

[229] C. K. Kroell, M. E. Pope, D. C. Viano, C. Y. Warner, and S. D. Allen, "Interrelationship of velocity and chest compression in blunt thoracic impact to swine," in *Proceedings of the 25th Stapp Car Crash Conference*, 1981.

[230] C. K. Kroell, S. D. Allen, C. Y. Warner, and T. R. Perl, "Interrelationship of velocity and chest compression in blunt thoracic impact to swine II," in *Proceedings of the 30th Stapp Car Crash Conference*, 1986.

[231] S. M. Kuppa and R. H. Eppinger, "Development of an improved thoracic injury criterion," in *Proceedings of the 42nd Stapp Car Crash Conference*, 1998, pp. 139–154.

[232] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Soft-tissue injury in robotics," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 3426–3433.

[233] F. Zacharias, *Knowledge representations for planning manipulation tasks.* Springer Science & Business Media, 2012.

[234] J. W. Burdick, "On the inverse kinematics of redundant manipulators: Characterization of the self-motion manifolds," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1989, pp. 264–270.

[235] C. L. Luck and S. Lee, "Redundant manipulator self-motion topology under joint limits with an 8-DOF case study," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1993, pp. 848–855.

[236] N. Mansfeld, M. Keppler, and S. Haddadin, "Speed gain in elastic joint robots: An energy conversion-based approach," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 3, pp. 4600–4607, 2021.

[237] T. Yoshikawa, "Translational and rotational manipulability of robotic manipulators," in *International Conference on Industrial Electronics, Control and Instrumentation*, 1991, pp. 1170–1175.

[238] P. Chiacchio, Y. Bouffard-Vercelli, and F. Pierrot, "Force polytope and force ellipsoid for redundant manipulators," *Journal of Robotic Systems*, vol. 14, no. 8, pp. 613–620, 1997.

[239] M. Cyrus and J. Beck, "Generalized two-and three-dimensional clipping," *Computers and Graphics*, vol. 3, no. 1, pp. 23–28, 1978.

[240] P. Corke, *Robotics, vision and control: Fundamental algorithms in MATLAB*. Springer, 2011.

[241] R. Konietschke, *Planning of Workplaces with Multiple Kinematically Redundant Robots*. Technische Universität München, 2008.

[242] M. Hamad, N. Mansfeld, S. Abdolshah, and S. Haddadin, "The role of robot payload in the safety map framework," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 195–200.

[243] O. Khatib, "Object manipulation in a multi-effector robot system," in *ISRR*, R. Bolles and B. Roth, Eds., 1988, pp. 137–144.

[244] R. J. Kirschner, N. Mansfeld, G. G. Peña, S. Abdolshah, and S. Haddadin, "Notion on the correct use of the robot effective mass in the safety context and comments on iso/ts 15066," in *IEEE International Conference on Intelligence and Safety for Robotics (ISR)*, 2021, pp. 6–9.

[245] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[246] R. F. Hartl, S. P. Sethi, and R. G. Vickson, "A survey of the maximum principles for optimal control problems with state constraints," *SIAM review*, vol. 37, no. 2, pp. 181–218, 1995.

[247] N. Mansfeld and S. Haddadin, "Reaching desired states time-optimally from equilibrium and vice versa for visco-elastic joint robots with limited elastic deflection," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[248] M. C. Özparpucu and S. Haddadin, "Optimal control for maximizing link velocity of visco-elastic joints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 3035–3042.

[249] S. Haddadin, M. Weis, S. Wolf, and A. Albu-Schäffer, "Optimal control for maximizing link velocity of robotic variable stiffness joints," in *World Congress of the International Federation of Automatic Control*, 2011, pp. 6863–6871.

[250] S. Haddadin, F. Huber, K. Krieger, R. Weitschat, A. Albu-Schäffer, S. Wolf, W. Friedl, M. Grebenstein, F. Petit, J. Reinecke, and R. Lampariello, "Intrinsically elastic robots: The key to human like performance," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 4270–4271.

