Lehrstuhl für Steuerungs- und Regelungstechnik
Technische Universität München
Prof. Dr.-Ing./Univ. Tokio Martin Buss

Development and Control of an Anthropomorphic Telerobotic System

Bartłomiej Stańczyk

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr.-Ing. Eckehard Steinbach

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing./Univ. Tokio Martin Buss
2. Univ.-Prof. Dr.-Ing. habil. Heinz Ulbrich

Die Dissertation wurde am 21.02.2006 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 17.07.2006 angenommen.

Foreword

This thesis is a result of four years of work in the research group of my thesis adviser Prof. Martin Buss; at first at the Technische Universität Berlin, then at the Technische Universität München. I would like to thank him for giving me the opportunity to conduct exciting research in one of the most honorable robotics and control groups in Germany, for providing me with great hardware and a wonderful research environment.

During my time spent with the group, many people gave me support and advice in numerous ways. I would like to thank Dr. Konstantin Kondak from TU Berlin for leading my first steps in Robotics. His help was invaluable during the design phase. Designing and building of the 7 DoF manipulator would not have been possible without Mr. Franz Bachmann (TU Berlin), who not only put the design on paper, but personally crafted all the aluminum elements. I also would like to thank Mr. Josef Gradl, Mr. Horst Kubick and Mr. Thomas Lowitz at the TU München for additional developments. Many thanks for your professionalism, creativity and enthusiasm.

I would like to express my gratitude to all the colleagues at the Lehrstuhl für Steuerungs- und Regelungstechnik, especially Sandra Hirche, Angelika Peer and Ulrich Unterhinninghofen, with whom I cooperated closely on the SFB453 Project: for the scientific input, advice, help and for the great working atmosphere. Special thanks go to Hasan Esen, who always supported and encouraged me as a friend.

I am also indebted to my students: Kwang-Kyu Lee, Hongquing Hu, Jiduo Wu, Klaas Klasing and Martin Ernst. Thank you very much for the contributions you brought to the project.

Finally, it is necessary to express my gratitude to my parents, my sister and my wife. They were continuously supporting me with their love and patience, especially when I needed it most.

Munich, October 2005. Bartłomiej Stańczyk

to my family
...

Abstract

This thesis presents and summarizes the research work carried out during the development and experimental evaluation of a dual arm anthropomorphic manipulator designed for teleoperation purposes. The results are generalizable to other types of manipulators, especially for robots acting in human environments and/or *physically* interacting with humans.

Such applications pose specific requirements that are in general different from those of traditional robot design. To assure safety and stable operation, "soft" robotic manipulators are absolutely essential. For this reason, in this work, compliance control methods were extensively studied and evaluated experimentally. As a result, an impedance control strategy with underlying stiffness controller in motion control loop is proposed. To achieve proper workspace matching between the human arm and the developed manipulator, an adequate singularity handling strategy must be applied. Experimental comparison of such solutions is performed and an innovative damped inverse kinematics method is introduced. This method allows for traversing through singularities with simultaneous weighing of the task space tracking error. Because the kinematic structures of the master/slave manipulators in a teleoperation system are in general dissimilar, there is a need for a universal kinematic interface, independent of the device structure. The classical approaches using Euler or Cardanian angles fail due to algebraic singularities introduced by the representation. To avoid this problem, the unit quaternion was used for describing orientation and corresponding rotational displacement. In order to avoid the collisions between the arms and the workspace limits a virtual forces concept was introduced. This application forms a basis for local optimization strategies and an intuitive force display for the human operator. Finally, the force - position teleoperation control architecture was analyzed and a tele-assembly experiment in 6 DoF was successfully performed. The experimental results confirm the high performance of the developed hardware and control strategies.

Kurzfassung

In der vorliegenden Dissertationsschrift werden die Forschungsarbeiten, die während der Entwicklung und experimentellen Bewertung eines speziell für Telepräsenzanwendungen konzipierten, zweiarmigen, anthropomorph gestalteten Manipulators durchgeführt wurden, vorgestellt und zusammengefasst. Die Ergebnisse lassen sich auf andere Arten von Manipulatoren übertragen, insbesondere auf jene, die in für Menschen gestalteten Umgebungen eingesetzt werden oder in direkten physikalischen Kontakt mit menschlichen Personen treten. Die besonderen Anforderungen der genannten Anwendungsgebiete unterscheiden sich grundsätzlich von denen klassischer Robotikanwendungen. Um Sicherheit und Stabilität zu gewährleisten, sind "nachgiebige" Roboterarme unabdingbar. Aus diesem Grund wurden Methoden nachgiebiger Positionsregelungen ausgiebig untersucht und experimentell bewertet. Als Ergebnis wird eine Impedanzregelungsstrategie mit unterlagerter Steifigkeitsregelung vorgeschlagen. Um die Arbeitsbereiche des menschlichen Armes und des entwickelten Manipulator aufeinander abzustimmen, muss eine angemessene Strategie zur Behandlung von Singularitäten angewandt werden. Ein experimenteller Vergleich derartiger Lösungen wird durchgeführt, und eine besonders attraktive Methode

zur gedämpften inversen Kinematik wird vorgestellt, welche es erlaubt, Singularitäten zu durchlaufen und dabei den kartesischen Regelfehler im Arbeitsraum abzuschätzen. Da die kinematischen Strukturen auf Operator- und Teleoperatorseite eines Teleoperationssystems üblicherweise verschieden sind, bedarf es einer allgemein verwendbaren, geräteunabhängigen Schnittstelle. Klassische, auf Euler- oder Kardan-Winkeln basierende Ansätze sind aufgrund der möglichen algebraischen Singularitäten ungeeignet. Um dieses Problem zu umgehen, werden Einheitsquaternionen zur Beschreibung von Orientierung und Orientierungsfehler verwendet. Zur Vermeidung von Kollisionen zwischen den beiden Armen und den Arbeitsraumgrenzen wird ein auf virtuellen Kräften beruhendes Konzept vorgestellt, welches als Grundlage für lokale Optimierungsstrategien und eine intuitive Kraftrückkopplung an den menschlichen Operator dient. Schließlich wird die Kraft-Positions-Teleoperationsstruktur untersucht und die erfolgreiche Durchführung eines Experiments zum telepräsenten Fügen gezeigt. Die experimentellen Ergebnisse besttigen die hohe Performanz der entwickelten Manipulatoren und Steuerungsmethoden.

Contents

1 Introduction **1**
- 1.1 Teleoperation: Definition and Motivation . . . 2
- 1.2 Requirements for the Telemanipulator Development . . . 5
- 1.3 Main Contributions and Outline of the Dissertation . . . 6

2 Telemanipulator Kinematics **9**
- 2.1 Telemanipulator Kinematic Design . . . 10
  - 2.1.1 Forward Kinematics for a 7 DoF Manipulator . . . 10
  - 2.1.2 Definition of the Elbow Angle: Self Motion Parametrization . . . 12
- 2.2 Closed Form Inverse Kinematics Algorithm for 7 DoF Anthropomorphic Arm 13
  - 2.2.1 Position-Based Inverse Kinematics Algorithm . . . 13
  - 2.2.2 Velocity-Based Inverse Kinematics for a Spherical Wrist . . . 17
- 2.3 Singularity-Robust Inverse Kinematics . . . 18
  - 2.3.1 Jacobian Methods . . . 18
  - 2.3.2 Singularity Robustness: State of the Art . . . 19
  - 2.3.3 The Least Square (Pseudoinverse) Method . . . 20
  - 2.3.4 Damped Least Square Method . . . 20
  - 2.3.5 Weighted DLS Method with Task Priority . . . 21
  - 2.3.6 Adjoint Jacobian Method . . . 22
  - 2.3.7 Jacobian Transpose Method . . . 23
- 2.4 Experimental Evaluation of the WDLS Method . . . 24
- 2.5 Redundancy Resolution . . . 27
  - 2.5.1 Pseudoinverse with Optimization Criteria . . . 28
  - 2.5.2 Extended Kinematics Method . . . 29
- 2.6 Summary . . . 30

3 Task Space Definition and Orientation Error **31**
- 3.1 Problem of Dissimilar Kinematics in Teleoperation . . . 31
- 3.2 Orientation Representation and Orientation Error . . . 32
  - 3.2.1 Euler Angles . . . 32
  - 3.2.2 Orientation Error with Nonminimal Representation . . . 35
  - 3.2.3 Euler Angles vs. Unit Quaternion: a Simulation Example . . . 37
- 3.3 Extended Task Space Formulation . . . 38
- 3.4 Summary . . . 39

4 Manipulator Control **41**
- 4.1 Manipulator Dynamic Modeling . . . 42
- 4.2 Compliant Control Methods: State of the Art . . . 42
  - 4.2.1 Stiffness Control . . . 44

4.2.2 Impedance Control in Task Space . . . 48
4.2.3 Mixed Stiffness-Impedance Approach . . . 50
4.3 Experimental Comparison of the Control Algorithms . . . 50
4.3.1 Criteria for Evaluation . . . 51
4.3.2 Trajectory Tracking . . . 52
4.3.3 Impedance Display Fidelity . . . 59
4.4 Summary . . . 63

5 Telemanipulation Control Loop 65
5.1 Telemanipulation Control Architectures . . . 65
5.1.1 Two-port Model of the Bilateral Teleoperation System . . . 66
5.1.2 Four Channel Bilateral Control Architecture . . . 66
5.1.3 Stability and Performance Analysis of a Two-port System . . . 68
5.1.4 Two Channel Bilateral Control Architectures . . . 69
5.2 Analysis of the Teleoperation Control Loop . . . 70
5.3 Telerobotic Response Requirement . . . 73
5.4 Controller Tuning . . . 74
5.5 Teleoperation Experiment . . . 77
5.6 Summary . . . 78

6 Collision Avoidance with Virtual Forces in Dual Arm Teleoperation 81
6.1 Collisions in Dual Arm Configuration . . . 82
6.2 Robot Modeling and Collision Detection . . . 84
6.3 Virtual Forces Concept . . . 84
6.4 Collision Avoidance with Redundancy Utilization . . . 85
6.5 Experimental Results . . . 86
6.6 Summary . . . 89

7 Conclusions and Future Work 91
7.1 Concluding Remarks . . . 91
7.2 Future Work . . . 92

A Appendix 95
A.1 Forward Kinematics for the 7 DoF Arm . . . 95
A.2 Analysis of the Elbow Motion . . . 97
A.3 Hardware and Implementation Details . . . 99
A.4 Mass Properties of the Manipulator Links . . . 100

Notations

Abbreviations

DoF	Degree of Freedom
FK	Forward Kinematics
IK	Inverse Kinematics
LS	Least Square
DLS	Damped Least Square
WDLS	Weighted DLS
SJA	Stiffness control with inverse dynamics in joint space
SJB	Modified stiffness control with inverse dynamics
STT	Stiffness control in the task space
RAC	Resolved acceleration control
PDJ	Joint space PD control
TO	Teleoperator

Conventions

Throughout this thesis the term "force" stands for both linear force and torque, while "motion" in terms of a generalization of position, velocity and acceleration refers to both translational and angular motion quantities.

Scalars, Vectors, and Matrices

Scalars are italicized in both upper and lower cases. *Vectors* are denoted by lower case letters in boldface style, e.g., the vector $\boldsymbol{x}$ is composed of elements x_i. *Matrices* are denoted by upper case letters in boldface type, e.g., the matrix $\boldsymbol{M}$ is composed of elements M_{ij} (i-th row, j-th column).

x	scalar
$\boldsymbol{x}$	vector
$\boldsymbol{X}$	matrix
$\dot{\boldsymbol{x}}$, $\ddot{\boldsymbol{x}}$	are equivalent to $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{x}$ and $\frac{\mathrm{d}^2}{\mathrm{d}t^2}\boldsymbol{x}$

One exception is reserved for the Cartesian wrench $\boldsymbol{F}$, which is a vector written in capital for consistence with the literature, e.g., [84].

Simplified Trigonometric Notation

s_α	$\sin(\alpha)$
c_α	$\cos(\alpha)$
s_i	$\sin(q_i)$
c_i	$\cos(q_i)$

Subscripts and Superscripts

x_a	apparent value of x
x_d	desired value of x
x_e	value x associated with the environment
x_f	value x associated with the force
x_E	value x associated with the extended kinematics formulation
x_h	value x associated with the human operator
x_j	value x associated with joint coordinates
x_k	value x associated with Cartesian coordinates
x_l	value x associated with the left arm
x_m	value x associated with the master manipulator
x_o	value x associated with rotational coordinates
x_p	value x associated with translational coordinates
x_r	value x associated with the right arm
x_s	value x associated with the slave manipulator
x_θ	value x associated with the elbow coordinate
f_x, f_y, f_z	components of vector $\boldsymbol{f}$ in $x-, y-, z-$direction

Symbols and Abbreviations

$\boldsymbol{D}_j$	damping matrix in joint coordinates
$\boldsymbol{D}_k$	damping matrix in Cartesian coordinates
$\boldsymbol{e}$	position of the elbow
$\boldsymbol{f}$	force
$\boldsymbol{F}$	Cartesian wrench
$\boldsymbol{g}$	gravity vector
G	transfer function
$\boldsymbol{H}$	Hybrid matrix of a two-port
$\boldsymbol{I}_k$	identity matrix of the size k
$\boldsymbol{J}$	Jacobian matrix
$\boldsymbol{K}$	stiffness matrix
l	link length
m	dimension of the task space
$\boldsymbol{M}$	mass (inertia) matrix
n	dimension of the joint space
$\boldsymbol{p}$	position vector
$\boldsymbol{q}$	vector of joint coordinates
$\mathcal{Q}$	unit quaternion
r	degree of redundancy
$\boldsymbol{R}$	rotation matrix
s	independent variable in Laplace domain
$\boldsymbol{s}$	position of the shoulder
$\boldsymbol{S}(\cdot)$	skew symmetric matrix operator
$\boldsymbol{t}$	position of the tool center point (end-effector)
t	time, independent variable in time domain
$\boldsymbol{T}$	homogeneous transformation matrix
$\boldsymbol{w}$	position of the wrist
$\boldsymbol{W}_q$	joint space weighing matrix
$\boldsymbol{W}_x$	task space weighing matrix
$\boldsymbol{x}$	position
$\dot{\boldsymbol{x}}$	velocity
$\ddot{\boldsymbol{x}}$	acceleration
Z	mechanical impedance
γ	damping constant in DLS
$\boldsymbol{\epsilon}$	vector part of the unit quaternion
ζ_K	stiffness fidelity
η	scalar part of the unit quaternion
μ	Cartesian torque
σ	matrix singular value
$\boldsymbol{\tau}$	joint space torque
$\boldsymbol{\varphi}$	vector of Euler angles
$\boldsymbol{\omega}$	angular velocity

List of Figures

1.1 Components of a multimodal telepresence system . . . 2
1.2 Vanguard, Eodperformance Co. . . . 3
1.3 EXPERT, PIAP . . . 3
1.4 thEODor, Telerob GmbH . . . 3
1.5 Enryu, tmsuk co., ltd . . . 4
1.6 ARMSII . . . 4

2.1 Design of the dual arm telerobotic manipulator . . . 10
2.2 Kinematic structure and link coordinate systems of the 7 DoF arm . . . 11
2.3 Photograph of the physical arm construction . . . 11
2.4 The elbow circle definition . . . 13
2.5 The 7 DoF arm at θ=0 . . . 14
2.6 Kinematic structure of a spherical wrist . . . 17
2.7 Joint angles for LS and WDLS methods . . . 25
2.8 Joint velocities for LS and WDLS methods . . . 25
2.9 Error Comparison between LS and WDLS algorithms . . . 26
2.10 Weighing properties of the WDLS method . . . 27

3.1 Orientation error . . . 35
3.2 Orientation error vs. angular displacement . . . 36
3.3 Euler angles vs. unit quaternions in impedance control . . . 38

4.1 Stiffness control in the task space (STT) . . . 45
4.2 Stiffness control with inverse dynamics in the task space (RAC) . . . 46
4.3 Stiffness control with inverse dynamics in the joint space (SJA) . . . 47
4.4 Stiffness control with modified inverse dynamics in the joint space (SJB) . . . 48
4.5 Impedance control strategy . . . 48
4.6 Block scheme of translational impedance equation . . . 49
4.7 Block scheme of rotational impedance equation . . . 50
4.8 Desired pulse trajectory with its derivatives . . . 52
4.9 Trajectory recorded with SJA . . . 53
4.10 Norm of the tracking errors in the joint and in the task space for SJA . . . 53
4.11 Trajectory recorded with SJB . . . 54
4.12 Norm of the tracking errors in the joint and in the task space for SJB . . . 54
4.13 Trajectory tracking with STT . . . 55
4.14 Norm of the tracking errors in the joint and in the task space for STT . . . 55
4.15 Trajectory tracking with RAC . . . 56
4.16 Norm of the tracking errors in the joint and in the task space for RAC . . . 56
4.17 Trajectory recorded with IMP-PDJ . . . 57
4.18 Norm of the tracking errors in the joint and in the task space for IMP-PDJ 57
4.19 Applied force vs. displacement for stiffness controllers . . . 60

4.20 Applied force vs. displacement for impedance controller 60
4.21 Applied force vs. translational displacement, simulation 61
4.22 Stiffness curves in 6 DoF for impedance controller 62

5.1 General two-port network model 66
5.2 General bilateral control architecture of Lawrence 67
5.3 Simulation model of the present teleoperation system 70
5.4 Simulation model of the present teleoperation system 71
5.5 Root locus of the closed loop teleoperation system 75
5.6 Experimental system architecture 78
5.7 Teleoperation experimental setup 79
5.8 Position tracking (Slave) during haptic exploration experiment 80
5.9 Force tracking (Master) during haptic exploration experiment 80
5.10 Position tracking (Slave) during screw tightening experiment 80
5.11 Force tracking (Master) during screw tightening experiment 80

6.1 Types of collisions in a dual arm system 83
6.2 Potential feld as “seen” by the left arm 85
6.3 Virtual forces and distances 87
6.4 Optimization of a multiple objective functions 88

A.1 Dual anthropomorphic system 99
A.2 Photograph of the elbow joint 99

List of Tables

2.1 Teleoperator Denavit-Hartenberg parameters 11

4.1 Control gains for the trajectory tracking 52
4.2 Maximum steady state error for different controllers 58
4.3 Mean tracking error 58
4.4 Apparent stiffness K_a and stiffness fidelity ζ_K 60
4.5 6 DoF impedance properties of the impedance controller 62
4.6 Impedance properties of joint space PD control 63
4.7 Comparison of all the candidate control schemes 63

5.1 The optimal Z_s for different quality functions 77

A.1 Hardware details of the 7 DoF arm 99

1 Introduction

Since more than two decades, various kinds of robots have been leaving the factory environments to be more and more present in human environments. Scientists envision introducing the robotic devices into almost every aspect of human life. We can find them already in medicine, science, military, entertainment and many other fields. However, it is necessary to stress the main difference between "industrial" and "human" robotics. The industrial robots work in a strictly structured world, where both the environment and the task can be measured, modeled, and planned prior to the task execution. Robots are required to fulfill their assignments with high precision, speed and endurance. Human and natural environments are, on the contrary, unstructured, difficult to model, and in several situations even hazardous. So the expectations for the robots in such cases are generally different.

Human beings have a phenomenal capability to manipulate objects using their arms, hands and fingers. Even the whole body is used to interact with the environment. Human manipulation apparatus consists of complex sensorimotor mechanisms utilizing visual and tactile information leading to the versatility level unmatched by technical systems even today. Making use of those extraordinary abilities, humans can precisely position objects, modulate grasp forces, detect objects and fine surface features. Consider how easily we can peel a breakfast egg, switch on a light, drive a screw with a screwdriver or play a musical instrument. We are also capable of dealing with uncertainties and adapting to changes in the environment, whereas robots have limited adaptation possibilities.

On the other hand, there are situations into which robots fit better than their human counterparts. Consider factories with robots assembling parts at greater speed, accuracy, and endurance than any human worker could achieve. Robots can be exploited in environments that are difficult to access (e.g., underwater or in space), hazardous for humans (e.g., nuclear sites), or not to a human scale (e.g., microsurgery). However, autonomous robot operation makes sense only in the case of factory automation or structured and known environments. Programming robotic systems to autonomously execute tasks in unstructured environments (known as *autonomous manipulation*) is extremely difficult. Today it is not yet possible to program machines for every unpredictable situation, and the robotic brains are still far from perfect.

An interesting alternative to *autonomous manipulation* is *telemanipulation*, in which human intelligence complements the advantages of distance/scaling/automatization offered by robotic systems. Telepresence can provide humans with the ability to see, touch, hear and feel remote environments that are hostile, dangerous, or difficult to access. The presence of the human "in the loop" reduces the complexity as compared to a purely autonomous system and the system benefits from natural human abilities of multi-sensory data fusion, logic reasoning, task planning, execution and adaptation. Telemanipulation is the part of telepresence, concentrating on the physical interaction with the environment.

1.1 Teleoperation: Definition and Motivation

In situations such as earthquakes, explosions, and road accidents, it is not only the catastrophe victims who are endangered, but also the rescue teams are exposed to extremely high risks. In such a critical situation technology can be very helpful. Robotic systems can be efficient tools to speed up search and rescue operations. They can explore the environment using various types of sensors, search for victims, report their location and plan a route to enable rescue teams to reach them. Eventually, if they had a possibility to interact physically with the environment, one could foresee robots totally replacing the human rescuers.

Relatively much has been done in the field of robot mobility in unstructured environments. Multi - legged walking robots and tracked vehicles are commonly used as the transporting platform [30, 39, 81]. The existing rescuing robots make also extensive use of communication and multimodal environment recognition. They also include various levels of "intelligence", autonomy and interaction with a human operator. However, they very seldom have the possibility of real physical interaction with the environment, whereas the need for doing physical work in the remote environment is reported by the practitioners [38, 81, 105, 123]. A robot capable of environment manipulation could be used for clearing and securing the way before the human rescuers enter the dangerous area. It could transport objects, remove or destroy debris and other obstacles, and build mechanical constructions to support the walls, ceilings or other objects from falling down. It is clear that those tasks require human intelligence; this is why instead of fully autonomous robots the technology of telepresence and teleoperation (TO) will be the key in rescue applications [31, 38, 81].

The basic components of a multimodal telepresence system are shown in Fig.1.1. In teleoperation scenarios tasks are performed by a mechanical manipulator, referred to as "telerobot" or "slave" manipulator. This robot is controlled remotely by a human operator provided with an arm/hand controller with force feedback, referred to as "master" or "haptic interface". Such an interface is technically also a robot equipped with a handle or a force reflecting glove. The operator commands are collected using the master device, and sent over the communication medium (usually Internet) to the telerobot working at the remote site. The sensory information, namely visual, audio and force signals are fed back to the operator site and displayed to the human operator.

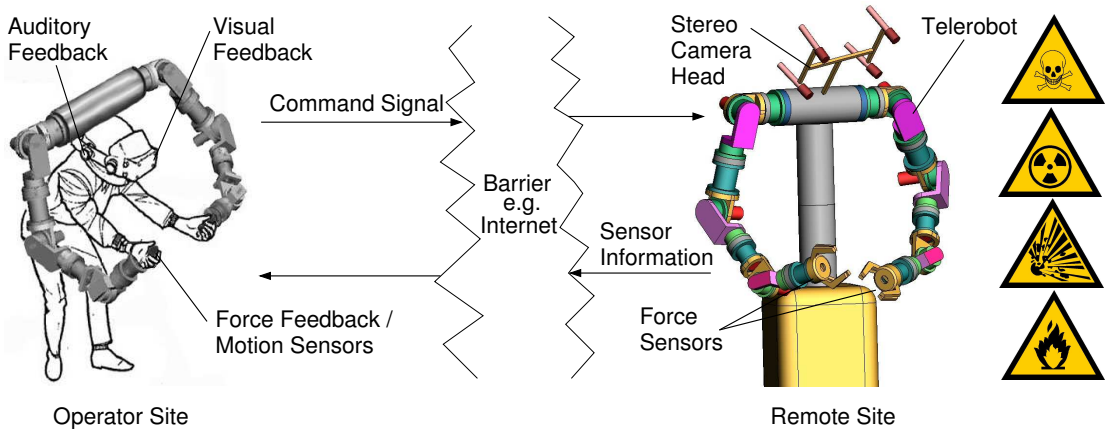


Figure 1.1: Components of a multimodal telepresence system

Below is a short summary of manipulation tasks that may be faced by a telerobot in a catastrophic scenario:

- explore the environment haptically
- remove mechanical obstacles: heavy and bulky objects like concrete or metal blocks
- dismantle obstacles mechanically using tools: cutting, unscrewing, breaking etc.
- secure ceilings and walls mechanically; build simple scaffolding-like constructions
- transport important or dangerous objects into/out of the dangerous area
- transport humans that are unconscious or otherwise not able to move.

Such tasks require specially designed devices, entirely different from the robots known from industry [81]. In the succeeding paragraphs, an overview of the telemanipulation systems will be given, with the particular focus on rescue applications.

Existing Telemanipulation Systems for Rescuing Applications

Conventional commercially available robotic systems that are used in rescue applications *and* have the possibility of physical interaction with the environment are represented by three examples shown in Figs. 1.2, 1.3 and 1.4. These are typical constructions differing

Figure 1.2: Vanguard ©Eodperformance Co.

Figure 1.3: EXPERT ©PIAP

Figure 1.4: thEODor ©Telerob GmbH

in size, ranging from 50 kg (Vanguard, [30]) to 150 kg (EXPERT, [101]) and more than 360 kg (thEODor, [39]), depending on the installed equipment. Their load capability ranges from 5 to 30 kg. Such robot units consist of a tracked crawler mobile platform, a manipulator with several kinds of attachable devices, and an operator station. The operator interface consists of a screen, joystick and buttons. The disadvantages of these constructions are their single arm configuration and a lack of force feedback. Moreover, most of the manipulators provide only joystick-driven joint space control and *not* Cartesian motion control.

To the best of authors' knowledge, only two commercial constructions exist that alleviate the aforementioned problems. These are the Japanese robot T-52 Enryu (literally "rescue dragon", [31]) shown in Fig. 1.5 and the American ARMSII (Articulated Remote Manipulator System) shown in Fig. 1.6. Both constructions are rather large and massive

Figure 1.5: Enryu Rescue Dragon Robot
©2004 tmsuk co., ltd

Figure 1.6: ARMS II
©Air Force Research Laboratory

(Enryu-5000 kg, ARMS-3600 kg) tractor platforms with hydraulically operated two arms ending in pincer "hands". Each Enryu arm is capable of lifting 500 kg with full arm extension of 10 m. The user interface consists in both cases of dual force feedback devices. Available information on these two constructions is classified as military (ARMS) or commercial (Enryu), hence scarce.

Laboratory based research is yielding more promising results than the commercially available products. Detailed overview of the relevant teleoperation systems may be found in [6, 71**?**]. Despite a large body of literature, most of the teleoperation related research concern general control methods for the time delayed systems, due to the inherent communication delay between the master and slave sites [27, 74, 143**?**]. For this reason uncomplicated hardware is used [47, 71] and the issues of the appropriate design and control of the telerobot are hardly addressed. In most cases, an industrial manipulator is used [57, 128, 129, 142]. The anthropomorphic manipulators built specially for this purpose are the constructions of KIST [23], DLR-light-weight-arm [3, 50], and NASA's ROBONAUT [4, 34, 35]. Work reported on the KIST design concentrated on the kinematic transformations and mapping between the operator's arm and the telerobot, without considering the requirements for the telemanipulator development listed in Sec. 1.2. The excellent design of DLR is exploited at the moment as a general purpose 7 DoF manipulator; the teleoperation specific works are still ongoing and their results have not been reported to date. The concepts of the compliant control, coming from [3], were extended and implemented within this thesis. The NASA design is very impressive, but the details of their implementation are not available to the research community. Another work presented in [18] describes teleoperation with a dual arm robot, but it addresses only the master side and the integration of multi sensory display.

We know only one telemanipulation system that is explicitly aimed at rescue applications. It is the dual 4 DoF construction built at the Technische Universität München, see [71, 72]. This work concentrates on the generic control of the teleoperation system and

the coordination of the finger-arm force display for the user interface. Due to the limited number of DoFs and small workspace, the aspects of the telerobot design and control are not addressed.

It is necessary to stress the contrast to i) typical large distance teleoperation systems (space or underwater), where significant time delays exclude real time operation, so that *wait-and-see* strategies are successfully implemented and ii) telemaintenance systems, where certain knowledge of the environment model may be exploited. In our opinion, there exist specific demands for a "rescue" type of system:

- no knowledge of the environment and the task is assumed
- all computations have to be performed online
- direct physical contact with humans is considered
- the time of the task completion is of vital importance.

Therefore security and intuitiveness of the operation, aiming at minimization of the human fatigue and attention, have priority over accuracy. For those reasons we have undertaken the effort of analyzing, designing and building a laboratory system to investigate the principles and implications of the teleoperation with a dual arm, anthropomorphic telerobot, aimed at technical aid in catastrophic scenarios.

The system that this thesis is concerned with is not limited to the aforementioned scenario and the methods developed here are transferable to other types of manipulators. However, it helps to focus the attention at high performance requirements.

1.2 Requirements for the Telemanipulator Development

The manipulator developed within the scope of this thesis has an anthropomorphic structure, i.e., it consists of a torso with two 7 DoF arms. Such a design is motivated by the strong belief that to transfer the manipulation skills of a human arm over a distance, the controlled robot must be equipped with manipulation capabilities that are at least comparable to these of the human arm. The majority of research in anthropomorphic robotics is still engrossed in locomotive issues of a humanoid, and the upper body plays a secondary role. At best, it is used for mirroring human appearance or being a contra-weight for walk control. In a teleoperation system driven by a human operator, it is the ability to explore the environment and to manipulate objects using arms and fingers that is particularly attractive and crucial for intuitive and efficient operation. The development of sufficiently large and powerful haptic input devices, together with the general technological progress in the last decades, has changed the perspective on the telemanipulation systems. The user is no longer operating the remote manipulators using joysticks, screens or buttons. Instead, it is now possible that he/she operates them as if they were his/her own arms. It is obvious that the tele-robots can no longer be industrial robots adapted somehow for teleoperation purposes. They have to be new developments addressing explicitly the specific requirements of anthropomorphic manipulation.

We can summarize the geometric/load requirements for the construction of the anthropomorphic telemanipulator as follows [2, 20]:

- arm workspace: convex semi-sphere with a radius of 80 cm
- arm static load: in average 5 kg
- dual arm configuration.

Those formulations are very general and will be investigated more in detail in the coming chapters. Nevertheless, it has to be emphasized that we are aiming at achieving not the anthropomorphic appearance, but the anthropomorphic *functionality*. As such we consider the following features:

1. *Arm redundancy*. Without changing the position of the hand, the arm configuration can be altered to achieve additional goals: reach through an opening or behind an obstacle, increase the structural stiffness or general manipulability.

2. One of the vital anthropomorphic features is the *ability to work in the singular configurations*. In order to increase the arm stiffness, or to make up for poor actuation, humans are exploiting the arm singularities very often. Especially, if it is necessary to produce (or compensate for) large forces.

3. The phenomenal dexterity of the human arm is mainly thanks to its *built-in (variable) compliance* (softness). Owing to the passive compliance of the tissues plus active compliance of the muscles, humans are able to manipulate a wide range of objects, also unmodeled and unknown before.

4. Human hands may take *arbitrary positions and orientations* within their workspace. Therefore all the control strategies must be applicable in 6 DoF. Limiting the number of DoFs is unacceptable.

1.3 Main Contributions and Outline of the Dissertation

This thesis summarizes the efforts undertaken to implement the aforementioned anthropomorphic prompts in a telerobotic system. The need of replication of human manipulation capabilities induced a re-thinking of almost every aspect of robot development. As already pointed out before, the performance of the early teleoperation systems was drastically limited by the performance of the input devices. Correspondingly, the manipulators used as telerobots were not very advanced: in many cases industrial manipulators were applied. It is the progress in the development of haptic interfaces that forces us to come back to the specific design of a *tele*-robotic torso with arms. As far as we know, this is one of the very few examples of robotic arm design for teleoperation purposes, and the only one with the anthropomorphic features as specified in previous paragraph. Because the robotic system presented in this thesis was built from scratch, the concepts presented here were verified experimentally at every stage of the development.

The thesis is organized in seven chapters regarding several aspect of the development of the anthropomorphic telerobotic system. Chapters 2 - 4 present the specific issues of the design and control of the 7 DoF manipulator. Chapters 5 and 6 concern the problems of implementation of the developed methods in the teleoperation system. Each chapter includes a detailed review of the relevant literature.

Chapter 2 is devoted to the kinematic design and control of the telemanipulator arms. First the kinematic structure is presented, with corresponding forward and inverse solutions in a closed algebraic form. For efficient redundancy handling, the extended kinematic concept is introduced with parametrization of the motion of the redundant degrees of freedom. The concept of the self motion parametrization of a redundant arm is well known [112, 147]. However, our method for describing the elbow motion from the perspective of the human operator (in the biophysically meaningful coordinates) is new in the robotic community. The elbow motion may be sensed and used as input for the control of the telerobot without resorting to the exoskeleton type of constructions. Such an approach makes it possible to solve the inverse kinematics problem analytically. The concept may be applied to all kinds of 7 DoF manipulators, and the solution to 7 DoF manipulators without joint offset.

To achieve proper workspace matching between the human arm and the developed manipulator, an adequate singularity handling strategy must be applied. A review and experimental comparison of such solutions is given in Sec. 2.3. We introduce a particularly attractive and novel Weighted Damped Inverse Kinematics with Task Priority in Sec. 2.3.5. This method allows for traversing through singularities with simultaneous weighing of the task space tracking error. This way it is possible to assign different levels of importance to the task coordinates, e.g., the elbow motion may be tracked in a less accurate way. The method is generic and may be used for all types of manipulators.

Chapter 3 describes the orientation representation and the orientation error definition. This is necessary for two reasons. First, because the kinematic structures of both master and slave manipulators are in general dissimilar, there is a need of a universal kinematic interface, independent of the device structure. Second, since the target tasks are strictly spatial (6 DoF) a suitable orientation representation must be used. The classical approaches using Euler or Cardanian angles fail in teleoperation scenarios due to algebraic singularities introduced by the representation. To avoid this problem, unit quaternions are used for describing orientation and corresponding rotational displacement. Simulation results comparing the performance of Euler angle and quaternion representation are presented. To our best knowledge, the usage of unit quaternions in teleoperation systems has not been reported elsewhere. Finally, a complete formulation of the task space vector is given.

Chapter 4 elaborates on the issues of manipulator control. The aim of this work is to find a control method assuring simultaneously good tracking and high fidelity of the desired impedance. A review of compliance control methods is given and several compliance control methods are experimentally compared. Among other issues, the tracking performance, impedance display fidelity, direction decoupling and the ability to control the null space compliance are taken as comparison criteria. The last feature is important in order to handle unexpected contacts that cannot be sensed by means of the wrist force/torque sensor. In order to control the redundant DoFs in a compliant manner, the extended kinematics concept was applied. As a result, an impedance control strategy with underlying

stiffness controller in motion control loop is proposed. The idea is new and was not reported in the literature to date.

Chapter 5 analyzes the teleoperation control loop as a benchmark for previously developed control methods. We show that the force/position architecture exploited in the experiments is a special case of the general four channel architecture. A method for tuning the impedance parameters of the slave robot is proposed. This method is based on the optimization of time-based or frequency-based objective function, which is derived from the transparency definition. Stability and transparency are analyzed and the teleoperation experiments in 6 DoF with the haptic input device ViSHaRD10 [131] are presented. Detailed description of the experimental setup and of the assignments is given. The experiments consist of haptic exploration of objects possessing different stiffness characteristics, and a of teleoperated screw-tightening task.

Chapter 6 introduces the concept of virtual forces for avoiding collisions between the two arms and the workspace boundaries. In a dual configuration, every arm becomes virtually a moving obstacle for the other one. Partitioning the workspace between the manipulators is not a solution for fine manipulation. Instead, the workspace should be shared in a dynamic way. The concept of virtual forces makes it possible; moreover, the advantage is twofold. First, exploiting the arm redundancy, certain types of collisions may be avoided with the escape null motions. This is achieved by the means of a local controller aiming at minimization of the virtual force. This is an example of the shared control in a teleoperation system. Second, the virtual forces corresponding to the unavoidable collisions, e.g., workspace boundaries or arm to arm collisions, are displayed to the human operator using the haptic interface. As a result, intuitive user involvement in collision avoidance is achieved. The basic idea of virtual forces comes from previous work in dual robot telemanipulation. The works presented in [71] used visual display to signal the collision danger, what was however reported to increase the user burden. This is unacceptable in a dexterous telemanipulation system. The concept presented in chapter 6 is simple and effective. It was not reported in other teleoperation system to date, except the preliminary work [71]. However, it is highly recommendable for multi robot teleoperation systems.

Chapter 7 closes the thesis with concluding remarks and directions for the future work.

For more information regarding this dissertation, including source code, detailed images and videos, the reader is kindly encouraged to visit the ***www.lsr.ei.tum.de/team/stan***.

2 Telemanipulator Kinematics

The need for human-equivalent manipulative capabilities has motivated the development of a dexterous, dual arm robotic system. Its main purpose is to work on the remote site of the teleoperation system. Since teleoperation systems are controlled by human operators, it is required that the telerobot mirrors the human motion as closely as possible.

Anthropomorphic robot designs have been challenging many researchers, as well as practitioners in robotics [3, 50, 69, 112, 113, 143] and computer animation community [86, 147], and within the last two decades have been accepted. However, due to high levels of task orientation, several aspects of the kinematic control are still unsolved. As mentioned before, one has to consider the difference between the industrial robots and an anthropomorphic structure. First, the industrial manipulators are built usually as maximum 6 DoF devices, whereas the human arm has significantly more DoFs, which results in a redundant structure. The redundancy is used by humans to reconfigure the arms in order to achieve additional tasks e.g., reach through an opening, avoid obstacles, increase the manipulability or perform a whole arm grip. Industrial robots are not assigned to such tasks; they work simply in a structured environment where the redundancy is not necessary and is avoided due to financial constraints. The second major difference is the contrary singularity handling. Industrial manipulators are configured to avoid the singularities due to mechanical and control problems at such configurations. On the contrary, humans exploit their singular-like configurations (hand and forearm aligned, forearm and upper arm aligned) very often because at these configurations the arm structural stiffness is highest. It is particularly important in tasks requiring exertion of large forces. Consider pushing a disabled car or carrying a heavy load: the arms are always outstretched since the arm bones are much more rigid than the muscles. Our teleoperation experiments, described in Sec. 5.5, showed that in fact, for large force generation, the robot was driven into the singular configuration, just as the human operator would drive his/her own arm.

In this chapter issues related to the kinematics of the anthropomorphic telemanipulator are presented. The kinematic design with the corresponding forward and inverse solutions in an analytical form are given in Sec. 2.1 and Sec. 2.2.1. Physically meaningful parametrization of the elbow motion resulting from the arm redundancy is described in Sec. 2.1.2. The analytical inverse kinematics introduce the problem of singular configurations, which are analyzed for the spherical wrist in Sec. 2.2.2. Second part of the chapter studies the velocity-based inverse kinematics algorithms with respect to their singularity robustness. Sec. 2.3 comprises a detailed review of the commonly used Jacobian methods, as a way of derivation of a particularly attractive Weighted Damped Least Square Method, which is outlined in Sec. 2.3.5, and evaluated experimentally in Sec. 2.4. Finally, two redundancy resolution methods are described: the optimization methods with the Jacobian pseudoinverse in Sec. 2.5.1, and the extended kinematics method in Sec. 2.5.2.

2.1 Telemanipulator Kinematic Design

The telemanipulator consists of two identical, human-scaled arms, connected in a mirrored configuration imitating anthropomorphic torso structure shown in Fig. 2.1. Kinematic analysis of human limbs reveals that the minimum number of DoFs used for their modeling is 7, although there exist approaches representing human arms as 8, 10 or more DoFs [2, 20, 69, 75, 86]. Since the kinematical structure of the human operator and the slave robot do not need to match exactly (on the condition of workspace matching), a 7 DoF structure is chosen as a trade-off between system complexity and performance. Investigations of possible 7 DoF structures result in the design shown in Fig. 2.1, which is proved in [55] to be optimal in the sense of "elimination of singularities, mechanical realizability, kinematic simplicity and workspace shape". The design consists of two spherical joints with three DoFs at the shoulder and the wrist, and one rotational joint at the elbow. The Denavit-

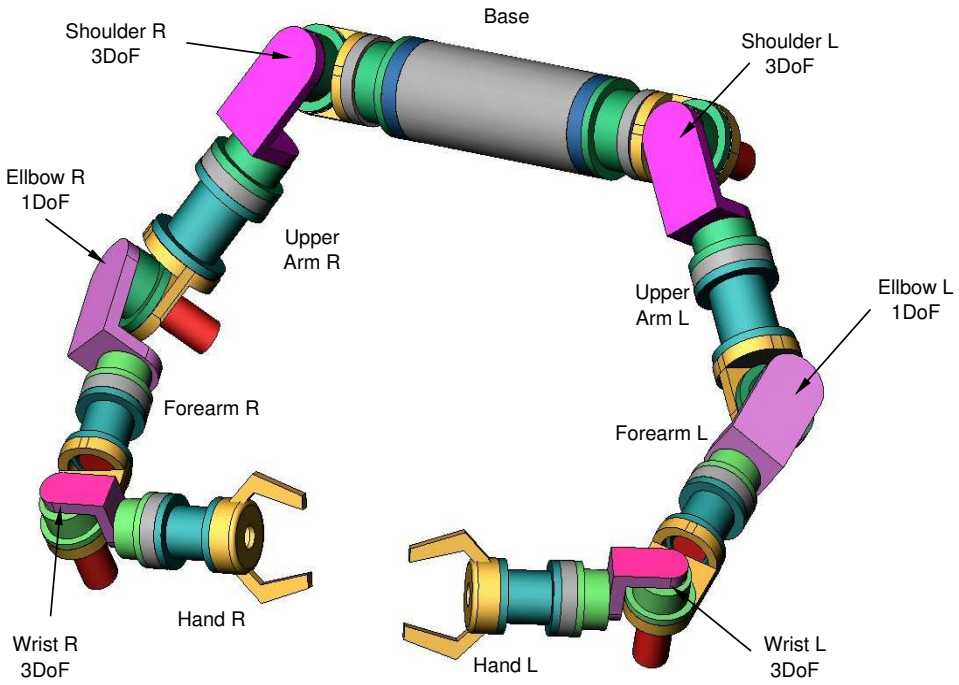


Figure 2.1: Design of the dual arm telerobotic manipulator: CAD drawing, front view

Hartenberg parameters for the arm, according to the convention in e.g., [111, 118], are listed in Tab. A.4, the corresponding set of frames is shown in Fig. 2.2.

Human shoulder and human wrist are highly complex kinematic structures, whereas a spherical joint is merely an approximation. Nevertheless, the Cartesian workspace does not suffer because of that, due to the increased angle ranges in joint space [2, 69].

2.1.1 Forward Kinematics for a 7 DoF Manipulator

The kinematic notation and the derivations are given as in standard robotic books [111, 118]. The position and orientation of the robot end-effector is described uniquely with a

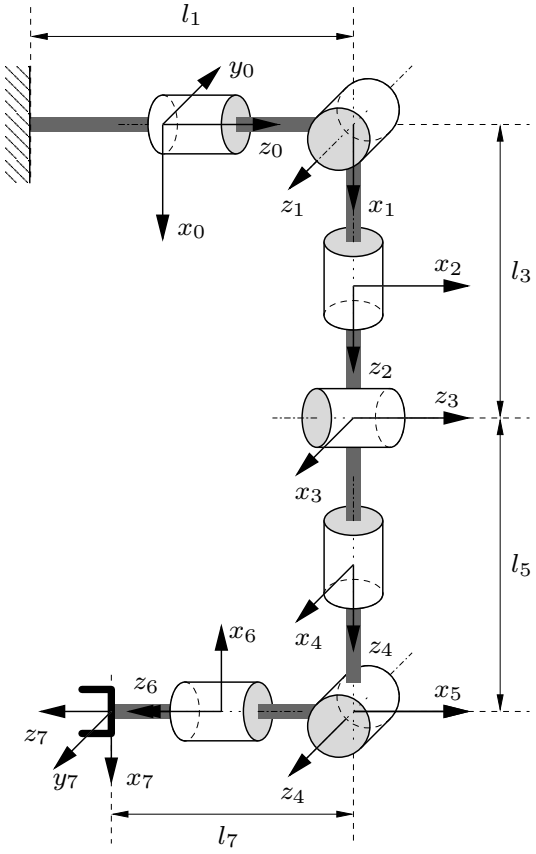


Figure 2.2: Kinematic structure and link coordinate systems of the 7 DoF arm

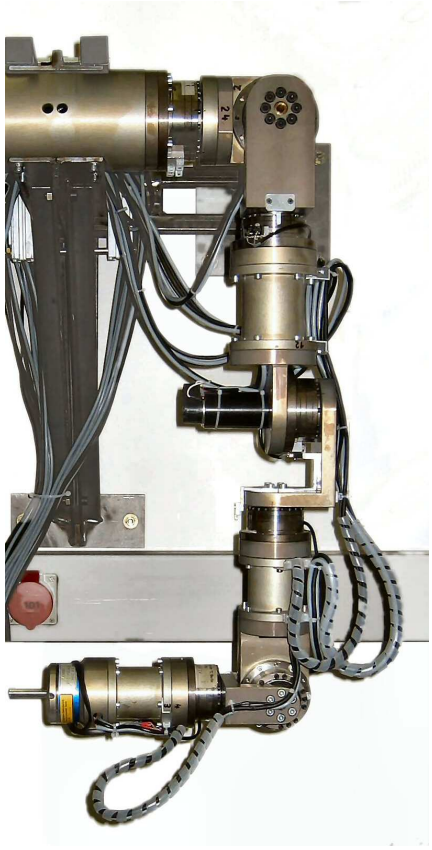


Figure 2.3: Photograph of the physical arm construction

Table 2.1: Teleoperator Denavit-Hartenberg parameters, $l_1 = 0.2655$ m for the right arm and $l_1 = -0.2655$ m for the left arm

Joint nr i	Link offset a_i	Link length l_i [m]	Link twist α_i [rad]
1	0	l_1	$\pi/2$
2	0	0	$\pi/2$
3	0	0.312	$-\pi/2$
4	0	0	$\pi/2$
5	0	0.312	$-\pi/2$
6	0	0	$\pi/2$
7	0	0.244	$\pi/2$

homogeneous transformation matrix

$$\boldsymbol{T}_0^7 = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix}, \tag{2.1}$$

where $\boldsymbol{p} = [p_x\, p_y\, p_z]^T$ will be referred to as Cartesian position vector and 3×3 matrix $\boldsymbol{R}$ as the orientation matrix. The matrix $\boldsymbol{T}_0^7$ is computed using the link homogeneous matrices as in [118]

$$\boldsymbol{T}_0^7 = \boldsymbol{A}_1 \boldsymbol{A}_2 \ldots \boldsymbol{A}_7. \tag{2.2}$$

The end-effector velocity is described by the time derivative of the vector $\boldsymbol{p}$, while its angular velocity $\boldsymbol{\omega}$ is expressed as the time derivative of the rotation matrix $\boldsymbol{R}$

$$\dot{\boldsymbol{R}} = \boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{R}, \tag{2.3}$$

where $\boldsymbol{S}(\cdot)$ is the skew symmetric matrix operator. The Cartesian and joint space velocities are related by the mapping

$$\begin{bmatrix} \dot{\boldsymbol{p}} \\ \boldsymbol{\omega} \end{bmatrix} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{2.4}$$

where $\boldsymbol{q}$ is the vector of joint coordinates. The matrix $\boldsymbol{J}$ used here is called *geometrical* [111] or *spatial* [84] Jacobian, because it refers to geometrical velocities in Cartesian space, independently from the orientation parametrization, see Sec. 3.2. For an n-link robot it is computed with the formulas [111, 118]:

$$\boldsymbol{J} = [\boldsymbol{J}_1 \boldsymbol{J}_2 \dots \boldsymbol{J}_n]$$

$$\boldsymbol{J}_i = \begin{bmatrix} \boldsymbol{z}_{i-1} \times (\boldsymbol{o}_n - \boldsymbol{o}_{i-1}) \\ \boldsymbol{z}_{i-1} \end{bmatrix}, \tag{2.5}$$

where $\boldsymbol{z}_i$ is the unit vector showing the rotation axis of the i-th link, and $\boldsymbol{o}_i$ denotes the vector from the origin of the frame 0 to the origin of the frame i. For the detailed forward kinematic formulation, see App. A.1. The *geometrical* Jacobian is much less computationally expensive than the *analytical* Jacobian obtained by differentiation of the forward kinematics solution [111]. In case of the analytical Jacobian, the corresponding relation between the rotational velocity and the derivatives of the task space parameters depends strongly on the chosen orientation representation, as explained in Chapter 3. This representation introduces additional algebraic singularities, which may cause severe stability problems. This property will be explained more in detail in Sec. 3.2.1. Algebraic singularities do not exist in case of the geometrical Jacobian. Its inversion is sensitive only to mechanical singularities, see Sec. 2.2.2.