[251] M. A. Patterson and A. V. Rao, "GPOPS-II: A MATLAB software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming," *ACM Transactions on Mathematical Software (TOMS)*, vol. 41, no. 1, pp. 1–37, 2014.

[252] D. C. Viano and V.-K. Lau, "Role of impact velocity and chest compression in thoracic injury," *Aviation, space, and environmental medicine*, vol. 54, no. 1, pp. 16–21, 1983.

[253] N. Mansfeld and S. Haddadin, "A comparison of braking strategies for elastic joint robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 789–796.

[254] S. Haddadin, M. C. Özparpucu, and A. Albu-Schäffer, "Optimal control for maximizing potential energy in a variable stiffness joint," in *IEEE Conference on Decision and Control (CDC)*, 2012, pp. 1199–1206.

[255] L. Meirovitch, *Principles and techniques of vibrations.* Prentice Hall, 1997.

[256] T. Kailath, *Linear systems.* Prentice-Hall, 1980.

[257] L. Meirovitch, H. Baruh, and H. Oz, "A comparison of control techniques for large flexible systems," *Journal of Guidance, Control, and Dynamics*, vol. 6, no. 4, pp. 302–310, 1983.

[258] J. Holterman and T. J. de Vries, "Active damping based on decoupled collocated control," *IEEE/ASME Transactions on Mechatronics*, vol. 10, no. 2, pp. 135–145, 2005.

[259] M. Heertjes, K. de Graaff, and J.-G. van der Toorn, "Active vibration isolation of metrology frames; a modal decoupled control design," *Journal of Vibration and Acoustics*, vol. 127, no. 3, pp. 223–233, 2005.

[260] J. De Schutter, D. Torfs, H. Bruyninckx, and S. Dutré, "Invariant hybrid force/position control of a velocity controlled robot with compliant end effector using modal decoupling," *The International Journal of Robotics Research*, vol. 16, no. 3, pp. 340–356, 1997.

[261] M. V. Kothare, V. Balakrishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.

[262] S. Tarbouriech and M. Turner, "Anti-windup design: an overview of some recent advances and open problems," *Control Theory and Applications*, vol. 3, no. 1, pp. 1–19, 2009.

[263] N. Mansfeld, C. Ott, and S. Haddadin, "Maximal input limits for independent siso control in modal space under consideration of actuator constraints," in *American Control Conference (ACC)*, 2016, pp. 6721–6728.

[264] F. Ma, M. Morzfeld, and A. Imam, "The decoupling of damped linear systems in free or forced vibration," *Journal of Sound and Vibration*, vol. 329, no. 15, pp. 3182–3202, 2010.

[265] L.-P. Ku, B. Liu, and W. Hsu, *Discovering Large Empty Maximal Hyper-Rectangle in Multi-Dimensional Space.* National University of Singapore, Department of Information Systems and Computer Science, 1997.

[266] B. Chazelle, R. Drysdale, and D. Lee, "Computing the largest empty rectangle," *SIAM Journal on Computing*, vol. 15, no. 1, pp. 300–315, 1986.

[267] N. Mansfeld and S. Haddadin, "Steuern und/oder regeln von motoren eines roboters," *PCT patent WO 2016/170144 AI*, 2015.

[268] H. Lee and V. I. Utkin, "Chattering suppression methods in sliding mode control systems," *Annual reviews in control*, vol. 31, no. 2, pp. 179–188, 2007.

[269] M. Iskandar and S. Wolf, "Dynamic friction model with thermal and load dependency: modeling, compensation, and external force estimation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7367–7373.

[270] M. G. Catalano, G. Grioli, M. Garabini, F. Bonomo, M. Mancini, N. Tsagarakis, and A. Bicchi, "VSA-CubeBot: A modular variable stiffness platform for multiple degrees of freedom robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 5090–5095.

[271] C. Wampler, "Inverse kinematic functions for redundant manipulators," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, 1987, pp. 610–617.

[272] J. Baillieul, J. Hollerbach, and R. Brockett, "Programming and control of kinematically redundant manipulators," in *IEEE Conference on Decision and Control (CDC)*, 1984, pp. 768–774.

[273] J. Luh and Y. Gu, "Industrial robots with seven joints," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1985, pp. 1010–1015.

[274] A. Liegeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 7, no. 12, pp. 868–871, 1977.

[275] C. Chevallereau and W. Khalil, "A new method for the solution of the inverse kinematics of redundant robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1988, pp. 37–42.

[276] L. Sciavicco and B. Siciliano, "A dynamic solution to the inverse kinematic problem for redundant manipulators," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1987, pp. 1081–1087.

[277] H. Hanafusa and Y. Nakamura, "Control of articulated robots with redundancy," *World Congress of the International Federation of Automatic Control*, vol. 19, no. 14, pp. 15–22, 1986.

[278] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *International Journal of Robotics Research*, vol. 6, no. 2, pp. 3–15, 1987.

[279] M. W. Gertz, J.-O. Kim, and P. K. Khosla, "Exploiting redundancy to reduce impact force," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1991, pp. 179–184.

[280] J. Kim, W. Chung, and Y. Youm, "Normalized impact geometry and performance index for redundant manipulators," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2000, pp. 1714–1719.

[281] N. Mansfeld, B. Djellab, J. Raldúa Veuthey, F. Beck, C. Ott, and S. Haddadin, "Improving the performance of biomechanically safe velocity control for redundant robots through reflected mass minimization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 5390–5397.

[282] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano, "Closed-loop inverse kinematics schemes for constrained redundant manipulators with task space augmentation and task priority strategy," *International Journal of Robotics Research*, vol. 10, no. 4, pp. 410–425, 1991.

[283] F. Flacco, A. D. Luca, and O. Khatib, "Control of redundant robots under hard joint constraints: Saturation in the null space," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 637–654, 2015.

[284] ——, "Prioritized multi-task motion control of redundant robots under hard joint constraints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 3970–3977.

[285] T. Petrič and L. Ẑlajpah, "Smooth continuous transition between tasks on a kinematic control level: Obstacle avoidance as a control problem," *Robotics and Autonomous Systems*, vol. 61, no. 9, pp. 948 – 959, 2013.

[286] H. Han and J. Park, "Robot control near singularity and joint limit using a continuous task transition algorithm," *International Journal of Advanced Robotic Systems*, vol. 10, no. 10, p. 346, 2013.

[287] A. Dietrich, A. Albu-Schäffer, and G. Hirzinger, "On continuous null space projections for torque-based, hierarchical, multi-objective manipulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 2978–2985.

[288] N. Mansfeld, Y. Michel, T. Bruckmann, and S. Haddadin, "Improving the performance of auxiliary null space tasks via time scaling-based relaxation of the primary task," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9342–9348.

[289] V. Padois, J.-Y. Fourquet, and P. Chiron, "Kinematic and dynamic model-based control of wheeled mobile manipulators: A unified framework for reactive approaches," *Robotica*, vol. 25, no. 2, pp. 157–173, 2007.

[290] J. M. Hollerbach, "Dynamic scaling of manipulator trajectories," in *American Control Conference*, 1983, pp. 752–756.

[291] O. Dahl and L. Nielsen, "Torque-limited path following by online trajectory time scaling," *IEEE Transactions Robotics and Automation*, vol. 6, no. 5, pp. 554–561, 1990.

[292] E. Szadeczky-Kardoss and B. Kiss, "Tracking error based on-line trajectory time scaling," in *IEEE International Conference on Intelligent Engineering Systems*, 2006, pp. 80–85.

[293] C. Wah Wu and L. Chua, "A unified framework for synchronization and control of dynamical systems," *International Journal of Bifurcation and Chaos*, vol. 4, no. 4, pp. 979–998, 1994.