2.1.2 Definition of the Elbow Angle: Self Motion Parametrization

In case of the anthropomorphic manipulator it is possible to describe the self motion in a physically meaningful way. The self motion results from the kinematic redundancy, and its physical meaning is as follows: if one holds the shoulder s, the wrist w in a fixed position, the elbow e is free to swivel about the axis from the shoulder to the wrist, as shown in Fig. 2.4. This phenomenon is referred to in the literature as "orbiting" [86, 147], and the corresponding parameter is called *arm* or *elbow angle*. The elbow position on the circle is determined by specifying the angle θ, which is the angle between the arm projection of the "zero position" $\boldsymbol{e}_0$ and the actual position $\boldsymbol{e}$ onto the plane of the circle. The choice of "zero position" is arbitrary; in our case, the elbow angle is computed as the angle between the planes $\triangle s_r s_l w_r$ and $\triangle s_r e_r w_r$

$$\sin\theta = -\frac{s_2 s_3 s_4}{|s_4|} \tag{2.6}$$

$$\cos\theta = -|s_4| \frac{s_4 c_2 + s_2 c_3 (1 + c_4)}{\sqrt{2 s_4 (1 + c_4)}} \tag{2.7}$$

$$\theta = \operatorname{atan2}(\sin\theta, \cos\theta), \tag{2.8}$$

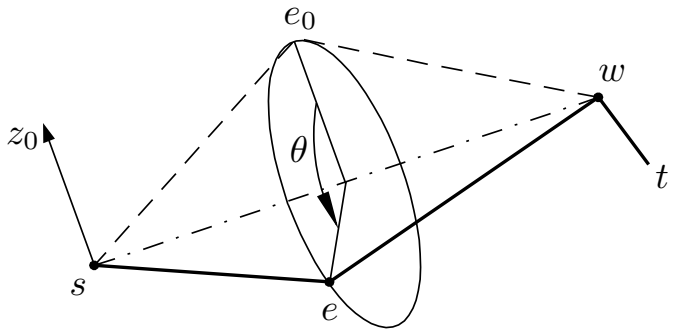


Figure 2.4: The elbow circle definition

where the right arm is taken as an example (index 'l' refers to the left arm, whereas 'r' or no index refers to the right arm). The variables $s_i = \sin q_i$ and $c_i = \cos q_i$ are the functions of the corresponding i-th joint variable. It is interesting to notice that the elbow angle is a physiological measure, independent from the arm size and the posture of the whole body. Therefore it may be used to control the arm self motion by different human operators without scaling or extensive calibration measures. Other ways of the self motion description base on the elbow positioning in absolute coordinates [69], which requires calibration for every operator, due to different arm size and geometry.

For the analytical proof that the elbow motion lies in the null space of the Jacobian, please refer to App. A.2.

2.2 Closed Form Inverse Kinematics Algorithm for 7 DoF Anthropomorphic Arm

The specific construction of the 7 DoF arm makes it possible that, despite the redundancy, we can obtain the analytical solution to the inverse kinematics problem. In the following sections, a method based on the parametrization of the self motion is outlined. First, at the position level and second, on the velocity level. Finally, the influence of the singularity on the solution is analyzed.

2.2.1 Position-Based Inverse Kinematics Algorithm

The inverse kinematics (IK) problem is stated as follows: find the joint coordinates $\boldsymbol{q} = [q_1, q_2, ..., q_7]^T$, given a homogeneous transformation matrix $\boldsymbol{T}_0^7$, defined as in (2.1). First, one needs to note that this equation, due to the arm redundancy, does not have a unique solution. In case of an anthropomorphic structure however, the specific mechanical construction of the arm makes it possible to find a solution, after having extended the task space with the elbow angle θ. Later on, we assume θ to be given. The wrist partitioning method described in [118] and [55] is used. This approach refers to manipulators, whose last three rotation axes intersect in one point, forming a spherical wrist. The movement of those three joints does not affect the wrist position $\boldsymbol{w}$, which depends only on the first four angles. The whole problem may be decoupled then into two simpler problems, namely *inverse position kinematics* and *inverse orientation kinematics*.

The algorithm used here may be summarized as follows. For the class of manipulators with the spherical wrist, and the wrist length l_7, the wrist position $\boldsymbol{w}$ is calculated by

$$\boldsymbol{w} = \boldsymbol{p} - l_7 \boldsymbol{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{2.9}$$

Having obtained $\boldsymbol{w}$, one finds the first four angles $(q_1, ..., q_4)$ that put the wrist at $\boldsymbol{w}$, and a set of Euler angles corresponding to the rotation matrix

$$\boldsymbol{R}_7^4 = (\boldsymbol{R}_0^4)^T \boldsymbol{R}, \tag{2.10}$$

where $\boldsymbol{R}_0^4$ is the orientation matrix of the frame 4 in reference to the world coordinates, and $\boldsymbol{R}_4^7$ is the orientation matrix of the tool in reference to the frame 4.

Solution for q_4

Because q_4 is the only variable that determines the distance between the shoulder and the wrist, it is calculated from the triangle formed by the shoulder $\boldsymbol{s}$, elbow $\boldsymbol{e}$ and the wrist $\boldsymbol{w}$, see Fig.2.5:

$$q_4 = \pi - \text{acos} \frac{l_3^2 + l_5^2 - |\boldsymbol{w} - \boldsymbol{s}|^2}{2 l_3 l_5}. \tag{2.11}$$

In the implementation it is assumed that the elbow bends only in one direction, i.e., $q_4 \in (0, \pi)$. It is possible to extend the solution with $q_4' = -q_4$ with a suitable selection algorithm.

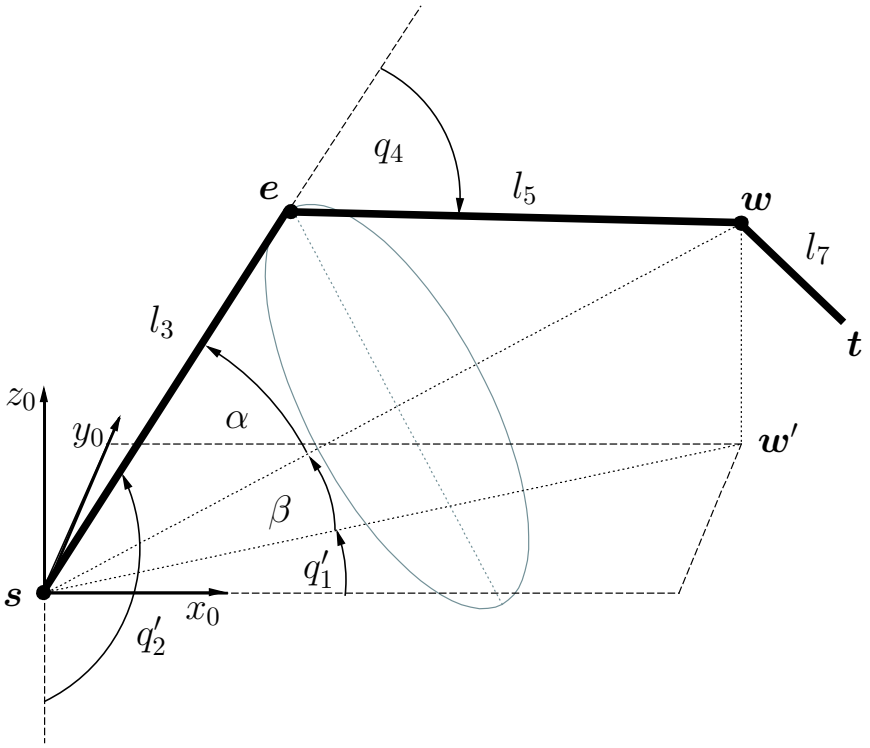


Figure 2.5: The 7 DoF arm at θ=0

Solution for q_1 and q_2

We start with finding the q_1^0 and q_2^0 that would result if $\theta = 0$, as shown in Fig. 2.5. In this position, both the upper arm l_3 and the lower arm l_5 lie in a plane normal to the plane determined by the axes $\boldsymbol{x}_0$ and $\boldsymbol{y}_0$. We use the plain trigonometry rules for those two planes, which results in:

$$q_1^0 = \text{atan2}(w_x, w_y) \tag{2.12}$$

$$q_2^0 = \alpha + \beta + \frac{\pi}{2} = \text{asin}\frac{w_z}{|\boldsymbol{w}|} + \text{asin}(l_5\frac{sin(\pi - |q_4|)}{|\boldsymbol{w}|}) + \frac{\pi}{2}, \tag{2.13}$$

where the superscript 0 denotes a quantity referred to $\theta = 0$. The elbow position $\boldsymbol{e}^0$ for q_1^0 and q_2^0 is given by:

$$\boldsymbol{e}^0 = \boldsymbol{A}_1^0\boldsymbol{A}_2^0 \begin{bmatrix} 0 \\ 0 \\ l_3 \end{bmatrix} = l_3 \begin{bmatrix} \cos(q_1^0)\sin(q_2^0) \\ \sin(q_1^0)\sin(q_2^0) \\ -\cos(q_2^0) \end{bmatrix}. \tag{2.14}$$

In order to find the new elbow position $\boldsymbol{e}$ we need to derive a rotation matrix $\boldsymbol{R}_{\boldsymbol{n},\theta}$, representing the rotation of θ degrees about the shoulder to wrist axis, so that

$$\boldsymbol{e} = \boldsymbol{e}^0\boldsymbol{R}_{\boldsymbol{n},\theta}. \tag{2.15}$$

Using the Rodrigues formula, the new elbow position is found

$$\boldsymbol{e} = \boldsymbol{n}(\boldsymbol{n}\cdot\boldsymbol{e}^0)(1-\cos\theta) + \boldsymbol{e}^0\cos\theta + \boldsymbol{n}\times\boldsymbol{e}^0\sin\theta \tag{2.16}$$

and the first two joint angles

$$q_1 = \text{atan2}(e_y, e_x) \tag{2.17}$$

$$q_2 = \text{atan2}(\sqrt{e_x^2 + e_y^2}, -e_z). \tag{2.18}$$

Solution for q_3

We can obtain q_3 by solving the wrist position equation

$$\mathbf{A_1}\mathbf{A_2}\mathbf{A_3}\mathbf{A_4} \begin{bmatrix} 0 \\ 0 \\ l_3 \\ 1 \end{bmatrix} = \boldsymbol{w}, \tag{2.19}$$

which takes the form

$$\begin{bmatrix} l_3c_1s_2 + l_5(c_1c_4s_2 + (c_1c_2c_3 + s_1s_3)s_4) \\ l_3s_1s_2 + l_5(c_4s_1s_2 + (c_2c_3s_1 - c_1s_3)s_4) \\ l_1 - l_3c_2 + l_5(-c_2c_4 + c_3s_2s_4) \end{bmatrix} = \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix}. \tag{2.20}$$

Solving for s_3 and c_3 yields

$$s_3 = \frac{-c_1l_3s_2 - c_1c_4l_5s_2 - c_1c_2c_3l_5s_4 + w_x}{l_5s_1s_4} \tag{2.21}$$

$$c_3 = \frac{-l_1 + c_2l_3 + c_2c_4l_5 + w_z}{l_5s_2s_4}, \tag{2.22}$$

hence the third angle is computed as

$$q_3 = \text{atan2}(s_3, c_3). \tag{2.23}$$

Wrist joint angles

Having obtained the first four angles, we find the hand orientation $\boldsymbol{U} = (\boldsymbol{R}_0^4)^T \boldsymbol{R}$ relative to the forearm. The wrist joint angles are found as a set of Euler angles by solving the following equation:

$$\boldsymbol{R}_4^7 = \boldsymbol{U} \tag{2.24}$$

$$\begin{bmatrix} c_5c_6c_7 - s_5s_7 & -c_5c_6s_7 - s_5c_7 & -c_5s_6 \\ s_5c_6c_7 + c_5s_7 & -s_5c_6s_7 - c_5c_7 & -s_5s_6 \\ s_6c_7 & -s_6s_7 & c_6 \end{bmatrix} = \boldsymbol{U}. \tag{2.25}$$

Due to the physical nature of spherical wrists, there exist always two solutions to (2.25) determined by the middle (in this case 6-th) joint. For $s_6 \neq 0$ one gets 2 sets of solutions distinguished by $(')$ or $('')$

$$\begin{aligned} q_6' &= \text{acos}(u_{33}) && (2.26)\\ q_5' &= \text{atan2}(-u_{23}/s_6'), -u_{13}/s_6')) && (2.27)\\ q_7' &= \text{atan2}(-u_{32}/s_6'), u_{31}/s_6')) && (2.28) \end{aligned}$$

or

$$\begin{aligned} q_6'' &= -\text{acos}(u_{33}) && (2.29)\\ q_5'' &= \text{atan2}(-u_{23}/s_6''), -u_{13}/s_6'')) && (2.30)\\ q_7'' &= \text{atan2}(-u_{32}/s_6''), u_{31}/s_6'')). && (2.31) \end{aligned}$$

For $s_6 = 0$ equation (2.25) takes one of the two forms:
CASE 1: $q_6 = 0$, $c_6 = 1$, $u_{33} = 1$

$$\begin{bmatrix} c_5c_7 - s_5s_7 & c_5s_7 - s_5c_7 & 0 \\ s_5c_7 + c_5s_7 & -s_5s_7 - c_5c_7 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_{5+7} & s_{5+7} & 0 \\ s_{5+7} & c_{5+7} & \\ 0 & 0 & 1 \end{bmatrix} = \boldsymbol{U}, \tag{2.32}$$

so that the sum

$$q_5 + q_7 = \text{atan2}(u_{21}, u_{11}) = \text{atan2}(-u_{12}, u_{22}); \tag{2.33}$$

CASE 2: $q_6 = \pi$, $c_6 = -1$, $u_{33} = -1$

$$\begin{bmatrix} -c_5c_7 - s_5s_7 & c_5s_7 - s_5c_7 & 0 \\ -s_5c_7 + c_5s_7 & s_5s_7 - c_5c_7 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -c_{5-7} & -s_{5-7} & 0 \\ s_{5-7} & c_{5-7} & \\ 0 & 0 & -1 \end{bmatrix} = \boldsymbol{U} \tag{2.34}$$

and

$$q_5 + q_7 = -\text{atan2}(-u_{12}, -u_{11}). \tag{2.35}$$

Both equations result in indefinite number of solutions. Because there is no numerically sound solution, we set by convention $q_5 = q_{5old}$, where q_{5old} is the solution at the previous iteration step and compute q_7 from (2.33).

Note on the implementation

For the proper choice of the solution between the ($'$) and ($''$) we define the continuity condition as a norm of the difference between previous (denoted '*old*') and current solution:

$$\text{IF } \sum_{\text{i=5}}^{\text{i=7}} |q_i' - q_{iold}| < \sum_{\text{i=5}}^{\text{i=7}} |q_i'' - q_{iold}| \text{ THEN } \mathrm{q_5 = q_5'};\ \mathrm{q_6 = q_6'};\ \mathrm{q_7 = q_7'} \tag{2.36}$$

$$\text{ELSE } \mathrm{q_5 = q_5''};\ \mathrm{q_6 = q_6''};\ \mathrm{q_7 = q_7''}. \tag{2.37}$$

However, this method fails exactly at the singularity, when one has to switch between the solutions, resulting either in oscillatory or high velocity behavior, or in locking the wrist at $q_6 = 0$, so that passing through the singularity becomes unreliable. This is why the solution on the velocity level is proposed.

2.2.2 Velocity-Based Inverse Kinematics for a Spherical Wrist

The inverse kinematics problem may be in general tackled also on the velocity level. Consider the a spherical robotic wrist of the 7 DoF arm shown in Fig. 2.6. It consists of 3 rotational axes numbered, as a part of a larger structure, 5 to 7, with the corresponding coordinates q_5 to q_7. All the rotational axes intersect in one point, same with the 6-th joint local coordinate system; this is also where we put our reference coordinate system.

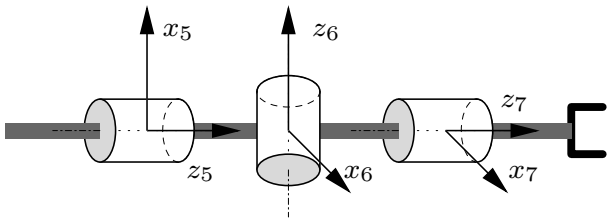


Figure 2.6: Kinematic structure of a spherical wrist

The rotational velocity $\boldsymbol{\omega}$ of the manipulator end-effector must be first expressed in link 4 coordinate system

$$\boldsymbol{\omega}_4 = \begin{bmatrix} s_1s_3s_4 + s_2c_1c_4 + s_4c_1c_2c_3 & s_3s_4c_1 - s_1s_2c_4 - s_1s_4c_2c_3 & c_2c_4 - s_2s_4c_3 \\ s_1c_3 - s_3c_1c_2 & c_1c_3 + s_1s_3c_2 & s_2s_3 \\ s_2s_4c_1 - s_1s_3c_4 - c_1c_2c_3c_4 & s_1s_2s_4 + s_3c_1c_4 - s_1c_2c_3c_4 & s_4c_2 + s_2c_3c_4 \end{bmatrix} \boldsymbol{\omega}. \tag{2.38}$$

Then the angular velocity $\boldsymbol{\omega}_4$ and the joint velocity $\dot{\boldsymbol{q}}_{567} = [q_5, q_6, q_7]^T$ are related by a nonlinear mapping

$$\boldsymbol{\omega}_4 = \boldsymbol{J}_{567}\dot{\boldsymbol{q}}_{567} = \begin{bmatrix} 1 & 0 & c_6 \\ 0 & c_5 & s_5s_6 \\ 0 & s_5 & -s_6c_5 \end{bmatrix} \begin{bmatrix} \dot{q}_5 \\ \dot{q}_6 \\ \dot{q}_7 \end{bmatrix}. \tag{2.39}$$

Solving this equation for joint velocities one gets

$$\dot{q}_5 = \omega_x - (\omega_z c_5 - \omega_y s_5)c_6/s_6 \tag{2.40}$$

$$\dot{q}_6 = -\omega_y c_5 - \omega_z s_5 \tag{2.41}$$

$$\dot{q}_7 = (\omega_z c_5 - \omega_y s_5)/s_6, \tag{2.42}$$

where ω_x, ω_y and ω_z are corresponding components of $\boldsymbol{\omega}_4$ vector.

Influence of Singularity

The above equations show clearly the existence of singularity in the solution for $\dot{q}_5$ and $\dot{q}_7$, where the denominator s_6 causes the velocities going to infinity as q_6 approaches zero. Since such high velocities are physically unrealizable, we cannot obtain accurate motion. This is the first deficiency of singular configurations. Second is that at a singular configuration ($q_6 = 0$) the rotational axes $\boldsymbol{z}_5$ and $\boldsymbol{z}_7$ of joints 5 and 7 are aligned. While it is possible to rotate freely about the $\boldsymbol{z}_6$, $\boldsymbol{y}_6$ axes, the rotation about the $\boldsymbol{x}_6$ is impossible, and one DoF is lost. Some inverse kinematics algorithms get either locked or unstable at such a configuration.

Using the Jacobian formalism, the singular configurations are found as the roots of $\det(\boldsymbol{J}) = 0$ (or $\det(\boldsymbol{J}\boldsymbol{J}^{\mathrm{T}}) = 0$ in a redundant case). Thus, singularities are identified by the rank deficiency of the matrix $\boldsymbol{J}$, which physically represents the inability of the robot to generate an arbitrary velocity in the task space.

2.3 Singularity-Robust Inverse Kinematics

This section introduces the Jacobian methods in inverse kinematics, with the goal of finding a most suitable algorithm for singularity handling. The state of the art methods are first reviewed and later on outlined more in detail. Weighted Damped Least Square Method is introduced as an extension to the known algorithms.

2.3.1 Jacobian Methods

The Jacobian methods are applied to solving the system of linear equations (2.4) by means of inverting the Jacobian matrix $\boldsymbol{J}$,

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^{-1}(\boldsymbol{q})\dot{\boldsymbol{x}}. \tag{2.43}$$

In a regular case, when the Jacobian is square and of full rank, this solution is equivalent to the analytical solutions

$$\boldsymbol{q} = IK(\boldsymbol{x}) \tag{2.44}$$
$$\dot{\boldsymbol{q}} = IK_v(\dot{\boldsymbol{x}}) \tag{2.45}$$

and all of them are complementary and may be used interchangeably. However, there are major differences between the analytical methods and Jacobian method. The great advantage of analytical solutions is, whether they are available, their computational simplicity. They are also a useful tool for the analysis of the singularities. On the other hand, all the singular configurations must be considered separately on a *case by case* basis, see Sec. 2.2.2. Moreover, there exists no systematic formalism to handle the redundancy and to include optimization methods. In comparison, the computational cost of Jacobian methods is much higher, but this drawback is compensated by their generality in handling the regular, singular and redundant cases in an consolidated manner.

2.3.2 Singularity Robustness: State of the Art

One problem at the singular configuration is that the manipulator Jacobian becomes non invertible. Efforts to remedy this shortcoming are undertaken by researchers for more than three decades and may be classified along three main lines reviewed briefly below. The most common solution is to avoid singularities using e.g., pseudoinverse [65, 76, 78, 88, 106] or other task specific singularity avoidance techniques [65, 76, 132]. Unfortunately, it causes severe restrictions on both the configuration and task space, because the singularity is splitting the configuration space into separate components. As the result, significant portion of workspace is lost; to make up for this highly complicated (usually redundant) kinematic structures are proposed [131]. For this reason, the algorithms capable of traversing through singularity became very attractive.

The normal form approach [124, 125] offers an exact path tracking by slicing the desired path, and planning the pieces close to singularity in the joint space. Finally, the pieces are joined. The main disadvantage of this kind of methods is their off line character: they cannot be applied in teleoperation. The extended Jacobian technique [95] supplements the original Jacobian with auxiliary kinematic functions to restore its well conditioning. An interesting example of such approach is the singularity-consistent approach [127, 128], which is suppressing the velocity, but preserving the direction of motion. However, it is developed for space applications of the "wait and see" type, and is also not applicable online. Moreover, for non-redundant manipulators with square Jacobian, the extended Jacobian becomes non square, which causes additional cost for the generalized inverse.

As a "pragmatic" way of passing through singularity, a so called "singularity-robust" inverse [11, 16, 24, 135] is used. Instead of inverting the original Jacobian at singularity, a well condition matrix is inverted. This method is further referred to as "Damped Least Square Method" and described more in detail in Sec. 2.3.4. The weakness of this method are tracking errors even far from singular configurations.

Another interesting idea of obtaining a singularity consistent motion is using the transposed Jacobian matrix $\boldsymbol{J}^T$ [11, 111] or a modified transposed matrix [110] in lieu of inverted Jacobian $\boldsymbol{J}^{-1}$. Despite numerical stability, the drawback of this method are significant tracking errors in compare with the $\boldsymbol{J}^{-1}$ method.

There exists a large body of literature describing methods based on neural networks and computational intelligence [36, 60, 104, 126]. However, they are normally predesignated for path planning in known environments and in strictly offline applications.

Special attention must be drawn to the scenarios, which include human-robot interaction: teleoperation or direct physical contact. This feature is explicitly addressed in the works of Kosuge [70]. They focus on traversing through singularity by means of switching the IK solutions, to provide an accurate solution. In such cases however, robot accuracy is less important than the issues of security and user comfort. In our preliminary user studies, we found that excessive joint velocities are dangerous and scary. This strengthens the motivation for limiting the joint velocities in the neighborhood of singularity, even on the cost of accuracy. In the following paragraphs we analyze a class of IK algorithms, capable of both singularity traversing and damping the joint velocities in a controllable way, for which the experimental results are presented.

2.3.3 The Least Square (Pseudoinverse) Method

The most widely proposed methods for solving the inverse kinematic problem near singularities involve the use of the least square (LS) inverse (or pseudoinverse) $\boldsymbol{J}^{\#}$. In this method, proposed by Whitney in 1969 [137] and used commonly for redundant manipulators, a solution is sought, which minimizes the quadratic cost function of the joint velocities

$$g(\dot{\boldsymbol{q}}) = \frac{1}{2}\dot{\boldsymbol{q}}^T\dot{\boldsymbol{q}}. \tag{2.46}$$

The solution is found with the *Lagrangian multipliers* method [111]. Take a modified cost function

$$g(\dot{\boldsymbol{q}}, \boldsymbol{\lambda}) = \frac{1}{2}\dot{\boldsymbol{q}}^T\dot{\boldsymbol{q}} + \boldsymbol{\lambda}^T(\dot{\boldsymbol{x}} - \boldsymbol{J}\dot{\boldsymbol{q}}), \tag{2.47}$$

where λ is $m \times 1$ vector of unknown multipliers, and set

$$\frac{\partial g}{\partial \dot{\boldsymbol{q}}}^T = \boldsymbol{0} \quad \Rightarrow \quad \dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}} \tag{2.48}$$

$$\frac{\partial g}{\partial \boldsymbol{\lambda}}^T = \boldsymbol{0} \quad \Rightarrow \quad \boldsymbol{J}\dot{\boldsymbol{q}} = \boldsymbol{J}\boldsymbol{J}^T\boldsymbol{\lambda}. \tag{2.49}$$

Solving for $\dot{\boldsymbol{q}}$ one gets the final result

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T)^{-1} = \boldsymbol{J}^{\#}\dot{\boldsymbol{x}}. \tag{2.50}$$

Main disadvantage of this method is that is produces large and discontinuous velocities near the singularities. It can be shown by expressing $\boldsymbol{J}^{\#}$ in terms of singular value decomposition

$$\boldsymbol{J}^{\#} = \sum \frac{1}{\sigma_i}\boldsymbol{V}_i\boldsymbol{U}_i^T. \tag{2.51}$$

The singular values of $\boldsymbol{J}$, denoted by σ_i, are calculated and ordered from largest to smallest so that

$$\sigma_1 \geq \sigma_2 \geq \ldots \sigma_m \geq 0. \tag{2.52}$$

While approaching a singularity, the smallest singular value σ_{min} becomes small leading to high joint velocities. At singularity σ_{min} becomes zero; consequently, it is not taken into account any more. The summation is carried out up to $m - 1$, and the joint velocity decreases significantly.

2.3.4 Damped Least Square Method

The damped least squares (DLS) method avoids many of the LS-inverse method's problems with singularities: the solution discontinuity and the excessive velocities. The DLS method is known as the Levenberg-Marquardt [88] stabilization method. This solution minimizes the modified const function

$$g(\dot{\boldsymbol{q}}, \dot{\boldsymbol{x}}) = ||\dot{\boldsymbol{x}} - \boldsymbol{J}\dot{\boldsymbol{q}}|| + \gamma^2||\dot{\boldsymbol{q}}||^2, \tag{2.53}$$

where γ is a constant. This new criterion means that the end-effector tracking error is weighted against the norm of joint velocity by using the factor γ also known as the damping factor. This solution is typically obtained as the least-squares solution of the following system:

$$\dot{\boldsymbol{q}} = (\boldsymbol{J}^T\boldsymbol{J} + \gamma^2\boldsymbol{I}_n)^{-1}\boldsymbol{J}^T\dot{\boldsymbol{x}}. \tag{2.54}$$

If $n > m$, the following equivalent relation is easier to compute :

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T + \gamma^2 \boldsymbol{I}_m)^{-1}\dot{\boldsymbol{x}} = \boldsymbol{J}^*\dot{\boldsymbol{x}}, \tag{2.55}$$

where $\boldsymbol{J}^*$ will be referred to as a "damped" inverse Jacobian. For $\gamma = 0$ this solution transforms into the LS solution (2.50). The SVD of this solution is given by

$$\boldsymbol{J}^* = \sum \frac{\sigma_i}{\sigma_i^2 + \gamma^2} \boldsymbol{V}_i \boldsymbol{U}_i^T. \tag{2.56}$$

One can clearly see that $\frac{\sigma_i}{\sigma_i^2+\gamma^2} \neq \frac{1}{0}$, and for $\sigma_i >> \gamma$, $\frac{\sigma_i}{\sigma_i^2+\gamma^2} \approx \frac{1}{\sigma_i}$.

The constant joint velocity weighting factor γ has the undesirable characteristic of producing end-effector and kinematic constraint errors, even when the manipulator is far from any singular configuration. In other words, predefined constant velocity weighting can guarantee bounded joint velocities and smooth transitions through singular configurations. However the task performance at well-conditioned configurations is unnecessarily compromised. As an attempt to alleviate this problem, variable joint velocity weighting can be adopted such that γ is adjusted automatically to have a large value in the neighborhood of singularities and a small value away from them. Such a modification still deteriorates the motion of *all* manipulator joints proportionally, whereas in typical situations velocity damping in only a number of joints is necessary. For such reason a *weighted* DLS is proposed.

Note on the use of DLS in projectors

Some authors, e.g., [16, 25], suggest the DLS being an universal replacement for the LS solution. Independently from the accuracy, which is addressed above, it is important to underscore that in certain cases, the simple replacement is not possible. This is due to the fact that the property $\boldsymbol{J}\boldsymbol{J}^*\boldsymbol{J} = \boldsymbol{J}$ does not hold for the DLS inverse. So the projecting features of the LS solution for redundant manipulators $\boldsymbol{J}(\boldsymbol{I} - \boldsymbol{J}^*\boldsymbol{J})\boldsymbol{q}_0 = 0$, see Sec. 2.5.1, are not fulfilled.

2.3.5 Weighted DLS Method with Task Priority

In this section we introduce the modification to the DLS method with the goal of assigning different level of importance to the task coordinates (in the sense of tracking accuracy) and introducing a matrix as a damping coefficient.Consider a modified cost function

$$g(\dot{\boldsymbol{q}}, \dot{\boldsymbol{x}}) = \frac{1}{2}(\dot{\boldsymbol{x}} - \boldsymbol{J}\dot{\boldsymbol{q}})^T \boldsymbol{W}_x (\dot{\boldsymbol{x}} - \boldsymbol{J}\dot{\boldsymbol{q}}) + \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{W}_q \dot{\boldsymbol{q}}, \tag{2.57}$$

where $\boldsymbol{W}_x$ and $\boldsymbol{W}_q$ are $m \times m$ and $n \times n$ constant symmetric positive-definite weighting matrices associated with the errors in the task space and joint velocities, respectively. Setting

$$\frac{\partial g}{\partial \dot{\boldsymbol{q}}} = -\boldsymbol{J}^T \boldsymbol{W}_x (\dot{\boldsymbol{x}} - \boldsymbol{J}\dot{\boldsymbol{q}}) + \boldsymbol{W}_q \dot{\boldsymbol{q}} = \boldsymbol{0} \tag{2.58}$$

yields

$$\dot{\boldsymbol{q}} = (\boldsymbol{J}^T \boldsymbol{W}_x \boldsymbol{J} + \boldsymbol{W}_q)^{-1} \boldsymbol{J}^T \boldsymbol{W}_x \dot{\boldsymbol{x}}. \tag{2.59}$$

It can be shown that for $\boldsymbol{W}_x = \boldsymbol{I}_m$ and $\boldsymbol{W}_q = \boldsymbol{0}$, the solution (2.59) equals to the LS (pseudoinverse) solution (2.50), and for $\boldsymbol{W}_x = \boldsymbol{I}_m$ and $\boldsymbol{W}_q = \gamma^2 \boldsymbol{I}_n$ (2.59) equals the standard DLS solution of (2.54).

The matrix $\boldsymbol{W}_x$ is assigned for every task coordinate *independently* introducing different levels of importance for different coordinates. For example, we can assign different priorities to position and orientation tracking. The weightings can also be used as normalizing/scaling factors for the different variables. This is important since the variables have different units and therefore need to be normalized. The matrix $\boldsymbol{W}_q$ has the same function as the parameter γ in the DLS solution, which is damping of the joint velocities. The major advantage of $\boldsymbol{W}_q$ is that we can assign the damping coefficients for every axis independently. However, keeping the weights in $\boldsymbol{W}_q$ constant, has the same negative influence suppressing the motion as the constant γ in (2.55). To remedy this, it is possible to specify $\boldsymbol{W}_x$ and $\boldsymbol{W}_q$ as a functions of a chosen configuration index, e.g., the manipulability measure $\mu(\boldsymbol{q})$ defined later in (2.71). Since $\mu(\boldsymbol{q})$ drops to 0 at the singularities, the joint velocity weighting factor $\boldsymbol{W}_q$ can be chosen as

$$\boldsymbol{W}_q = \begin{cases} 0, & for \quad \mu(\boldsymbol{q}) > \mu_0 \\ c_0(1 - \frac{\mu}{\mu_0})^2 \boldsymbol{I}_n, & for \quad \mu(\boldsymbol{q}) \leq \mu_0, \end{cases} \tag{2.60}$$

where μ_0 is a specified threshold and c_0 is a positive constant that sets the maximum damping coefficient. In this case, damping is highest for $\mu = 0$ (at the singularity) and vanishes for $\mu > \mu_0$. More sophisticated techniques for selection of $\boldsymbol{W}_q$ based on the minimum singular value of $\boldsymbol{J}$ are possible. One disadvantage of those is the difficulty in assigning appropriate numerical values to the thresholds for the manipulability measure and the minimum singular values.

Implementation of the described method to a rotary manipulator exampled by the spherical wrist from Sec. 2.2.2 showed that it is more convenient to scale the $\boldsymbol{W}_x$ and $\boldsymbol{W}_q$ with respect to its configuration. Spherical wrists have their singularity when the first and the last rotation axes are aligned. As the configuration index, the value of the middle angle may be chosen in the same way as the manipulability measure described above. Although less general than SVD and $\mu(\boldsymbol{q})$ methods, this possibility is more intuitive in the practical designs.

2.3.6 Adjoint Jacobian Method

The adjoint Jacobian method [129] is a convenient tool for singularity handling. Denote the end-effector velocity as

$$\boldsymbol{\omega} = v\boldsymbol{u}, \tag{2.61}$$

where the unit vector $\boldsymbol{u} \in \boldsymbol{R}^3$ denotes the end-effector instantaneous-motion direction, while the scalar variable v stands for the end-effector velocity. Vector $\boldsymbol{u}$ varies in the workspace in accordance with the commanded velocity. Equation (2.43) can be rewritten as

$$\dot{\boldsymbol{q}} = \frac{v}{\det \boldsymbol{J}} \mathrm{adj} \boldsymbol{J} \boldsymbol{u}, \tag{2.62}$$

where det$\boldsymbol{J}$ and adj$\boldsymbol{J}$ denote the determinant and the adjoint matrix of the Jacobian, respectively. The term (adj$\boldsymbol{J}\boldsymbol{u}$) determines the direction of the joint velocity vector, and (det$\boldsymbol{J}$) its value. In order to overcome the singularity problem, [127, 129] used the fact that this representation effectively separates the magnitude from the direction of motion modifying (2.62) as:

$$\dot{\boldsymbol{q}} = \xi b(\text{adj}\boldsymbol{J})\boldsymbol{u}, \tag{2.63}$$

where $\xi = \pm 1$ is a sign variable, and $b \geq 0$ is a scalar value. With a proper design of ξ and b, the robot may be controlled at and around a singularity without any error in the direction, and with feasible joint velocity. Crucial for the velocity-based considerations is the design of the parameter ξ. At regular points, ξ agrees basically with the sign of the Jacobian determinant

$$\xi = \text{sign}(\det(\boldsymbol{J})). \tag{2.64}$$

Going through the singularity has the effect of (det$\boldsymbol{J}$) changing its sign. If ξ were changed accordingly, then obviously, the continuity of motion would be disrupted. To avoid this, [129] proposes keeping ξ constant while crossing the singularity. As a result, the end-effector motion direction becomes exactly opposite to the commanded direction. Thus in practice "the operator is required to stop the motion briefly after moving through the singularity; then the system adjusts the sign automatically". This approach was applied first in a space telemaintenance system, where preserving the motion accuracy prevails over the completion time and the operator fatigue. In rescue applications the design criteria are exactly contrary. This is why the adjoint Jacobian method will not be practically implemented, in spite of its potentially attractive feature of preserving the direction of motion.

2.3.7 Jacobian Transpose Method

This method minimizes the cost function

$$\begin{aligned} g(\boldsymbol{x}) &= \frac{1}{2}(\boldsymbol{x}_d - \boldsymbol{x})^T(\boldsymbol{x}_d - \boldsymbol{x}) = \\ &= \frac{1}{2}(\boldsymbol{x}_d - FK(\boldsymbol{q}))^T(\boldsymbol{x}_d - FK(\boldsymbol{q})) \end{aligned} \tag{2.65}$$

with respect to $\boldsymbol{q}$ by gradient decent

$$d\boldsymbol{q} = -a\frac{\partial g}{\partial \boldsymbol{q}} = a(\boldsymbol{x}_d - \boldsymbol{x})\frac{\partial FK(\boldsymbol{q})}{\partial \boldsymbol{q}} = a\boldsymbol{J}^T(\boldsymbol{q})d\boldsymbol{x}, \tag{2.66}$$

where $\boldsymbol{x}_d$ is the desired position and a is an arbitrary positive constant. The attractiveness of this method consists of the fact it does not include Jacobian inversion. The operating principle resembles the stiffness control algorithms. Namely, it projects the difference vector $\boldsymbol{x}_d - \boldsymbol{x}$ on those directions of $\boldsymbol{q}$ which can reduce it the most. The advantage of this method is its simple computation and numerical robustness due to no matrix inversion. However, it yields unpredictable joint configurations, and needs many interactions until it converges, especially for the small Jacobian coefficients, which is mostly the case at singular configurations. This method is reported [110, 111] to perform poorly in compare to the Jacobian inverse method. Further, whereas $\boldsymbol{J}^{\#}$ has the nice property that the solution has minimum norm of the joint velocities at every step, $\boldsymbol{J}^T$ does not have this property. Joints that are far from end-effector experience larger displacements, hence take disproportionately large steps.

2.4 Experimental Evaluation of the WDLS Method

In the following experiment we are going to validate the theoretical conclusions and compare the LS, WDLS and a combined inverse kinematics methods, with respect to their potential of damping the excessive velocities in the singularity neighborhood. To test this feature, we need to generate the trajectories driving the manipulator into the interesting region. It is important to underscore that we assume no previous knowledge about the trajectories and they can be completely arbitrary. Since the robot is equipped with a force/torque sensor on its wrist, we apply a force controller for generating the robot motion, as in a typical teach in mode. The robot is driven by hand through the singular configuration at the wrist $q_6 = 0$. The goal is to go smoothly through the singularity without generating excessive velocities. In practice, we apply one successful IK method for driving the real robot, and the alternative algorithms are solved in parallel and the joint trajectories stored for comparison.

Traversing through Singularity

We compare the LS and the WDLS methods. For simplicity, we investigate only the last three joints: 5, 6 and 7, see Fig. 2.6. We define the weighting matrix $\boldsymbol{W}_x$ in (2.59) as

$$\boldsymbol{W}_x = \begin{bmatrix} \boldsymbol{V}_p & 0 & 0 \\ 0 & \boldsymbol{V}_\omega & 0 \\ 0 & 0 & V_\theta \end{bmatrix}, \tag{2.67}$$

where $\boldsymbol{V}_p, \boldsymbol{V}_\omega, V_\theta$ are weighting matrices for Cartesian position, orientation and θ angle. We assign equal priorities to the task coordinates $\boldsymbol{V}_p = \mathrm{diag}(1,1,1)$, $\boldsymbol{V}_\omega = \mathrm{diag}(1,1,1)$ and $V_\theta = 1$. We would like to introduce the damping only at the vicinity of the singularity, and only to the joints 5 and 7, so that we set the damping matrix $\boldsymbol{W}_q$ as follows:

$$\boldsymbol{W}_q = k\ \mathrm{diag}(0,0,0,0,1,0,1), \tag{2.68}$$

where k equals zero for $q_6 > 0.3$ rad and $k = (1 - \frac{q_6}{0.3})^2$ otherwise.

Fig. 2.7 illustrates the joint trajectories obtained with the LS method (left) and WDLS method (right) for the joints q_5 (top), q_6 (middle) and q_7 (down). The corresponding joint velocities are shown in in Fig. 2.8. As we can see, in Fig. 2.7/left, the sixth joint is bumped away from singularity $q_6 = 0$, without going through. As the result, the joints 5 and 7 must travel an angle of almost $\pi/2$ with a high velocity shown in Fig. 2.8/left. On the other hand, the WDLS method offers a smooth transition through the singularity, see Fig. 2.7/right with small velocities in Fig. 2.8/right.

Error Introduced by Damping

In the consequent experiment, we would like to find out the amount of error introduced by damping in WDLS method. Fig. 2.9 illustrates the Cartesian reconstruction errors as a difference between the trajectories generated by the LS and WDLS algorithms. The top two figures show the translational and rotational velocity difference, respectively. The factor k and the angle q_6 are given as a reference in the lower figures. Excluding the peaks, for which the LS methods produces not feasible high velocities, the difference is small and is not perceived in practice by the human operators.