[294] I. Blekhman, A. Fradkov, H. Nijmeijer, and A. Pogromsky, "On self-synchronization and controlled synchronization," *Systems And Control Letters*, vol. 31, no. 5, pp. 299 – 305, 1997.

[295] C. Ott, A. Kugi, and Y. Nakamura, "Resolving the problem of non-integrability of nullspace velocities for compliance control of redundant manipulators by using semi-definite lyapunov functions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1999–2004.

[296] J. Park, W. Chung, and Y. Youm, "On dynamical decoupling of kinematically redundant manipulators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1999, pp. 1495–1500.

[297] M. Z. Huang and H. Varma, "Optimal rate allocation in kinematically-redundant manipulators-the dual projection method," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1991, pp. 702–707.

[298] B. Nemec, L. Zlajpah, and D. Omrcen, "Stability of null-space control algorithms," in *12th International Workshop on Robotics in Alpe-Adria-Danube Region Cassino*, 2002.

[299] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1006–1028, 2014.

[300] A. Dietrich, C. Ott, and A. Albu-Schäffer, "Multi-objective compliance control of redundant manipulators: Hierarchy, control, and stability," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 3043–3050.

[301] J. Park, W. Chung, and Y. Youm, "Unified motion specification and control of kinematically redundant manipulators," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2000, pp. 3945–3951.

[302] N. Mansfeld, F. Beck, A. Dietrich, and S. Haddadin, "Interactive, intuitively interpretable null space motions for redundant mobile manipulators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 5368–5375.

[303] G. Schreiber and G. Hirzinger, "An intuitive interface for nullspace teaching of redundant robots," in *Advances in Robot Kinematics*. Springer, 2000, pp. 209–216.

[304] G. Grunwald, G. Schreiber, A. Albu-Schäffer, and G. Hirzinger, "Programming by touch: The different way of human-robot interaction," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 4, pp. 659–666, 2003.

[305] A. Dietrich, *Whole-Body Impedance Control of Wheeled Humanoid Robots*, ser. Springer Tracts in Advanced Robotics. Springer International Publishing, 2016.

[306] S. Haddadin, M. Suppa, S. Fuchs, T. Bodenmüller, A. Albu-Schäffer, and G. Hirzinger, "Towards the robotic co-worker," in *Robotics Research*. Springer, 2011, pp. 261–282.

[307] T. Ende, S. Haddadin, S. Parusel, T. Wüsthoff, M. Hassenzahl, and A. Albu-Schäffer, "A human-centered approach to robot gesture based communication within collaborative working processes," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 3367–3374.

[308] S. Haddadin, "Towards safe robots: approaching asimov's 1st law," Ph.D. dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2011.

[309] N. Vahrenkamp, T. Asfour, and R. Dillmann, "Robot placement based on reachability inversion," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013, pp. 1970–1975.

[310] P. Svarny, J. Rozlivek, L. Rustler, and M. Hoffmann, "3D collision-force-map for safe human-robot collaboration," *arXiv preprint arXiv:2009.01036*, 2020.

[311] S. Herbster, R. Behrens, and N. Elkmann, "A new approach to estimate the apparent mass of collaborative robot manipulators," in *Experimental Robotics*, B. Siciliano, C. Laschi, and O. Khatib, Eds. Springer International Publishing, 2021, pp. 211–221.

[312] K. Fujiwara, F. Kanehiro, S. Kajita, K. Kaneko, K. Yokoi, and H. Hirukawa, "Ukemi: Falling motion control to minimize damage to biped humanoid robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2002, pp. 2521–2526.

[313] X. Chen, W. Ding, Z. Yu, L. Meng, M. Ceccarelli, and Q. Huang, "Combination of hardware and control to reduce humanoids fall damage," *International Journal of Humanoid Robotics*, vol. 17, no. 1, pp. 2 050 002.1–2 050 002.23, 2020.

[314] A. Dietrich and C. Ott, "Hierarchical impedance-based tracking control of kinematically redundant robots," *IEEE Transactions on Robotics*, vol. 36, no. 1, pp. 204–221, 2019.