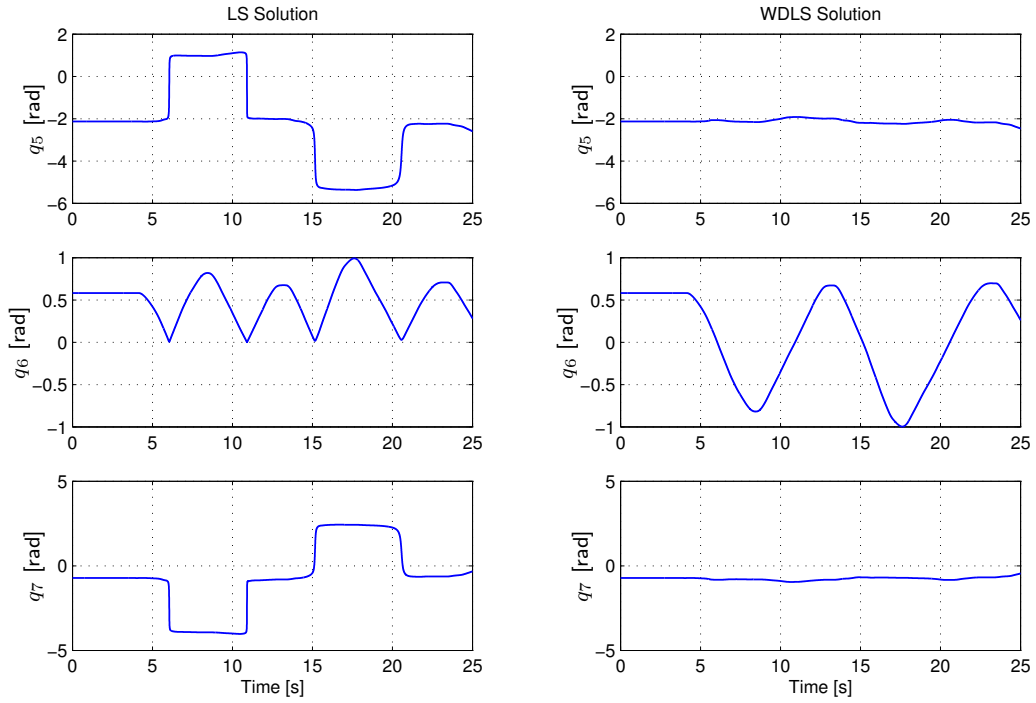


Figure 2.7: Joint angles for LS and WDLS methods: singular configuration at $q_6 = 0$

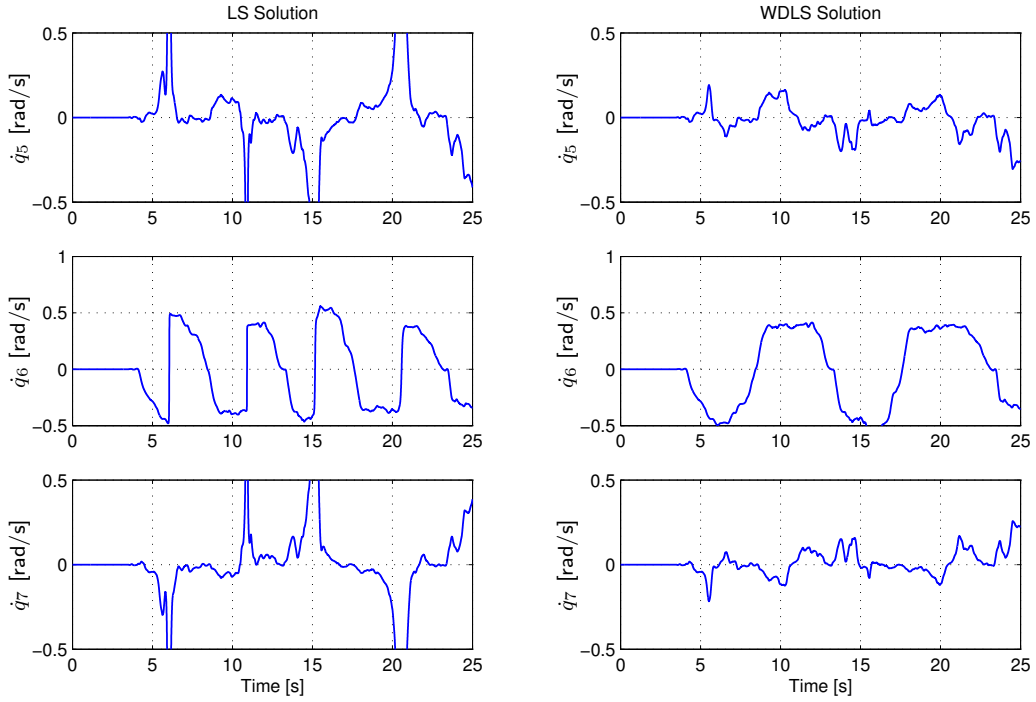


Figure 2.8: Joint velocities for LS and WDLS methods

Weighing Properties of WDLS Method

In order to show the features of the task space error weighing with WDLS method, we solve the IK for the following 3 sets of weighing factors:

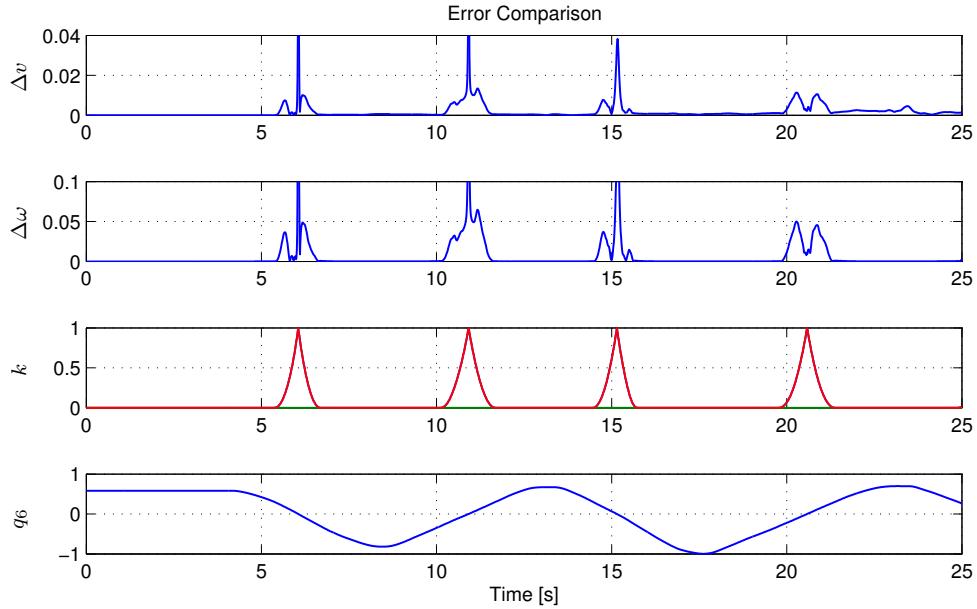


Figure 2.9: Error Comparison between LS and WDLS algorithms, $\Delta v = |v_{ls} - v_{wdls}|, \Delta\omega = |\omega_{ls} - \omega_{wdls}|$

A: Position has a ten times higher priority than the orientation and elbow angle.
$\boldsymbol{W}_x = 10^3 \text{diag}(10, 10, 10, 1, 1, 1, 1)$

B: Orientation has a higher priority than the position, the weighing for the elbow angle is 30% of the previous one.
$\boldsymbol{W}_x = 10^3 \text{diag}(1, 1, 1, 10, 10, 10, 0.3)$

C: As a reference, we investigate a mixed IK solution, where the first 4 joint angles are found using the closed form solution from Sec. 2.2.1 and the wrist joint angles are found using WDLS method. Within the current framework, this settings are equivalent to
$\boldsymbol{W}_x = \text{diag}(1, 1, 1, 1, 1, 1, \infty)$

The joint weighing matrix is in all experiments constant $\boldsymbol{W}_q = 0.1 \text{ diag}(0, 0, 0, 0, 1, 0, 1)$. Those weighing matrices are designed like that to show the highest priority in position tracking (settings A), orientation tracking (settings B) and elbow tracking (settings C). The results are illustrated in Fig. 2.10, where we can see the tracking errors for translational velocity $\Delta\boldsymbol{v}$, rotational velocity $\Delta\boldsymbol{\omega}$ and elbow angle $\Delta\theta$. As a reference, in the lowest row, the norm of joint velocities is given.

The experimental findings confirm the outstanding performance of the WDLS method. We conclude, that it is a powerful tool for solving the inverse kinematics problem. Within one unform framework, the problems of singularity traversing and weighing the task space error are solved.

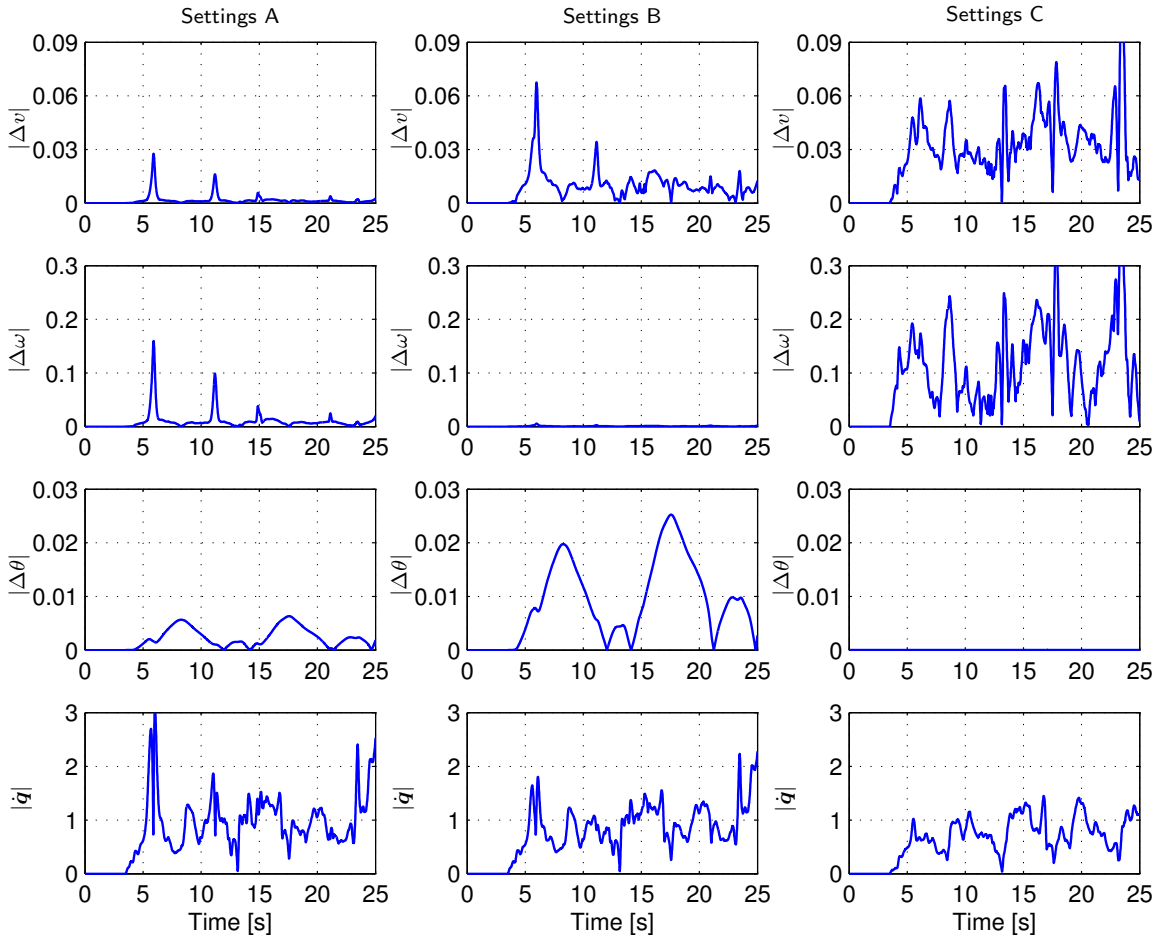


Figure 2.10: Weighing properties of the WDLS method: reconstruction error for different priority matrices

2.5 Redundancy Resolution

A robot manipulator is redundant when its number of degrees of freedom n is greater than the dimension of the workspace m. The difference $r = n - m$ represents the degree of redundancy. Consequently, the Jacobian is non square and cannot be directly inverted and the inverse kinematic model gives an infinite number of solutions. In teleoperation scenario, the end-effector trajectory is generated by the human operator and it does not provide the complete information about the robot pose. In the case of the 7 DoF manipulator, the information regarding the elbow angle θ is simply missing. There are two possible solutions to this problem. One can sense the motion of the user elbow using various kind of trackers and send this information to the telemanipulator together with the end-effector coordinates. Applying the IK solution from Sec. 2.2.1 and the extended Jacobian method from Sec. 2.1.2 one get the unique description of the manipulator configuration. The second option is that, the self motion is governed by a local controller according to the specific predefined criteria. This possibility will be discussed more in detail in the next section.

2.5.1 Pseudoinverse with Optimization Criteria

The redundancy may be exploited to optimize a secondary performance criteria. The pseudoinverse (2.50) has an interesting property that the matrix $(\boldsymbol{I} - \boldsymbol{J}^{\#}\boldsymbol{J})$ projects an arbitrary vector $\dot{\boldsymbol{q}}_0$ onto the null space $\mathcal{N}(\boldsymbol{J})$ of the manipulator Jacobian such that the secondary task does not affect the primary task of the end-effector. Therefore, for any arbitrary vector $\dot{\boldsymbol{q}}_0$, $\boldsymbol{J}(\boldsymbol{I} - \boldsymbol{J}^{\#}\boldsymbol{J})\boldsymbol{q}_0 = \boldsymbol{0}$. This means one can set

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^{\#}\dot{\boldsymbol{x}} + (\boldsymbol{I} - \boldsymbol{J}^{\#}\boldsymbol{J})\dot{\boldsymbol{q}}_0 \tag{2.69}$$

and still obtain a solution that minimizes $\|\dot{\boldsymbol{q}} - \boldsymbol{J}^{\#}\dot{\boldsymbol{x}}\|$. The second term defines the self motion of the telemanipulator, so by suitably choosing $\boldsymbol{q}_0$ one can achieve secondary goals like minimizing the norm of the joint velocities, avoiding obstacles, singular configurations or joint limits, or minimizing driving joint torques, etc. The objective function w to be optimized may be expressed as a function of the joint coordinates $\boldsymbol{q}$. The vector $\dot{\boldsymbol{q}}_0$ is obtained with the derivative of the objective function

$$\dot{\boldsymbol{q}}_0 = k(\frac{\partial w_i}{\partial \boldsymbol{q}})^T, \tag{2.70}$$

where $k > 0$ is a constant. There exist a large body of literature concentrating on optimizing various performance critera [21, 88, 89, 144, 146]. Usual objective functions are:

- the manipulability measure which equals zero at singular configurations, and thus may be used to avoid singularities

$$w(\boldsymbol{q}) = \sqrt{\det(\boldsymbol{J}\boldsymbol{J}^T)}, \tag{2.71}$$

- the distance from mechanical joint limits

$$w(\boldsymbol{q}) = \sum_{i=1}^{n}(\frac{q_i - \overline{q}_i}{q_{iM} - q_{im}})^2, \tag{2.72}$$

  where q_{iM}, q_{im} denotes maximum (minimum) limit for q_i and $\overline{q}_i$ the middle of the joint range;

- the distance to an obstacle or the other arm;

- minimum gravity compensation torque defined as

$$w(\boldsymbol{q}) = ||\boldsymbol{G}(q)||, \tag{2.73}$$

  where $\boldsymbol{G}(\boldsymbol{q})$ is the gravitional part of the robot dynamic model.

In tasks requiring high forces, the redundancy may be used also to increase the natural stiffness of the manipulator by reconfiguring the mechanical structure.

The null space method can be also used to assign different priorities to different tasks. However, little attention has been paid to *simultaneous* optimization of *multiple* performance criteria [29, 99], where particular degrees of redundancy are assigned to particular additional tasks. There is even less literature available on optimizing multiple criteria with 1 DoF of redundancy. In such case the multiple performance criteria must be weighted according to the significance function

$$w(\boldsymbol{q}) = \sum_{i=1}^{s} c_i w_i(\boldsymbol{q}) \ , \tag{2.74}$$

where $\overline{w}_i(\boldsymbol{q})$ is a normalized objective function of the i-th performance criterion and c_i is a weighting factor, usually configuration dependent. For example, for optimization of two objective functions "arm to arm collision avoidance" and "joint limits avoidance", the distance to collision and the distance to joint limit may be taken as the weighing factor. For the experimental results the reader is asked to see Sec. 6.5.

The general problem of this method is that it produces a noncyclic and/or non-repeatable motions. This means that the closed task space trajectories result in not necessarily closed joint trajectories. In other words, the pose of the robot is governed by the optimization criteria and cannot be controlled in a direct way. This drawback can be overcame with the extended kinematics method described below.

2.5.2 Extended Kinematics Method

In the extended kinematics approach, proposed in [8] and [106], an appropriate joint motion is selected from this infinite set in a way that the manipulator performs an additional user-specified task. To achieve that, the user selects a set of $r = n - m$ linearly independent kinematic functions $\Psi(\cdot)$ in Cartesian or joint space

$$\boldsymbol{x}_c = \Psi(\boldsymbol{q}), \tag{2.75}$$

where the vector $\boldsymbol{x}_c$ is an $(r \times 1)$ reference vector. These functions can represent either physical constraints of the robot or constrains related to the environment, and can be any arbitrary function of the physical configuration and the geometrical parameters of the manipulator. The user also specifies the desired variation of the kinematic functions denoted by the $\boldsymbol{x}_c(t)$, and thus defines the *additional task* that will be performed utilizing the manipulator redundancy. Differentiating equation (2.75) with respect to time gives

$$\dot{\boldsymbol{x}}_c = \boldsymbol{J}_\Psi(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{2.76}$$

where

$$\boldsymbol{J}_\Psi(\dot{\boldsymbol{q}}) = \frac{\partial \Psi(\cdot)}{\partial \boldsymbol{q}} \tag{2.77}$$

is the $(r \times n)$ Jacobian matrix of $\Psi(\boldsymbol{q})$. Combining this equation with the kinematic model we obtain an $(n \times n)$ extended Jacobian matrix $\boldsymbol{J}_E$ and new task space vector $\boldsymbol{x}_E$ such that:

$$\dot{\boldsymbol{x}}_E = \begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{x}}_c \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}(\boldsymbol{q}) \\ \boldsymbol{J}_\Psi(\boldsymbol{q}) \end{bmatrix} \dot{\boldsymbol{q}} \tag{2.78}$$

The additional task is completely general and unrestricted to any particular application, and can be formulated either as the equality constraint or the inequality constraint. For the concrete implementation of the extended kinematics method the reader is asked to see the Sec. 3.3, where the complete task space is defined after having defined the suitable orientation representation.

2.6 Summary

This chapter was devoted to the kinematic design and control of the anthropomorphic 7 DoF telemanipulator. The kinematic model was presented including standard Denavit-Hartenberg parameters and a forward kinematics solution was given. The self motion of the manipulator was parameterized in a physically meaningful way in order to produce anthropomorphic type of motions. The corresponding solution to the inverse kinematics problem was also given analytically.

Further, singularity-robust algorithms for inverse kinematics were extensively studied. The purpose was to mimic the human motion which requires stable operation in the neighborhood of the singular configurations. A Weighted Damped Least Square method was introduced that allows for both singularity traversing by limiting the joint velocities with simultaneous weighing of the tracking error. Experimental comparison of this method to other inverse kinematics algorithm showed its superior performance.

Finally, fundamentals of the redundancy resolution with multiple optimization criteria were summarized and the extended kinematics method was formulated.

3 Task Space Definition and Orientation Error

In majority of robotic applications, the task the robot must perform is given in terms of the desired trajectory of the end-effector frame. The correct execution of the end-effector motion is entrusted to the motion controller which shall provide appropriate commands to the joint actuators such that the end-effector follows the desired trajectory as exact as possible. Therefore it is necessary to define the position and orientation of the end-effector as well as their deviation from the desired path. It is straightforward to define the position of end-effector and position error. To define the orientation and orientation error of the end-effector is not a trivial task. The rotational motion is naturally parameterized on the differential level with the rotational velocity $\boldsymbol{\omega}$, which is consistent (parallel) to the driving torque. However its integral of $\boldsymbol{\pi} = \int \boldsymbol{\omega} \mathrm{d}t$ does not have any physical meaning. The $\boldsymbol{\pi}$ parameter is considered in analytical mechanics a "quasi coordinate"[1] and according to [13, 37] cannot be used in orientation control. Due to this deficiency, in the majority of teleoperation systems, it is the velocity that is commanded to the teleoperator and not the position/orientation. On the other hand, it has been shown that in case of significant time delay, position/orientation feed forward is essential to remove the position drift, see [49, 122]. A proper definition of the orientation error is also crucial for interaction control, namely the definition of the stiffness matrix.

In this chapter considerations regarding the orientation representation will be given. This issue is crucial for mutual kinematic connection of robotic manipulators of dissimilar kinematics, as well a for the proper definition of mechanical compliance. Unit quaternion representation will be described, which is well suited for both purposes. The usefulness of this representation over classical Euler angles will be shown in a simulation of an interaction scenario. Finally, the complete task space definition, applied for the controll of the 7 DoF anthropomorphic telemanipulator will be given.

3.1 Problem of Dissimilar Kinematics in Teleoperation

In teleoperation scenarios, tasks are performed by a mechanical manipulator (slave) controlled remotely by a human operator provided with an arm/hand controller with force feedback (haptic interface or master), see Sec. 1.1. The movement transmission between the two robots may be achieved using generally two approaches: joint to joint control or Cartesian space control with force feedback. Joint to joint control is limited to devices with the same kinematics. It allows only for relatively simple control strategies. Impedance/stiffness or shared control are not possible. Besides, redundant slave manipulators require redundant master robots, which complicates the design and is not necessarily a good solution for force reflection purposes. So in general, the kinematical structure of both master/slave manipulators are different, and the devices communicate in task, mostly Cartesian, space. That is why both devices need to be equipped with an universal kinematic interface, in-

[1] quasi coordinate is valid only at differential level

dependent from the device structure. There are very few works addressing this subject directly. Usually, the manipulators work in somehow limited workspace, so that the full spatial immersion is not achieved or limited [57, 143]. Moreover, due to the unmatched workspace of the two coupled manipulators, indexing or shifting is used [72, 128]. The resulting system is neither transparent nor intuitive to the user, and is not acceptable in the current scenario. As long as the transformation of positions or translational velocities between the devices is trivial, transforming the orientation is a much more complex problem. Within the scope of this work, experiments with 2 DoF [48, 49, 122], 6 DoF [119, 121] and 10 DoF [1, 15, 98] input devices were performed, and the experience gained confirm the necessity of the proper orientation representation.

3.2 Orientation Representation and Orientation Error

The first widely accepted method in robot kinematics is based on Denavit-Hartenberg notation of spatial mechanisms [33, 97, 118]. The orientation of a rigid body that rotates freely in space is described uniquely by a (3×3) rotation matrix $\boldsymbol{R}$ with positive determinant, so the configuration manifold of this motion is the special orthogonal group SO(3), and the time derivative of the matrix $\boldsymbol{R}$ is

$$\dot{\boldsymbol{R}} = \boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{R}, \tag{3.1}$$

where $\boldsymbol{S}(\cdot)$ is the skew-symmetric matrix

$$\boldsymbol{S}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}, \tag{3.2}$$

and $\boldsymbol{\omega} = [\omega_x \quad \omega_y \quad \omega_z]^{\mathrm{T}}$ is the rotational velocity. Since $\boldsymbol{R} \in \mathrm{SO}(3)$ and $\boldsymbol{\omega} \in \boldsymbol{R}^3$, there exists no direct relation corresponding to conventional differential-integral operations. Researchers in robot kinematics tried alternative methods in order to represent rigid body transformations based on concepts introduced by mathematicians and physicists such as screw theory [40, 59, 84], Lie algebra [9, 10, 42, 43], Epsilon algebra [134] matrix exponentials [103] or unit quaternions [17, 28, 138, 145]. The analysis presented in [41, 67] proved the superior computational efficiency and compactness of the unit quaternions in compare to the other methods. In the following sections a comparative analysis of the three parameter representations and the unit quaternion is presented.

3.2.1 Euler Angles

One of the classical forms is obtained by using a set of Euler angles $\boldsymbol{\varphi} = [\alpha \ \beta \ \gamma]^T$. Consider the rotation matrix expressing the elementary rotation about one of the coordinate axes as a function of a single angle. Those three parameters denote three sequential elementary rotation angles. Then, a generic rotation matrix is obtained by three sequential elementary rotations while guaranteeing that any of two successive rotations are not performed about one parallel axis. It implies that only 12 sets of angles are allowed out of all 27 possible combinations. It can be divided into two groups according to the sequence of the successive rotations: *Eulerian* and *Cardanian*. The Eulerian type involves repetition of rotations

about one particular axis, e.g.,: XYX, XZX, YXY. The Cardanian type is characterized by the rotations about all three axes, e.g.,: XYZ, XZY, YZX. Even though the Cardanian type is different from the Eulerian type in terms of the combination of the rotations, they both use very similar approach to compute the orientation angles. One may treat the Cardanian sequences as a subset of the Eulerian. These are often called altogether as Euler angles representation. For instance, the XYZ representation of orientation in terms of Euler angles is described by the rotation matrix

$$\begin{aligned} \boldsymbol{R}(\boldsymbol{\varphi}) &= \boldsymbol{R}_x(\alpha)\boldsymbol{R}_y(\beta)\boldsymbol{R}_z(\gamma) = \\ &= \begin{bmatrix} c_\beta c_\gamma & -c_\beta s_\gamma & s_\beta \\ s_\alpha s_\beta c_\gamma + c_\alpha s_\gamma & -s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & -s_\alpha c_\beta \\ -c_\alpha s_\beta c_\gamma + s_\alpha s_\gamma & c_\alpha s_\beta s_\gamma + s_\alpha c_\gamma & c_\alpha c_\beta \end{bmatrix}, \end{aligned} \tag{3.3}$$

where $\boldsymbol{R}_x$, $\boldsymbol{R}_y$,$\boldsymbol{R}_z$ are the elementary rotation matrices about the three coordinate axes, and the notations c_ϕ and s_ϕ are the abbreviations for $\cos\phi$ and $\sin\phi$, respectively. The angular velocity $\boldsymbol{\omega}$ and the time derivative of Euler angels $\dot{\boldsymbol{\varphi}}$ are connected by the transformation matrix $\boldsymbol{T}$, relating the vectors expressed in orthogonal Cartesian space with vectors expressed in non-orthogonal spaces of Euler angles.

$$\boldsymbol{\omega} = \begin{bmatrix} 1 & 0 & s_\beta \\ 0 & c_\alpha & -s_\alpha c_\beta \\ 0 & c_\alpha & c_\alpha c_\beta \end{bmatrix} \dot{\boldsymbol{\varphi}} = \boldsymbol{T}(\boldsymbol{\varphi})\dot{\boldsymbol{\varphi}}, \tag{3.4}$$

The most natural way to define an orientation error is to take the difference of representation parameters of orientation between the desired and actual end-effector. When using Euler angles, the orientation error is expressed as follows:

$$\boldsymbol{\varphi}_{de} = \boldsymbol{\varphi}_d - \boldsymbol{\varphi}_e \tag{3.5}$$

where $\boldsymbol{\varphi}_d$ and $\boldsymbol{\varphi}_e$ are Euler angles extracted respectively from the rotation matrices $\boldsymbol{R}_d$ and $\boldsymbol{R}_e$ describing the orientation of desired end-effector frame (Σ_d) and actual one (Σ_e). When it comes to orientation error using Euler angles, there are two disadvantages: representation singularity and task geometric inconsistency.

Representation Singularity

Such orientation and error representation using Euler angles has been widely applied for the majority of conventional robots. However, using minimal representations including Euler angles for angular displacement that use only three parameters suffer from representation singularities. Algebraically representation singularities arise when the second rotation angle β is 0 or π for Eulerian type and $\pm\pi/2$ for Cardanian type. The problem appears when a physical vector (ω, forces, torques) has to be expressed in Euler angles space, since then the inverse of $\boldsymbol{T}(\boldsymbol{\varphi})$ must be used, e.g., $\dot{\boldsymbol{\varphi}} = \boldsymbol{T}^{-1}(\boldsymbol{\varphi})\boldsymbol{\omega}$. One can see that $\det(\boldsymbol{T}(\boldsymbol{\varphi})) = 0$ for $\beta = \pm\pi/2$. This is a parasite representation singularity, which has to be taken care of in TO scenario.

Another usual problem with the Euler angles is that, as far as the transformation $\boldsymbol{R}(\boldsymbol{\varphi})$ in (3.3) is unique, the inverse transformation has always 2 solutions (see Sec. 2.2.1, solution for the wrist joint angles). Moreover, it is indefinite at $\beta = \pm\pi/2$, which means that without limiting the workspace, it is impossible to obtain a unique solution.

There are three solutions to remedy those drawbacks:

1. Choose the representation singularity together with kinematic singularity and apply a singularity avoidance kinematic method. The disadvantage of this method is that large portion of the workspace (around the singularity) is lost.

2. Switch between different representations, e.q. of Eulerian and Cardanian type, which unfortunately introduces artificial discontinuities and additional problems with the switching conditioning.

3. Define the orientation error together with the desired impedance not in the task frame, but in the end-effector frame. The advantage of this approach is that the relative angular displacement in the end-effector frame is small, and a singularity-free Euler angles set may be found.

None of the suggested choices solves the next related problem: the task geometric inconsistency.

Task Geometric Inconsistency

From a physical point of view, the meaning of $\boldsymbol{\omega}$ is more intuitive than that of $\dot{\boldsymbol{\varphi}}$. The three components of $\boldsymbol{\omega}$ represent the components of the end-effector angular velocity with respect to the base frame. On the other hand, the three elements of $\dot{\boldsymbol{\varphi}}$ represent nonorthogonal components of angular velocity defined with respect to the axes of a frame, which varies together with the orientation of the end-effector. The clear advantage of $\dot{\boldsymbol{\varphi}}$ is that its integral over time $\boldsymbol{\varphi}$ has clear physical meaning, while the integral of $\boldsymbol{\omega}$ does not provide a physical interpretation at all. However, when it comes to correlate the orientation error with the resulting torque, it is not as straightforward as for translational error. For example, consider impedance control described more in detail in Sec. 4.2.2. The impedance equation is written as

$$\boldsymbol{M}_p\ddot{\boldsymbol{x}}_{de} + \boldsymbol{D}_p\dot{\boldsymbol{x}}_{de} + \boldsymbol{K}_p\boldsymbol{x}_{de} = \boldsymbol{f} \tag{3.6}$$

with translational error $\boldsymbol{x}_{de} = \boldsymbol{x}_d - \boldsymbol{x}_e$ and a translational contact force $\boldsymbol{f}$. Subscript 'd' and 'e' denote involving about desired and actual posture of the end-effector, respectively; $\boldsymbol{M}_p$ is a desired equivalent mass, $\boldsymbol{D}_p$ is a damping matrix and $\boldsymbol{K}_p$ a stiffness matrix. From the above equation, it is noted that the resulting translational contact force $\boldsymbol{f}$ is parallel to the translational error and its derivatives, which allows the resulting contact force to directly correlate the corresponding translational error $\boldsymbol{x}_{de}$ and its derivatives. Such property preserves the linearity of the mass - spring - damper system, and enables to design control law intuitively. Now consider impedance equation for orientation (rotation) error using Euler angles.

$$\boldsymbol{M}_o\ddot{\boldsymbol{\varphi}}_{de} + \boldsymbol{D}_o\dot{\boldsymbol{\varphi}}_{de} + \boldsymbol{K}_o\boldsymbol{\varphi}_{de} = \boldsymbol{T}^{-1}(\boldsymbol{\varphi})\boldsymbol{\mu} \tag{3.7}$$

where $\boldsymbol{M}_o$, $\boldsymbol{D}_o$, and $\boldsymbol{K}_o$ is a virtual mass, damping, and stiffness matrix for orientation, respectively, $\boldsymbol{\mu}$ denotes the resulting contact moment. Unlikely to the translational motion, the direction of resulting Cartesian contact moment is parallel neither to the vector of the orientation error $\boldsymbol{\varphi}_{de}$ nor to its derivative vectors. These can be correlated only by means of transformation matrix $\boldsymbol{T}(\boldsymbol{\varphi})$ in (3.4) and the task geometric consistency is lost. Furthermore, the resulting Cartesian contact moment is not always possible to be transformed into

the operational space, in which Euler angles are defined, since the transformation matrix $\boldsymbol{T}(\boldsymbol{\varphi})$ suffers from the representation singularities as noticed in the previous section.

3.2.2 Orientation Error with Nonminimal Representation

To overcome the drawbacks of the minimal representation such as representation singularity and task geometric inconsistency one needs to resort to nonminimal representations. To formulate the orientation error using nonminimal representation it is necessary to revise the orientation error in the Cartesian space. Although the orientation error treated here is not new, its physical meaning can be hardly found. Hence it is a useful attempt to explain the physical meaning of the orientation error.
First, consider two frames in which one frame is rotated by an angle α about only one axis, see Fig. 3.1A. How can the amount of orientation error and its direction be defined?

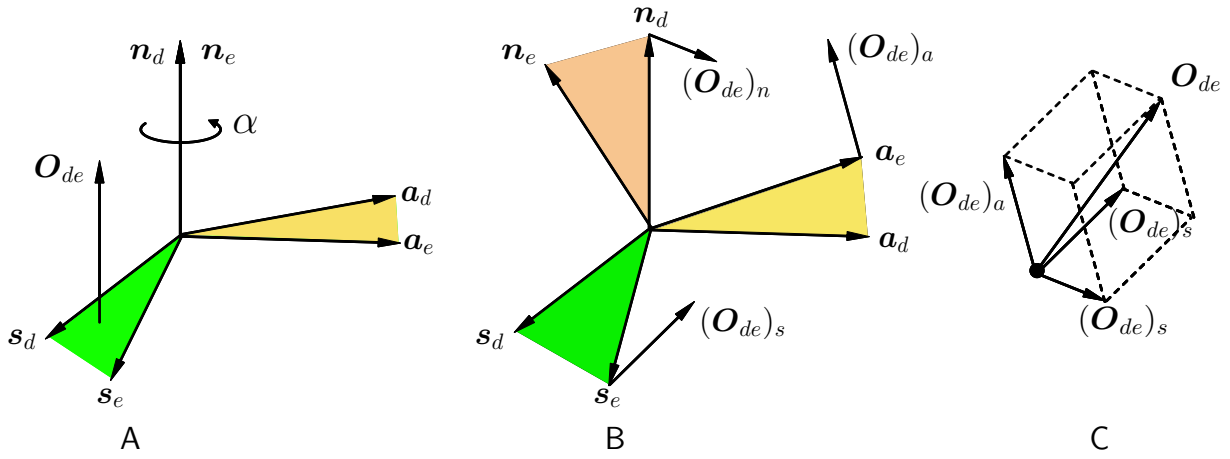


Figure 3.1: Orientation error: A - misalignment about one axis, B - misalignment about three axes, C - overall orientation error corresponding to B

There are several possible choices. One of the intuitive choices is to denote the amount of orientation error as the sum of shaded area of two triangles generated by the misalignment by the angle α. Letting the direction of orientation error be the same as the rotation angle, which is perpendicular to both shaded triangles, the orientation error is compactly expressed with cross product of two pairs of unit direction vectors:

$$\boldsymbol{O}_{de} = \frac{1}{2}\left(\boldsymbol{s}_d \times \boldsymbol{s}_e + \boldsymbol{a}_d \times \boldsymbol{a}_e\right). \tag{3.8}$$

The length of the vector $\boldsymbol{O}_{de}$ corresponds to the shaded area, and its direction is defined as the sum of the two vector products. More generally, consider two frames that are arbitrary deviated from each other, as illustrated in Fig. 3.1B. Similarly to the orientation error with respect to one axis, we can build three orientation error vectors whose lengths are the same with the area of the corresponding plane. In this case, the overall orientation error is a vector composed of those three vectors:

$$\boldsymbol{O}_{de} = \frac{1}{2}\left(\boldsymbol{n}_d \times \boldsymbol{n}_e + \boldsymbol{s}_d \times \boldsymbol{s}_e + \boldsymbol{a}_d \times \boldsymbol{a}_e\right) \tag{3.9}$$

It should be noted that rotation angle belongs to the interval $[-\pi, \pi]$.
Such concept is directly used in *angle/axis* representation for orientation error, where the orientation error is expressed as follows:

$$\boldsymbol{O}_{de} = \sin(\theta_{de})\boldsymbol{r}_{de}\,, \qquad \theta \in [-\pi, \pi]. \tag{3.10}$$

The symbol $\boldsymbol{O}_{de}$ is orientation error with respect to the end-effector frame, $\boldsymbol{r}_{de}$ is the unit vector which denotes the direction of the orientation error in (3.9), and $\sin(\theta_{de})$ corresponds to the length of $\boldsymbol{O}_{de}$ and further to the sum of area of the three triangles in Fig. 3.1B.

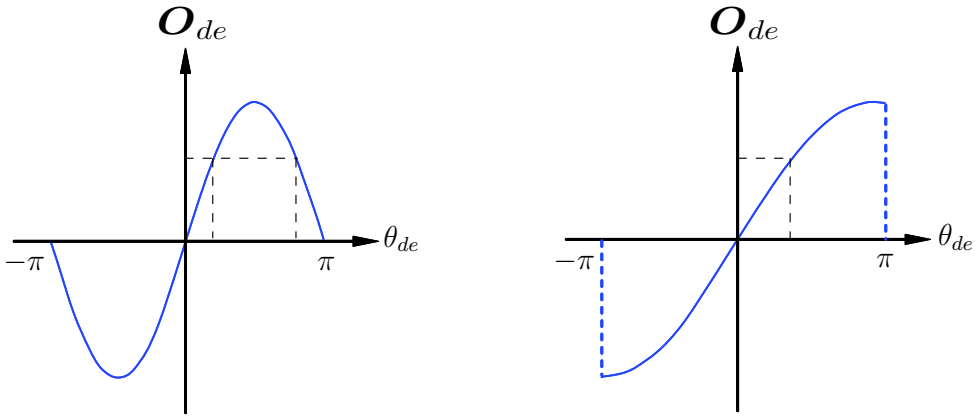


Figure 3.2: Orientation error $\boldsymbol{O}_{de}$ vs. angular displacement θ_{de}

Taking the whole interval of rotation angle into account makes the singularity still possible to occur. Furthermore, it is possible that two different angular displacement can lead to the same orientation error, see Fig. 3.2/left . Such drawbacks can be overcome by modifying the orientation error (3.10) such that it becomes a one-to-one mapping over the whole possible angular displacements as follows:

$$\boldsymbol{O}_{de} = \sin\left(\frac{\theta_{de}}{2}\right)\boldsymbol{r}_{de}\,, \qquad \theta_{de} \in [-\pi, \pi]. \tag{3.11}$$

As shown in Fig. 3.2/right, any angular displacement corresponds to only one orientation error so that only zero angular displacement leads to zero orientation error.
The above modification leads to the singularity free orientation representation, the so-called *unit quaternion* representation. The orientation error (3.11) denotes the vector part of the quaternion. To make it unitary norm, the scalar part of quaternion must be $\cos(\theta_{de}/2)$. This way we have defined the orientation error in geometric consistent way and express the orientation error using *unit quaternion* which doesn't suffer from representation singularities.

If we denote the complete *unit quaternion* as $\mathcal{Q} = \{\eta, \boldsymbol{\epsilon}\}$, then it is derived as follows:

$$\eta_{de} = \frac{1}{2}\sqrt{r_{11} + r_{22} + r_{33} + 1} \tag{3.12}$$

$$\boldsymbol{\epsilon}_{de} = \frac{1}{2}\begin{bmatrix} \mathrm{sign}(r_{32} - r_{23})\sqrt{r_{11} - r_{22} - r_{33} + 1} \\ \mathrm{sign}(r_{32} - r_{23})\sqrt{r_{22} - r_{33} - r_{11} + 1} \\ \mathrm{sign}(r_{32} - r_{23})\sqrt{r_{33} - r_{11} - r_{22} + 1} \end{bmatrix} \tag{3.13}$$

where r_{ij} denotes a i-th row, j-th column element of the corresponding rotation matrix $\boldsymbol{R}_d$.
The relationship between the time derivative of the *unit quaternion* and the body angular velocity $\boldsymbol{\omega}$ is established by the so-called *quaternion propagation* rule:

$$\dot{\mathcal{Q}} = \begin{bmatrix} \dot{\eta} \\ \dot{\boldsymbol{\epsilon}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\boldsymbol{\epsilon}^{\mathrm{T}} \\ \eta \boldsymbol{I} - \boldsymbol{S}(\boldsymbol{\epsilon}) \end{bmatrix} \boldsymbol{\omega} = \boldsymbol{T}(\eta, \boldsymbol{\epsilon}) \boldsymbol{\omega} \tag{3.14}$$

with

$$\boldsymbol{T}(\eta, \boldsymbol{\epsilon}) = \frac{1}{2} \begin{bmatrix} -\boldsymbol{\epsilon}^{\mathrm{T}} \\ \eta \boldsymbol{I} - \boldsymbol{S}(\boldsymbol{\epsilon}) \end{bmatrix}. \tag{3.15}$$

Reversely, the angular velocity of the end-effector in the base coordinates is written as

$$\boldsymbol{\omega} = 2\boldsymbol{T}^T(\eta, \boldsymbol{\epsilon}) \dot{\mathcal{Q}}. \tag{3.16}$$

The matrix $\boldsymbol{T}(\eta, \boldsymbol{\epsilon})$ has the property

$$\boldsymbol{T}^T(\eta, \boldsymbol{\epsilon}) \boldsymbol{T}(\eta, \boldsymbol{\epsilon}) = \boldsymbol{I}_3; \quad \boldsymbol{T}^T(\eta, \boldsymbol{\epsilon}) \mathcal{Q} = 0, \tag{3.17}$$

so that it is free from representation singularities.

3.2.3 Euler Angles vs. Unit Quaternion: a Simulation Example

To visualize the problem of the representation singularity of the Euler angles, we perform a simulational experiment. We apply two concepts of the orientation error to the impedance control method (see Sec. 4.2.2 for details). Euler angles and the unit quaternion are used to describe the rotational motion of the manipulator. The impedance control modifies the desired manipulator trajectory (subscript 'd') according to the specified dynamics giving as the result the compliant trajectory (subscript 'c'). The desired dynamics is defined by the equation

$$\boldsymbol{M}_{Eo} \ddot{\boldsymbol{\varphi}}_c + \boldsymbol{D}_{Eo} \dot{\boldsymbol{\varphi}}_c + \boldsymbol{K}_{Eo} (\boldsymbol{\varphi}_d - \boldsymbol{\varphi}_c) = \boldsymbol{T}^{-1}(\boldsymbol{\varphi}) \boldsymbol{\mu} \tag{3.18}$$

in case of the Euler angles, and

$$\boldsymbol{M}_{Qo} \dot{\boldsymbol{\omega}}_c + \boldsymbol{D}_{Qo} \boldsymbol{\omega}_c + \boldsymbol{K}'_{Qo} \boldsymbol{\epsilon}_c = \boldsymbol{\mu} \tag{3.19}$$

in case of the unit quaternion, where $\boldsymbol{M}$, $\boldsymbol{D}$ and $\boldsymbol{K}$ are mass, damping and stiffness matrices, and $\boldsymbol{\mu}$ is the measured torque vector. In teleoperation, the above equations need to be solved for every $\boldsymbol{\mu}$ and for every manipulator pose. Consider the desired Euler angles trajectory

$$\boldsymbol{\varphi}_d = [0 \ 1 \ 0]^T \sin(\frac{\pi}{5} t) \tag{3.20}$$

which is to pass through the singular configuration $\beta = \pm\frac{\pi}{2}$, see Fig. 3.3A. During the motion, a small constant Cartesian torque $\boldsymbol{\mu} = [1 \ 0 \ 0]^T$ Nm is sensed. At the representation singularity, the torque represented in the Euler angles space goes to infinity as the $\det(\boldsymbol{T}(\boldsymbol{\varphi}))$ goes to zero, see Fig. 3.3B. This yields in case of Euler angles an unstable motion, expressed in terms Euler angles (Fig. 3.3F) and unit quaternion (Fig. 3.3D). The quaternion-based impedance control offers stable motion with smooth transition through the representation singularity, as seen in Fig. 3.3C and D.

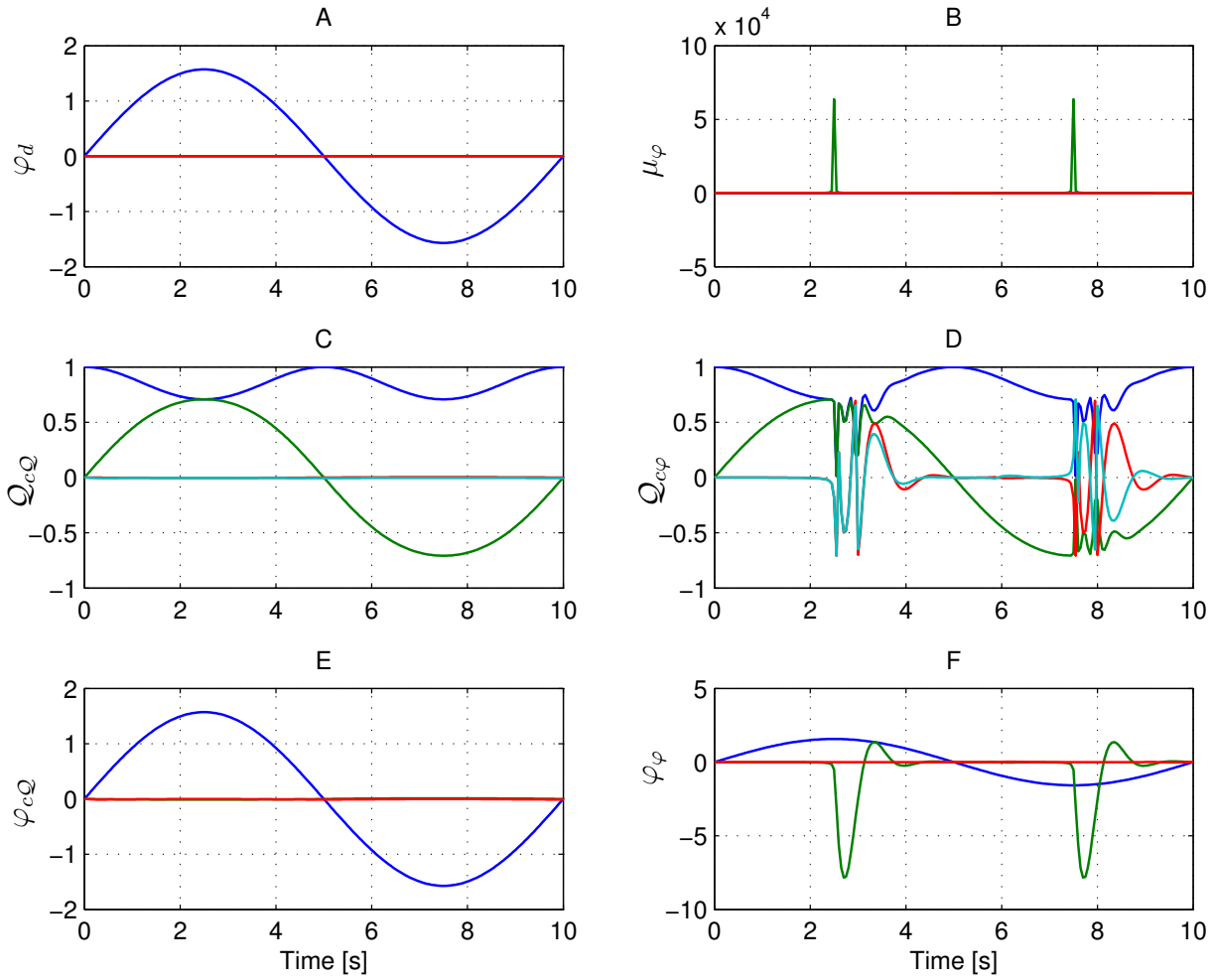


Figure 3.3: Euler angles versus unit quaternions in impedance control: A - desired Euler angles trajectory, B - torque in Euler space, C, E - quaternion-based compliant trajectory expressed in terms of unit quaternion and Euler angles respectively, D, F - Euler angles-based compliant trajectory expressed in terms of unit quaternion and Euler angles respectively

3.3 Extended Task Space Formulation

Having defined the suitable orientation representation, one can formulate the complete definition of the task space vector. We apply the extended kinematics method described in Sec. 2.5.2. As the additional task Ψ, we use the angle $\theta(\boldsymbol{q})$ defined in Sec. 2.1.2. The extended task space vector $\boldsymbol{x}_E$ is given as follows:

$$\boldsymbol{x}_E = \left[\boldsymbol{p}^T(\boldsymbol{q})\ \mathcal{Q}^T(\boldsymbol{q})\ \theta(\boldsymbol{q})\right]^T , \tag{3.21}$$

where $\boldsymbol{p}$ and $\mathcal{Q}$ is the position vector and orientation of the end-effector using the *unit quaternion*, respectively. It is noted that the vector $\boldsymbol{x}_E$ is (8×1) vector due to 4 parameters used to describe the orientation. However, only 3 of them are independent, so that the velocity vector reduces to a (7×1) vector:

$$\dot{\boldsymbol{x}}_E = \left[\dot{\boldsymbol{p}}^T(\boldsymbol{q})\ \boldsymbol{\omega}^T(\boldsymbol{q})\ \dot{\theta}(\boldsymbol{q})\right]^T , \tag{3.22}$$

where $\dot{\boldsymbol{\omega}}$ denotes the rotational velocity of the end-effector. The corresponding error vector $\tilde{\boldsymbol{x}}_E$ is also a (7×1) vector

$$\tilde{\boldsymbol{x}}_E = \left[\tilde{\boldsymbol{p}}^T(\boldsymbol{q})\ \boldsymbol{\epsilon}^T(\boldsymbol{q})\ \tilde{\theta}(\boldsymbol{q})\right]^T . \tag{3.23}$$

The angle θ parameterizes the self motion of the manipulator uniquely, so that the IK function is given as a function of the extended task space vector $\boldsymbol{x}_E$. The resulting manipulator Jacobian is redefined as a (7×7) matrix

$$\dot{\boldsymbol{x}}_E = \boldsymbol{J}_E(\boldsymbol{q})\dot{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{J}(\boldsymbol{q}) \\ \boldsymbol{J}_\theta(\boldsymbol{q}) \end{bmatrix}\dot{\boldsymbol{q}}, \tag{3.24}$$

where $\boldsymbol{J}(\boldsymbol{q})$ is the (6×7) manipulator geometric Jacobian and $\boldsymbol{J}_E(\boldsymbol{q})$ is the (7×7) augmented Jacobian extended by the (1×7) elbow Jacobian

$$\boldsymbol{J}_\theta(\boldsymbol{q}) = \frac{\partial\theta(\boldsymbol{q})}{\partial\boldsymbol{q}}. \tag{3.25}$$

The contact wrench $\boldsymbol{F}_E$ in the extended task space is defined as

$$\boldsymbol{F}_E = \begin{bmatrix} \boldsymbol{F} \\ 0 \end{bmatrix} = \left[\boldsymbol{f}^T\ \boldsymbol{\mu}^T\ 0\right]^T, \tag{3.26}$$

where $\boldsymbol{F}$ is the 6 dimensional Cartesian wrench, consisting of force $\boldsymbol{f}$ and torque $\boldsymbol{\mu}$, and the 0 in the last row corresponds to the contact force of elbow motion. Since there is no physical possibility of measuring its value, it is set to zero by default. However, it is possible to obtain its value using observer methods and manipulator dynamic model. This method has not been implemented to date, and is subject to further study.

3.4 Summary

The goal of this chapter was to define a proper orientation representation, and the resulting task space vector for motion and interaction control of the 7 DoF telemanipulator. The problem of coupling of the devices of dissimilar kinematics was introduced. It was also explained why the minimum three parameter representations such as Euler or Cardan angles are not suitable for teleoperation purposes. A four parameter representation based on the unit quaternion was introduced as the singularity-free parametrization of both the orientation and the angular displacement. Simulative comparison of an interaction control scheme for Euler angles and the unit quaternion was performed. The results confirm that the unit quaternion representation is superior to the Euler angles representation. Finally, a complete task space was defined using the extended kinematics concept from Chapter 2 and the self motion parametrization.

4 Manipulator Control

This chapter explores the fundamentals of a biologically inspired control of an anthropomorphic manipulator. The remarkable dexterity that humans exhibit in various manipulation tasks, especially considering unknown objects or environments, is a long mimic target of robots. The existing industrial as well as anthropomorphic manipulators in the research community are still inferior compared to humans. One can ask the question how humans, with limbs clearly softer, and with a fraction of robot's accuracy can achieve such a remarkable level of manipulation capabilities. There is a number of ongoing works [51, 85, 93, 102] in biomechanics aiming at understanding these phenomena. Without going into details, one can formulate the statement that those capabilities are due to the arm (variable) compliance, redundancy and sensory (kinesthetic, tactile and visual) feedback. We focus on the compliant aspect of the anthropomorphic manipulations, and on the possibilities of endowing compliance into an otherwise rigid manipulator structure. A long list of tasks can be stated where the compliant action of the manipulator is of advantage. One may start with industrial applications, like contour following, polishing, painting, assembling, etc., where the robot has to follow the desired path while exerting force on the environment. When the manipulator comes in contact with the environment, pure position controls tend to cause excessive contact forces and severe vibrations, which usually results in the loss of contact or even physical damage. In order to control the contact force with purely position controlled robots, a precise model of the manipulator, the environment and the task must be provided. The robot itself must be a high accuracy device, which can be manufactured only at the expense of monetary cost, weight and size. Otherwise, a small mismatch in the model or task specification can lead to large, destructive forces. On the other hand, pure force control is not possible in free space motion and is ill-suited for impact situations [26, 53, 133].

In applications including physical interaction with humans (service and rehabilitation robotics, robo-medicine, robo-entertainment), the human factor must be taken into account. It is already widely accepted that in such applications, "soft" robotic arms are absolutely essential [12, 54, 70, 120, 148]. Recent works on cooperative load transportation or cooperative writing report that in such applications the robot must both be safe and "feel like human" [12, 58, 130, 140].

The purpose of this chapter is to investigate the control strategies for a redundant serial manipulator, in order to select the most appropriate one. The dynamic model of the manipulator is given in Sec. 4.1. Sec. 4.2 contains a detailed review of several compliant control strategies, with the special focus on those that control the redundant DoFs in a compliant manner. To achieve that, the concept of extended kinematics from Sec. 2.5.2 is applied for motion control resulting in an innovative concept of impedance control with underlying stiffness control outlined in Sec. 4.2.3. The results of their practical implementations on a 7 DoF manipulator are presented in Sec. 4.3 and a comparative evaluation is performed. The comparison criteria are, among others, the tracking performance, impedance display fidelity, direction decoupling and the ability to control the null space compliance.

4.1 Manipulator Dynamic Modeling

The dynamic model of robots plays an important role in their design and operation. In the robot design phase, the model is used to select the actuators and for simulations with the purpose of testing the performance and possible control schemes. For example, it is used for linearization purposes or in the observer design.

Several methods have been proposed to model the dynamics of a robot. The most frequent ones are the Euler-Lagrange and Newton-Euler formulations [111, 118]. Despite their conceptual simplicity, both methods get computationally too expensive for manipulators consisting of more than 4 bodies, which practically excludes their online application. Within the scope of this work, the Kane method is used, implemented in commercially available multi body simulation package AUTOLEV, see [62, 63]. The advantage of the Kane method is that it is not iterative, like the other two, but delivers analytical expressions for the robot dynamic model, and for its inverse. Although the mathematical expressions are complex, the generated C code is very efficient and allows real time (2 kHz sampling rate) execution.

Independently from the modeling method, the resulting dynamic model has the form:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{V}(\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau} - \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}, \tag{4.1}$$

where $\boldsymbol{q}$ is the vector of joint variables, $\boldsymbol{M}(\boldsymbol{q})$ is the symmetric positive definite inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ is the vector of Coriolis and centrifugal torques, $\boldsymbol{V}(\dot{\boldsymbol{q}})$ is the vector of friction torques, $\boldsymbol{g}(\boldsymbol{q})$ is the vector of gravitational torques, $\boldsymbol{\tau}$ is the vector of driving torques, $\boldsymbol{F}$ is the Cartesian force exerted by the manipulator's end-effector on the environment and $\boldsymbol{J}(\boldsymbol{q})$ is the Jacobian matrix relating joint velocities $\dot{\boldsymbol{q}}$ to the vector of end-effector velocities $\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{q}}$. Substituting $\boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{V}(\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q})$ we obtain a more compact form of (4.1)

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{\tau} - \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}. \tag{4.2}$$

This model will be used frequently within this thesis, in various derivations of the classical inverse dynamics formulation. Namely, setting the driving torque to be

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}_d + \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F} \tag{4.3}$$

leads to cancelation of the nonlinear and coupling elements and yields a double integrator behavior

$$\ddot{\boldsymbol{q}} = \ddot{\boldsymbol{q}}_d, \tag{4.4}$$

where $\ddot{\boldsymbol{q}}_d$ is the desired joint space acceleration. In most variations of the inverse dynamics control, the design objective is the proper $\ddot{\boldsymbol{q}}_d$ generation [64, 111, 136, 139].

4.2 Compliant Control Methods: State of the Art

The controllers used in robotic systems are usually grouped into motion and force controllers. In pure motion control, the user has to completely specify the robot trajectory, which implies free space motion. The absence of any contact prevents the exertion of forces. On the other hand, in pure force control, the robot end-effector is constrained by the environment and there is no motion at all [82, 116]. Both strategies suffer from severe stability

problems during transition from free space motion to contact with rigid environments. Between those extremes, there exist constraint surfaces, termed C-surfaces by Mason [80], where motion is possible along the C-surface tangents and force can be exerted along the C-surface normals. This formulation results in a so called hybrid control [5, 26, 139, 140]. The main disadvantage of this method is that it treats the force and motion control as conflicting goals and partitions the workspace into motion- and force-controlled directions. Since the constrained and unconstrained directions must be determined before the task begins, such a control is not applicable in teleoperation due to the lack of a model of the environment and the task. Further we shall consider only those control strategies that *do not assume prior knowledge of a geometric contact description.*

An alternative to hybrid control is the wide class of compliance controllers. These approaches aim at providing the designer with means of specifying and controlling motion and forces in a non-conflicting manner. To start the review of the available compliant control methods, let us define the concept of mechanical impedance first.

Definition of Mechanical Impedance

We define mechanical impedance as a dynamic relationship between the generalized force $\boldsymbol{F}$ and motion coordinates, resulting in the spatial mass-spring-damper equation

$$\boldsymbol{F} = \boldsymbol{M}\ddot{\boldsymbol{x}} + \boldsymbol{D}\dot{\boldsymbol{x}} + \boldsymbol{K}\tilde{\boldsymbol{x}}, \tag{4.5}$$

where motion may be expressed in terms of generalized position $\boldsymbol{x}$, velocity $\dot{\boldsymbol{x}}$ or acceleration $\ddot{\boldsymbol{x}}$. The matrices $\boldsymbol{M}$, $\boldsymbol{D}$ and $\boldsymbol{K}$ are inertia, damping and stiffness matrices, and $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{x}_0$, where $\boldsymbol{x}_0$ is an equilibrium position. Such impedance may be viewed as *second order impedance*, contrary to the *first order impedance*: $\boldsymbol{F} = \boldsymbol{D}\dot{\boldsymbol{x}} + \boldsymbol{K}\tilde{\boldsymbol{x}}$ and *zeroth order impedance*: $\boldsymbol{F} = \boldsymbol{K}\tilde{\boldsymbol{x}}$. In some works, aiming at achieving compliant behavior without force measurement, equation (4.5) takes the form: $\boldsymbol{0} = \boldsymbol{M}\ddot{\boldsymbol{x}} + \boldsymbol{D}\dot{\boldsymbol{x}} + \boldsymbol{K}\tilde{\boldsymbol{x}}$, with the force component simply set to zero.

Review of Compliant Control Methods

A state of the art review of compliant control methods is found in [32, 90, 108, 116]. Compliance control algorithms generally fall into one of the two categories:

- methods involving force measurement at the robotic end-effector, where the robot motion is modified according to the measured forces; consequently, the compliant action is active only in a contact situation,
- methods that do not require force measurement; the robot internal impedance is maintained independently of the contact situation; these methods are equally applicable to free space as well as to constrained manipulation.

The first class derives from position based impedance control introduced by Hogan [52, 53] and is based on modifying the trajectory according to the measured forces. The second one bases on Salisbury's active stiffness control formulation and modulates the motion error either in joint space [79, 109] with properly shaped nonlinear gains, or directly in the task space. Modulating the error in the task space is also known as "task space PD control" [111].

Special attention needs to be drawn to the redundant manipulators. Whereas the dynamic behavior at the end-effector may be controlled using the force sensor, the self motion dynamics cannot. In order to control null space dynamics, [56] proposes a dynamic control law for redundant manipulators, extended in [91, 92, 94, 96, 100] by means of augmented kinematics. Worth noticing is a more recent, excellent implementation of compliant control strategies for a redundant anthropomorphic robot in [3, 50]. However, to achieve such an excellent performance, torque sensors in manipulator joints are necessary, which in our current implementation are not available. In the following sections we present a selection of compliant control methods with focus on a practical implementation on a redundant teleoperated manipulator.

Note on the Nomenclature

In the coming section, we are going to describe various control strategies. Despite the fact that they can all be viewed as compliance controllers, they have very different names, which may be somewhat misleading. The reason is that they originated at different stages of robotic research, or are derived from different backgrounds (e.g., from motion control techniques). We have decided to leave them in their original for the sake of consistence with the literature, despite their sometimes deceptive character (e.g., stiffness control is in fact first order impedance control). The term *compliance*, used commonly in the robotic community, denominates the inverse of *stiffness* $\boldsymbol{C} = \boldsymbol{K}^{-1}$ and represents the amount of body *softness*, so that the both terms may be used interchangeably, yet having in mind their contrary character.

4.2.1 Stiffness Control

Stiffness control is a special case of impedance control, where the robot impedance is shaped as a spring and damper system. As mentioned before, it does not require measurements of the contact force. That is why stiffness control is potentially expected to manage accidental contacts with the environment also for those degrees of freedom, where such measurements are not available for technical reasons. In other words, stiffness control can endow structural or internal compliance independently from the contact situation and measured (or not) forces. This property makes it particularly attractive for service robots, where the interaction with humans involves physical contact; not only at the end-effector, but also with the whole manipulator body. A very attractive feature of the control methods described below is that using the extended kinematic formulation and the extended Jacobian, it is possible to assign impedance behavior to the manipulator self motion, i.e., to control the redundant DoFs as additional Cartesian coordinates.

Stiffness control can be categorized into two groups based on the domain in which the control error is defined: in the joint space or in the task (Cartesian) space.

Task Space Stiffness Control (STT)

Because the contact force $\boldsymbol{F}$ is sensed in the Cartesian space and the task is generally defined also in the Cartesian space, specifying the impedance relationship in Cartesian space is clearly consistent with the task and enables intuitive parameter tuning. Such stiffness control can be realized in the task space by $\boldsymbol{J}^T$ mapping of the task space error onto

the manipulator torques. Together with gravity and friction compensation, the stiffness control law in the task space is written as:

$$\boldsymbol{\tau} = \boldsymbol{J}_E^T(\boldsymbol{q})(\boldsymbol{D}_k\dot{\tilde{\boldsymbol{x}}}_E + \boldsymbol{K}_k\tilde{\boldsymbol{x}}_E) + \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}, \tag{4.6}$$

where the subscript 'E' denotes quantities defined according to the extended space formulation from Sec. 3.3; $\boldsymbol{J}_E$ and $\boldsymbol{x}_E$ denote the extended Jacobian in (3.24) and the extended task space vector in (3.21); $\tilde{\boldsymbol{x}}_E$ and $\dot{\tilde{\boldsymbol{x}}}_E$ denote the position and velocity error in the task space, respectively. Fig. 4.1 shows the block scheme of the stiffness control law (4.6), where FK denotes the forward kinematics solutions and the subscript 'd' relates to the desired values. The task space error, consisting of both position and orientation error (depicted with 'Pose Error' block) is computed according to (3.23).

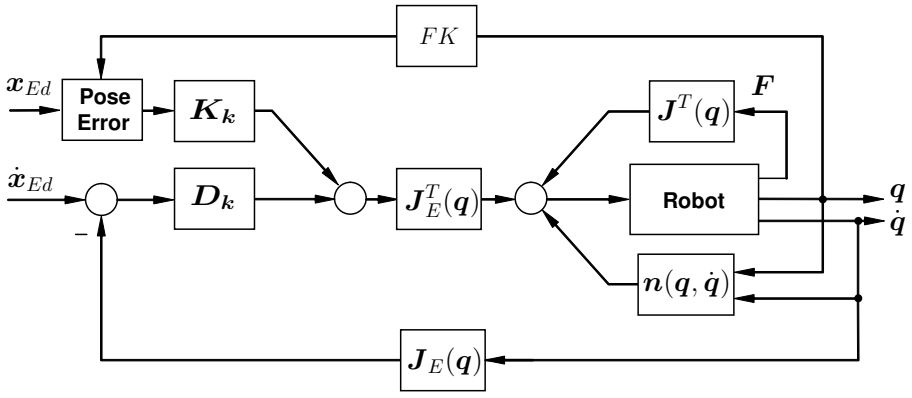


Figure 4.1: Stiffness control in the task space with gravity and friction compensation (STT)

Task Space Stiffness Control with Inverse Dynamics (RAC)

One of the disadvantages of the control in (4.6) is its limited performance in high velocity dynamics, since the control law is obtained from the static model-based compensation. For high velocity motion, it is necessary to compensate the full manipulator dynamics, including its inertia. For this purpose, the acceleration signals must be generated and converted from the task space into the joint space:

$$\dot{\boldsymbol{q}}_d = \boldsymbol{J}_E^{-1}\dot{\boldsymbol{x}}_{Ed} \tag{4.7}$$

$$\ddot{\boldsymbol{q}}_d = \boldsymbol{J}_E^{-1}(\ddot{\boldsymbol{x}}_{Ed} - \dot{\boldsymbol{J}}_E(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}). \tag{4.8}$$

Combining the desired accelerations with the manipulator inverse dynamic model (4.3) we obtain the dynamic model in the task space as:

$$\boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}_E^{-1}(\boldsymbol{q})(\ddot{\boldsymbol{x}}_{Ed} - \dot{\boldsymbol{J}}_E(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}) + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{V}(\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau} - \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}. \tag{4.9}$$

Including the Cartesian error in the acceleration profile gives the control law for the stiffness control with inverse dynamics in the task space as

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}_E^{-1}(\boldsymbol{q})\left(\ddot{\boldsymbol{x}}_{Ed} + \boldsymbol{D}_k\dot{\tilde{\boldsymbol{x}}}_E + \boldsymbol{K}_k\tilde{\boldsymbol{x}}_E - \dot{\boldsymbol{J}}_E(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}\right) + \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}. \tag{4.10}$$

The corresponding block scheme is illustrated in Fig. 4.2. The task space stiffness control of this form is essentially identical with the resolved acceleration control, known from motion control techniques [77, 111].

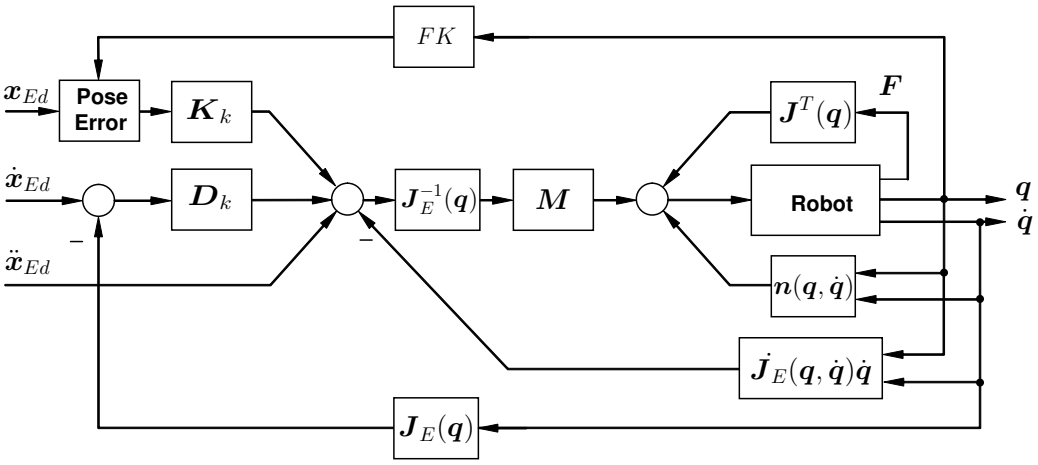


Figure 4.2: Stiffness control with inverse dynamics in the task space (RAC)

Salisbury Stiffness Control (SJA & SJB)

The desired impedance may be specified both in joint space and in Cartesian space. For example, the most common choice for the desired acceleration in (4.4) is $\ddot{\boldsymbol{q}}_d = \boldsymbol{K}_j\tilde{\boldsymbol{q}} + \boldsymbol{D}_j\dot{\tilde{\boldsymbol{q}}}$, which is known as the joint space PD control law with inverse dynamics [64, 111, 118]. The matrices $\boldsymbol{K}_j$ and $\boldsymbol{D}_j$ are joint space stiffness and damping matrices. However, it is possible to find a relationship between the Cartesian impedance and the joint space impedance. This idea proposed in [14, 79, 109] for mapping the Cartesian stiffness ($\boldsymbol{K}_k$, $\boldsymbol{D}_k$) into the joint space, makes use of the manipulator Jacobian $\boldsymbol{J}$ and may be derived by generalization of the linear relationship $f = k\mathrm{d}x$ to a six dimensional spring

$$\boldsymbol{F} = \boldsymbol{K}_k\delta\boldsymbol{x}, \tag{4.11}$$

where $\boldsymbol{K}_k$ is the stiffness matrix and $\delta\boldsymbol{x}$ is the generalized displacement from the commanded pose $\boldsymbol{x}_0$. Recall the Jacobian definition [111, 118]

$$\delta\boldsymbol{x} \approx \boldsymbol{J}\delta\boldsymbol{q} \tag{4.12}$$

and the static relationship

$$\boldsymbol{\tau} = \boldsymbol{J}^T\boldsymbol{F}. \tag{4.13}$$

Blending 4.11 to 4.13, one gets

$$\boldsymbol{\tau} = \boldsymbol{J}^T\boldsymbol{K}_k\boldsymbol{J}\delta\boldsymbol{q}. \tag{4.14}$$

Since the mappings (4.11-4.13) are true for all types of manipulators, including redundant ones with extended kinematics formulation, one can map the Cartesian stiffness and damping matrices into joint space using the extended Jacobian $\boldsymbol{J}_E$

$$\boldsymbol{K}_j = \boldsymbol{J}_E^T\boldsymbol{K}_k\boldsymbol{J}_E \tag{4.15}$$

$$\boldsymbol{D}_j = \boldsymbol{J}_E^T\boldsymbol{D}_k\boldsymbol{J}_E. \tag{4.16}$$

The joint space stiffness and damping matrices $\boldsymbol{K}_j$ and $\boldsymbol{D}_j$ are then intentionally non diagonal, but still symmetric.

Combining the stiffness control law with inverse dynamics one gets

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})(\ddot{\boldsymbol{q}}_d + \boldsymbol{D}_j\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_j\tilde{\boldsymbol{q}}) + \boldsymbol{n} + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}, \tag{4.17}$$

where the gain matrices $\boldsymbol{K}_j$ and $\boldsymbol{D}_j$ are mapped from the Cartesian gain matrix $\boldsymbol{K}_k$ and $\boldsymbol{D}_k$ as in (4.15, 4.16). The vector $\tilde{\boldsymbol{q}}$ and $\dot{\tilde{\boldsymbol{q}}}$ denote the error of corresponding vectors: $\boldsymbol{q}_d - \boldsymbol{q}$ and $\dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}$. Matrix $\boldsymbol{J}_E$ is the extended Jacobian matrix defined in (3.24). The corresponding control block scheme is shown in Fig. 4.3.

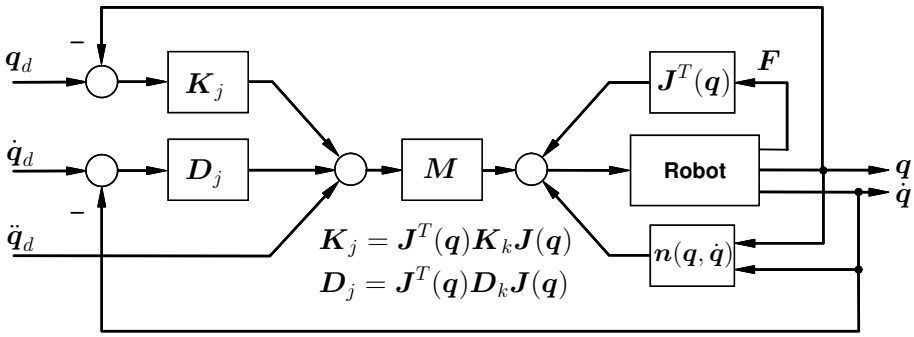


Figure 4.3: Stiffness control with inverse dynamics in the joint space (SJA)

This structure aims at compensating the natural dynamics of the manipulator by the inverse dynamics. In case of an accurate dynamic model of manipulator, the control is expected to be independent from the manipulator dynamics and to provide good performance also for high velocities. It is remarkable that all the control actions in this control scheme go through the inertia matrix of the manipulator, so that the overall performance of this control scheme will strongly depend on the modeling accuracy. In case of modeling inaccuracy, all the control action is deteriorated in a way that the error of the inertia matrix is further propagated.

As an alternative, to reduce sensitivity to model accuracy, the Wen-Bayard [136] modification to (4.17) may be applied as follows:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}_d + \boldsymbol{D}_j\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_j\tilde{\boldsymbol{q}} + \boldsymbol{n} + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}. \tag{4.18}$$

This control law is exactly the same as the previous one (4.17), except for a separation of the control actions from the inertia matrix. Consequently, it is less dependent on the modeling accuracy of the inertia matrix. Substituting this control law (4.18) into (4.1), we obtain the following dynamics of position error under assumption of a perfect compensation:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\tilde{\boldsymbol{q}}} + \boldsymbol{D}_j\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_j\tilde{\boldsymbol{q}} = 0, \tag{4.19}$$

which shows that the position error converges to zero depending of the positive definite inertia matrix $\boldsymbol{M}$, damping matrix $\boldsymbol{D}_j$ and stiffness matrix $\boldsymbol{K}_j$. The corresponding block scheme is illustrated in Fig. 4.4.

Joint Space PD Control (PDJ)

By setting the matrices $\boldsymbol{K}_j$ and $\boldsymbol{D}_j$ in equations 4.17 and 4.19 as constant diagonal matrices, one obtains the basic joint space PD control algorithm [111, 118]. Since impedance parameters are specified in joint space, the resulting Cartesian impedance is nonlinear and configuration dependent. For this reason, this algorithm is not suitable for interaction control. However, it will be used extensively as a motion control algorithm, due to its simplicity and robustness against modeling errors. The arm impedance will be governed by the impedance control algorithm in the outer loop, as described in the next section.

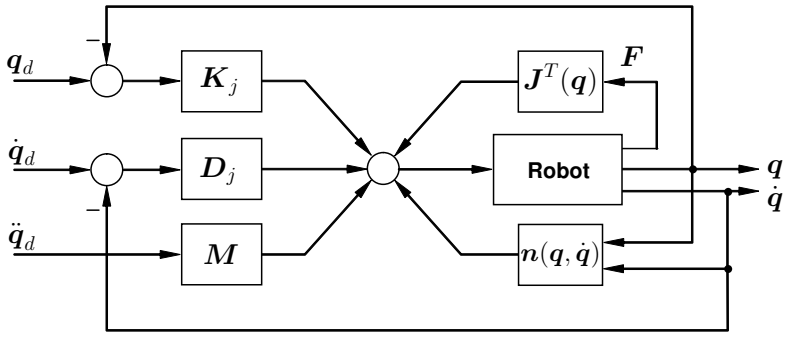


Figure 4.4: Stiffness control with modified inverse dynamics in the joint space (SJB)

4.2.2 Impedance Control in Task Space (IMP)

The core idea behind the impedance control (IMP) [52] is that instead of controlling individual positions and forces directly, it controls the dynamic response of the robotic manipulator by establishing a virtual mass-spring-damper system on the end-effector in the form of (4.5). This equation may be solved for force, given motion variables, leading to so a called *force-based impedance control*. Another possibility is to solve the equation for motion variables, given measured force, which leads to *position-based impedance control* or *admittance control*. To formally define a mechanical impedance, besides the actual end-effector frame Σ_e and the desired end-effector frame Σ_d, it is worth introducing the compliant frame Σ_c, as shown in Fig. 4.5. The compliant frame describes the end-effector

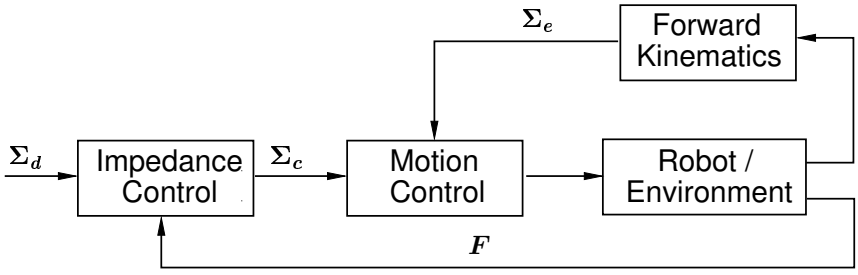


Figure 4.5: Impedance control strategy; $\Sigma_d, \Sigma_c, \Sigma_e$ are desired, compliant and end-effector frames, F - measured wrench (forces and torques)

position and orientation when it is in contact with the environment, so that the contact force and moment are governed by the user specified impedance, a so-called target or desired impedance. In other words, when the end-effector moves in free space, the impedance controller does not modify the desired end-effector frame Σ_d so that the end-effector follows the desired trajectory. When the end-effector is in contact with the environment, the impedance controller modifies the frame Σ_d according to the contact wrench F and returns the compliant frame Σ_c, which must be followed by the end-effector during the contact. Consequently, the impedance control is active only in contact. This means that in situations when no contact force/torque is sensed, the compliant frame Σ_e is identical with the desired frame Σ_d.

Translational Impedance

As mentioned earlier, the goal of the impedance control is to make the end-effector behave as a linear and decoupled mechanical impedance characterized by a virtual mass $\boldsymbol{M}$, damping $\boldsymbol{D}$ and stiffness $\boldsymbol{K}$ matrices with regards to the measured contact force $\boldsymbol{f}$. This can be written for the translational impedance as follows:

$$\boldsymbol{M}_p\ddot{\boldsymbol{p}}_{dc} + \boldsymbol{D}_p\dot{\boldsymbol{p}}_{dc} + \boldsymbol{K}_p\boldsymbol{p}_{dc} = \boldsymbol{f}, \tag{4.20}$$

the subscript 'p' denotes the translational parameters; $\boldsymbol{p}_{dc}$, $\dot{\boldsymbol{p}}_{dc}$ and $\ddot{\boldsymbol{p}}_{dc}$ denote the difference $\boldsymbol{p}_{dc} = \boldsymbol{p}_c - \boldsymbol{p}_d$ between the desired and compliance frames (for translational part $\boldsymbol{p}$ only) and its first and second derivative, respectively. In case of free space motion ($\boldsymbol{f} = 0$)

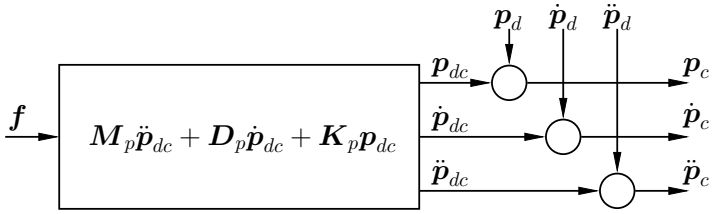


Figure 4.6: Block scheme of translational impedance equation

$\boldsymbol{p}_{dc}$ converges to zero, so that the end-effector trajectory completely relies on the inner position control loop. In case of constrained motion ($\boldsymbol{f} \neq 0$), the desired trajectory $\boldsymbol{p}_d$ is modified by the difference $\boldsymbol{p}_{dc}$. This results in the compliant trajectory $\boldsymbol{p}_c = \boldsymbol{p}_d + \boldsymbol{p}_{dc}$, which serves as a desired trajectory for the inner motion control loop. Such control scheme is illustrated in Fig. 4.6. In this way, we can specify the dynamic response of the end-effector to a certain contact force. Since the impedance parameters are matrices, different dynamic characteristics can be assigned for different directions if necessary.

Rotational Impedance

As already pointed out in Chapter 3.2.2, the angular displacement (orientation error) $\boldsymbol{O}_{dc}$ between the frames Σ_d and Σ_c, expressed by the *unit quaternion*, is the vector part of the following product

$$\mathcal{Q}_{dc} = \{\eta_{dc}, \boldsymbol{\epsilon}_{dc}\} = \mathcal{Q}_c^{-1} * \mathcal{Q}_d. \tag{4.21}$$

The quaternion $\mathcal{Q}_{dc}$ expresses the mutual orientation between those two frames. Similarly to the translational impedance, the rotational impedance is defined as

$$\boldsymbol{M}_o\dot{\boldsymbol{\omega}}_{dc} + \boldsymbol{D}_o\boldsymbol{\omega}_{dc} + \boldsymbol{K}_o^{'}\boldsymbol{\epsilon}_{dc} = \boldsymbol{\mu}, \tag{4.22}$$

where $\boldsymbol{\mu}$ is the contact moment with respect to the compliant frame Σ_c. The stiffness matrix $\boldsymbol{K}_o'$ is defined as

$$\boldsymbol{K}_o' = 2\boldsymbol{E}^T(\eta_{dc}, \boldsymbol{\epsilon}_{dc})\boldsymbol{K}_o, \tag{4.23}$$

with $\boldsymbol{E}$ as in (3.15). The rotational impedance equation (4.22) is illustrated in Fig. 4.7. It is noted that quaternion product is used to calculate the orientation of the compliant frame. Rotational velocity and acceleration are computed in the same way, as for the translational impedance. Similarly to the case of translational impedance, different parameters can be assigned to different directions.

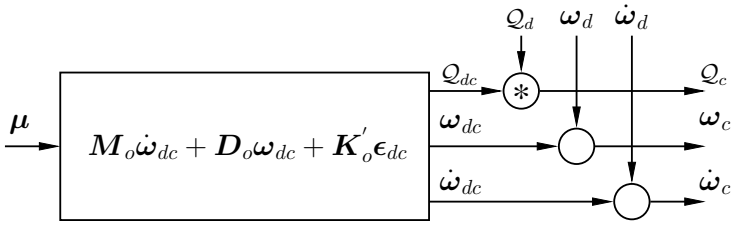


Figure 4.7: Block scheme of rotational impedance equation

4.2.3 Mixed Stiffness-Impedance Approach

As discussed in the preceding sections, impedance equation modifies the desired trajectory according to the measured contact force measured at the wrist. Since force measurement on the robot elbow is not available, it is not possible to apply the impedance equation (4.5) to the redundant DoFs of the manipulator. Nonetheless, the self motion can also be controlled in a compliant manner by combining the impedance controller with a stiffness controller (e.g., STT) as the motion controller in the inner loop. For the coordinates, for which the force measurement is available (position and orientation of the end-effector), the gains in the inner stiffness control loop must be set considerably high in order to achieve good tracking of the compliant coordinates $[\boldsymbol{p}_c\ \mathcal{Q}_c]$. The null space impedance parameters K_θ and D_θ are set in the inner loop and $\theta_c = \theta_d$. Other approaches to compliant control of redundant manipulators use optimization procedures that do not allow the explicit control of the self motion; it is governed by the local optimizer [90, 94, 100].

4.3 Experimental Comparison of the Control Algorithms

Extensive experimental work was carried out to effectively compare the performance of various control schemes with regard to teleoperation. In the succeeding sections the candidate control schemes will be summarized, the comparison criteria will be discussed and the corresponding test scenarios and experimental results will be examined.

Candidate Control Schemes for Experiments

The candidate control laws for experimental evaluation are listed as follows:

STT : Stiffness control in the task space with gravity and friction compensation. The corresponding control law of Fig. 4.1 is

$$\boldsymbol{\tau} = \boldsymbol{J}_E^T(\boldsymbol{q})(\boldsymbol{D}_k\dot{\tilde{\boldsymbol{x}}}_E + \boldsymbol{K}_k\tilde{\boldsymbol{x}}_E) + \boldsymbol{f} + \boldsymbol{g} + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}.$$

RAC : Stiffness control in the task space with inverse dynamics. The corresponding control law of Fig. 4.2 is

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}_E^{-1}(\boldsymbol{q})\left(\ddot{\boldsymbol{x}}_E + \boldsymbol{D}_k\dot{\tilde{\boldsymbol{x}}}_E + \boldsymbol{K}_k\tilde{\boldsymbol{x}}_E - \dot{\boldsymbol{J}}_E(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}\right) + \boldsymbol{n} + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}.$$

SJA : Stiffness control with inverse dynamics in the joint space in which Cartesian stiffness and damping matrices are mapped into the joint space. The corresponding control law of Fig. 4.3 is

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})(\ddot{\boldsymbol{q}}_d + \boldsymbol{D}_j\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_j\tilde{\boldsymbol{q}}) + \boldsymbol{n} + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}.$$

SJB : This control law is obtained by modification of SJA with the aim of a reduction of sensitivity to modeling accuracy. The corresponding control law of Fig. 4.4 is

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}_d + \boldsymbol{D}_j\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_j\tilde{\boldsymbol{q}} + \boldsymbol{n} + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}.$$

IMP-SJB : Impedance control with SJB as inner position control.

IMP-STT : Impedance control with STT as inner position control.

IMP-RAC : Impedance control with resolved acceleration scheme (inverse dynamics control) in the task space.

IMP-PDJ : Impedance control with conventional PD control in joint space. This control scheme will also be referred to as IMP only.

In all control schemes except IMP-PDJ, the extended task space vector $\boldsymbol{x}_E$ (3.21) and the (7×7) augmented manipulator Jacobian $\boldsymbol{J}_E$ (3.24) are used. To represent orientation and orientation error, the *unit quaternion* is applied in all schemes.

4.3.1 Criteria for Evaluation

The tasks that the teleoperator is required to execute can be categorized into two groups: free space motion, and constrained motion during contact. All candidate controllers should serve as a possibly accurate position control as long as the manipulator is in free space motion. For this reason a tracking experiment is carried out. Since the manipulator is a highly complex kinematic structure, the mapping between joint and task space is extremely nonlinear, which contributes to coupling effects in Cartesian coordinates. This may be observed especially during fast motions. Ideally, controllers should excite motion only in the desired directions and the other coordinates should not be influenced. This feature may be well seen in step responses. The impedance parameters are crucial for stability and performance of the teleoperation control loop, see chapter 5. Teleoperation requires that the apparent manipulator impedance matches precisely the desired impedance. Next important issue is the ability to control self motion in a compliant manner. Motion of the redundant DoFs (self motion) is a key to handle unexpected contacts that cannot be recognized by means of the wrist force - torque sensors. These are typically collisions of the robot links with the environment or another manipulator. In such a case, soft elbow can yield its current posture without causing any end-effector motion and afterward recover its original configuration. In human environments, this is at the first stage a safety measure to handle impact of the manipulator on humans.

From what has been discussed, *trajectory tracking*, *direction decoupling* and *impedance display fidelity* are chosen as basic criteria for performance comparison. The *computational burden* and *robustness against singularities* are chosen as side criteria. The following experimental test scenarios are described:

- **Trajectory Tracking & Direction Decoupling:** The tracking error and the coupling between the coordinates will be investigated by means of motion step responses.
- **Impedance Display Fidelity:** The fidelity of the impedance display is judged by comparing the apparent manipulator impedance to the impedance specified in the controller.

Table 4.1: Control gains for the trajectory tracking

Control Scheme	$\boldsymbol{K}$ [N/m]	$\boldsymbol{D}$ [Ns/m]
SJA	10^3·[6 6 6 3.5 3.5 3.5 3.5]	[70 70 70 40 40 40 40]
SJB	10^3·[6 6 6 3.5 3.5 3.5 3.5]	10^2·[1 1 1 0.6 0.6 0.6 0.6]
STT	10^3·[6 6 6 3.5 3.5 3.5 3.5]	10^2·[1 1 1 0.6 0.6 0.6 0.6]
RAC	10^3·[6 6 6 3.5 3.5 3.5 3.5]	10^2·[1 1 1 0.6 0.6 0.6 0.6]
PDJ	10^3·[49 39 39 39 11 11 11]	10·[9 9 9 9 5.4 5.4 5.4]

- **Robustness to Singularities:** Every control using inversion of the manipulator Jacobian suffers from the singularity problem, since there are certain configurations in which the manipulator Jacobian cannot be inverted. Hence the control methods using the inverse Jacobian, such as the resolved acceleration scheme, will not be implemented in teleoperation scenarios and are considered here only as a reference to compare the performance of the candidate controls.

4.3.2 Trajectory Tracking

To demonstrate the coupling effects in the task space, the step responses for every task space coordinate are recorded. However, most of the investigated controllers require acceleration input, which means that the desired trajectory needs to be differentiated twice. To avoid problems caused by the differentiation noise, a smooth approximation of rectangular pulses is generated using a fifth-order polynomial with zero initial and final velocities and accelerations. The rising time of the pulse is 0.5 s, as seen in Fig. 4.8. The maximum step

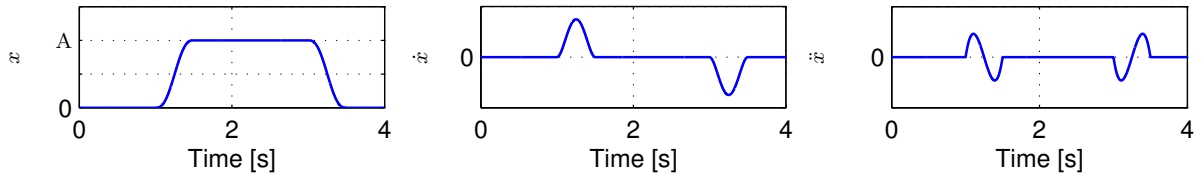


Figure 4.8: Desired pulse trajectory with its derivatives

value A for end-effector position is 0.1 m, the corresponding orientation is 0.087 rad, and the elbow angle defined as 0.1 rad. Such a choice of parameters, resulting from maximal velocity/acceleration properties of the motion of human arms, is actually over-scaled for precise manipulation. Dexterous manipulation with tools has velocity/bandwidth requirements of 10 % of the aforementioned requirements, see [7, 87].

During the experiment there is no contact with the environment, so that the impedance control will not modify the desired trajectory. In this case, the control schemes IMP-SJB, IMP-STT and IMP-RAC are exactly identical as with SJB, STT and RAC, respectively. Therefore, the following control schemes are tested: SJA, SJB, STT, RAC, and IMP-PDJ. The control gains used in the above control schemes are chosen to be high and are summarized in Tab. 4.1. The control gains $\boldsymbol{K}$ and $\boldsymbol{D}$ are defined in task space, only for PDJ they are specified in joint space. The gains are chosen high to achieve good tracking. The PDJ controller is tuned according to [118], aiming at a critically damped closed loop system.

- **Recorded Trajectory: SJA**

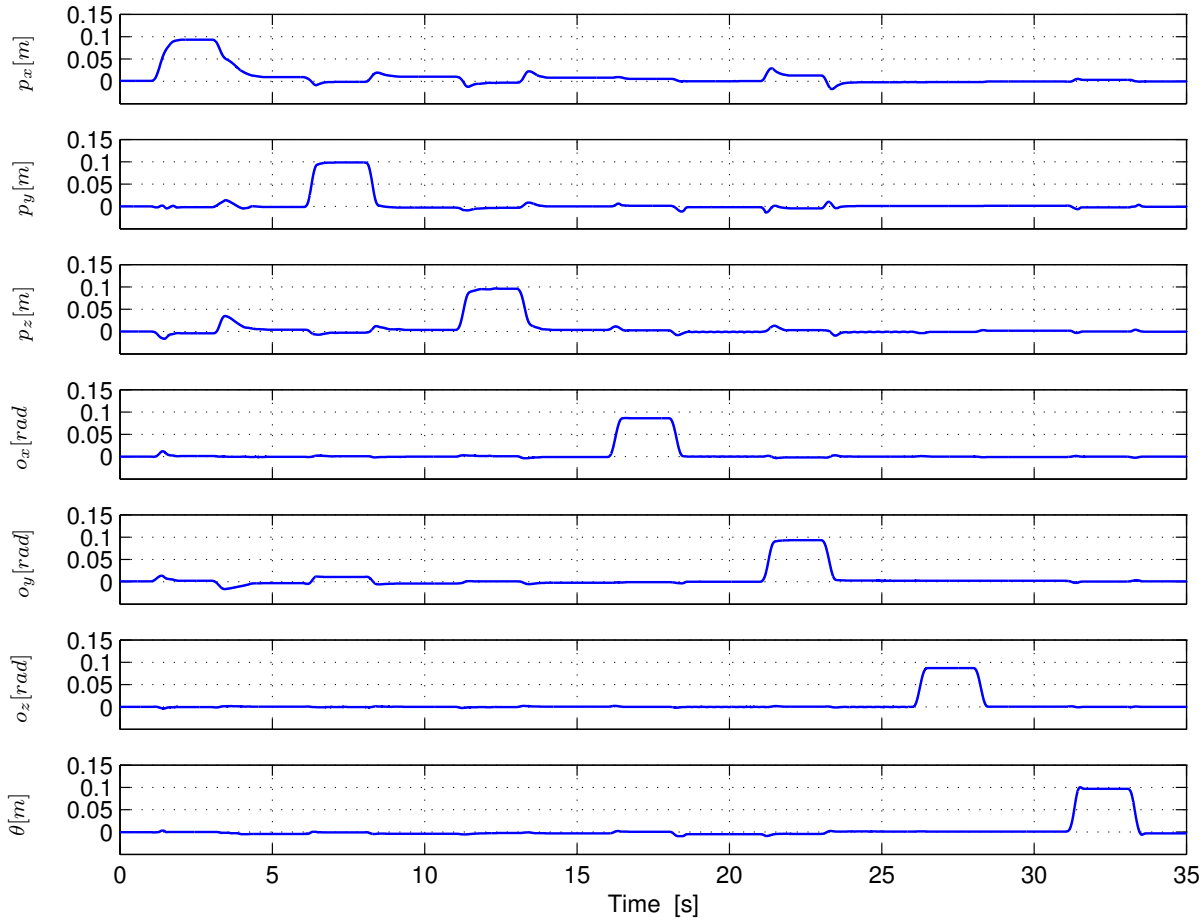


Figure 4.9: Trajectory recorded with SJA

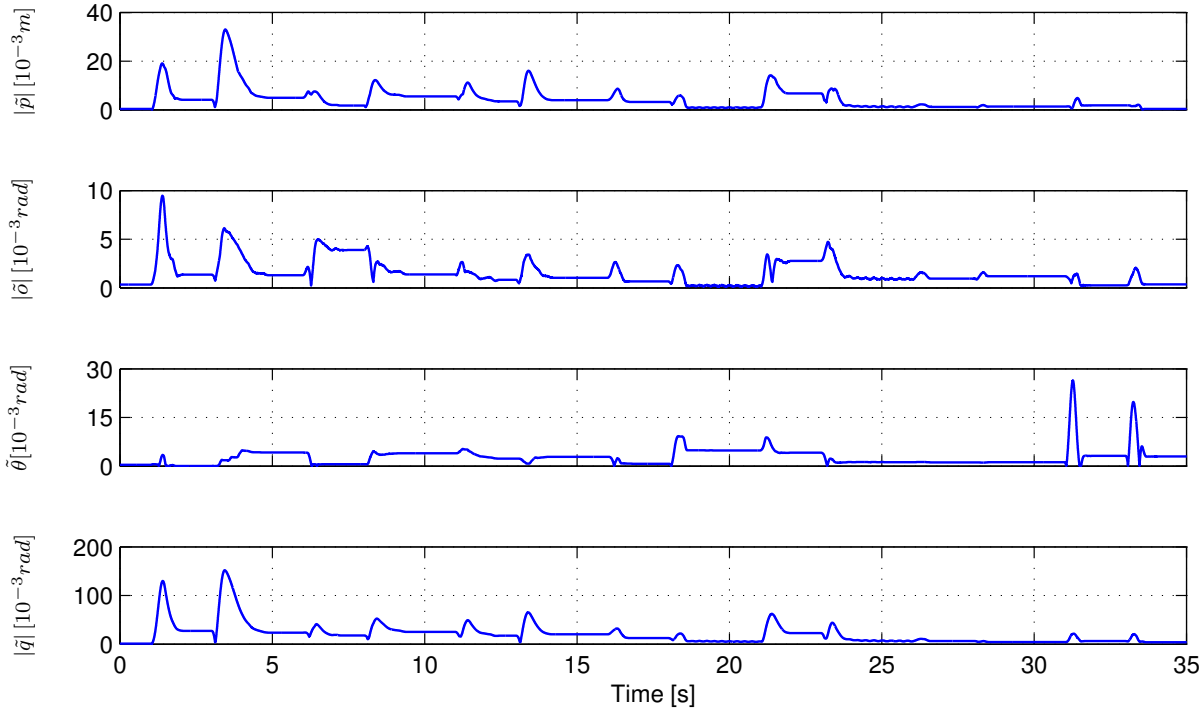


Figure 4.10: Norm of the tracking errors in the joint and in the task space for SJA

- **Recorded Trajectory: SJB**

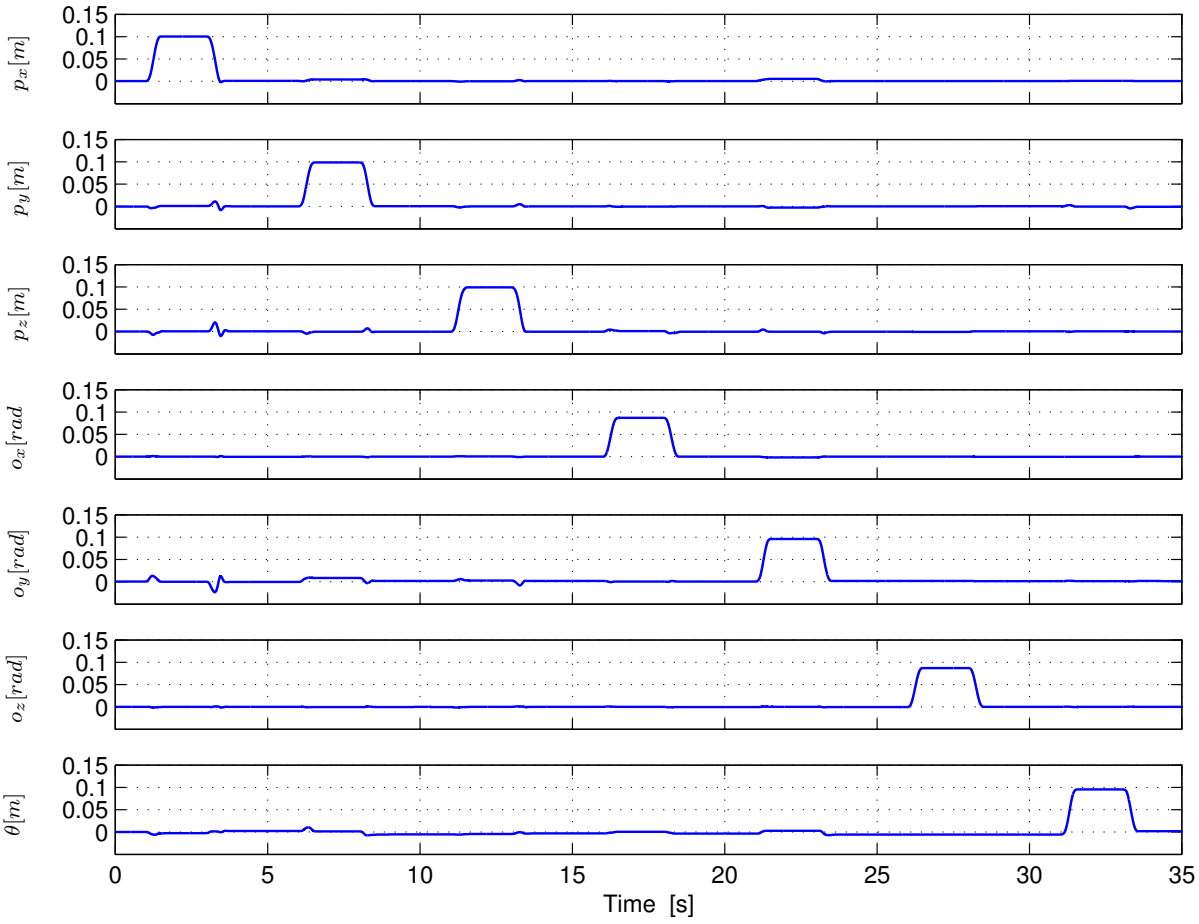


Figure 4.11: Trajectory recorded with SJB

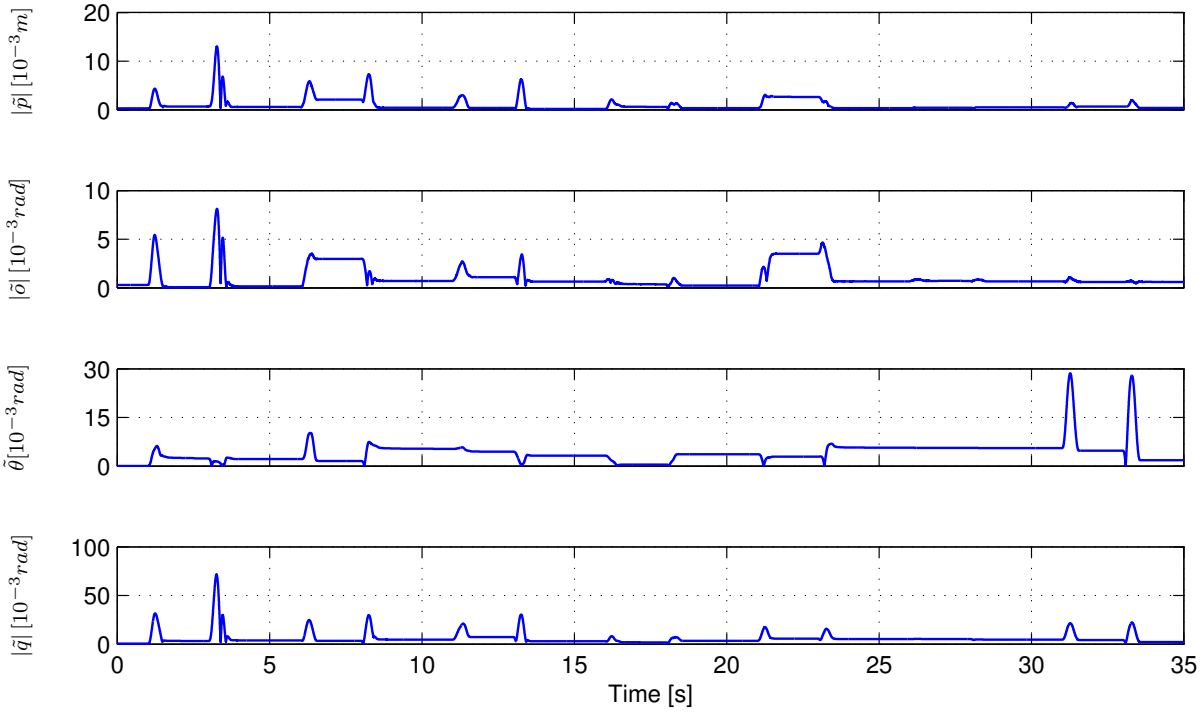


Figure 4.12: Norm of the tracking errors in the joint and in the task space for SJB

- **Recorded Trajectory: STT**

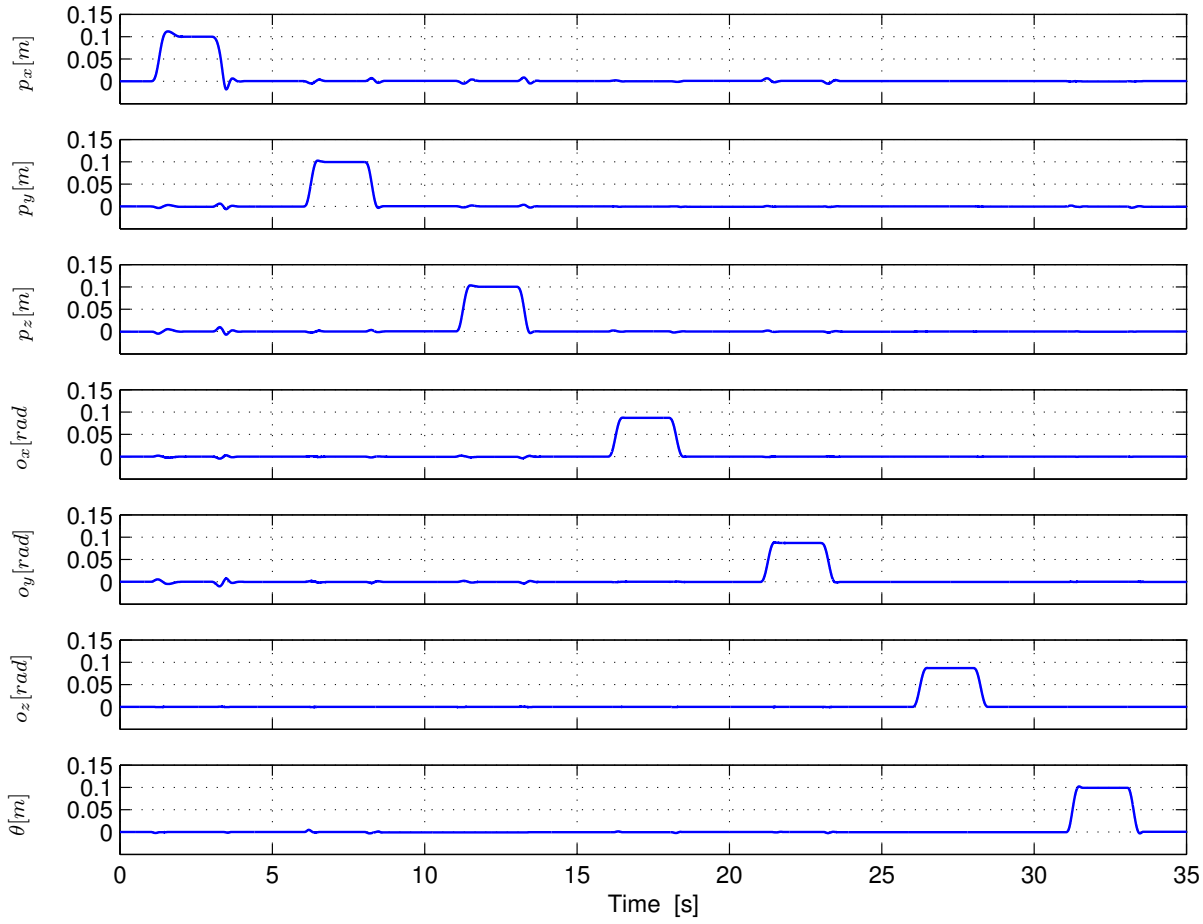


Figure 4.13: Trajectory tracking with STT

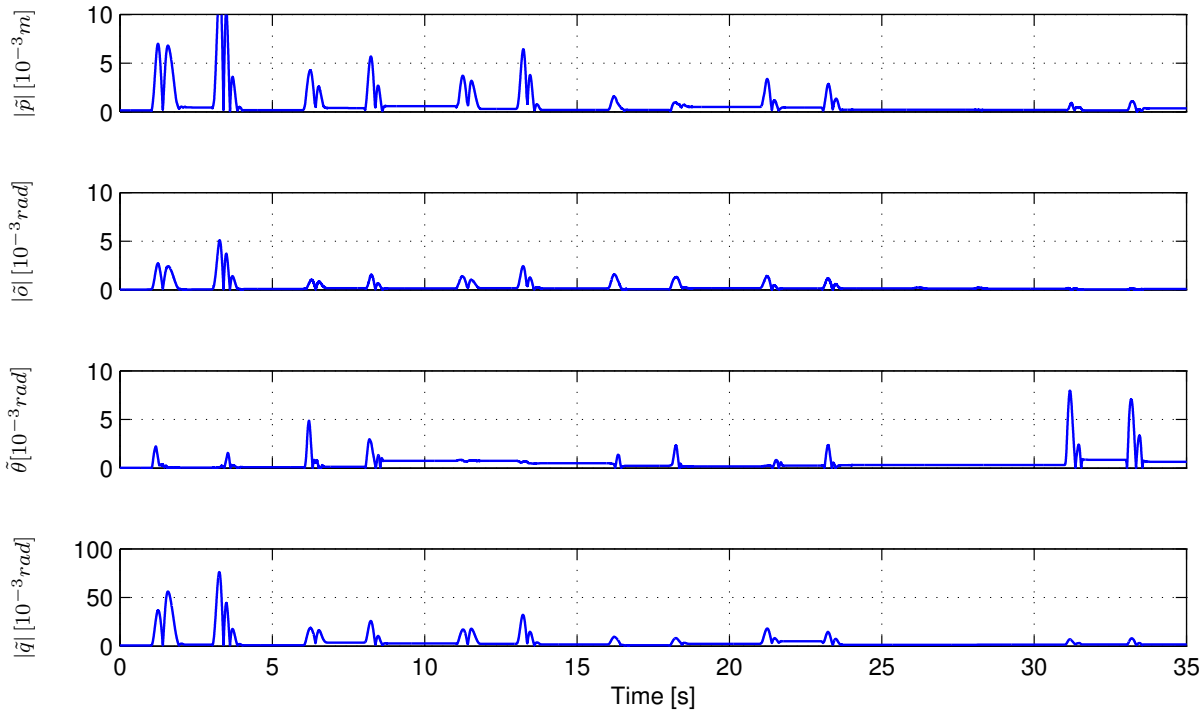


Figure 4.14: Norm of the tracking errors in the joint and in the task space for STT

- **Recorded Trajectory: RAC**

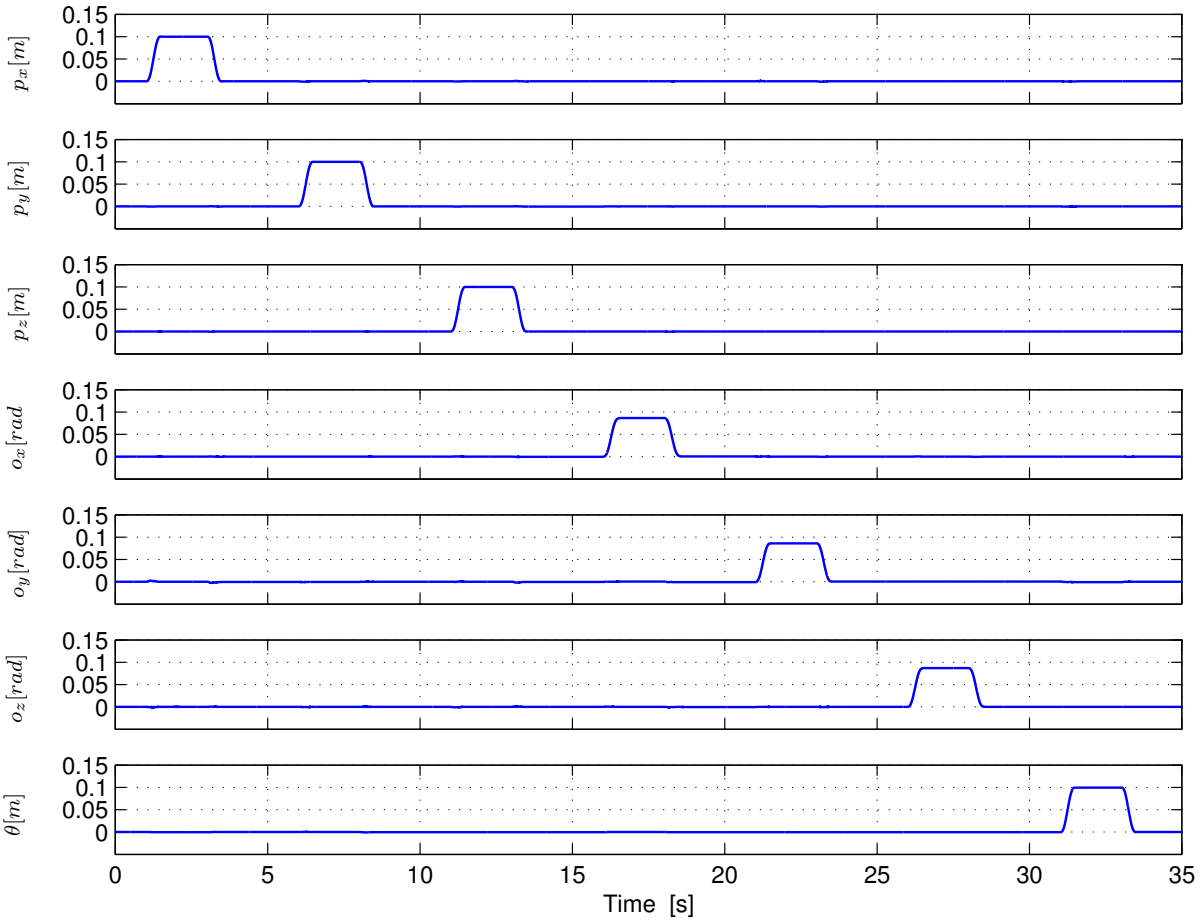


Figure 4.15: Trajectory tracking with RAC

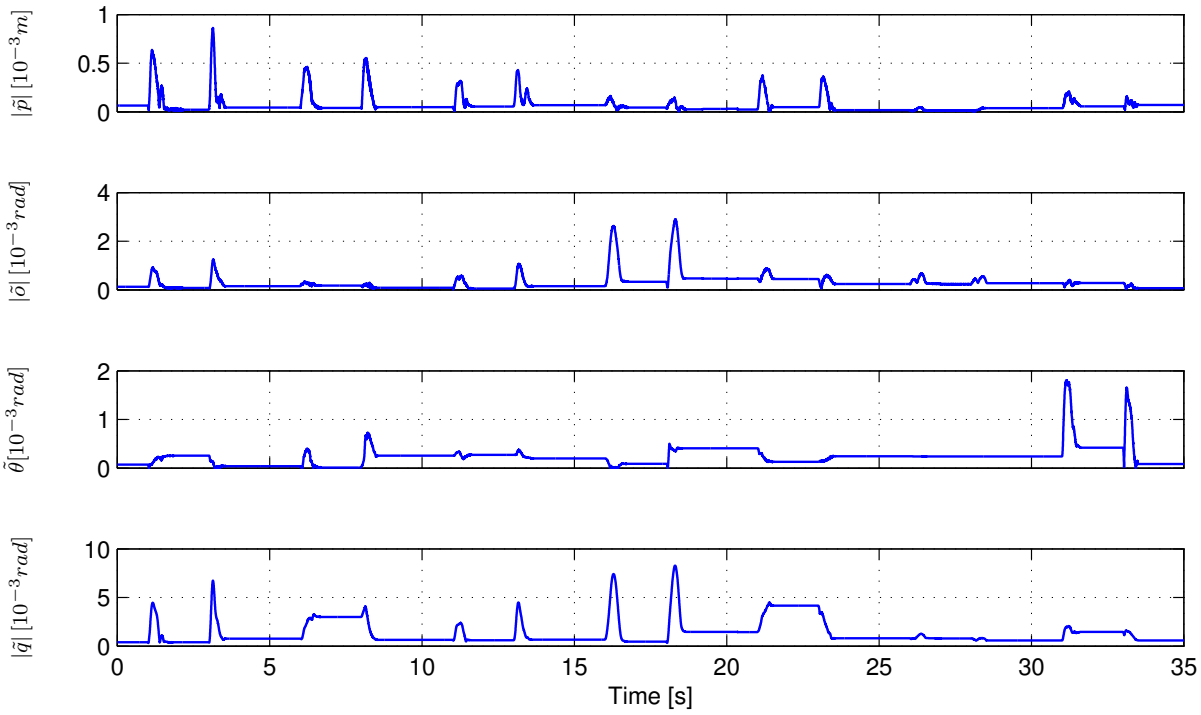


Figure 4.16: Norm of the tracking errors in the joint and in the task space for RAC

- **Recorded Trajectory: IMP-PDJ**

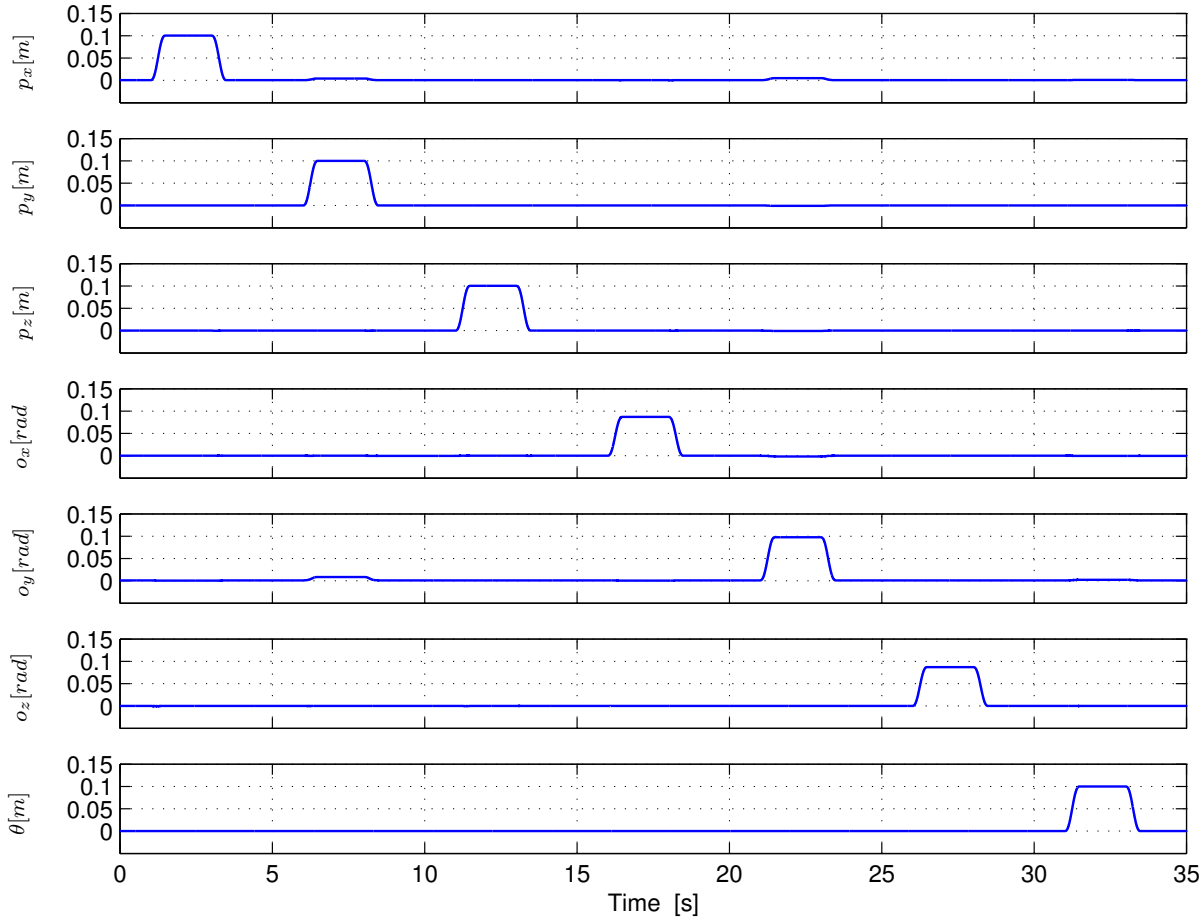


Figure 4.17: Trajectory recorded with IMP-PDJ

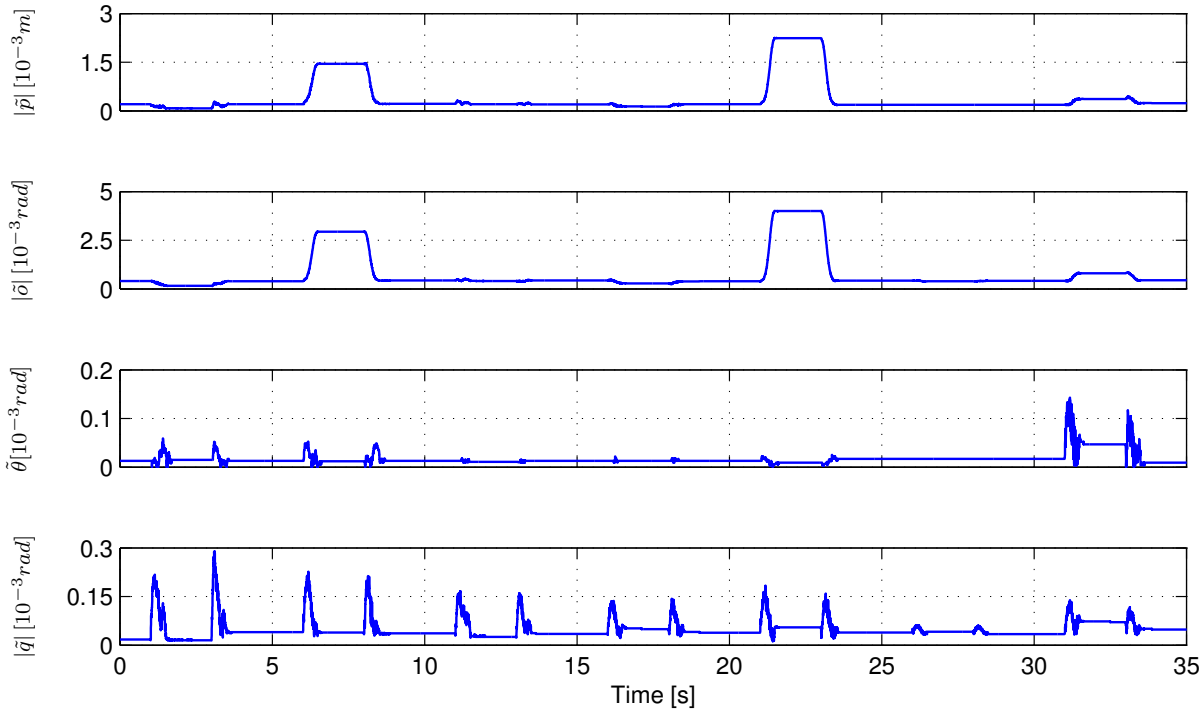


Figure 4.18: Norm of the tracking errors in the joint and in the task space for IMP-PDJ

The step responses are collected in Fig. 4.9-4.18. The maximum steady state errors along each direction are listed in Tab. 4.2.

Table 4.2: Maximum steady state error for different controllers

Control Scheme	Maximum steady state error [m] ([rad]) $[\tilde{\boldsymbol{p}}_x \;\; \tilde{\boldsymbol{p}}_y \;\; \tilde{\boldsymbol{p}}_z \;\; o_x \;\; o_y \;\; o_z \;\; \tilde{\theta}\,]$						
SJA	[0.0133	0.0080	0.0042	0.0013	0.0071	0.0006	0.0055]
SJB	[0.0026	0.0015	0.0009	0.0002	0.0007	0.0001	0.0014]
STT	[0.0008	0.0009	0.0004	0.0002	0.0001	0.0005	0.0009]
RAC	[0.0002	0.0001	0.0000	0.0004	0.0003	0.0012	0.0004]
PDJ	[0.0010	0.0010	0.0007	0.0003	0.0001	0.0022	0.0001]

The best performance is found in RAC control, followed by STT and IMP-PDJ. Thus a natural choice for the tracking controller is RAC. However, there are certain practical problems with its implementation in a teleoperation scenario. First, the inversion of the manipulator Jacobian is sensitive to mechanical singularities, and the damped inverses, developed in Sec. 2.3.4, have not been tested for control at this point. Second, RAC controllers require the desired acceleration signal. Whereas with the other control schemes the acceleration may be set to zero for low velocities, here, it is used for motion generation and cannot simply be omitted. In the future implementation the desired trajectories will be generated using an input device that is equipped with encoders for motion sensing. The acceleration signal is obtained by twin differentiation and, due to the noise, is practically useless. So this control scheme is used here just as a reference. Outstanding tracking is achieved with STT and IMP-PDJ controllers, what makes them good candidates for further investigation. Poor tracking performance found in Fig. 4.9 (SJA) can be explained by the fact that all control actions in this scheme have to go through the inertia matrix and its inaccuracy deteriorates the control performance. Therefore, SJA cannot be practically used as long as the modeling accuracy is not guaranteed. The SJB method, a modified version of SJA, where the sensitivity to the inertia matrix was reduced, yielded much better tracking performance. It has only about one fifth of the maximum steady state error of SJA. These findings force us to conclude that the inertia matrix developed and used throughout this thesis has a certain degree of inaccuracy.

Comparing SJB with STT shows that STT has about one half of the SJB maximum steady state error. It requires also less computational power, since it is executed without mapping of the gain matrices into the joint space $\boldsymbol{J}_E^T \boldsymbol{D}_k \boldsymbol{J}_E$ and $\boldsymbol{J}_E^T \boldsymbol{K}_k \boldsymbol{J}_E$.

Table 4.3: Mean tracking error

Control Scheme	Mean tracking error $\tilde{\boldsymbol{p}}\,[10^{-3}m]$	$\tilde{\boldsymbol{o}}\,[10^{-3}rad]$	$\tilde{\theta}\,[10^{-3}rad]$	$\tilde{\boldsymbol{q}}\,[10^{-3}rad]$
SJA	1.2075	0.5083	1.7756	1.6692
SJB	0.2820	0.4674	2.1931	0.2415
STT	0.1593	0.0587	0.2030	0.0808
RAC	0.0115	0.0037	0.1788	0.5498
PDJ	0.2799	0.4406	0.0150	0.0079

The coupling effects may be clearly seen in the step responses in Fig. 4.9 - 4.18. As a quantitative measure, the mean of the tracking error is exploited. Tab. 4.3 shows the mean values for position, orientation and elbow error, as well as the joint space error. Again, the best performance is found for RAC controller, then for STT and IMP-PDJ. Concluding, the stiffness control in the task space STT offers the second best trajectory tracking performance after RAC. The mapping of the control gains into the joint space is not effective.

4.3.3 Impedance Display Fidelity

The goal of this experiment is to test the compliance controllers with respect to the quality of the impedance display they provide. The testing method consists of identifying the apparent manipulator impedance and comparing it to the target impedance specified by the controller. The apparent manipulator impedance is found in the following experiment: the manipulator is required to hold a constant position, while a force is manually applied to its end-effector. The experiment comprises a series of manual "push and release" actions performed on the end-effector. The position $\boldsymbol{x}$ and the force $\boldsymbol{F}$ readings are used for identification of the mass, damping and stiffness parameters, as defined in (4.5). Since the impedance itself is a dynamic system, identification of the impedance parameters is not an easy task. The parameters that depend on the position derivatives especially require high velocities and accelerations, which are difficult to realize in the described setup. As a result of this, we have to accept certain amount of error in the identified mass and damping parameters. A more sophisticated apparatus is necessary to find those parameters correctly. However, in contact situations, one can approximate the impedance as pure stiffness, which can be found with high accuracy. Concluding, the force/displacement relationship is identified as zeroth (stiffness only) impedance for slow motion and first and second order impedance for fast motion. Matlab Identification Toolbox is used as the identification tool.

Stiffness Fidelity

For simplicity of presentation, we consider a single DoF case. At the first stage, the apparent stiffness is identified as a quasi-static relationship $f = K\tilde{x}$. The stiffness coefficient K is a good approximation of zeroth order impedance. A relatively low velocity of 0.5 cm/s is applied, so that the damping and inertial terms in the manipulator impedance may be neglected. Although this method is valid only for low velocities, it is robust and gives a good insight into the stiffness properties of the investigated controllers. To compare them quantitatively, a *stiffness fidelity* ζ_K is defined as relationship between the apparent stiffness K_a and the stiffness specified by the controller K_d:

$$\zeta_K = 1 - \frac{|K_d - K_a|}{K_d}. \tag{4.24}$$

In case of perfect stiffness display, $\zeta_K = 1$ is achieved. This test is carried out for the following control schemes: SJB, STT and the combinations IMP-PDJ, IMP-SJB, IMP-STT. It turned out that it is not crucial what kind of motion controller is combined with the impedance controller. The only assumption is that its bandwidth and stiffness are much higher than the desired impedance. Since a high gain joint space controller fulfills this

requirement, we present the results obtained for the IMP-PDJ combination (the results for IMP-SJB and IMP-STT are similar).

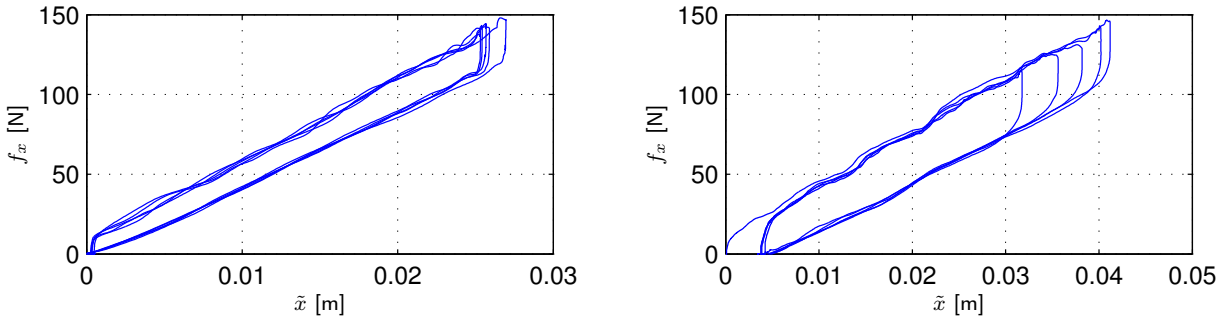


Figure 4.19: Applied force f vs. translational displacement $\tilde{x}$. Left: SJB, $K_d = 10^4$ N/m and $K_a = \frac{140}{0.026} \approx 5384$ N/m. Right: STT, $K_d = 6 \times 10^3$ N/m and $K_a = \frac{140}{0.04} \approx 3500$ N/m

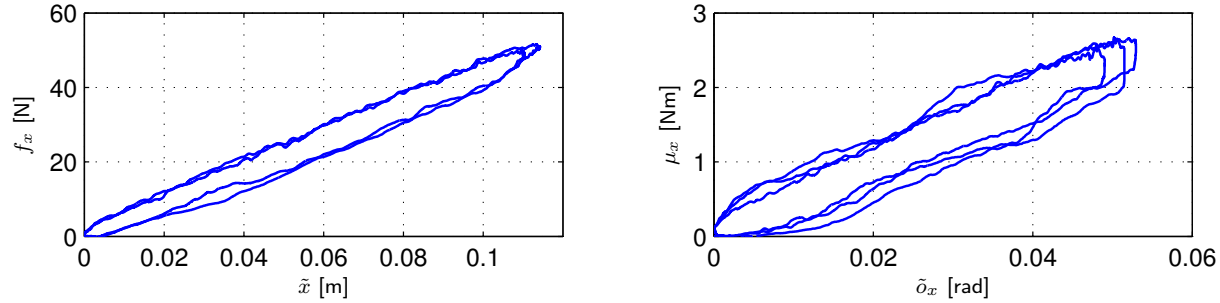


Figure 4.20: Left: applied force f vs. translational displacement $\tilde{x}$ for IMP, $K_d = 450$ N/m and $K_a = \frac{50}{0.11} \approx 454$ N/m. Right: applied moment μ vs. rotational displacement $\tilde{\varphi}$ for impedance controller, $K_d = 50$ Nm/rad and $K_a = \frac{2.5}{0.047} \approx 53$ Nm/rad

Fig. 4.19 illustrates the force/displacement relationship for the x coordinate obtained with the SJB and STT controllers. Fig. 4.20 shows corresponding results for the IMP controller: for one translational and for one rotational DoF. Ideally, the relationship between the applied force (torque) and the resulting translational (rotational) displacement is linear. Instead, we can see a number of curves corresponding to push and release actions. For both stiffness controllers we can see a hysteresis, and for the impedance controller an ellipsoid. The apparent stiffness and the stiffness fidelity measure ζ_K are calculated and listed in Tab. 4.4. As one can see, the performance of the impedance controller is superior

Table 4.4: Apparent stiffness K_a and stiffness fidelity ζ_K

Control Scheme	K_d	K_a	ζ_K
SJB	10000 N/m	5384 N/m	0.5384
STT	6000 N/m	3500 N/m	0.5833
IMP-PDJ	450 N/m	454 N/m	0.991
IMP-PDJ	50 Nm/rad	53 Nm/rad	0.94

to both of the stiffness controllers.

The stiffness fidelity ζ_K in SJB and STT is less than 0.6, which means the apparent stiffness at the end-effector is less than 60 % of the desired stiffness. Such limited stiffness fidelity can be explained by the fact that the manipulator Jacobian has only local meaning; hence the converted stiffness $\boldsymbol{K}_j = \boldsymbol{J}^T \boldsymbol{K}_k \boldsymbol{J}$ can also give only a local relationship between the joint torques and the joint angles. Although this reasoning is correct, it does not explain the limited stiffness fidelity ζ_K with STT, in which the stiffness matrix is not converted into the joint space. The other possible reason for the limited ζ_K is a limited tracking performance. Both stiffness controls still have nonzero steady state error after releasing the applied force, which results in the hysteresis seen on the stiffness plots. This is the result of the imperfect dynamic model of the manipulator used for compensation. Both stiffness controllers rely strictly on the proper generation of the motor torques. In practice, certain part of these torques is wasted for the compensation of model uncertainties, and not used for shaping of the end-effector impedance. Hence the apparent stiffness is smaller than the desired one. This can be proved in a numeric simulation: we simulate the behavior of the SJB and STT controllers for the complete dynamic model of the 7 DoF arm with 100%, 90% and 80% accuracy in the friction part. The results are shown in Fig. 4.21. We observe the

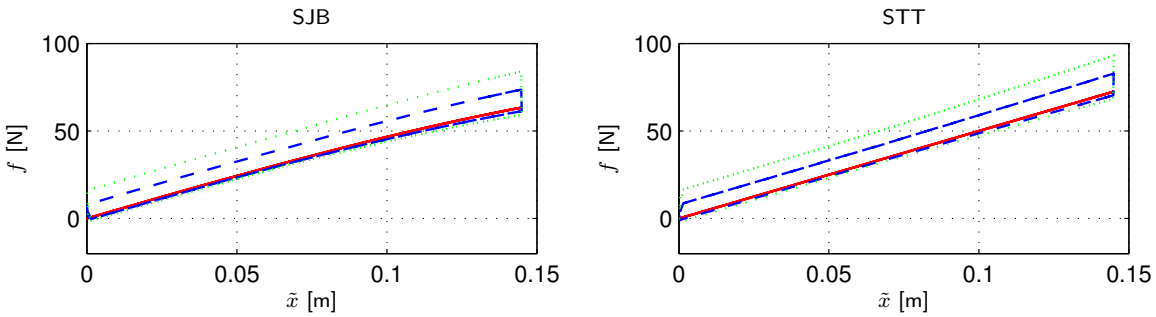


Figure 4.21: Applied force vs. translational displacement for STT and SJB controllers, for 100% (solid line), 90% (dashed line) and 80% (dotted line) accuracy in the friction model, $K = 500$ N/m.

hysteresis proportional in size to the model uncertainty. Moreover, for the SJB controller, we can see the slight bending of the curve for larger displacement. This is due to the only local validity of the $\boldsymbol{K}_j = \boldsymbol{J}^T \boldsymbol{K}_k \boldsymbol{J}$ mapping. For those reasons the desired stiffness values for both the stiffness controllers must have been chosen considerably higher than for the impedance controllers, otherwise we could not obtain sufficient tracking performance.

On the contrary, IMP-PDJ exhibits outstanding fidelity both in translational direction (0.9911) and in rotational direction (0.94), and the end-effector comes back to the initial position after releasing the force. This is due to the fact that the desired impedance is shaped in an outer loop, by means of motion commands and not torque commands. Motion is executed independently in the inner motion control loop by a controller, which is robust against the model inaccuracy. As mentioned before, the experiment confirmed that any motion controller can be applied, provided its sufficient motion bandwidth.

Fig. 4.22 shows the stiffness curves for the stiffness in 6 DoF for a IMP-PDJ combination. The motion controller is a high gain controller, so that modeling errors vanish. Again,we can see outstanding performance of this controller. The elliptic character of the curves is a result of non-zero velocities during the experiment. However, the stiffness fidelity is very good: the desired stiffness is $\boldsymbol{K}_d = \mathrm{diag}(900\ 900\ 900\ 50\ 50\ 50)$ N/m (Nm/rad). The identi-

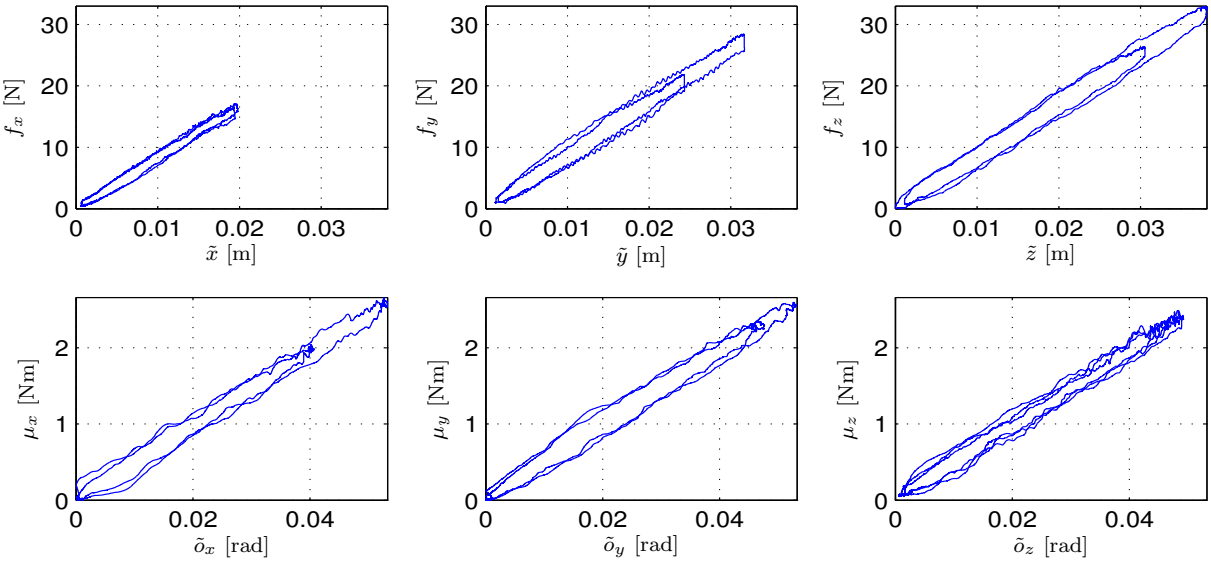


Figure 4.22: Stiffness curves in 6 DoF for impedance controller

fied stiffness in 6 DoF is $\boldsymbol{K}_a = \text{diag}(839.46\ 877.22\ 856.50\ 49.12\ 49.05\ 49.52)$ N/m (Nm/rad) which corresponds to the stiffness fidelity of (0.93 0.97 0.95 0.98 0.98 0.99) for the corresponding coordinates.

Higher Order Dynamics with Impedance Controller

In order to identify the higher order dynamics (first and second order impedance), we perform a similar experiment, but with motion of higher velocity. Although the experiments were performed for all the investigated controllers, the results for STT, SJA and SJB are omitted here for the space reason. The results for IMP controller are gathered in Tab. 4.5. As we can see, the desired stiffness and damping are displayed with a very good

Table 4.5: 6 DoF impedance properties of the impedance controller, the indices d, a, p, o refer to the desired and apparent values of position and orientation impedance parameters

Experiment I								Experiment II							
	M_{pd}	D_{pd}	K_{pd}		M_{od}	D_{od}	K_{od}		M_{pd}	D_{pd}	K_{pd}		M_{od}	D_{od}	K_{od}
	16	240	900	o_x	0.7	10	50		1.6	120	90		0.07	5	5
	M_{pa}	D_{pa}	K_{pa}		M_{oa}	D_{oa}	K_{oa}		M_{pa}	D_{pa}	K_{pa}		M_{oa}	D_{oa}	K_{oa}
x	2.64	236	839	o_x	0.60	10.8	48.1	x	1.30	118.3	90.2	o_x	0.14	4.87	5.37
y	0.77	232	859	o_y	0.59	10.7	49.0	y	1.96	120.2	89.1	o_y	0.10	4.90	5.22
z	8.39	243	856	o_z	0.56	10.5	49.5	z	1.68	120.3	89.1	o_z	0.08	5.00	5.00

fidelity. The identification results suggest that the inertia (mass) part is reproduced less accurately. This is not the case: the large error in the inertia part is a result of the identification method (Matlab Identification Toolbox). In order to identify the inertia properly, higher accelerations are necessary, which in the current setup are not available.

The conclusion drawn from the data for stiffness and damping display fidelity is that the impedance controller works as a very efficient impedance display, and it proved to be the best of all the tested controllers.

Impedance Properties of Joint Space Control

To make sure the motion controller used in combination with the impedance controller acts as a high impedance (stiff) block, it is necessary to investigate the impedance properties of the applied joint space control law. We use a joint space high gain PD controller, with its gains tuned to be $\boldsymbol{K}_j = 10^3\text{diag}(49, 39, 39, 39, 11, 11, 11)$ and $\boldsymbol{D}_j = \text{diag}(90, 90, 90, 90, 54, 54, 54)$. The gains were found empirically, as the highest gains within the stability margin. The resulting Cartesian impedance is configuration-dependent and varies between the different workspace points. For this reason we perform the same experiment at the same position as for the other controllers. This will give us at the insight into the structural impedance of the manipulator. The 6 DoF impedance parameters identified in this experiment are shown in the Tab. 4.6.

Table 4.6: Impedance properties of joint space PD control

	$\boldsymbol{M}_p$	$\boldsymbol{D}_p$	$\boldsymbol{K}_p$		$\boldsymbol{M}_o$	$\boldsymbol{D}_o$	$\boldsymbol{K}_o$
x	13.3	13 463	12 280	o_x	0.51	513	1 773
y	39.3	39 334	26 729	o_y	0.41	1 410	394
z	7	7 490	16 721	o_z	0.6	608	698

As in the previous experiments, the identified mass and damping, due to relatively low velocities, are burdened with considerable amount of error. However, we are interested more in the order of the impedance than in its exact value. The identified impedance is composed of the impedance of the actuators and of the mechanical structure. The resulting values are sufficiently high in compare to those of the impedance set in the IMP controller, see Tab. 4.5. These findings allow to conclude that the PDJ motion controller is stiff enough (has sufficient bandwidth) to be combined with the impedance controller in the outer loop.

4.4 Summary

All candidate control schemes have been tested with respect to the prescribed criteria and the results are summarized in Tab. 4.7, and ranked in a 1 (best) - 5 (worst) scale.

Table 4.7: Comparison of all the candidate control schemes

Control Scheme	Impedance Order	Tracking Performance	Stiffness Fidelity	Null-space Impedance	Computation Burden
SJA	1	5	-	definable	3
SJB	1	4	0.54	definable	3
STT	1	3	0.58	definable	2
RAC	1	1	-	definable	5
IMP-SJB	2	4	$\simeq$1	definable	4
IMP-STT	2	2	$\simeq$1	definable	3
IMP-RAC	2	1	$\simeq$1	definable	5
IMP-PDJ	2	3	$\simeq$1	undefinable	1

Regarding the tracking performance, it is not possible to state any difference between the stiffness control and impedance control with inner position loops, since the impedance control does not modify the desired trajectory in free space motion. The best tracking performance is achieved when the inverse dynamics control is employed in the task space RAC, followed by STT and PDJ. Comparing the control schemes SJB, in the joint space and STT in the task space, the joint space controller has no merits over the task space control. Moreover, converting the stiffness matrix from task space into joint space requires higher computational burden.

In the stiffness control strategies SJA, SJB, STT, high stiffness is required to achieve satisfactory tracking performance, so the manipulator may still experience excessive contact forces. The impedance fidelity for lower impedances is poor. On the contrary, the impedance control IMP-SJB, IMP-STTcan provide superb impedance fidelity in a wide range of displacement, as long as the bandwidth of the underlying motion controller is sufficient.

In order to estimate the computation burden, the execution time of a single computation loop was measured. Since the exact numbers vary in different phases of the experiments, the results are ranked relative to the IMP-PDJ control scheme.

In the succeeding teleoperation experiments, see Sec. 5.5, the IMP-PDJ controller will be used for the cases that do not require null space compliance; otherwise the IMP-STT combination will be applied. RAC method will not be implemented in the teleoperation system, due to its requirement of acceleration input and excessive computational load.

5 Telemanipulation Control Loop

In order to achieve the goals specified in the introduction, i.e., to control the telemanipulator remotely, it is necessary to develop an appropriate control structure. This chapter presents the analysis of the telemanipulation control loop consisting of the haptic input device, the telerobot and a latency-free communication link, see Fig. 1.1. We implement the methods developed within the preceding work at our group [71**?**], and use the system for testing the control algorithms specific to the telerobot. The new aspect in compare to [71] is that we use only one force/position architecture, and one universal controller for both motion in free space and constrained motion in contact. In the work [71] different control strategies were used for different contact situations. Switching between them was reported to be a source of severe stability problems. Our strategy provides a smooth transition from free space to constrained motion. This is possible thanks to the powerful impedance controllers developed in the previous chapter.

This chapter is organized as follows. In Sec. 5.1 the teleoperation control architectures and the methods for their evaluation are summarized. It is shown that the two channel force/position architecture used in the current setup is a special case of the general four channel architecture. The complete control structure and measures of stability and performance are defined in Sec. 5.2. Sections 5.3 and 5.4 introduce the requirements for telerobotic systems that come from psychophysical analysis of human perception. Tuning methods for the impedance parameters of the slave manipulator is formulated as an optimization problem in time and frequency domain, with the purpose of assuring system stability and high transparency. Finally, in Sec. 5.5, the experimental results for a tele-assembly experiment (tightening a screw) in 6 DoF are presented.

Such an experiment is intended to be a benchmark topping the control strategies developed in the previous chapters. Although the analysis shown below is performed for a single DoF case, results may be extended to a 6 DoF case. Thanks to the high performance local controllers, the Cartesian space may be viewed as six independent, non coupled directions.

5.1 Telemanipulation Control Architectures

In order to evaluate the above-mentioned control architectures, it is essential to recall a model of the human arm, the environment, and both master and slave manipulators. The relation between the variables force $f(t)$ and position $x(t)$ is similar to the relation between voltage $u(t)$ and current $i(t)$, which is named as mechanical impedance, see (4.5). The dynamic characteristic of the system that is described by its mass M, viscosity D and stiffness K and represented in Laplace domain as

$$F_i(s) = (M_i s^2 + D_i s + K_i) X_i(s) = Z_i(s) X_i(s), \quad i = \{m, s, h, e\} \tag{5.1}$$

where the subscripts $\{m, s, h, e\}$ stand for master, slave, human and environment. $X_i(s)$ and $F_i(s)$ are the position and the force, and $Z_i(s)$ is the mechanical impedance.

5.1.1 Two-port Model of the Bilateral Teleoperation System

The teleoperator and operator can be modeled as a two-port network element relating force and position of the master manipulator F_m and X_m, to the force and position of the slave manipulator, F_s and X_s, see Fig. 5.1. The master, slave, and communication channel models are lumped into a linear-time-invariant (LTI) master-slave two-port network (MSN) block. According to Hannaford [44] it can be represented by a two-port network hybrid

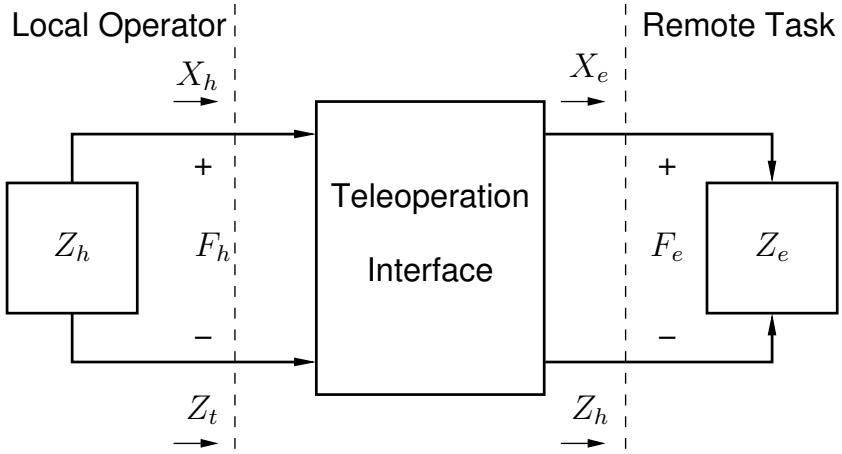


Figure 5.1: General two-port network model

matrix:

$$\begin{bmatrix} F_h \\ -X_e \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} X_h \\ F_e \end{bmatrix} = \boldsymbol{H} \begin{bmatrix} X_h \\ F_e \end{bmatrix}, \tag{5.2}$$

where F_h, F_e are the force exerted on the master and on the slave, X_h, X_e are the operator and the environment positions, respectively; h_{ij} $(i, j = 1, 2)$ represent the hybrid parameters that are functions of the master, the slave dynamics, and their control parameters. Note that the impedances are not defined as force/velocity but force/position relationship. Although the force/velocity representation has the advantage that the power is immediately given by the terminal variables of the two port, it introduces a pole/zero pair at the origin that causes complications in stability analysis, which is purely an artifact of the representation [19]. Here, the force/position representation is used to avoid these complications. In the following a linear latency-free model will be considered as the primary physical model throughout the analysis. The two-port network representation will be used to analyze the transparency of the teleoperation system.

5.1.2 Four Channel Bilateral Control Architecture

Important issues for a haptic system are the performance evaluation and controller design for providing a stable high-fidelity system.

- Stability is of primary concern in feedback control systems. In a teleoperation system, instability can cause an undesirable feeling to the user that distorts the transparent interaction with the environment. It can also be dangerous if the manipulator can output high force or velocities.

- Once the stability criterion is satisfied, the performance of the system is evaluated. Position/force tracking and fidelity of the displayed impedance are the two measures employed to determine the performance of the system. Tracking refers to a measure of how well the slave (master) manipulator can follow the position (force) commanded by the master (slave) manipulator. Transparency measure is the degree of distortion of the feeling between the operator and the remote environment.

In 1993, Lawrence proposed a unified four-channel bilateral control architecture [74] that communicates the sensed forces and positions from the master to the slave, and vice versa. Fig. 5.2 shows a block diagram of a four-channel teleoperation system with master, slave

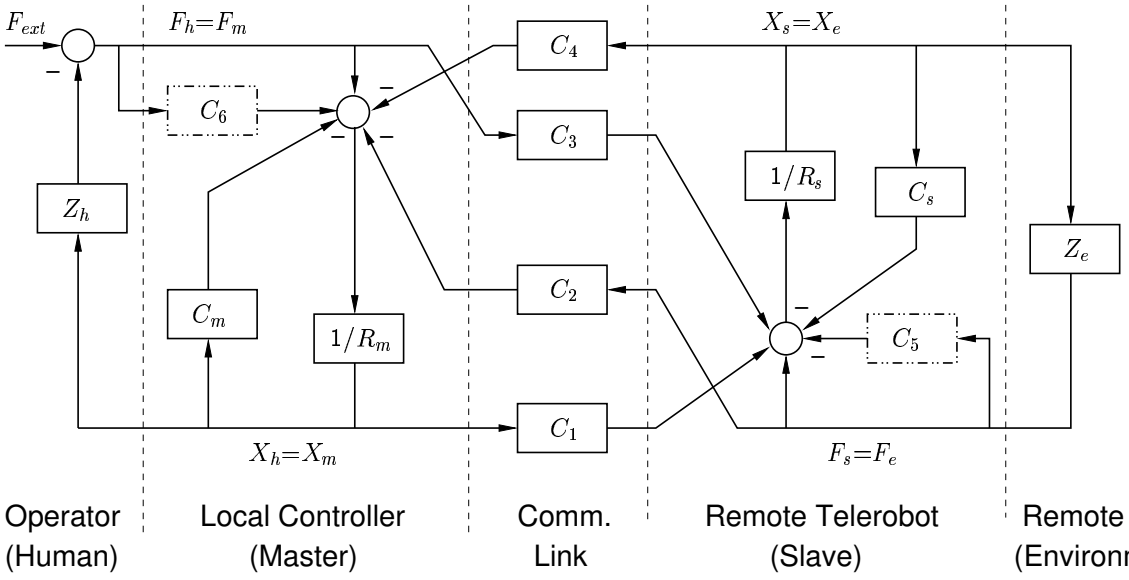


Figure 5.2: General bilateral control architecture of Lawrence

and communication link models, as well as, operator and environments models. Lawrence asserted that all four channels should be used to obtain transparency. By using all four channels, the dynamic equations of the master and the slave are obtained:

$$R_m X_m = F_m - C_m X_m - C_4 X_s - C_2 F_s \tag{5.3}$$
$$R_s X_s = C_1 X_m + C_3 F_m - C_s X_s - F_s \tag{5.4}$$

where R_m and R_s denote the linear dynamic models of the master and slave respectively, and C_m and C_s represent local position controllers of master and slave. Independently of Lawrence, Yokokohji [142] has developed a general control architecture that is similar to that of Lawrence. The exclusive difference is the local force feedback (the dashed blocks) that is appended on the master and slave sides. Therefore the equations are transformed as follows:

$$R_m X_m = F_m + C_6 F_m - C_m X_m - C_4 X_s - C_2 F_s \tag{5.5}$$
$$R_s X_s = C_1 X_m + C_3 F_m - C_s X_s - C_5 F_s - F_s \tag{5.6}$$

The hybrid parameters h_{ij} in (5.2) are represented then as:

$$h_{11} = \frac{(R_m + C_m)(R_s + C_s) + C_1 C_4}{(1 + C_6)(R_s + C_s) - C_3 C_4} \tag{5.7}$$

$$h_{12} = \frac{C_2(R_s + C_s) - C_4(1 + C_5)}{(1 + C_6)(R_s + C_s) - C_3 C_4} \tag{5.8}$$

$$h_{21} = -\frac{C_3(R_m + C_m) + C_1(1 + C_6)}{(1 + C_6)(R_s + C_s) - C_3 C_4} \tag{5.9}$$

$$h_{22} = \frac{(1 + C_5)(1 + C_6) - C_2 C_3}{(1 + C_6)(R_s + C_s) - C_3 C_4} \tag{5.10}$$

5.1.3 Stability and Performance Analysis of a Two-port System

Transparency and stability are the two most important factors in teleoperation systems, on the other hand the relation between them is a conflicting issue. There are many different criteria to judge the stability and performance of a teleoperation system. According to [45], an LTI two-port network (5.2) is absolutely stable if and only if:

- the hybrid parameters h_{11} and h_{22} have no poles in the open right-half-plane
- any poles of h_{11} and h_{22} on the imaginary axis are simple and have real and positive residues
- the inequalities hold:

$$\mathrm{Re}\{\mathrm{h}_{11}\} \geq 0 \quad \wedge \quad -\frac{\mathrm{Re}\{\mathrm{h}_{12}\mathrm{h}_{21}\}}{|\mathrm{h}_{12}\mathrm{h}_{21}|} + 2\frac{\mathrm{Re}\{\mathrm{h}_{11}\}\mathrm{Re}\{\mathrm{h}_{22}\}}{|\mathrm{h}_{12}\mathrm{h}_{21}|} \geq 1. \tag{5.11}$$

Those criteria are called *Llewellyn's absolute stability criteria* and depends on the network parameters alone and are not subject to the operator or environment.

Besides stability, transparency is the principal goal of the teleoperation system. According to Lawrence, perfect transparency is achieved if the impedance transmitted to the human operator is equivalent to the environmental impedance [74]

$$Z_t = Z_e, \tag{5.12}$$

where Z_t, Z_e represent the transmitted impedance perceived by the operator through the teleoperation system and the environment impedance, respectively. Z_t can be expressed in the light of the hybrid parameters as [45]:

$$Z_t = \frac{h_{11} + \Delta h \cdot Z_e}{1 + h_{22} Z_e}, \tag{5.13}$$

where $\Delta h := h_{11}h_{22} - h_{12}h_{21}$. If the hybrid parameters are not functions of Z_h and Z_e, the complete transparency condition can be expressed as:

$$\boldsymbol{H} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{5.14}$$

In order to judge the performance, one have to investigate the concrete form of each hybrid parameter. To simplify the analysis of the telepresence and to quantify the transparency, one can use Z_t as a criterion, which is examined for extreme values of Z_e:

$$Z_{tmin} := Z_t|_{Z_e=0} = h_{11} \tag{5.15}$$

$$Z_{twidth} := Z_t|_{Ze\to\infty} - Z_{tmin} = \frac{-h_{12}h_{21}}{h_{22}} \tag{5.16}$$

Ideally, the good performance (complete transparency) is characterized by $|Z_{tmin}| \to 0$ and $|Z_{twidth}| \to \infty$. This expression is equal to the condition (5.14).

5.1.4 Two Channel Bilateral Control Architectures

Two-channel control architectures are the simplest and most intuitive architectures. In contrast to the four-channel architecture, the two-channel architecture means the relevant constraints are equal to zero. In the following paragraph four typical two-channel bilateral control architectures will be analyzed [45], in order to select the most appropriate one.

1. **Force-Position** architecture
   In **F-P** architecture, which is known as flow forward or force feedback, the constraints $C_3 = 0 \wedge C_4 = 0$, see Fig. 5.2. Namely, the master position is sent as a command to the slave, while the interaction force at the slave is sent back directly as a reaction force to the master. In terms of the two-port model and the complete transparency condition (5.14), the control parameters for satisfying perfect transparency must be set as follows:
   $$\begin{cases} C_1 = R_s + C_s \neq 0; \\ C_2 = 1 + C_6 \neq 0; \\ C_5 = -1; \\ C_m = -R_m. \end{cases} \tag{5.17}$$

2. **Position-Force** architecture
   Similarly, in **P-F** architecture the constraints $C_1 = 0 \wedge C_2 = 0$; the idea is to send the interaction force at the master as a reaction force to the slave, and the slave position is passed to the master. The transparency condition (5.14) is specified as follows:
   $$\begin{cases} C_3 = 1 + C_5 \neq 0; \\ C_4 = -(R_m + C_m) \neq 0; \\ C_6 = -1; \\ C_s = -R_s. \end{cases} \tag{5.18}$$

   We can see that the **complete transparency can be achieved by using the F-P and P-F architectures**, if and only if we select appropriate control parameters $C_1, ..., C_6$ and C_S.

3. **Position-Position** architecture
   This architecture means the constraints $C_2 = 0 \wedge C_3 = 0$. To satisfy the transparency condition, C_5 should be '-1', but it will result in $h_{12} = 0$ and **the transparency condition (5.14) cannot be satisfied**.

4. **Force-Force** architecture
   Similarly, this architecture means the constraints $C_1 = 0 \wedge C_4 = 0$. Likewise, to satisfy the transparency condition, C_m should be '-1', but it will result in $h_{21} = 0$ and **the transparency condition (5.14) cannot be satisfied.**

From the above-mentioned two architectures, we can see that **transparent teleoperation is impossible in the P-P and F-F architectures**. As a conclusion, in the practical implementation presented in the following section, the **F-P** architecture will be exploited. On the contrast to the previous research on the teleoperation conducted at the Technische Universität München, see [71], where different controllers are used for different contact situations, the method applied here avoids switching of the control mode applying only one controller for both free space motion and for the contact.

5.2 Analysis of the Teleoperation Control Loop

The present bilateral model is constructed on the basis of the above-mentioned analysis of the **F-P** two-channel architecture with local force feedback, namely $C_3 = C_4 = 0$, C_5 and C_6 are not equal to zero at the same time. To simplify the analysis, only one degree of freedom will be taken into account, from which the conclusion is applied to the other dimensions.

Fig. 5.3 shows the functional model of the present teleoperation system. The master

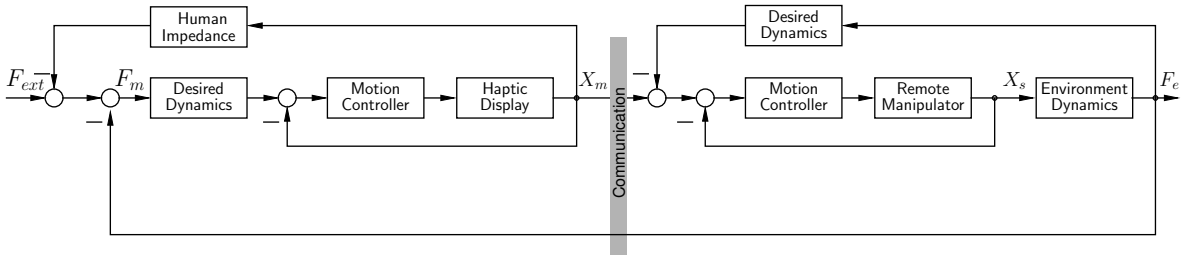


Figure 5.3: Simulation model of the present teleoperation system

site consists of the 'Haptic Display' with corresponding 'Motion Controller', the dynamic behavior of the haptic display is governed by the 'Desired Dynamics' block. The human operator is modeled as a force source F_{ext} and passive impedance of the arm, named 'Human Impedance'. The slave site has a similar structure, plus the environment modeled as 'Environment Dynamics' block. The basic functionality of this system may be described as follows. The human operator exerts a force F_{ext} on the master force sensor. According to the desired dynamics, a motion is generated and executed on the haptic display using the motion controller. The actual motion X_m of the master is measured and transmitted via communication link to the remote manipulator and executed there by the slave motion controller. When the remote manipulator comes in contact with the remote environment, a force F_e is sensed, transmitted back to the master controller and displayed to the operator. According to the desired slave dynamics force F_e is also used for modifying the X_m trajectory, resulting in the X_s trajectory.

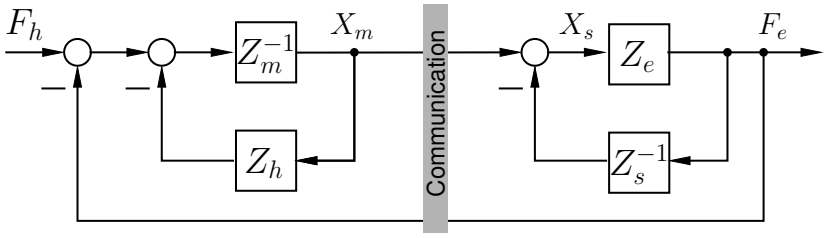


Figure 5.4: Simulation model of the present teleoperation system

Provided we have fast, high bandwidth motion controllers and perfect sensors and actuators, we can considerably simplify the structure, to the one shown in Fig. 5.4, where the Z_m and Z_s are the desired master and slave impedances, Z_h is the impedance of the operator arm (human), Z_e is the environment impedance. All impedances Z_i have the form of (5.1). The following relations may be written:

- closed loop transfer function

$$G_f = \frac{F_e}{F_{ext}} = \frac{Z_e Z_s}{Z_e Z_s + (Z_h + Z_m)(Z_s + Z_e)} \tag{5.19}$$

- open loop transfer function

$$G_o = \frac{Z_e Z_s}{(Z_h + Z_m)(Z_s + Z_e)} \tag{5.20}$$

- impedance displayed to the operator (transmitted impedance)

$$Z_t = Z_h + Z_m + \frac{Z_e Z_s}{Z_s + Z_e} \tag{5.21}$$

- transparency of the whole system

$$G_t = \frac{Z_e}{Z_t} = \frac{Z_e(Z_e + Z_s)}{(Z_h + Z_m)(Z_s + Z_e) + Z_e Z_s}. \tag{5.22}$$

We will show that the presented structure is a special case of the general bilateral control architecture from Sec, 5.1.2. From (5.7) and the precondition of the **F-P** architecture (5.14), the hybrid parameters are written as follows:

$$\begin{array}{ll} h_{11} = \frac{R_m + C_m}{1 + C_6} & h_{12} = \frac{C_2}{1 + C_6} \\ h_{21} = \frac{-C_1}{R_s + C_s} & h_{22} = \frac{1 + C_5}{R_s + C_s} \end{array}. \tag{5.23}$$

To find the relationship between the general bilateral control architecture and the practically implemented model, we perform a comparison of Fig. 5.2 and Fig. 5.4. On the slave side, the relationship between the X_s and X_m are obtained through the calculation of the transfer functions:

$$\frac{X_s}{X_m} = \frac{C_1}{R_s + C_s + C_5 + Z_e} \tag{5.24}$$

$$\frac{X_s}{X_m} = \frac{Z_s}{Z_s + Z_e}. \tag{5.25}$$

Comparing (5.24) and (5.25) , and from the transparency condition $h_{22} = 0$, we obtain $C_1 = Z_s$, $C_5 = -1$, $R_s + C_s = Z_s + 1$. Similarly, we can write the signal equations for the master side as follows:

$$(R_m + C_m)X_m = (F_{ext} - Z_h X_m)(1 + C_6) - C_2 F_e \quad (5.26)$$

$$Z_m X_m = F_{ext} - Z_h X_m - F_e, \quad (5.27)$$

which leads to $C_2 = 1$, $C_6 = 0$ and $R_m + C_m = Z_m$. The hybrid matrix of our present system takes the form

$$\boldsymbol{H} = \begin{bmatrix} Z_m & 1 \\ \frac{-Z_s}{Z_s+1} & 0 \end{bmatrix}. \quad (5.28)$$

Note on the implementation

The $Z_s = M_s s^2 + D_s s + K_s$ and $Z_m = M_m s^2 + D_m s + K_m$ are the desired impedances of the master and slave subsystems. We assume they are freely tunable, on the assumption that perfect sensor/actuator and controllers are present in the system. At this point it is necessary to mention that in order to achieve the functionality of a force controlled device, the stiffness K_m and damping D_m of the master must be set to zero, so that $Z_m = M_m s^2$.

Transparency Analysis

According to (5.12), to achieve perfect transparency, the transmitted impedance should be the same as the environment impedance $Z_t = Z_e$. To achieve this, the hybrid parameters in matrix (5.28) must fulfill the following conditions: $h_{11} = h_{22} = 0$ and $h_{12}h_{21} = -1$, so the first conclusion regarding transparency may be drawn. Perfect transparency is not attainable in the present system, because it requires setting $Z_m = 0$ and $Z_s = \infty$. This is not possible, due to hardware constraints. Further we will analyze how close we can get to the perfect transparency using only Z_m and Z_s as tuning parameters. The transparency function in the frequency domain is defined as follows:

$$G_t = \frac{Z_t}{Z_e} = \frac{(Z_m + Z_h)(Z_e + Z_s) + Z_e Z_s}{Z_e(Z_e + Z_s)}, \quad (5.29)$$

where Z_t is given by $Z_t{=}F_{ext}/X_m$. Ideally, complete transparency in the single degree-of-freedom case requires that $G_t{=}1$. Practically, sufficient transparency is such that the magnitude of the transparency function G_t is unity and the phase is zero within a certain bandwidth larger than the sensory and motor bandwidth of the human operator [117], which is usually up to 20 Hz.

It is interesting to analyze the transparency from the steady-state point of view, which may also be seen as a low frequency approximation. According to the final-value theorem

$$\lim_{t\to\infty} G_t = \frac{K_e K_s + K_h(K_e + K_s)}{K_e(K_e + K_s)}, \quad (5.30)$$

and the requirement $G_t = 1$, the parameter K_s, which is the only tunable variable should satisfy the following equation:

$$K_s = \frac{K_e^2 - K_h K_e}{K_h}. \quad (5.31)$$

The conclusion is drawn that high stiffness of the slave side leads to better transparency in the case of contact with very stiff environment, and with respect to contact with soft environment, the stiffness of the slave side should not be high.

If the impedance of human can be neglected, the above formula is rewritten as

$$\lim_{t\to\infty} G_t = \frac{K_s}{K_e + K_s} = \frac{1}{\frac{K_e}{K_s} + 1}, \tag{5.32}$$

so that the best transparency is achieved if the $K_e = 0$ or $K_s = \infty$.

Since we are only interested in the certain frequency domain, which is determined by the sensory and motor bandwidth of the operator [117], the above-mentioned method can be expressed in the frequency domain as follows: the magnitude response should be equal zero dB and the phase response to zero degree. In order to find out his frequency range, we review shortly the human psychophysic characteristics.

5.3 Telerobotic Response Requirement

Telepresence represents an upper bound on the required response of a teleoperation system because the human cannot utilize or command data beyond one's capacity. Namely, there is a frequency range or bandwidth, within which one can perceive the force, and additional bandwidth or response capability would simply go unused. The operator in the telepresence system acts in dual roles. On the one hand, as a signal generator that sends position signal to the telerobot. On the other hand, as a receiver with an inherent input signal bandwidth limitation. According to [73], humans have asymmetrical input/output capabilities. Namely, the bandwidth of the human output is more narrow than of the human input. The haptic perception is a complex exploratory act integrating two distinct modes:

- Tactile sensors provides information about compressive stress ($\sim$ 10 Hz), skin motion stimulus ($\sim$ 30 Hz), vibration ($50 - 400$ Hz) and skin stretch (very low frequency).

- Proprioceptors sense the muscle contraction and tension, while kinesthetic receptors sense the angle and velocity of joint movement which is useful for the detection of the transmitted impedance. About the exact bandwidth of these forms of sensory there is no explicit assertion, and the estimates of proprioceptive/kinesthic bandwidths are in the range of 20-30 Hz. The actual bandwidth is considerably less, and reported to be 7 Hz [73].

- It is known that the long time behavior is determined by low frequencies. If the operator moves very slowly, then this behavior may be considered as the quasi steady-state, with the bandwidth of maximum 1.2 Hz [73].

On the basis of the above analysis, different frequency ranges will be selected to define the quality function to achieve better transparency.

5.4 Controller Tuning

The goal of the present controller design is the increase of the performance of the closed loop telemanipulation system. The performance will be assessed by means of the transparency defined as

- fidelity of the transmitted impedance $Z_t = Z_e$
- force tracking in contact with the environment $F_m = F_s$
- position tracking in free space motion $X_m = X_s$.

It is important to mention that due to the built-in compliance in the slave manipulator, the position tracking in contact will not be perfect. Though in constrained motion, it is the force tracking that dominates the system performance, so in the following only the first two criteria will be considered.

A number of simplifications are executed to make the analysis tractable:

- The analysis is performed for 1 DoF only, based on the assumption that the DoFs are not internally coupled. This feature is provided by the local impedance controllers.
- Due to the specific implementation of the impedance controller on the mater side, see [131, 132], the minimum impedance that does not destabilize the master controller is $Z_m = M_m s^2$ with $M_m = 10$ kg for translational DoFs. The damping D_m and stiffness K_m are set to zero.
- The actuators and sensors are taken for perfect, i.e., the apparent impedance is identical with the one assigned in the controller, and the signals read from sensors are noise-free and not delayed. For an analysis of the system with imperfect actuators/sensors, see [71].
- We assume the impedance parameters of the human arm are constant $M_h = 0.8$, $D_h = 5$, $K_h = 400$ and come from biomechanical research of [83].
- Because we concentrate on the *tele-assembly* tasks (see the experimental section), the main assignment for the whole system is to provide the user with sufficient haptic information about the shape and stiffness of the environment. We do not plan transportation tasks or working in a viscous environments. As a result, we assume the environment impedance consists of pure stiffness $Z_e = K_e$.

In the following we find the optimal values of the only tunable parameters in the loop: the impedance parameters of the slave manipulator Z_s.

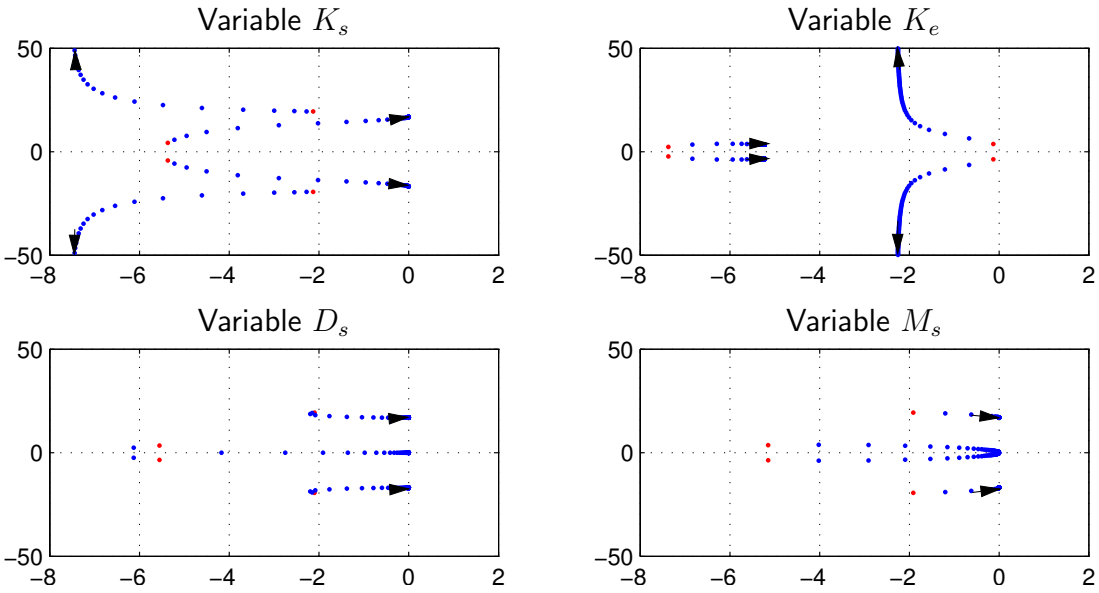


Figure 5.5: Influence of slave impedance parameters, and the environment stiffness on the root locus of the closed loop teleoperation system.

Stability Analysis

Assuring the stability of the whole system is, in case of negligible latencies, conceptually very straightforward. Either we check if the Llewelyn stability condition in (5.11) is fulfilled, or we prove if the denominator of the closed loop system in (5.19) is Hurwitz. Since the problem is multidimensional, it is not obvious how to choose the tunable parameters. Therefore it is advantageous to investigate how they influence the closed loop root locus, shown in Fig. 5.5. As we can see, all parameters may destabilize the closed loop after having exceeding some boundaries. The above method allows for checking if the chosen controller parameters result in a stable system. However they do not provide any tool for assessing and/or increasing the system performance. This shall be the subject of the following sections.

The Quality Function in Frequency Domain

To find the optimized parameters, it is important to define an appropriate quality function. To optimize the transparency, the quality function originates directly from the equation

$$G_t(j\omega) \equiv 1. \tag{5.33}$$

Ideal transparency means that the amplitude $A(\omega) =\mid G_t(j\omega) \mid = 0$ dB and the phase shift $\varphi(\omega) = \mathrm{Arg}(G_t(j\omega)) = 0$ rad within definite frequency domain $0 \leq \omega \leq \omega_n$. Since G_t may be seen as a dynamic system that is a minimum phase system (it does not have zeros in the right half s plane), and according to the character of the minimum phase system, stating that $\varphi(\omega)$ can be calculated from $A(\omega)$, we can considerably simplify the analysis focusing only on the amplitude curve. So the quality function is given as follows:

$$I_A = \int_0^{\omega_n} \mid 20 \log(A(G_t(j\omega))) \mid \mathrm{d}(\log \omega). \tag{5.34}$$

According to [114], the transparency measure allows to determine the frequency range within which one would expect the system to accurately transmit impedance. Hence, the main task is to select a rational frequency range ω_n. Through the simulation and the practical experience, one can see that the bandwidth of the present bilateral telemanipulation system and transparency transfer function is narrow, which is lower than 6 Hz in the absence of time delay. Additionally, these two bandwidths depend on the environment, i.e., rigid objects and sharp edges require large bandwidth.

- According to the previous section, from the psychophysic point of view, an upper bound on human force control bandwidth is about 20 Hz. So the frequency range that will be used in quality function is limited to $0 - 20$ Hz.

- Since such bandwidth is not actually necessary, and not achievable with the present hardware, we restrict the frequency range to $0 - 1.2$ Hz, similarly to [71].

The Quality Function in Time Domain

In time domain, the ideal transparency is equivalent to ideal force or position tracking on both sides. However, force tracking plays a dominating role in the case of contact with environment. Hence, in order to investigate the influence of the force tracking error on the transparency, we can also define the force error as a new quality function. From Fig. 5.4 we can deduce the error transfer function as follows:

$$G_{error} = \frac{1 + Z_h Z_m + Z_e Z_s + Z_e Z_h Z_m Z_s}{1 + Z_e Z_m + Z_h Z_m + Z_e Z_s + Z_e Z_h Z_m Z_s} \tag{5.35}$$

Because of the contradiction between the steady-state error and the dynamic character, only the tracking error within two seconds is taken into account, and the settling time is not taken into account. Hence, the quality function can be expressed as:

$$I_f = \int_0^2 \mid e_f \mid^2 \mathrm{d}t, \tag{5.36}$$

where

$$e_f = f_m - f_s.$$

On the basis of the identified environment impedance, the corresponding optimized parameters of the slave impedance are found using the optimization toolbox in MATLAB (The MathWorks, Inc.). Six optimization results have been obtained by employing different quality functions for two different environment objects are shown in Tab. 5.1.

The impedance parameters of the slave robot must be tuned for a concrete task, i.e., the environment impedance must be known in advance. Strategies handling multiple environment types, and very rigid environments are subject to further research. For the assembly task described in the next section, a heuristically tuned controller is used, as a low bandwidth, critically damped impedance $Z_{sp} = 16s^2 + 240s + 900$ for translational DoFs and $Z_{so} = 0.7s^2 + 10s + 100$ for rotational DoFs.

Table 5.1: The optimal Z_s for different quality functions

Environment	Quality function	Z_s
sponge $K_e \approx 850$ N/m	I_A $(0 - 1.2$ Hz)	$30s^2 + 328.76s + 932.92$
	I_A $(0 - 20$ Hz)	$6.56s^2 + 60s + 278.20$
	I_f	$20s^2 + 94.6s + 2634.7$
rubber $K_e \approx 2400$ N/m	I_A $(0 - 1.2$ Hz)	$30s^2 + 3000s + 9845.4$
	I_A $(0 - 20$ Hz)	$10.3s^2 + 189.1s + 1566.2$
	I_f	$14.3s^2 + 193.8s + 4136.7$

5.5 Teleoperation Experiment

The experimental setup, see Fig. 5.6, consists of the haptic input device VisHaRD10, the 7 DoF slave manipulator and the stereo vision system. According to the bilateral control structure, the motion of the operator is read by the master device and sent as desired positions to the slave impedance controller. The measured contact forces are sent back as the input to the master admittance controller. The devices communicate over the UDP network with a sampling rate of 1 kHz, which is the same as for the local loop control.

Both manipulators are built using commercially available components combined with aluminum/steel construction elements. The actuation torque is provided by DC-motors coupled with harmonic drive gears offering zero backlash. The motors are actuated by PWM-amplifiers supplying a control of the motor current at a bandwidth of 2.5 kHz. In order to permit force feedback control the devices are equipped with a six-axis JR3 force-torque sensor providing a bandwidth of 8 kHz at a comparatively low noise level. The joint angles are measured by digital MR-encoders with a resolution of 4 096 counts per revolution, resulting in a comparatively high position resolution when multiplied with the corresponding gear ratio. For the particulars regarding the slave construction see App. A.3 and for the details of the haptic interface VisHaRD10, see [131, 132].

The vision system consists of two CCD cameras placed on a 3 DoF camera head. The recorded video streams are displayed on the head mounted display (HMD) carried by the operator. The HMD has a built in tracker, which is used for controlling of the motion of the camera head. Such a setup provides the operator with a realistic visual information about the location of the objects, the environments, and the telemanipulator. Here the anthropomorphic construction of the telemanipulator plays an important role: the operator can drive it as if it were his/ her own arm. The visual information is useful not only for motion generation, but also for handling the contact and minimizing the effects of impact.

The experiment consists of three tasks:

- tracking of free space motion
- haptic exploration of different materials (soft and stiff), see Fig. 5.7a
- driving a screw with an aluminum tool, see Fig. 5.7b. This last experiment consists of three phases: contact with extreme stiff materials, a classic peg - in- hole operation and manipulation in a constrained environment.

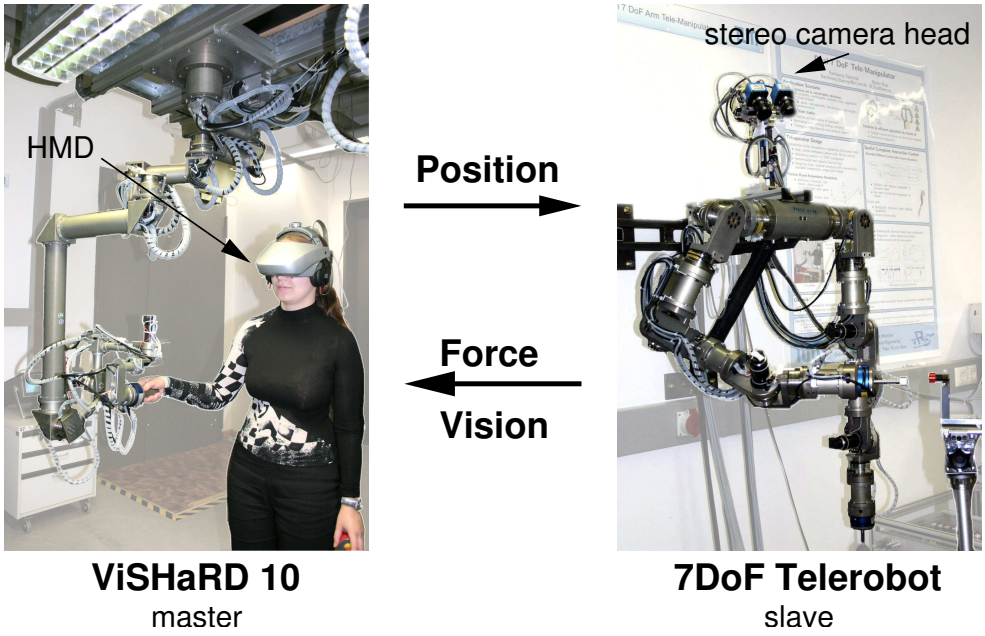


Figure 5.6: Experimental system architecture

Fig. 5.8 and 5.9 show the position and force tracking performance during haptic exploration of different materials (see Fig. 5.7a). The shaded areas indicate several contact phases. One can see that during free space motion the position tracking of the slave arm works very well, while in the contact situation, as a consequence of the implemented impedance controller, the slave position differs from the master position. Please note that as the force tracking is very good, this position displacement influences the displayed and felt environmental impedance in such a way that hard objects are perceived softer then they are. As the master controller is of admittance type, which reacts on the human force input, non-zero forces during free space motion are necessary to change the actual end-effector position.

Fig. 5.10 and 5.11 show the position and force tracking performance during the two screwing phases. Screw tightening differs from the simple exploration scenario as more than one translational and rotational constraint is active at the same time. One can see that the position displacement is small in comparison with the displacement during the exploration experiments. Also it can be noticed that the force tracking is outstanding in all translational and rotational directions. As the y-direction represents the actual screwing axis, torques around it should be interpreted as human torque inputs necessary to change the robot end-effector orientation. The active compliance introduced by the impedance control of the slave arm emulates a human like compliant behavior when interacting with the environment and enables screw tightening without destruction of the screw and the tool.

5.6 Summary

This chapter introduced the concepts of bilateral teleoperation. Issues regarding stability and performance in a two channel architecture were discussed. A simple tuning method for the slave impedance based on the time and frequency domain performance indices was given. As a benchmarking experiment for the motion and interaction control strategies,

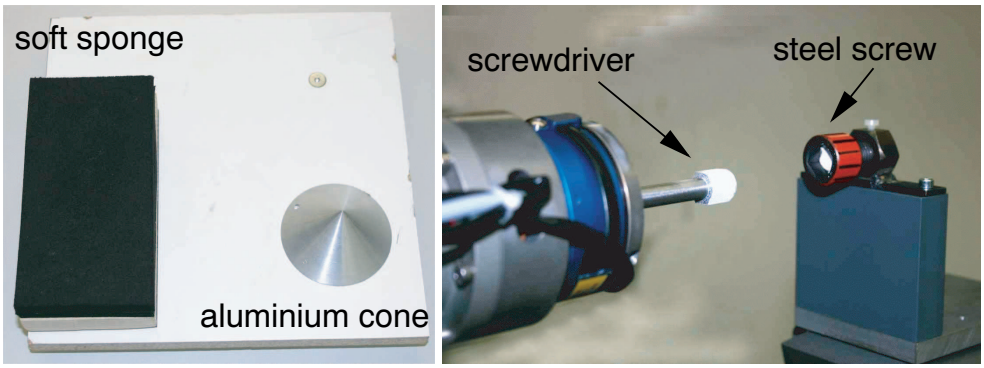


Figure 5.7: Slave side: a) different materials for haptic exploration b) the screw and the screwdriver

a tele-assembly experiment in 6 DoF was performed. As the human - robot interface, a 10 DoF haptic interface was used. The experimental results show very good tracking of positions in the free space motion and forces in contact states. The deterioration of the tracking performance in contact is the result of relatively low impedance of the slave robot and could be avoided at the moment. Future work in the direction of variable impedance control is necessary.

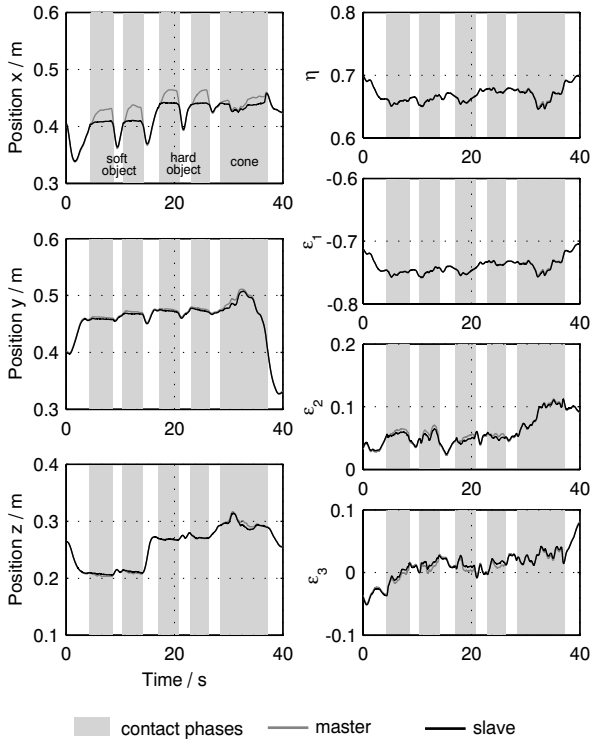


Figure 5.8: Position tracking (Slave) during haptic exploration experiment

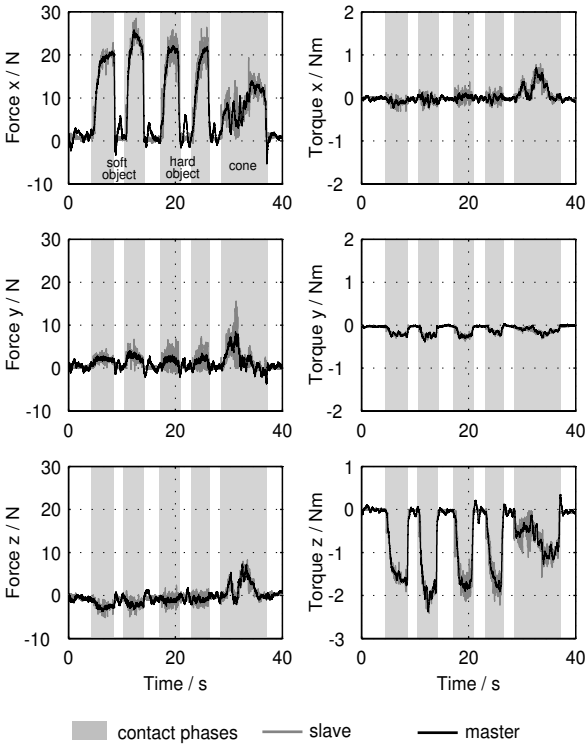


Figure 5.9: Force tracking (Master) during haptic exploration experiment

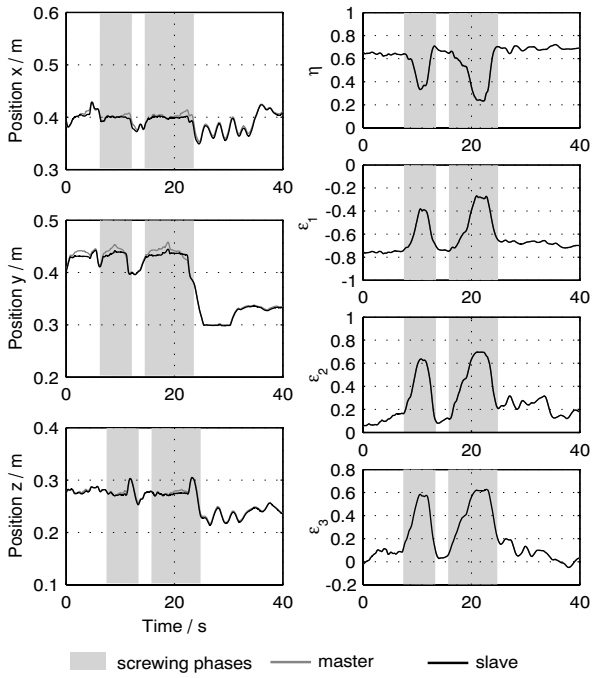


Figure 5.10: Position tracking (Slave) during screw tightening experiment

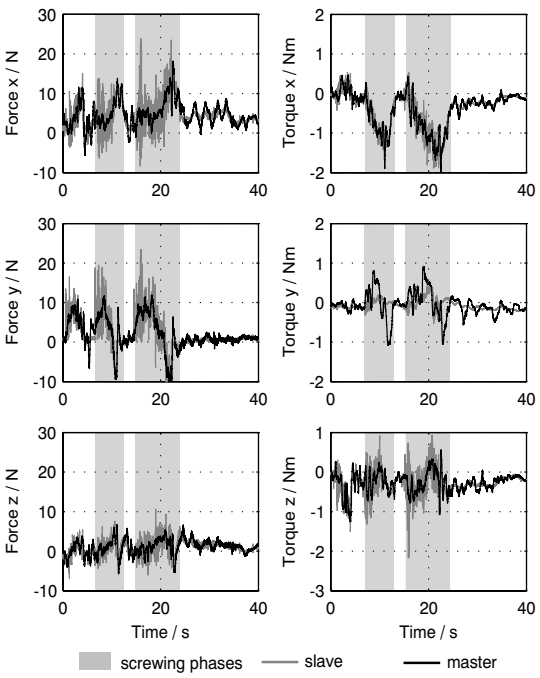


Figure 5.11: Force tracking (Master) during screw tightening experiment

6 Collision Avoidance with Virtual Forces in Dual Arm Teleoperation

This chapter is devoted to coordinated control of a dual-arm robotic system in teleoperation specific manner. Robotic task capability can be greatly expanded by having two arms work together. Although this is an inherent capability of humans and animals, dual arm collaboration is an unproven robotic technology that requires complex geometric reasoning and motion control. In multi-robotic systems *sharing the common workspace*, the major operational constraint is that of *collision detection and avoidance*. Due to fatal consequences of collisions, avoiding them is crucial in practical applications. It is necessary to underscore the major difference between the teleoperated and autonomous modes in cooperative control of manipulators. In teleoperation, tasks related to motion planning and force closure are by definition entrusted to the human operator, and the telerobotic system is to execute them as closely as possible. This may be the reason that collision avoidance is hardly addressed in the teleoperation related literature, as being not necessary. This is true as long as a single non-redundant telemanipulator is considered. In a redundant multi robotic system, detecting and avoiding collisions on a local level is absolutely essential. This is due to the fact that the manipulators are driven by means of the end-effector coordinates only. The operator controls exclusively the motion of the robotic "hands". The motion of the other manipulation links is controlled by a local controller, and the redundant manipulator structure may lead to collisions.

Research in the area of collision avoidance, viewed from the perspective of teleoperation, may be broadly divided into two categories: *global* and *local*. The *global* methodologies are applied on the task planning level and require that the task and environment model are known prior to the task execution. Because those conditions are not fulfilled in a teleoperation system, such methods will not be considered further on. As *local* methods we understand those, which can be potentially applied to unknown and dynamically changing tasks; e.g., in a dual arm system, each arm becomes a moving obstacle for the other arm.

The teleoperation-related approaches to the collision avoidance problem [72, 143] include user involvement by means of a visual display to alert one to the collision danger. Except the conceptual simplicity, such methods are unacceptable in high performance teleoperation. They draw the user attention away from the main task, tending to increase of the user burden. In addition to that, the reaction time to visual information is high, in compare to force or tactile sensing.

In the following chapter a method for collision avoidance between the teleoperated arms is introduced, utilizing the intuitive kinesthetic feedback for collision display. Since in our current teleoperation system the workspace of the master device is larger than this of the telerobot, the collisions with the workspace boundaries must also be handled. The method described here considers all types of collisions within on uniform framework.

This chapter has the following structure. First, the possible collisions are classified into *avoidable* and *unavoidable* collisions. The avoidable collisions are those which can be escaped exploiting the self motion of the redundant manipulator, e.g., hand to elbow

collision. It is important that the main task, i.e., tracking the end-effector trajectories, must not be influenced by this motion. All other collisions are classified as *unavoidable*. Second, an algorithm for computing virtual forces related to the distance to collision is introduced. Such a force mitigates against and finally stops the robot motion in the endangered directions. It is also sent to the human operator so that it feels like virtual spring keeping him/her from entering the forbidden regions. It should be stressed that this approach works very intuitively in bilateral control systems, which was successfully tested in the experiments. Third, a method for treating the *avoidable* collisions is described, which defines the collision avoidance problem as an optimization task aiming at minimization of the virtual force. The optimization is formulated on the position level of the inverse kinematics and the extended kinematics formalism from Sec. 3.3, and executed locally by the telerobot controller.

Finally, we present the experimental results for i) teleoperated collision avoidance with the force display to the operator, and ii) collision avoidance with self motion and optimization of a multiple objective functions from Sec. 2.5.1.

6.1 Collisions in Dual Arm Configuration

The problem of moving in space in the presence of environmental constraints, originated initially in mobile robots applications. A mobile robot is to move from an initial position to a target position without colliding with the obstacles. For a manipulator however, the problem is more complicated. Not only must the end-effector avoid collisions with the obstacles, but the whole robot body as well. This is why the prior work in collision avoidance for manipulators is mainly concerned with robot link modeling and with methods of minimum distance calculation [46, 61, 107, 115]. In the case of multiple manipulators sharing the same workspace, each robot is effectively a moving obstacle to the other. Hence each of them is influenced by the presence of the other one so that they must avoid each other in the execution of their main tasks.

Collisions can be classified according to different criteria. In the current scenario the characteristics of *avoidability* is most important. A collision is *avoidable* if it can be prevented by executing a special motion generated by a local controller, while the robot is still performing its main task, which is following the user generated trajectories. According to this restriction, the escape motion must be the self motion of the redundant manipulator. Correspondingly, a collision is *unavoidable*, if no such escape motion can be found.

Another possible classification is that with respect to the objects colliding. The range of collisions the controller has to handle are shown in Fig. 6.1 and listed below:

A self collisions, including arm to arm and arm to base collisions: tool to tool ($t2t$), tool to wrist ($t2w$), etc.

B special case of the arm to arm collisions: tool to elbow ($t2e$)

C collisions with the joint limits specified in Tab. A.1

D collisions with workspace boundaries

Regarding the workspace boundaries, the reachable workspace of the manipulator is defined by the distance shoulder to wrist, that is the sum of lengths of the upper and lower arms. The hand with the tool can take an arbitrary orientation.

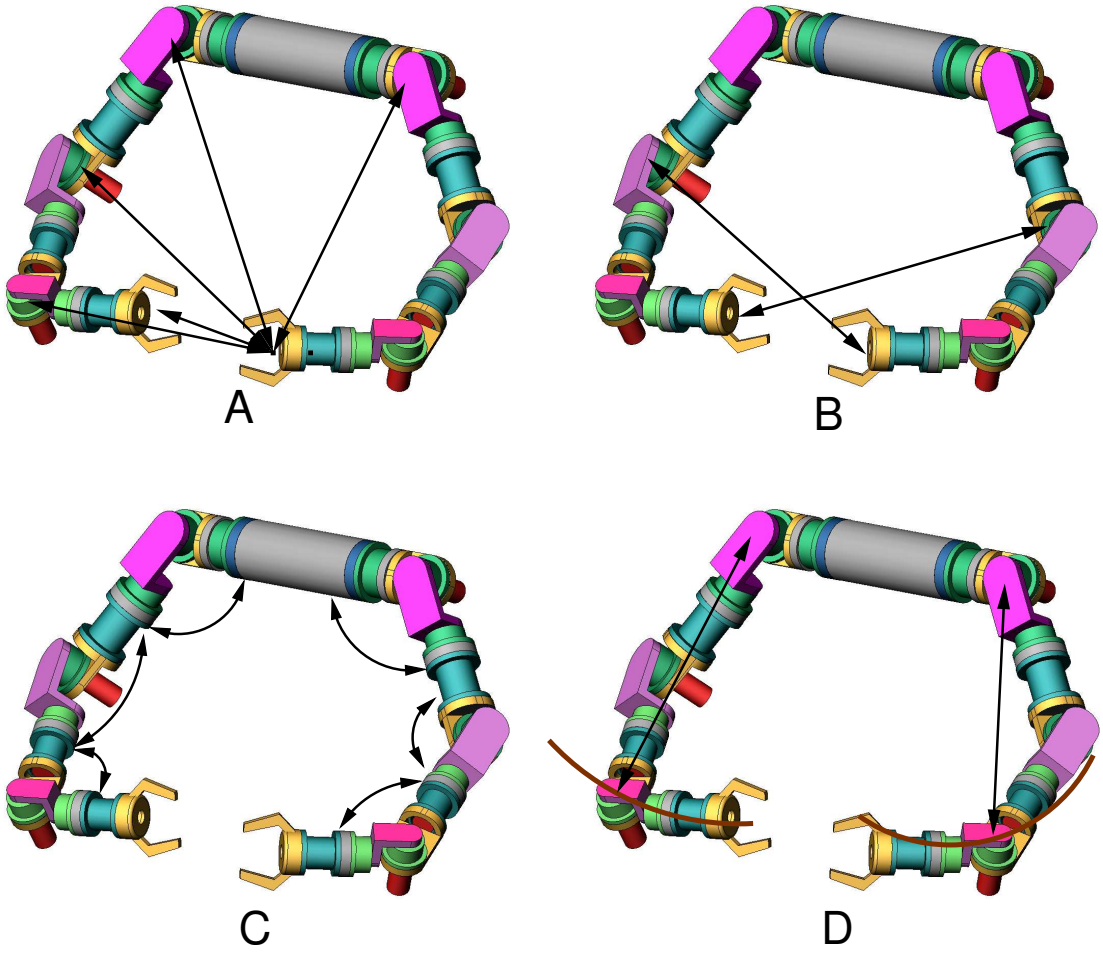


Figure 6.1: Types of collisions in the dual arm system: A - tool to tool ($t2t$), tool to wrist ($t2w$), etc., B- tool to elbow ($t2e$), C - collisions with the joint limit, D - collisions with workspace boundaries.

Desired Strategy for Collision Handling

Before proceeding, the *teleoperation specific* strategy for collision handling should be characterized.

- The *avoidable* collisions should be avoided by the local controller, without involving the operator on the master side. This must happen without disturbing the end-effector trajectories. The escape motion must lie in the arm self motion: the elbow rotation, see Sec. 2.1.2. As an example of this type of collisions are tool to elbow and joint limits collisions.

- In case of the *unavoidable* collisions entailing the end-effector, a suitable information should be displayed to the operator, with possible escape directions. The natural choice for the user involvement is the force display.

- Violating the security distances should stop the manipulator motion preventing physical damages.

In the following section we present a method fulfilling all those requirements in a unique framework.

6.2 Robot Modeling and Collision Detection

Collisions are detected based on the knowledge of the manipulator model and its current configuration. We use the collision detection algorithm based on minimum distance function, as introduced in [22]. The minimum distance function, defined by the Euclidean norm, relates to the danger of collision between two objects. In case of multibody manipulators, the minimum distance between the parts is considered. Although the CAD model of the robot is available, the irregular shapes of the robot parts (motors, gears, cables, etc.) make the exact modeling computationally expensive. A simpler representation of the parts in 3D is necessary. In [141] several polyhedra are extensively investigated: cuboids, prisms, cylinders, polyhedrons, ellipsoids, spheres and cyclispheres. It has been found that spheres and cyclispheres are an excellent choice for use in modeling objects in a robot's workspace because of their computational simplicity. Very little information has to be stored in order to fully define such shapes. The drawback of modeling with simple geometric shapes is that these shapes may not be able to provide a sufficiently detailed and accurate model. Since computational burden is an important factor, the spherical representation will be used in practical implementation. The robot links are represented by three spheres with their centers at the link two ends and at its middle.

6.3 Virtual Forces Concept

The collision avoidance problem is solved using the virtual forces concept, based on potential fields similar to electrical potential field; the force is a function of the distance. The goal is to prevent intrusion of the arm into the safety zone of other objects. For every object a *safety zone* is defined, displaced from the object surface and described by means of a *safety distance* d_{min}. Those distances are strictly related to the manipulator model and established experimentally. The proximity to the objects, $\boldsymbol{d}$ is computed in real time, and in case of $|\boldsymbol{d}| < d_{min}$ a virtual intrusion force $\boldsymbol{f}_v(\boldsymbol{d})$ is generated and displayed to the human operator. Several functions $\boldsymbol{f}_v(\boldsymbol{d})$ are possible, one can distinguish two main methods:

- function $\boldsymbol{f}_v$ is a polynomial type function of the distance $\boldsymbol{f}_v = \boldsymbol{p}(\boldsymbol{d})$,
- a dynamic relation is assigned to the force, for example a spring-damper system $\boldsymbol{f}_v = \boldsymbol{k}_p\boldsymbol{d} - \boldsymbol{k}_v\dot{\boldsymbol{d}}$.

Heuristically, a quadratic potential field is chosen

$$\boldsymbol{f}_v(d) = \begin{cases} 0, & \text{for} \quad |\boldsymbol{d}| > d_{min} \\ -\frac{\boldsymbol{d}}{d_{min}} \frac{2d_{min}(d_{min}-|\boldsymbol{d}|)}{|\boldsymbol{d}|^2}, & \text{for} \quad |\boldsymbol{d}| \leq d_{min} \end{cases} . \tag{6.1}$$

The goal is to create an impression of a fictitious spring located at the object safety zone pushing the manipulator end-effector into the "right" direction. In order to nullify this force, the operator drives the robot into the direction of the free space. In Fig. 6.2 an example force field is shown. It illustrates the xy projection of the virtual force as "seen" by the end-effector of the left arm. A simple *point to point* distance computation with spherical geometry is utilized. The distances to the right end-effector ($\boldsymbol{t}_r$), wrist ($\boldsymbol{w}_r$), shoulders ($\boldsymbol{s}_r$ and $\boldsymbol{s}_l$) as well as to link middle points approximate sufficiently well the desired force field, while demanding minimum computation load. This is important, because the potential field is being computed online due to the movement of the arms. The force field in Fig. 6.2 is a superposition of all the *point to point*-based forces and the

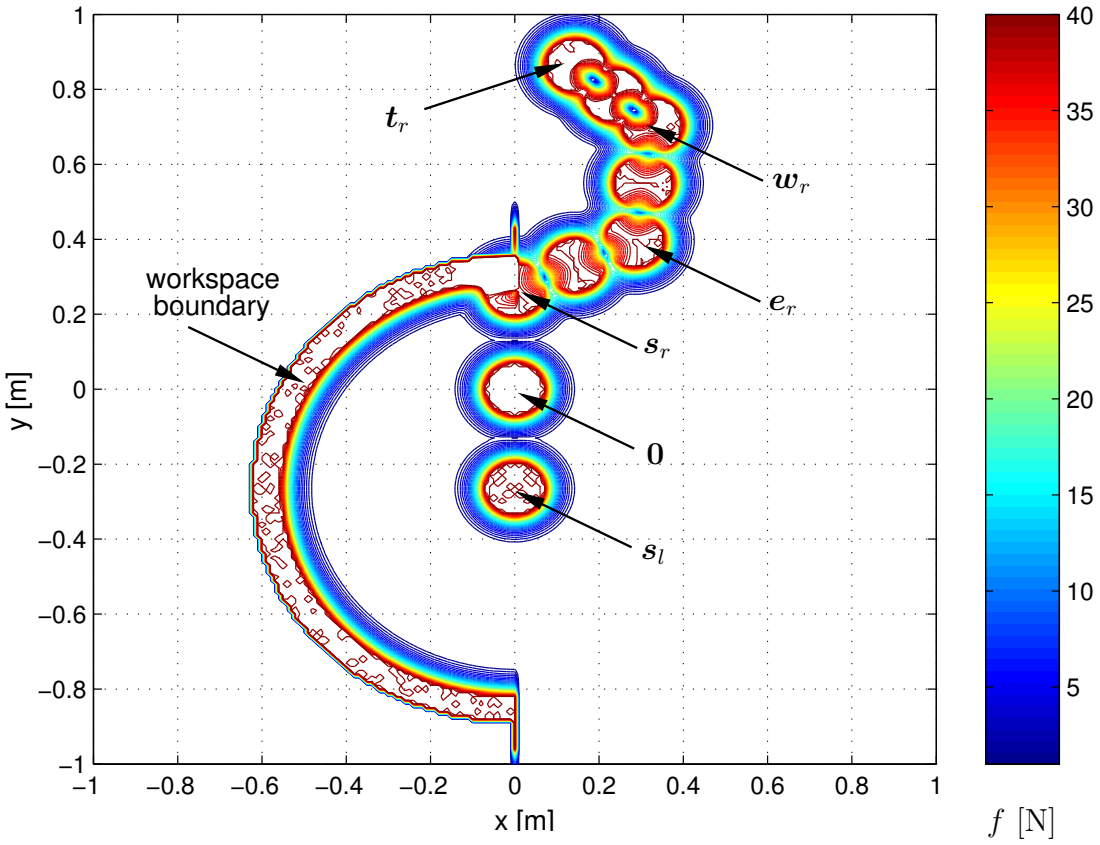


Figure 6.2: Potential feld as "seen" by the left arm: projection on thr xy plane

force generated by the workspace boundary. As we can see, it is splitting the reachable workspace into safe (white) zones and forbidden zones, marked by colors corresponding to the generated force.

6.4 Collision Avoidance with Redundancy Utilization

Certain types of collisions may be escaped by means of the self motion of the redundant kinematic structure, without perturbing the main robot trajectories. By self motion we consider the elbow motion as described in Sec. 2.1.2. To achieve this, we formulate the collision avoidance problem as an optimization problem aiming at minimization of the virtual forces. Consider the *tool to elbow* type of collisions depicted in Fig. 6.1B. Using the

virtual forces concept we can generate the motion so as the virtual forces acting on one of the arms are minimized by the self motion the *other* arm. Let's consider the left arm. The norm of the virtual force, as "seen" by its end-effector may be defined as a collidability measure we would like to minimize

$$w_{CAl} = || \sum_i k_i \boldsymbol{f}_i(d_i) ||, \tag{6.2}$$

where $\boldsymbol{f}_i(d_i)$ are the forces corresponding to i chosen points on the right arm, d_i are the distances to them, and k_i is a weighing factor. Since the virtual force is in fact a function of the robot configuration vector, it can be expressed as

$$w_{CAl} = w_{CAl}(\boldsymbol{q}_r, \boldsymbol{q}_l), \tag{6.3}$$

where $\boldsymbol{q}_r$ and $\boldsymbol{q}_l$ are the joint coordinates of the right and left arm correspondingly. The joint vector $\boldsymbol{q}_l$ is a function of the desired Cartesian position $\boldsymbol{x}_{El}$, defined using the extended kinematics as combination of the end-effector position $\boldsymbol{x}_l$ and the elbow angle θ_l

$$\boldsymbol{q}_l = IK([\boldsymbol{x}_l \; \theta_l]^T). \tag{6.4}$$

Usually such problems are solved on velocity level as described in sec 2.5.1. In the current optimization problem, $\boldsymbol{x}_l$ and $\boldsymbol{q}_r$ are considered constant and the optimal θ_l is found be means of a numerically computed gradient of w_{CAl}

$$\theta_{l\,new} = \theta_{l\,old} - \mu \frac{\Delta w_{CAl}(\cdot)}{\Delta \theta_l}, \tag{6.5}$$

where μ is a constant. Applying the velocity-based gradient optimization method from Sec. 2.5.1, one can achieve similar results with

$$\dot{\boldsymbol{q}}_{\theta l} = (\boldsymbol{I} - \boldsymbol{J}^{\#}\boldsymbol{J})(\frac{\partial w_{CAl}}{\partial \boldsymbol{q}_l})^T \tag{6.6}$$

and

$$\dot{\theta}_l = \boldsymbol{J}_\theta \mu' \dot{\boldsymbol{q}}_{\theta l}, \tag{6.7}$$

where $\boldsymbol{J}_\theta$ is the elbow Jacobian defined in (3.24), and μ' is a positive constant.

6.5 Experimental Results

Teleoperated Collision Avoidance with Virtual Forces

Despite its simplicity, the presented method has proved its efficiency in the following experiment. First, the collisions with the workspace limits are investigated. The user is asked to drive the arm toward the workspace limit, using VisHaRD10 as haptic input device, see Sec. 5.5. Fig. 6.3 shows the distance to collision d (A), the virtual force f (C) and the so called security index s (E). The security index is set to 0 in a secure space, and to 1 in the endangered zone. The manipulator is stopped then. In free space motion ($d > 5$ cm) no force is generated, $s = 0$ and the user is allowed to move freely. Starting at the distance of $d_{min} = 5$ cm to the workspace limit, see Fig. 6.3A, the virtual force is generated according

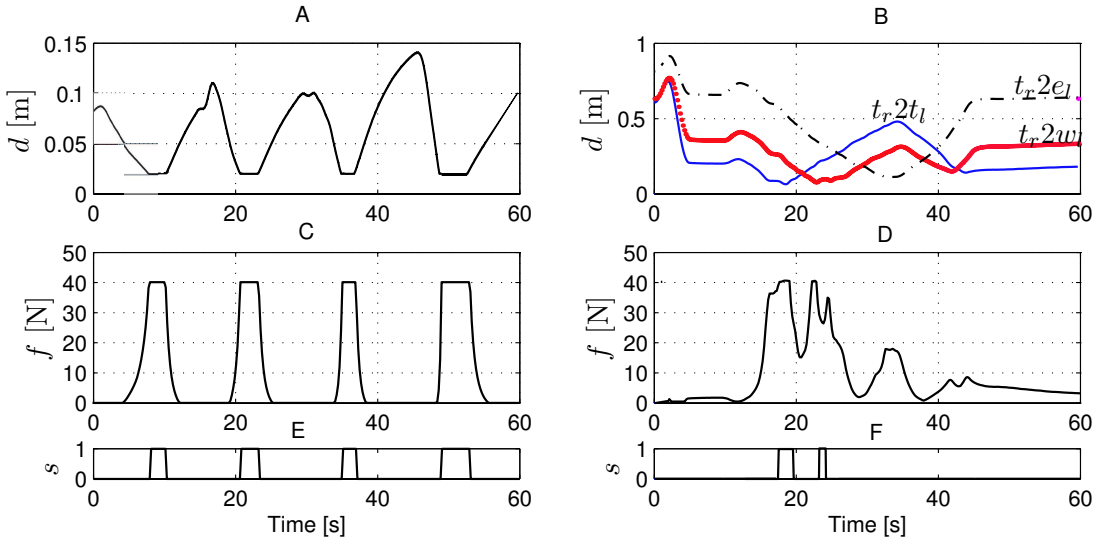


Figure 6.3: Virtual forces and distances during collisions with workspace boundary (left) and arm to arm collisions (right)

to the spring model of (6.1). The force is sent to the user, directing him/her in the direction of the free space. Since in the current experiment the maximum force is set to 40 N, it is still possible to push into the workspace limit. As a security measure, an emergency stop is executed when d reaches 1 cm, the security index is switched to 1 and the robot stops waiting for the next plausible command.

Analogous situation is observed in case of arm to arm collisions. Fig. 6.3B shows the distances for $t2t$, $t2w$ and $t2e$ types of collisions. The right arm is fixed in a chosen configuration, and the left arm is approaching it at different positions. The virtual force generated during this task is shown in plot D, and the security index in plot F. As we can see, the method works very well, and it is also confirmed by the operators that it is intuitive and not distracting.

Collision Avoidance with Self Motion

The following experiment illustrates the redundancy utilization for the purposes of fulfilling additional kinematic criteria. First, it is necessary to avoid the collisions with the other arm. Second, it is desirable to keep the current joint configuration as far as possible from its joint limits without disturbing the primary task of the end-effector. Differently to the previous experiment, the collision avoidance process should proceed without informing the user about the collision danger, so only local optimizers are applied, controlling the elbow motion. Therefore, joint limit avoidance and collision avoidance between both arms are chosen as the objective functions, see Sec. 2.5.1 for more details.

Recall the multiple performance criteria formulation. A slightly modified objective function for joint limits avoidance is given as follows, [146]:

$$w_{JL}(\boldsymbol{q}) = \frac{1}{4}\sum_{i=1}^{7}\frac{(q_{iM}-q_{im})^2}{(q_{iM}-q_i)(q_{im}-q_i)} \quad , \tag{6.8}$$

where w_{JL} denotes the objective function for the joint limit avoidance, q_{iM} and q_{im} denote the maximum and the minimum joint limits for the i-th joint angle q_i, respectively. If all the joints are in the middle of their ranges, the value of $w_{JL} = -7$, so the normalized objective function

$$\overline{w}_{JL}(\boldsymbol{q}) = \exp(7 + w_{JL}(\boldsymbol{q})) \tag{6.9}$$

equals zero when all joint variables stand in the middle of their feasible ranges, and goes to one at either limit.

In order to reduce the computational load necessary for the optimization, a simplified objective function for collision avoidance is applied. Since we are interested only in distances that involve the elbow, we take the shortest distance $d(\boldsymbol{q}) = |\boldsymbol{d}(\boldsymbol{q})|$ between the elbow and the other arm as the basis for the objective function w_{CA}:

$$\overline{w}_{CA}(\boldsymbol{q}) = \begin{cases} 1 & \text{for } d_M < d \\ \frac{d-d_m}{d_M-d_m} & \text{for } d_m \le d \le d_M \\ 0 & \text{for } d < d_m \end{cases} . \tag{6.10}$$

Such objective function is a modified virtual force model of (6.1) with normalization on the 0-1 range corresponding to minimum d_m and maximum d_M distances. These two objective functions are combined using the formulation of Sec. 2.5.1 to define the self motion.

The experimental results are illustrated in Fig. 6.4. It shows the elbow motion described with the angle θ and its derivative $\dot{\theta}$, the objective functions $\overline{w}_{CA}$ and $\overline{w}_{JL}$ with the corresponding weighing factors c_{jl} and c_{ca}, as defined in (2.74). Initially the left arm is

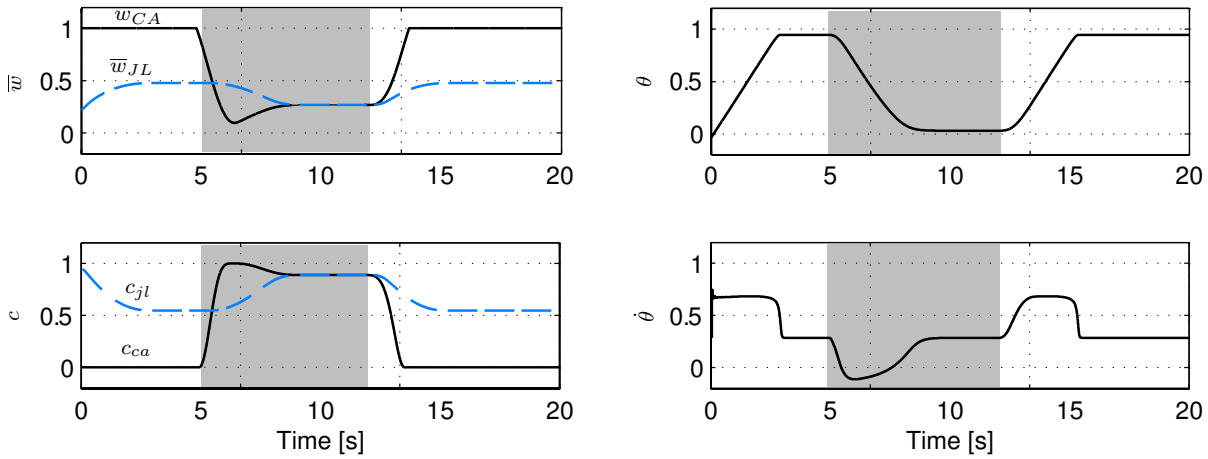


Figure 6.4: Optimization of a multiple objective functions: arm to arm collision avoidance (no background) with joint limits avoidance (shaded background)

placed far enough from the right elbow. After 7 s the left arm starts to move toward the elbow of the right arm and at the time of 9 s it starts to move back. Therefore, the whole process can be divided into three phases. In the first phase (left side of the shaded area in the figures), there is no danger of collision, so that the criteria for joint limit avoidance is dominant and $\overline{w}_{JL}$ is minimized through the motion of the redundant DoFs (self motion). However, since collision possibilities are induced in the second phase due to the approach of the left arm (shaded ares), the collision avoidance $\overline{w}_{CA}$ prevails and the self motion is resolved to avoid the collision with the left arm while $\overline{w}_{JL}$ is reduced. In the third phase,

after the left arm moved back, $\overline{w}_{JL}$ dominates again and the angle from the first phase is recovered.

6.6 Summary

This section introduced the concept of virtual forces in teleoperation, aimed at solving the problems of collision avoidance in dual configuration and workspace matching between the master and slave devices. The virtual force is generated as a function of the distance to the other arm and to the workspace limit. The advantage of this concept is twofold. First, the human operator is informed about the danger of collision in a very intuitive way using the force feedback (reaction time to force stimuli is much shorter than to visual ones). Second, the virtual force is used as a quality index for the local controller. This controller, aiming at minimizing of the virtual force, reduces the collision danger without involving the operator. If the virtual force exceeds certain threshold, the security systems are activated and the arms are stopped. The concept was tested for collisions of the arm with the workspace and the other arm in a teleoperation scenario. Optimization of multiple objective functions (arm to arm collision avoidance and joint limits avoidance) was applied for the purposes of redundancy resolution on the local controller.

7 Conclusions and Future Work

7.1 Concluding Remarks

This thesis describes the complete design and control process of a dual arm anthropomorphic telemanipulator. The purpose of such a robot is to act on the remote site of a telemanipulation system, mimicking the manipulation capabilities of human arms. Extensive experimental work evaluated the outcomes of every development stage, considering the requirements of the telemanipulator specified in Sec. 1.2. Thorough study of the design prerequisites and of the human biomechanical charcteristics shows that the desired manipulator must be different from the commercially available industrial robots. Even more so that the control methods are much more demanding and challenging.

The need for humanoid *functionality* forced us to construct manipulators of redundant structure, possibly close to the kinematic structure of the human arm. Several sources of biomechanical data were studied to extract the necessary kinematic and geometrical information. Since the human arm is a complex apparatus (the human shoulder itself may be modeled as more than 5 DoF), it was necessary to make certain simplifications. However, the 7 DoF structure applied here, thanks to its large joint ranges, fulfills those geometrical requirements very well. The Cartesian workspace does not suffer from choosing only one degree of redundancy, and the characteristic elbow motion can be perfectly reproduced.

With the weight of about 13 kg, the arm reach (shoulder to force/torque sensor) of 0.86 m and the payload of 6 kg, this structure may be referred to as lightweight, but first and foremost as *human scaled*. Efforts to mimic the power of the actuation of human limbs has long been a target for constructors. Better results only have been achieved by the Robotic Institute of the German Aerospace Center DLR, see [3, 50], at the cost of incomparably greater engineering effort. The arms developed within this thesis are built using commercially available components together with aluminium construction elements, see App. A.3 for details.

Except the geometry, one of the most important anthropomorphic features is the ability to work in the singular configurations, see Sec. 1.2. Within this thesis, a series of singularity-robust kinematic algorithms is investigated with respect to teleoperation, resulting in proposing a very effective Weighted Damped Least Squares method that allows for singularity traversing by limiting the joint velocities with simultaneous weighing of the tracking error.

The problem of the arm redundancy is resolved using the extended kinematics concept, whereas the arm self motion is defined in a physically meaningful way and the coordinate representing it is added to the task space vector. This way the arm configuration is uniquely defined so that the problems with non-repeatability of the redundant motion are eliminated. The characteristics of the self motion of the human arm are described in coordinates relative to the body, which are common across all the individuals. This is a major difference from other approaches describing the elbow motion in absolute coordi-

nates. Our method presented here does not require user calibration and is easy transferable to anthropomorphic manipulators.

Major achievement in this thesis is the closed form inverse kinematic solution. Since it is presented in an analytical way, the resolving of the task space trajectories into the joint space is computationally very efficient. It thus gives thorough understanding of various aspects of the arm kinematics such as singularities.

The compliance control methods are also comprehensively investigated. Performance of several impedance and stiffness control algorithms is evaluated. The experimental identification method made it possible to choose the most efficient algorithm. Due to the inherent properties of the impedance control methods based on the wrist force measurement, the redundant coordinates cannot be controlled in a compliant manner. The impedance control with stiffness control in the inner motion control loop solves this problem, offering the exact impedance display on the robot end-effector together with the compliance of the self motion. This is possible thanks to a concept of the extended kinematics and closed form inverse kinematics solution. Namely, the self motion (elbow motion) coordinate is included into the Jacobian as an extra row that lies in the space orthogonal to the classic Jacobian, so that the mapping between the joint space torques and the null space torques is feasible.

Large part of the presented work consists of experiments related to teleoperation. Since the 7 DoF arms are used in combination with different input devices resulting in a system of dissimilar kinematics, there is a need to specify a unique kinematic interface for connecting them. Mapping of positions, velocities, forces and torques is straightforward, but mapping the orientations is not trivial. The first attempts using Euler angles turned out to be unsuccessful due to the native representation singularities of Euler angles. Such approaches limit the attainable workspace and are not acceptable in a dexterous manipulation system. The four parameter representation based on unit quaternions is exploited both for motion generation and for the interaction control.

The problems of workspace matching and the collision avoidance between the two cooperating arms are solved using the concept of virtual forces. Approaching the other arm or the workspace boundary generates a virtual force displayed to the operator using the haptic interface and directing him/her into the safe zone. This method has the advantage over visual or acoustic displays in that the reaction time is much shorter and the user burden is lower.

It needs to be underscored that, although the presented manipulator has been developed for teleoperation purposes, practically all strategies are applicable to other kinds of robots. The teleoperation scenario is chosen as a benchmark scenario of extremely high requirements. Especially robots that are to work in human environments and have direct physical contact with humans (service, rehabilitation, entertainment robots) may profit from the singularity-robust kinematics and compliant interaction methods.

7.2 Future Work

A natural extension of the presented work is further hardware development with more anthropomorphic features. First, hand and fingers should be combined with the arm, to enable gripping of objects and fine manipulation. Second, in order to increase the workspace and general manipulability, the complete upper body should be constructed with a movable torso. A ball joint at the height of the human waist would be a good

embodiment of such an idea. Third, the mobility of the whole system should be assured using a mobile platform. Independently of the telerobot development, the master side of the teleoperation system should be equipped with a wearable sensor system reading the elbow motion of the human operator. Preliminary experiments proved that this is possible even with relatively simple hardware.

The performance of the current manipulator may be increased by more accurate modeling of the manipulator and friction. The stiffness control methods investigated in chapter 4, despite their promising features and the existing torque interface to the robot joints, suffer severely from modeling inaccuracy. Another possibility is to apply robust control algorithms that are less sensitive to the modeling errors. An example of such an approach is presented in our original work applying a version of sliding mode [68**?**].

Although the compliance controllers developed in this thesis are powerful and exact, the question of how the desired robot impedances should actually be tuned is still open. In all presented experiments, the impedance is adjusted to assure system stability and optimize chosen quality functions, and it is kept *constant* during the experiments. Our preliminary work [66] showed promising results with variable impedance during various manipulation phases, e.g., transition from free space to constrained motion.

It would be also advantageous to exploit additional information about the manipulated objects and the environment by extracting this information from the existing sensors. Vision/force/motion sensors may support the local controllers with data about the distances to contact, environment impedance, motion, etc. An example of local motion control is the method of simultaneously avoiding collisions and joint limits introduced in Sec. 6.4. This method works without involving the user in the collision avoidance process and is purely local.

Teleoperation experiments regarding direct dual telemanipulation, due to the lack of dual input device, have not been performed. Also other teleoperation experiments including load transportation or working in viscous environments are subject to future work.

A Appendix

A.1 Forward Kinematics for the 7 DoF Arm

The location of the robot end-effector is described with a homogeneous transformation matrix

$$\boldsymbol{T}_0^7 = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix}, \tag{A.1}$$

where $\boldsymbol{T}_0^7$ is computed using the link homogeneous matrices as in [118]

$$\boldsymbol{T}_0^7 = \boldsymbol{A}_1 \boldsymbol{A}_2 \dots \boldsymbol{A}_7. \tag{A.2}$$

The elements r_{ij} of matrix $\boldsymbol{R}$ are given with the following formulas:

$$\begin{aligned}
r_{11} =& s_1(c_3c_6c_7s_5 + c_7s_3s_4s_6 + c_3c_5s_7 + c_4s_3(c_5c_6c_7 - s_5s_7)) + \\
& c_1(s_2(-c_5c_6c_7s_4 + c_4c_7s_6 + s_4s_5s_7) + c_2(c_7(c_3c_4c_5c_6 - c_6s_3s_5 + c_3s_4s_6) - (c_5s_3 + c_3c_4s_5)s_7)) \\
r_{12} =& c_7(c_5(c_3s_1 - c_1c_2s_3) - (c_4s_1s_3 + c_1(c_2c_3c_4 - s_2s_4))s_5) - \\
& (s_1(c_4c_5c_6s_3 + c_3c_6s_5 + s_3s_4s_6) + c_1(s_2(-c_5c_6s_4 + c_4s_6) + c_2(c_3c_4c_5c_6 - c_6s_3s_5 + c_3s_4s_6)))s_7 \\
r_{13} =& s_1(c_6s_3s_4 - (c_4c_5s_3 + c_3s_5)s_6) + \\
& c_1(c_6(c_4s_2 + c_2c_3s_4) + (c_5s_2s_4 + c_2(-c_3c_4c_5 + s_3s_5))s_6) \\
r_{21} =& s_1s_2(-c_5c_6c_7s_4 + c_4c_7s_6 + s_4s_5s_7) - c_1(c_7(c_4c_5c_6s_3 + c_3c_6s_5 + s_3s_4s_6) + (c_3c_5 - c_4s_3s_5)s_7) + \\
& c_2s_1(-s_3(c_6c_7s_5 + c_5s_7) + c_3(c_4c_5c_6c_7 + c_7s_4s_6 - c_4s_5s_7)) \\
r_{22} =& c_1(c_3(-c_5c_7 + c_6s_5s_7) + s_3(c_4c_7s_5 + c_4c_5c_6s_7 + s_4s_6s_7)) + \\
& s_1(s_2(c_7s_4s_5 + c_5c_6s_4s_7 - c_4s_6s_7) - c_2(c_7(c_5s_3 + c_3c_4s_5) + (c_3c_4c_5c_6 - c_6s_3s_5 + c_3s_4s_6)s_7)) \\
r_{23} =& c_6(c_4s_1s_2 + (c_2c_3s_1 - c_1s_3)s_4) + (c_5s_1s_2s_4 + c_1(c_4c_5s_3 + c_3s_5) + c_2s_1(-c_3c_4c_5 + s_3s_5))s_6 \\
r_{31} =& c_7(-c_6s_2s_3s_5 + c_2(c_5c_6s_4 - c_4s_6) + c_3s_2(c_4c_5c_6 + s_4s_6)) - (c_5s_2s_3 + (c_3c_4s_2 + c_2s_4)s_5)s_7 \\
r_{32} =& -c_7(c_3c_4s_2 + c_2s_4)s_5 + (c_6s_2s_3s_5 + (c_2c_4 - c_3s_2s_4)s_6)s_7 - c_5(c_7s_2s_3 + c_6(c_3c_4s_2 + c_2s_4)s_7) \\
r_{33} =& -c_2(c_4c_6 + c_5s_4s_6) + s_2(s_3s_5s_6 + c_3(c_6s_4 - c_4c_5s_6))
\end{aligned} \tag{A.3}$$

and the elements of the vector $\boldsymbol{p} = [p_x\, p_y\, p_z]^T$

$$\begin{aligned}
p_x =& s_1((l_5 + l_7c_6)s_3s_4 - l_7(c_4c_5s_3 + c_3s_5)s_6) + \\
& c_1(s_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + c_2(c_3(l_5 + l_7c_6)s_4 + l_7(-c_3c_4c_5 + s_3s_5)s_6)) \\
p_y =& c_1(-(l_5 + l_7c_6)s_3s_4 + l_7(c_4c_5s_3 + c_3s_5)s_6) + \\
& s_1(s_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + c_2(c_3(l_5 + l_7c_6)s_4 + l_7(-c_3c_4c_5 + s_3s_5)s_6)) \\
p_z =& l_1 - c_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + s_2(c_3(l_5 + l_7c_6)s_4 + l_7(-c_3c_4c_5 + s_3s_5)s_6)
\end{aligned} \tag{A.4}$$

The manipulator Jacobian is computed as given in [111, 118]

$$\boldsymbol{J} = [\boldsymbol{J}_1 \boldsymbol{J}_2 \dots \boldsymbol{J}_n]$$

$$\boldsymbol{J}_i = \begin{bmatrix} \boldsymbol{z}_{i-1} \times (\boldsymbol{o}_n - \boldsymbol{o}_{i-1}) \\ \boldsymbol{z}_{i-1} \end{bmatrix},$$

where $\boldsymbol{z}_i$ is the unit vector showing the rotation axis of the i-th link, and $\boldsymbol{o}_i$ denotes the vector from the origin of the frame 0 to the origin of the frame i. The elements of $\boldsymbol{J}$ are given below.

$$\begin{aligned}
j_{11} =& c_1((l_5 + l_7c_6)s_3s_4 - l_7(c_4c_5s_3 + c_3s_5)s_6) - \\
& s_1(s_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + c_2(c_3(l_5 + l_7c_6)s_4 + l_7(-c_3c_4c_5 + s_3s_5)s_6)) \\
j_{12} =& c_1(c_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + s_2(-c_3(l_5 + l_7c_6)s_4 + l_7(c_3c_4c_5 - s_3s_5)s_6)) \\
j_{13} =& (l_5 + l_7c_6)(c_3s_1 - c_1c_2s_3)s_4 + l_7(c_3(-c_4c_5s_1 + c_1c_2s_5) + s_3(c_1c_2c_4c_5 + s_1s_5))s_6 \\
j_{14} =& (l_5 + l_7c_6)(c_4s_1s_3 + c_1(c_2c_3c_4 - s_2s_4)) + l_7c_5(s_1s_3s_4 + c_1(c_4s_2 + c_2c_3s_4))s_6 \\
j_{15} =& l_7(c_4s_1s_3s_5 + c_3(-c_5s_1 + c_1c_2c_4s_5) + c_1(c_2c_5s_3 - s_2s_4s_5))s_6 \\
j_{16} =& -l_7(s_1(c_4c_5c_6s_3 + c_3c_6s_5 + s_3s_4s_6) + c_1(s_2(-c_5c_6s_4 + c_4s_6) + c_2(c_3c_4c_5c_6 - c_6s_3s_5 + c_3s_4s_6))) \\
j_{17} =& 0 \\
j_{21} =& s_1((l_5 + l_7c_6)s_3s_4 - l_7(c_4c_5s_3 + c_3s_5)s_6) + \\
& c_1(s_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + c_2(c_3(l_5 + l_7c_6)s_4 + l_7(-c_3c_4c_5 + s_3s_5)s_6)) \\
j_{22} =& s_1(c_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + s_2(-c_3(l_5 + l_7c_6)s_4 + l_7(c_3c_4c_5 - s_3s_5)s_6)) \\
j_{23} =& -(l_5 + l_7c_6)(c_1c_3 + c_2s_1s_3)s_4 + l_7(c_2s_1(c_4c_5s_3 + c_3s_5) + c_1(c_3c_4c_5 - s_3s_5))s_6 \\
j_{24} =& (l_5 + l_7c_6)(c_2c_3c_4s_1 - c_1c_4s_3 - s_1s_2s_4) + l_7c_5(c_4s_1s_2 + (c_2c_3s_1 - c_1s_3)s_4)s_6 \\
j_{25} =& l_7(c_1(c_3c_5 - c_4s_3s_5) + s_1(-s_2s_4s_5 + c_2(c_5s_3 + c_3c_4s_5)))s_6 \\
j_{26} =& l_7(s_1s_2(c_5c_6s_4 - c_4s_6) + c_1(c_4c_5c_6s_3 + c_3c_6s_5 + s_3s_4s_6) + c_2s_1(c_6s_3s_5 - c_3(c_4c_5c_6 + s_4s_6))) \\
j_{27} =& 0 \\
j_{31} =& 0 \\
j_{32} =& s_2(l_3 + c_4(l_5 + l_7c_6) + l_7c_5s_4s_6) + c_2(c_3(l_5 + l_7c_6)s_4 + l_7(-c_3c_4c_5 + s_3s_5)s_6) \\
j_{33} =& s_2(-(l_5 + l_7c_6)s_3s_4 + l_7(c_4c_5s_3 + c_3s_5)s_6) \\
j_{34} =& (l_5 + l_7c_6)(c_3c_4s_2 + c_2s_4) + l_7c_5(-c_2c_4 + c_3s_2s_4)s_6 \\
j_{35} =& l_7(c_5s_2s_3 + (c_3c_4s_2 + c_2s_4)s_5)s_6 \\
j_{36} =& -l_7(-c_6s_2s_3s_5 + c_2(c_5c_6s_4 - c_4s_6) + c_3s_2(c_4c_5c_6 + s_4s_6)) \\
j_{37} =& 0 \\
j_{41} =& 0 \\
j_{42} =& s_1 \\
j_{43} =& c_1s_2 \\
j_{44} =& c_3s_1 - c_1c_2s_3 \\
j_{45} =& s_1s_3s_4 + c_1(c_4s_2 + c_2c_3s_4) \\
j_{46} =& c_4s_1s_3s_5 + c_3(-c_5s_1 + c_1c_2c_4s_5) + c_1(c_2c_5s_3 - s_2s_4s_5) \\
j_{47} =& s_1(c_6s_3s_4 - (c_4c_5s_3 + c_3s_5)s_6) + c_1(c_6(c_4s_2 + c_2c_3s_4) + (c_5s_2s_4 + c_2(-c_3c_4c_5 + s_3s_5))s_6)
\end{aligned}$$

$$
\begin{aligned}
j_{51} =& 0 \\
j_{52} =& -c_1 \\
j_{53} =& s_1 s_2 \\
j_{54} =& -c_1 c_3 - c_2 s_1 s_3 \\
j_{55} =& c_4 s_1 s_2 + (c_2 c_3 s_1 - c_1 s_3) s_4 \\
j_{56} =& c_1(c_3 c_5 - c_4 s_3 s_5) + s_1(-s_2 s_4 s_5 + c_2(c_5 s_3 + c_3 c_4 s_5)) \\
j_{57} =& c_6(c_2 c_3 s_1 - c_1 s_3) s_4 + (c_5 s_1 s_2 s_4 + (c_1 c_3 + c_2 s_1 s_3) s_5) s_6 + c_4(c_6 s_1 s_2 + c_5(-c_2 c_3 s_1 + c_1 s_3) s_6) \\
j_{61} =& 1 \\
j_{62} =& 0 \\
j_{63} =& -c_2 \\
j_{64} =& -s_2 s_3 \\
j_{65} =& -c_2 c_4 + c_3 s_2 s_4 \\
j_{66} =& c_5 s_2 s_3 + (c_3 c_4 s_2 + c_2 s_4) s_5 \\
j_{67} =& -c_2(c_4 c_6 + c_5 s_4 s_6) + s_2(s_3 s_5 s_6 + c_3(c_6 s_4 - c_4 c_5 s_6))
\end{aligned}
\tag{A.5}
$$

A.2 Analysis of the Elbow Motion

The elbow motion described in Sec. 2.1.2 does not influence the end-effector position, therefore it is compelling to investigate the Jacobian null space analytically. To simplify the problem, we consider only the positioning chain, consisting of the first 4 links. The forward kinematics is described with the wrist position equation:

$$
\boldsymbol{w} = \begin{bmatrix} l_5 c_4 s_3 + c_1(l_5 c_2 c_3 c_4 + s_2(l_3 - l_5 s_4)) \\ l_5 c_2 c_3 c_4 s_1 - l_5 c_1 c_4 s_3 + s_1 s_2(l_3 - l_5 s_4) \\ l_1 + l_5 c_3 c_4 s_2 + c_2(-l_3 + l_5 s_4) \end{bmatrix}. \tag{A.6}
$$

After differentiation we obtain the wrist Jacobian $\boldsymbol{J}_{wp}$

$$
\begin{aligned}
j_{11} &= -l_5 c_2 c_3 c_4 s_1 + l_5 c_1 c_4 s_3 - s_1 s_2(l_3 - l_5 s_4) \\
j_{12} &= c_1(-(l_5 c_3 c_4 s_2) + c_2(l_3 - l_5 s_4)) \\
j_{13} &= l_5 c_4(c_3 s_1 - c_1 c_2 s_3) \\
j_{14} &= -l_5(s_1 s_3 s_4 + c_1(c_4 s_2 + c_2 c_3 s_4)) \\
j_{21} &= l_5 c_4 s_1 s_3 + c_1(l_5 c_2 c_3 c_4 + s_2(l_3 - l_5 s_4)) \\
j_{22} &= s_1(-l_5 c_3 c_4 s_2 + c_2(l_3 - l_5 s_4)) \\
j_{23} &= -(l_5 c_4(c_1 c_3 + c_2 s_1 s_3)) \\
j_{24} &= -l_5(c_4 s_1 s_2 + (c_2 c_3 s_1 - c_1 s_3) s_4) \\
j_{31} &= 0 \\
j_{32} &= l_5 c_2 c_3 c_4 + s_2(l_3 - l_5 s_4) \\
j_{33} &= -l_5 c_4 s_2 s_3 \\
j_{34} &= l_5(c_2 c_4 - c_3 s_2 s_4)
\end{aligned}
\tag{A.7}
$$

as the function of $\boldsymbol{q}_w = [q_1\, q_1\, q_3\, q_4]^T$. Computing the null space of $\boldsymbol{J}_{wp}$ one gets the self motion vector field

$$\boldsymbol{f}_{self} = \boldsymbol{I} - \boldsymbol{J}_{wp}^{\#}\boldsymbol{J}_{wp} = c\,[l_5 c_3 c_4,\, l_5 c_4 s_2 s_3,\, l_5 c_2 c_3 c_4 + s_2(l_3 - l_5 s_4),\, 0], \tag{A.8}$$

where c is an arbitrary constant. The self motion vector field has the property

$$\boldsymbol{0} = \boldsymbol{J}_{wp}\boldsymbol{f}_{self}^T. \tag{A.9}$$

Applying the extended kinematics concept, we compute the extended task space vector and the extended Jacobian $\boldsymbol{J}_{Ew}$ for the wrist as

$$\begin{bmatrix} \dot{\boldsymbol{w}} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{wp} \\ \frac{\partial\theta}{\partial \boldsymbol{q}_w} \end{bmatrix} \dot{\boldsymbol{q}}_w = \boldsymbol{J}_{Ew}\dot{\boldsymbol{q}}_w. \tag{A.10}$$

Solving the inverse kinematics by means of the inversion of the extended Jacobian for $\dot{\boldsymbol{w}} = 0$ and $\dot{\theta} \neq 0$ we obtain the joint velocities that perform the self motion

$$\dot{\boldsymbol{q}}_{w,\,self} = \boldsymbol{J}_{Ew}^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \end{bmatrix} = c\boldsymbol{f}_{self}^T, \tag{A.11}$$

where

$$\begin{aligned} c \;=\; & \frac{\dot{\theta}}{s_2\sqrt{l_3^2 + l_5^2 + 2l_3 l_5 c_4}} \cdot \\ & \frac{1}{(l_3 + l_5 c_4 - l_5 s_4)(2l_3 s_2^2 + l_5 c_3 s2_2(c_4 + s_4)) + l_5^2(c2_2(3 + c2_3) + 2s_3^2)s2_4} \cdot \\ & (4l_3^2 + 5l_5^2 - c2_2(4l_3^2 + l_5^2 + 3l_5^2 c2_4) + \\ & l_5(-2(l_5 c2_3 - 8l_3 c_4)s_2^2 + l_5 c2_4(-1 + 2c2_3 s_2^2) + 8c_3(l_3 + l_5 c_4)s2_2 s_4)). \end{aligned} \tag{A.12}$$

As we can see, the described "elbow motion" is a special parametrization of the self motion resulting from Jacobian analysis.

A.3 Hardware and Implementation Details

The arms are built using commercially available components together with aluminium/steel construction elements. The arm reach (shoulder to force/torque sensor) is 0.86 m, weight in total approx. 13.5 kg, "worst case" payload is 6 kg. The motors are Maxon motors type RE40, gears Harmonic Drive, see Tab. A.1 and photographs below for details.

Table A.1: Hardware details of the 7 DoF arm

Joint i	Link length l_i [m]	Link twist α_i [rad]	Joint Limits [rad]	Gear Type	Gear Reduction
1	± 0.2655	$\pi/2$	-	HFUC-25-160	160
2	0	$\pi/2$	[0.886, 5.397]	HFUC-20-160	160
3	0.312	$-\pi/2$	-	HFUC-20-160	160
4	0	$\pi/2$	[-2.377, 2.377]	HFUC-20-160	160
5	0.312	$-\pi/2$	-	HFUC-17-100	100
6	0	$\pi/2$	[-0.174, 0.174]	HFUC-17-100	100
7	0.244	$\pi/2$	-	HFUC-17-100	100
All motors: Maxon Motor RE40					

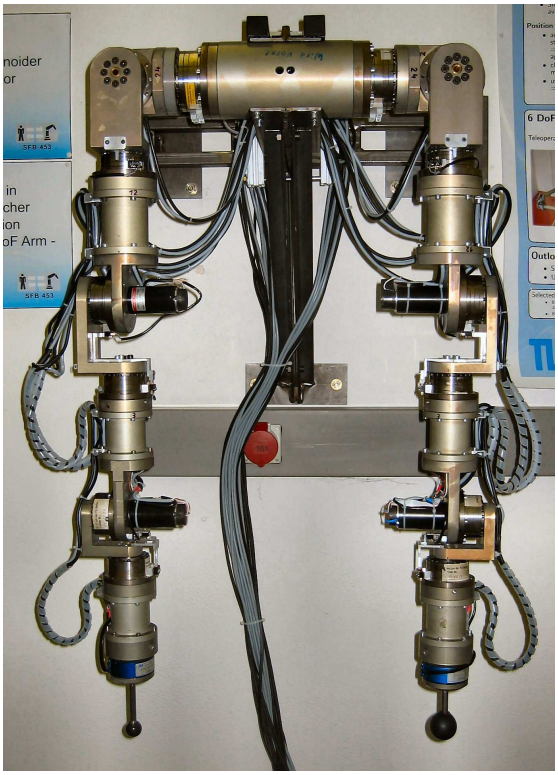


Figure A.1: Dual anthropomorphic system

Figure A.2: Photograph of the elbow joint

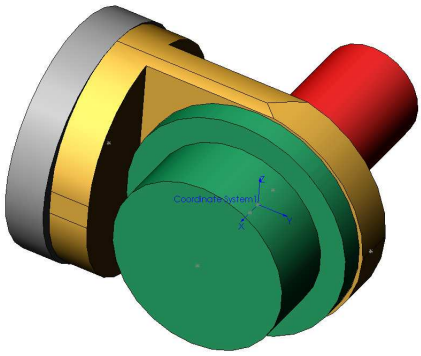
The motor torques are controlled by PWM amplifiers, operating in current control mode with the reference given by a voltage from the D/A converter output of the I/O board (Sensoray 626). The position of each joint is measured by an optic pulse incremental encoder on the motor shaft and then processed by a quadrature encoder on the I/O board. The force/torque is measured using a JR3 force/torque sensor. The control loops are

composed of MATLAB/SIMULINK blocksets; standalone realtime code for RT Linux is automatically generated from the SIMULINK model.

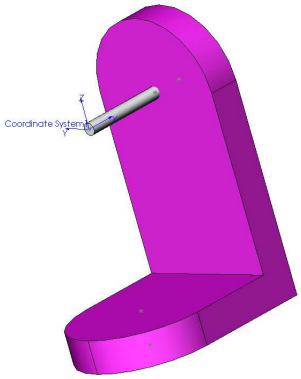
A.4 Mass Properties of the Manipulator Links

The mass properties of the manipulator links are defined according to the standard robotic notation of [97, 111, 118]. The coresponding coordinate systems are shown in Fig. 2.2 and the Denavit-Hartenberg parameters in Tab. A.4.

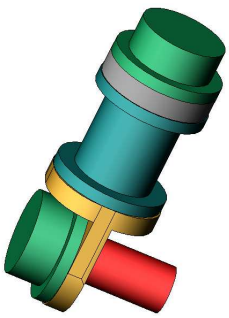
Link	1	R	L
Mass [kg]		2.563	2.563
Gravity Center [m]	l_1	0	0
	l_2	0.00021051	0.00021051
	l_3	0.06052543	-0.0605254
Inertia Matrix [kgm^2]	I_{xx}	0.00526792	0.00526792
	I_{yy}	0.00411581	0.00411581
	I_{zz}	0.00343615	0.00343615
	I_{xy}	0	0
	I_{xz}	0	0
	I_{yz}	0.00001325	-0.0000136



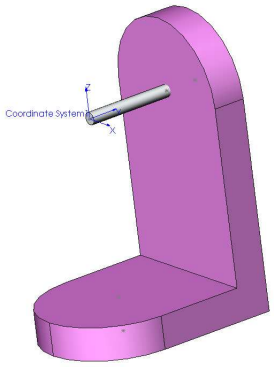
Link	2	R	L
Mass [kg]		0.741	0.7435
Gravity Center [m]	l_1	0.05767568	0.05767568
	l_2	0.03592568	0.03592568
	l_3	0.00050000	0.00050000
Inertia Matrix [kgm^2]	I_{xx}	0.00074811	0.00075063
	I_{yy}	0.00155721	0.00156246
	I_{zz}	0.00172566	0.00173148
	I_{xy}	-0.0004307	-0.0004322
	I_{xz}	0	0
	I_{yz}	0	0



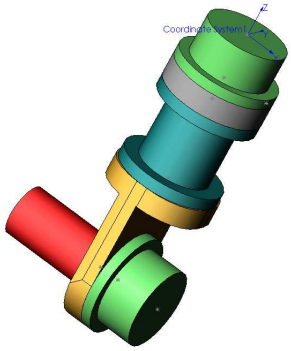
Link	3	R	L
Mass [kg]		4.347	4.348
Gravity Center [m]	l_1	0.12636277	0.12624533
	l_2	0.00021867	0.00021931
	l_3	-0.0000409	-0.0000408
Inertia Matrix [kgm^2]	I_{xx}	0.00488360	0.00488555
	I_{yy}	0.02990713	0.02992364
	I_{zz}	0.03107129	0.03108881
	I_{xy}	0.00008151	0.00008186
	I_{xz}	-0.0000146	-0.0000146
	I_{yz}	-0.0000030	-0.0000030



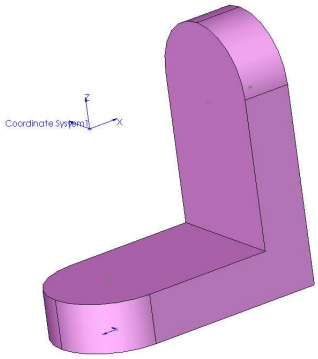
Link	4	R	L
Mass [kg]		0.718	0.719
Gravity Center [m]	l_1	0.05790828	0.05790828
	l_2	0.03590828	0.03590828
	l_3	0	0
Inertia Matrix [kgm^2]	I_{xx}	0.00073228	0.00073299
	I_{yy}	0.00150488	0.00150635
	I_{zz}	0.00166032	0.00166194
	I_{xy}	-0.0004152	-0.0004156
	I_{xz}	0	0
	I_{yz}	0	0



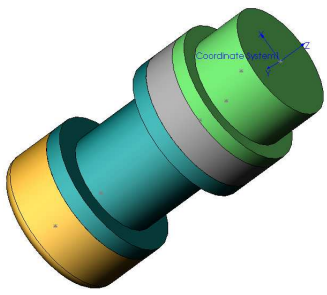
Link	5	R	L
Mass [kg]		3.19	3.2
Gravity Center [m]	l_1	0.12642271	0.12651493
	l_2	-0.0033447	-0.0033432
	l_3	0	0
Inertia Matrix [kgm^2]	I_{xx}	0.00312505	0.00313099
	I_{yy}	0.02029221	0.02032211
	I_{zz}	0.02152843	0.02155613
	I_{xy}	-0.0008537	-0.0008540
	I_{xz}	0	0
	I_{yz}	0	0



Link	6	R	L
Mass [kg]		0.292	0.292
Gravity Center [m]	l_1	0.03263961	0.03263961
	l_2	0.03540350	0.03540350
	l_3	0	0
Inertia Matrix [kgm^2]	I_{xx}	0.00018275	0.00018275
	I_{yy}	0.00021136	0.00021136
	I_{zz}	0.00031704	0.00031704
	I_{xy}	-0.0000883	-0.0000883
	I_{xz}	0	0
	I_{yz}	0	0



Link	7	R	L
Mass [kg]		2.278	2.279
Gravity Center [m]	l_1	0.08768685	0.08768685
	l_2	0	0
	l_3	0	0
Inertia Matrix [kgm^2]	I_{xx}	0.00133700	0.00133700
	I_{yy}	0.00839289	0.00839289
	I_{zz}	0.00839289	0.00839289
	I_{xy}	0	0
	I_{xz}	0	0
	I_{yz}	0	0



Bibliography

[1] B. Stanczyk A. Peer and M. Buss. Haptic telemanipulation with dissimilar kinematics. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2483–2488, Edmonton, Canada, 2005. Edmonton, Canada, August 2-6, Nominated to "Best Paper Award".

[2] K. Abdel-Malek, R. Yang, J. Brand, and E. Tanbour. Towards understanding the workspace of human limbs. *Ergonomics*, 47:1386–1405, 2004.

[3] A. Albu-Schaffer, C. Ott, U. Frese, and G. Hirzinger. Cartesian impedance control of redundant robots: Recent results with the dlr-light-weight-arms. *Proc. of the IEEE International Conference of Robotics and Automation*, 2003.

[4] R. O. Ambrose, H. Aldridge, R. S. Askew, R. R. Burridge, W. Bluethmann, M. Diftler, C. Lovchik, D. Magruder, and F. Rehnmark. Robonaut: NASA's space humanoid. *IEEE Journal on Intelligent Systems*, (4), 2000.

[5] R. J. Anderson and Mark Spong. Hybrid impedance control for robotic manipulators. *IEEE Journal of Robotics and Automation*, 4(5):549–556, October 1988.

[6] P. Arcara and C. Melchiorri. Comparison and improvement of control schemes for robotic teleoperation systems with time delay. *MISTRAL: Methodologies and Integration of Subsystems and Technologies for Anthropic Robots and Locomotion*, 2003.

[7] C. G. Atkeson and J. M. Hollerback. Kinematic features of unrestrained arm movements. (AIM-790), July 6 1984.

[8] J. Baillieul. Avoiding obstacles and resolving kinematic redundancy. *In IEEE International Conference Robotics and Automation*, pages 1698–1704, 1986.

[9] J. M. Pardos; C. Balaguer. Humanoid robot kinematics modeling using lie groups. *7th International Conference on Climbing and Walking Robots (Clawar'2004).*, Sep 2004.

[10] J. M. Pardos; C. Balaguer. Lie groups and lie algebras in robotics. *University Carlos III of Madrid - ROBOTICSLAB SEMINAR*, May 2004.

[11] A. Balestrino, G. De Maria, and L. Sciavicco. Robust control of robotic manipulators. *9th IFAC World Congress*, July 1984.

[12] R. Bischoff and V. Graefe. Hermes - a versatile personal robotic assistant. In *Proc. of the IEEE International Conference of Robotics and Automation*, volume 92, pages 1759– 1779, 2004.

[13] F. Bullo and R. M. Murray. Proportional derivative (PD) control on the Euclidean group. In *Proc. of European Control Conference*, volume 2, pages 1091–1097, Rome, Italy, June 1995.

[14] M. Buss. Multi-Fingered Regrasping Using a Hybrid Systems Approach. In *Proceedings of the 2nd IMACS/IEEE International Multiconference on Computational Engineering in Systems Applications (CESA'98)*, volume 4, pages 857–861, Hammamet, Tunisia, 1998.

[15] M. Buss, K. K. Lee, N. Nitzsche, A. Peer, B. Stanczyk, and U. Unterhinninghofen. Advanced telerobotics: Dual-handed and mobile remote manipulation. In R. Aracil, C. Balaguer, M. Buss, M. Ferre, and C. Melchiorri, editors, *Advances in Telerobotics: Human Interfaces, Control, and Applications*. Springer, STAR series, 2006, *to appear*.

[16] F. Caccavale, S. Chiaverini, and B. Siciliano. Second-order kinematic control of robot manipulators with jacobian damped least-squares inverse: theory and experiments. *IEEE/ASME Transactions on Mechatronics*, 2:188–194, 1007.

[17] F. Caccavale and B. Siciliano. Quaternion-based kinematic control of redundant spacecraft manipulator systems. In *IEEE International Conference of Robotics and Automation*, pages 435–440, 2001.

[18] Darwin G. Caldwell, Andrew Wardle, and Michael Goodwin. Tele-presence: Visual, audio and tactile feedback and control of a twin-armed mobile robot. In *ICRA*, pages 244–249, 1994.

[19] M. C. Cavusoglu, A.a Sherman, and F. Tendick. Bilateral controller design for telemanipulation in soft environments. In *Proc. of the IEEE International Conference of Robotics and Automation*, pages 1045–1052, 2001.

[20] D. B. Chaffin and G. Andersson. *Occupational Biomechanics*. New York: Wiley, 1991.

[21] T. F. Chan and R. V. Dubey. A weighted least norm solution based scheme for avoiding joint limits for redundant manipulators. *IEEE International Conference on Robotics and Automation*, 3:395–402, 1993.

[22] C. Chang, M. Chung, and Z. Bien. Collision-free motion planning for two articulated robot arms using minimumdistance function. *Robotica*, 8:137–144, 1990.

[23] S. Chang, J. Kim, I. Kim, J. Borm, C. Lee, and J. Park. Kist teleoperation system for humanoid robot. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 1198–1203, Kyongju, South Korea, 1999.

[24] S. Chiaverini. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Transactions on Robotics and Automation*, 13(3):398–410, 1997.

[25] S. Chiaverini, B. Siciliano, and O. Egeland. Review of the damped least-squares inverse kinematics with experiments on an industrial robot manipulator. *IEEE Transactions on Control Systems Technology*, 2:123–134, 1994.

[26] S. Chiaverini, B. Siciliano, and L. Villani. Force/position regulation of compliant robot manipulators. *IEEE Transactions on Automatic Control*, 39:647–652, 1994.

[27] N. Chopra, M. W. Spong, S. Hirche, and M. Buss. Bilateral Teleoperation over Internet: the Time Varying Delay Problem. In *Proceedings of the American Control Conference*, Denver, CO, 2003.

[28] J. C. Chou. Quaternion kinematic and dynamic differential equations. *IEEE Transaction on Robotics and Automation*, 8:53–64, 1992.

[29] K. Cleary. Incorporating multiple criteria generation for redundant robots. *IEEE International Conference on Robotics and Automation*, pages 994–999, 1990.

[30] Eodperformance Co. Vanguard. Available at www. eodperformance. com.

[31] Tsumuk Co. Enryu. Available at www. enryu. jp.

[32] J. De Schutter, H. Bruyninckx, W-H. Zhu, and M. W. Spong. *Force Control: A Bird's Eye View*. Spinger-Verlag, London, Dec 1997.

[33] J. Denavit and R. S. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. *ASME Journal of Applied Mechanics*, 22:215–221, 1955.

[34] M. Diftler, C. Culbert, R. Ambrose, R. Platt, and W. Buethmann. Evolution of the nasa/darpa robonaut control system. *Proc. of the IEEE International Conference on Robotics Automation*, 2003.

[35] M. A. Diftler and Robert O. Ambrose. Robonaut: A robotic astronaut assistant. In *Proc. of the ISAIRAS Conference*, Montreal, Canada, 2001.

[36] A. D'Souza, S. Vijayakumar, and Schaal S. Learning inverse kinematics. In *International Conference on Intelligence in Robotics and Autonomous Systems*, pages 298–303, 2001.

[37] I. Ebert-Uphoff and K. Kozak. Review of the role of quasi-coordinates for the kinematic and dynamic modeling of parallel manipulators. *Proceedings of the WORKSHOP on Funddamental Issues and Future Research Directions for Parallel MEchanisms and Manipulators, Quebec, Canada*, pages 133 –137, October 3-4,2002.

[38] U. Elger. A fire fighting robot in emergency cases, (basf factory fire brigade). Invited Talk at the IEEE International Workshop on Safety, Security, and Rescue Robotics, Bonn, Germany, 24-26 May 2004.

[39] Telerob Gesellschaft fuer Fernhantierungstechnik. thEODor - telerob explosive ordnance disposal. Available at www. telerob. de.

[40] J. Funda and R. P Paul. A computational analysis of screw transformations in robotics. *IEEE Transactions on Robots*, 6,3:348–356, 1990.

[41] J. Funda, R. H. Taylor, and R. P. Paul. On homogeneous transforms, quaternions, and computational efficiency. *IEEE Journal of Robotics and Automation*, 6(3):382–388, June 1990.

[42] J. S. Gu, Y. L. Ho. A unified kinematic representation and its applications to robotic control. In *Proc. of the IEEE International Conference on Robotics and Automation*, volume 1, pages 98–1031, 1990.

[43] Y. L. Gu. Analysis of orientation representations by lie algebra in robotics. volume 2, pages 874–879, 1988.

[44] B. Hannaford. A design framework for teleoperators with kinesthetic feedback. *IEEE Transactions on Robotics and Automation*, 5(4):426–434, 1989.

[45] K. Hashtrudi-Zaad and S. E. Salcudean. Analysis of control architectures for teleoperation systems with impedance/admittance master and slave manipulators. *International Journal Robotics Research*, 20(6):419–445, 2001.

[46] D. Henrich and X. Cheng. Fast distance computation for on-line collision detec-tion with multi-arm robots. *IEEE International Conference on Robotics and Automation*, 1992.

[47] S. Hirche and M. Buss. Study of Teleoperation using Realtime Communication Network Emulation. In *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Kobe, Japan, 2003.

[48] S. Hirche, B. Stanczyk, and M. Buss. Transparent Exploration of Remote Environments by Internet Telepresence. In *Proc. of the International Workshop on High-Fidelity Telepresence and Teleaction, jointly with the IEEE conference HUMANOIDS'2003*, Munich, Germany, 2003.

[49] S. Hirche, B. Stanczyk, and M. Buss. Haptic tele-assembly over the internet. In *Proceedings of the EuroHaptics'2004 Conference*, pages 417–421, Munich, Germany, 2004.

[50] G. Hirzinger, A. Albu-Schaffer, M. Hahnle, I. Schaefer, and N. Sporer. On a new generation of torque controlled light-weight robots. In *Proc. of the IEEE International Conference of Robotics and Automation*, pages 3356–3363, 2001.

[51] A. J. Hodgson and N. Hogan. A Model-Independent Definition of Attractor Behaviour Applicable to Interactive Tasks. *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, 30(1):105–118, February 2000.

[52] N. Hogan. Impedance control: An approach to manipulation, part i - theory, part ii - implementation,part iii - applications. *Journ. of Dyn. Systems, Measurement and Control*, 1985.

[53] N. Hogan. On the stability of manipulators performing contact tasks. *IEEE Journal of Robotics and Automation*, 4(6):677–686, December 1988.

[54] N. Hogan, H. I. Krebs, A. Sharon, and J. Charnnarong. *Interactive robotic therapist*. Massachusetts Institute of Technology, 1995. U. S. Patent 5 466 213.

[55] J. M. Hollerbach. Optimum kinematic design for a seven degree of freedom manipulator. *International J. Robotics Research*, pages 341–349, 1984.

[56] P. Hsu, S. Sastry, and J. Hauser. Dynamic control of redundant manipulators. *Journal of Robotic Systems*, 4, 1989.

[57] D. Hwang. *Teleoperation Performance with a Kinematically Redundant Slave Robot*. PhD thesis, University of Washington, Department of Electrical Engineering, 1995.

[58] R. Ikeura, T. Moriguchi, and K. Mizutani. Optimal variable impedance conrol for a robot and its application to lifting an object with a human. In *Proc. of the IEEE International Workshop on Robot and Human Interactive Communication*, pages 500–505, Berlin, Germany, 2002.

[59] J.Funda and R. P. Paul. On homogeneous transforms, quaternions and computational efficiency. *IEEE Transactions on Robots*, 6,3:382–388, 1990.

[60] M. Jordan and D. E. Rumelhard. Forward models: Supervised learning with a distal teacher. *Cognitive Science*, 16:307–354, 1992.

[61] J. Juang. Robot collision avoidance control using distance computation. In *Proc.of the IEEE International Conference on Robotics and Automation*, 1995.

[62] T. R. Kane. *AUTOLEV - A New Approach to Multibody Dynamics, Multibody Systems Handboook*. Springer–Verlag, Berlin, 1990.

[63] T. R. Kane. *Dynamics Online: Theory and Implementation with AUTOLEV*. OnLine Dynamics, Inc., 1996.

[64] P. K. Khosla and T. Kanade. Real-time implementation of the computed-torque scheme. *IEEE Transactions Rob. Automat.*, 5(2):245–253, 1989.

[65] M. V. Kirćanski and T. M. Petrović. Combined analytical-pseudoinverse kinematic solution for simple redundant manipulators and singularity avoidance. *International Journal of Robotics Research*, 12(2):188, April 1993.

[66] K. Klaasing. Incorporating human factors into the design of manipulator impedance in a bilateral telemanipulation system. Master's thesis, Lehrstuhl für Steuerungs- und Regelungstechnik, Technische Universität München, 2005.

[67] A. R. Klumpp. Singularity-free extraction of a quaternion from a direction cosine matrix. *Journal of Spacecraft*, 13(12):754–755, December 1976.

[68] K. Kondak, G. Hommel, B. Stanczyk, and M. Buss. Robust motion control for robotic systems using sliding mode. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3182–3187, 2005. Edmonton, Canada, August 2-6.

[69] J. U. Korein. *A Geometric Investigation of Reach*. MIT Press, 1985.

[70] K. Kosuge, K. Nakai, and Y. Hirata. Control of robot in singular configurations for human robot coordination. In *IEEE International Workshop on Robot and Human Interactive Communication, Berlin, Germany*, pages 356–361, 2002.

[71] A. Kron. *Beiträge zur bimanuellen und mehrfingrigen haptischen Informationsvermittlung in Telepraesenzsystem*. PhD thesis, Lehrstuhl für Steuerungs- und Regelungstechnik, Technische Universität München, 2005.

[72] A. Kron and G. Schmidt. A bimanual haptic telepresence system - design issues and experimental results. *Proceedings of International Workshop on High-Fidelity Telepresence and Teleaction jointly with the conference HUMANOIDS, Munich*, 2003.

[73] Brooks. T. L. Telerobotic response requirements. In *Proc. of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 113–120, 1990.

[74] D. A. Lawrence. Stability and transparency in bilateral teleoperation. *IEEE Transactions on Robotics and Automation*, 9(5):624–637, 1993.

[75] J. Lenarcic. Should robots copy humans. *Proceedings of 1997 IEEE International Conference on Intelligent Engineering Systems*, 1997.

[76] G. L. Long, S. W. Reed, and J. P. Donoghue. Regional structure kinematic modules for singularity avoidance. In *Proc. of the IEEE International Conference of Robotics and Automation*, pages 988–993, 1993.

[77] J. Luh, M. Walker, and R. Paul. Resolved-acceleration control of mechanical manipulators. *Automatic Control, IEEE Transactions on*, 25:468– 474, Jun 1980.

[78] G. Marani, J. Kim, J. Yuh, and W. K. Chung. A real-time approach for singularity avoidance in resolved motion rate control of robotic manipulators. In *Proc. of the IEEE International Conference of Robotics and Automation*, pages 1973–1978, 2002.

[79] M. Mason and J. K. Salisbury. *Robot Hands and the Mechanics of Manipulation*. MIT Press, 1985.

[80] M. T. Mason. Compliance and force control for computer controlled manipulators. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(6), June 1981.

[81] E. Messina, A. Jacoff, J. Scholtz, C. Schlenoff, H. Huang, A. Lytle, and J. Blitch. Statement of requirements for us&r robot performance standards. Technical report, National Institute of Standards and Technology, Department of Homeland Security, 2005. available at www.isd.mel.nist.gov.

[82] J. K. Mills and A. A. Goldenberg. Force and position control of manipulators during constrained motion tasks. *IEEE Journal of Robotics and Automation*, 5(1):30–46, February 1989.

[83] M. Minsky, M. Ouh-Young, and O. Steele. Feeling and seeing: Issues in force display. *Computer Graphics*, 24(4):235–243, 1990.

[84] R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. ISBN 0-8493-7981-4. CRC Press, 1993.

[85] F. A. Mussa-Ivaldi, N. Hogan, and E. Bizzi. Neural, mechanical, and geometric factors subserving arm posture in humans. *The Journal of Neuroscience*, 5(10):2732–2743, October 1985.

[86] D. Tolani N. Badler, A. Goswami. Real-time kinematics techniques for antropomorphic limbs. *Graphical Models 62, pages 353-388*, 2000.

[87] H. Nagasaki. Asymmetric velocity and acceleration profiles of human arm movements. *Experimental Brain Research*, 74(Jan2):319 – 326, Jan 741989.

[88] Y. Nakamura. *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley Publishing, 1991.

[89] Y. Nakamura and H. Hanafusa. Optimal redundancy control of robot manipulators. *International Journal of Robotics Research*, 6(1):34–42, 1987.

[90] C. Natale. *Interaction Control of Robotic Manipulators*. Springer Verlag, Berlin, 2003.

[91] C. Natale, B. Siciliano, and L. Villani. Control of moment and orientation for a robot manipulator in contact with a compliant environment. In *IEEE International Conference of Robotics and Automation*, pages 1755–1760, 1998.

[92] W. S. Newman and M. E. Dohring. Augmented impedance control: An approach to compliant control of kinematically redundant manipulators. *Proc. of the IEEE International Conference on Robotics and Automation*, 1991.

[93] N. Ogihara and N. Yamazaki. Generation of human reaching movement using a recurrent neural network model. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 692–697, Tokyo, Japan, 1999.

[94] T. H. Oh, W. K Chung, and Y. Youm. Extended impedance control of redundant manipulators based on weighted decomposition of joint space. *Journal of Robotic System*, 15:231–258, 1998.

[95] K. A. Oneil, Y. C. Chen, and J. Q. Seng. On the existence and characteristics of solution paths at algorithmic singularities. *IEEE Transactions on Robotics and Automation*, 14:336–342, 1998.

[96] G. Paul, M. Backes, and K. Long. Merging concurrent behaviors on a redundant manipulator. *IEEE International Conference on Robotics and Automation*, pages 638–645, 1993.

[97] R. P. Paul. *Robot Manipulators: Mathematics, Programming and Control*. The MIT Press, 1981.

[98] A. Peer, U. Unterhinninghofen, K. K. Lee, B. Stanczyk, and M. Buss. Haptic telemanipulation in extensive remote environments. In *Proceedings of the Joint International COE/HAM SFB-453 Workshop on Human Adaptive Mechatronics and High-Fidelity Telepresence*, pages 57–62, Tokyo, Japan, 2005.

[99] J. A. Pemanes and S. Zeghloul. Optimal placement of robotic manipulators using multiple kinematics criteria. *IEEE International Conference on Robotics and Automation*, pages 933–938, 1991.

[100] C. Pholsiri, D. Rabindran, M. Pryor, and C. Kapoor. Extended generalized impedance control for redundant manipulators. *Proceedings of the IEEE Conference on Decision and Control*, 2003.

[101] PIAP. Explosive ordnance disposal robot expert. Available at www. piap. pl.

[102] M. C. Pierre and R. F. Kirsch. Measurement and reliability of 3D end-point stiffness of the human arm. In *Proceedings of the 25th Annual International Conference of the IEEE EMBS*, pages 1433–1436, Cancun, Mexico, 2003.

[103] A. K. Pradeep, Paul J. Yoder, and Rangaswamy Mukundan. On the use of dual-matrix exponentials in robotic kinematics. *International Journal Robotic Research*, 8(5):57–66, 1989.

[104] A. Ramdane-Cherif, A. Benallegue, and B. Daachi. Memory neural network for kinematic inversion of constrained redundant robot manipulators. In *Proc. of the 11th International Conference on Advanced Robotics*, 2003.

[105] I. Raupach. Teleoperated handling of improvised explosive devices by police forces. Invited Talk at the IEEE International Workshop on Safety, Security, and Rescue Robotics, Bonn, Germany, 24-26 May 2004.

[106] R. Roberts and A. Maciejewski. Repeatable generalized inverse control strategies for kinematically redundant manipulators, 1993.

[107] H. J Rossmann. *Echtzeitfaehige, kollisionsvermeidende Bahnplanung fuer Mehrrobotersysteme*. PhD thesis, Institut fuer Roboterforschung, Universitaet Dortmund, 1993. In German.

[108] L. Villani S. Chiaverini, B. Siciliano. A survey of robot interaction control schemes with experimental comparison. *IEEE/ASME Transactions on Mechatronics*, 4:273–285, 1999.

[109] J. K. Salisbury. Active stiffness control of a manipulator in cartesian coordinates. In *Proc. of the 19th IEEE Int. Conference on Decision and Control*, pages 95–100, Albuquerque, New Mexico, USA, sep 1980.

[110] G Schreiber and G. Hirzinger. Singularity consistent inverse kinematics by enhancing the jacobian transpose. In *ARK 1998 Strobl/Wolfgangsee Advances in Robot Kinematics: Analysis and Control*, pages 475–483. Kluwer Academic Publishers, 1998.

[111] L. Sciavicco, B. Siciliano, and B. Sciavicco. *Modelling and Control of Robot Manipulators*. Springer-Verlag, 2000.

[112] H. Seraji and K. Kreutz-Delgado. Kinematic analysis of 7 dof antropomorphic arms. *IEEE Transactions on Robotics and Automation*, 1990.

[113] H. Seraji, M. K. Long, and T. Lee. Motion coontrol of 7-dof arms: The configuration control approach. *IEEE Transactions on Robotics and Automation*, 9(2):125–139, 1989.

[114] A. Sherman, M. C. Cavusoglu, and F. Tendick. Comparison of teleoperator control architectures for palpation task. In *Proceedings of ASME Dynamic Systems and Control Division*, pages 1261–1268, Nov. 2000.

[115] S. Shiang, J. Liu, and Y. Chien. Estimate of minimum distance between convex polyhedra based on enclosed ellipsoids. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2000.

[116] B. Siciliano and L. Villani. *Robot Force Control*. Kluwer Academic Publishers, Boston, MA, 1999.

[117] J. E. Speich, K. Fite, and M. Goldfarb. A method for simultaneously increasing transparency and stability robustness in bilateral telemanipulation. In *Proc. of the IEEE International Conference of Robotics and Automation*, pages 2671–2676, 2000.

[118] M. Spong and M. Vidyasagar. *Robot Dynamics and Control*. Wiley&Sons, 1989.

[119] B. Stanczyk and M. Buss. Development of a telerobotic system for exploration of hazardous environments. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2532–2537, Sendai, Japan, 2004.

[120] B. Stanczyk and M. Buss. Stiffness control of a redundant manipulator in a teleoperation scenario. In *Proc. of 13th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD)*, 2004. Brno, Czech Republuc, June 1-6.

[121] B. Stanczyk and M. Buss. Towards teleoperated exploration of hazardous environments: Control and experimental study of kinematically dissimilar master-slave structure. In *VDI-Berichte Nr. 1481, ROBOTIK*, pages 657–664, 2004.

[122] B. Stanczyk, S. Hirche, and M. Buss. Telemanipulation over the internet: a teleassembly experiment. In *Proc. of the IEEE International Conference on Mechatronics and Robotics, MechRob2004*, pages 315–320, Aachen, Germany, 2004.

[123] S. Tadokoro. Problem domain of japan national project (ddt project) on rescue robotics. *Proc. of the IEEE International Workshop on Safety, Security, and Rescue Robotics, Bonn, Germany*, 24-26 May 2004.

[124] K. Tchon and K. Muszynski. A normal form solution to the singular inverse kinematic problem for robotic manipulators: The quadratic case. In *Proc. of the IEEE International Conference of Robotics and Automation*, pages 3222–3227, 1998.

[125] K. Tchon and R. Muszynski. Singularities of nonredundant robot kinematics. *International Journal of Robotic Reasearch*, 16(1):60–76, 1997.

[126] G. Tevatia and S. Schaal. Inverse kinematics for humanoid robots. In *Proc. of the IEEE International Conference on Robotics and Automation*, volume 3, pages 294–299, 2000.

[127] Y. Tsumaki, P. Fiorini, G. Chalfant, and H. Seraji. A numerical sc approach for teleoperated 7-dof manipulator. *Proc. of the IEEE International Conference of Robotics and Automation*, 2001.

[128] Y. Tsumaki, S. Kotera, D. Nenchev, and M. Uchiyama. Advanced experiments with a teleoperation system based on the sc approach. *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1196–1201, 1998.

[129] Y. Tsumaki, D. N. Nenchev, S. Kotera, and M. Uchiyama. Teleoperation based on the adjoint jacobian approach. *IEEE Control Systems Magazine*, 17:53–62, 1997.

[130] T. Tsumugiwa, R. Yokogawa, and K. Hara. Variable impedance control based on estimation of human arm stiffness for human-robot cooperative calligraphic task. In *Proceedings of the 2002 IEEE International Conference Robotics and Automation*, pages 644–650, Washington, DC, 2002.

[131] M. Ueberle, N. Mock, and M. Buss. Vishard10, a novel hyper-redundant haptic interface. In *Proc. of the 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 58–65, Chicago, USA, 2004.

[132] M. Ueberle, N. Mock, A. Peer, C. Michas, and M. Buss. Design and control concepts of a hyper redundant haptic interface for interaction with virtual environments. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, Workshop on Touch and Haptics, (Sendai, Japan)*, September 28 - October 2, 2004.

[133] R. Volpe. *Real and Artificial Forces in the Control of Manipulators: Theory and Experiments*. PhD thesis, Carnegie Mellon University, Department of Physics, 1990.

[134] M. W. Walker. Manipulator kinematics and the epsilon algebra. *IEEE Journal of Robotics and Automation*, 4(2):186–192, 1988.

[135] C. W. Wampler. Manipulator inverse kinematic solution based on vector formulations and damped least squares methods. *IEEE Transactions on System Man and Cybernetics*, 16(1):93–101, 1986.

[136] J. T. Wen and D. Bayard. A new class of stabilizing control laws for robotic manipulators, part i: Nonadaptive case. *International Journal of Control*, 47(5):1361–1385, 1988.

[137] D. E. Whitney. Resolved motion rate control of manipulators and human prostheses. *IEEE Transactions Man-Machine Systems*, 10(2):4753, 1969.

[138] B. Xian, M. S. de Queiroz, D. M. Dawson, and L. Walker. Task-space tracking control of redundant robot manipulators via quaternion feedback. *IEEE Transactions on Robotics and Automation*, 20:160–167, 2004.

[139] T. Yabuta. Nonlinear basic concept of the hybrid position/force control scheme for robot manipulators. *IEEE Transactions on Robotics and Automation*, 8:663670, 1992.

[140] X. Yangsheng and R. P. Paul. Hybrid position force control of robot manipulator with an instrumented compliant wrist. *Experimental Robotics*, pages 244–270, 1989.

[141] Y. Yelgin. Kollisionsvermeidungsalgorithmus für einen bimanipulator mit siebenfreiheitsgraden. Master's thesis, Technical University of Berlin, Control Systems Group, 2004.

[142] Y. Yokokohji and T. Yoshikawa. Bilateral control of master-slave manipulators for ideal kinesthetic coupling-formulation and experiment. *IEEE Transactions on Robotics and Automation*, 10(1042-296X):605–620, 1994.

[143] W. Yoon, Y. Tsumaki, , and M. Uchiyama. Construction of an experimental system for dual-arm robot teleoperation in space with concepts of virtual grip and ball. *The Ninth International Conference on Advanced Robotics*, 1999.

[144] T. Yoshikawa. Analysis and control of robot manipulators with redundancy. *First International Symp. on Robotics Research*, 6(2):3–15, 1987.

[145] J. S. Yuan. Closed-loop manipulator control using quaternion feedback. *IEEE Journal of Robotics and Automation*, 4(4):434–440, August 1988.

[146] H. Zghal, R. V. Dubey, and J. A. Euler. Efficient gradient projection optimization for manipulators with multiple degrees of redundancy. *IEEE International Conference on Robotics and Automation*, pages 1006–1011, 1990.

[147] X. Zhao. *Kinematic control of human postures for task simulation.* PhD thesis, University of Pennsylvania, 1996.

[148] R. Zoellner, T. Asfour, and R. Dillmann. Programming by demonstration: Dual-arm manipulation tasks for humanoid. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004), Sendai, Japan, September 28 - October 2*, 2004.