

Two-arm Manipulation: From Manipulators to Enhanced Human-Robot Collaboration

Bruno Vilhena Adorno

► **To cite this version:**

Bruno Vilhena Adorno. Two-arm Manipulation: From Manipulators to Enhanced Human-Robot Collaboration. Automatic. Université Montpellier II - Sciences et Techniques du Languedoc, 2011. English. tel-00641678

HAL Id: tel-00641678

<https://tel.archives-ouvertes.fr/tel-00641678>

Submitted on 16 Nov 2011

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

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

THÈSE

pour obtenir le grade de
DOCTEUR DE L'UNIVERSITÉ MONTPELLIER II

Discipline : Génie Informatique, Automatique et Traitement du Signal
Formation Doctorale : Systèmes Automatiques et Microélectroniques
École Doctorale : Information, Structures et Systèmes

présentée et soutenue publiquement par

Bruno Vilhena Adorno

le 2 octobre 2011

Titre:

**Two-arm Manipulation: From Manipulators to Enhanced
Human-Robot Collaboration**

**(Contribution à la manipulation à deux bras : des manipulateurs à la
collaboration homme-robot)**

Jury:

Bruno Siciliano	Professeur, Università degli Studi di Napoli Federico II, Italie	<i>Rapporteur</i>
Masaru Uchiyama	Professeur, Tohoku University, Japan	<i>Rapporteur</i>
Oussama Khatib	Professeur, Stanford University, États-Unis	<i>Examineur</i>
Philippe Fraisse	Professeur, Université Montpellier II, France	<i>Directeur de thèse</i>
Philippe Poignet	Professeur, Université Montpellier II, France	<i>Examineur</i>
Véronique Perdereau	Professeur, Université Pierre et Marie Curie, France	<i>Examineur</i>

TWO-ARM MANIPULATION: FROM MANIPULATORS TO ENHANCED HUMAN-ROBOT COLLABORATION

BRUNO VILHENA ADORNO



Department of Robotics
Laboratoire d'Informatique, de Robotique et
de Microélectronique de Montpellier (LIRMM)
Université Montpellier 2
October 2, 2011

Bruno Vilhena Adorno: *Two-arm Manipulation: From Manipulators to Enhanced Human-Robot Collaboration* (Contribution à la manipulation à deux bras : des manipulateurs à la collaboration homme-robot), © October 2, 2011

*To Helena, Alice, and Polyana
for standing by my side
in this journey called life*

ACKNOWLEDGMENTS

A PhD thesis is written by only one author, and sometimes this is a lonely task. It consists of many hours of intense thinking, struggling to understand and develop abstract (but hopefully useful) concepts, and trying to present them in a way that prepares the next scientist to further advance the theory and techniques introduced over a hundred pages. However, today my thesis exists, such as it is, and this is thanks to innumerable people. I once read somewhere that “behind a great man, there is always a great woman,” and I think that this could be rephrased as “behind a (great, or not) thesis, there are great people supporting the author.” This page is dedicated to them.

First, I would like to thank Prof. Philippe Fraisse, who accepted me as his student. Always cordial in manner, he has never failed to show himself as open to discussion (technical or philosophical) and has provided me with the very best conditions to work in. Hard working and an exceptional project manager, he has participated in all aspects of my research—including the technical ones (which I think is quite rare among the supervisors of today)—and for this I am very grateful. His ethics and his way of living the academic life will always be a great source of inspiration.

I would also like to thank Sébastien Druon and Stéphanie for helping me when I arrived in Montpellier. Their guidance made it much easier for me to navigate my way through the French bureaucracy.

A special acknowledgement goes to Ahmed Chemori, Christine Azevedo, and Mitsuhiro Hayashibe. In addition to their agreeable presence in my daily life, they are models of hard workers—of people who make things happen. It is thanks to men and women like them that I continue to have great hope for the future of science.

I cordially acknowledge Dr. Etienne Dombre for the very helpful remarks and encouragement he gave me whenever he was able to attend my presentations. I sincerely acknowledge Prof. André Crosnier and Prof. Philippe Poinet for their kindness and their faith in the “Brazilian team.”

I cannot forget LIRMM’s administrative staff, in particular Nicole Gleizes and Olivier Minarro, who were never less than helpful and efficient. Without them, going to a conference would have been so much more difficult. Also, I would like to point out the remarkable job done by LIRMM’s technical staff. It’s thanks to their work behind the scenes that I was able to have all the infrastructure to execute my job.

A special thanks goes to Prof. Véronique Perdereau for her cordiality and kindness. Our discussions about the project were all fruitful, and I was always warmly received in Paris. The same applies to Hoang-Lân Pham, with whom I have had the most stimulating discussions.

This work wouldn’t be what it is without the keen remarks of Prof. Bruno Siciliano and Prof. Masaru Uchiyama. In addition to accepting to be reviewers of my thesis, their works have been an infinite source of inspiration.

I’m sincerely grateful that Prof. Oussama Khatib accepted to be a member of my jury, despite his huge responsibility as the general chair of the IROS’11 conference. I am also thankful for the infrastructure he provided for my PhD defense at Stanford.

I would like to thank Catherine Carmeni for helping to improve the text and my English writing. She did a remarkable job of transforming a barbarian text into a decent one, always with great efficiency and professionalism.

The time that I spent in the lab wouldn't have been so pleasant if the other PhD students hadn't been present. The mix of cultures and ideas definitely has made me a better person. Even at the risk of forgetting to acknowledge some of my colleagues and friends (if I do forget, I will make it up to them by offering "pamonha", "pão de queijo", and "cachaça" in Belo Horizonte, I promise), I must write down a list of the important people who shared their friendship with me over these last three years: Alain Hassoun, Ashvin Sobhee, Azhar Hadmi, Baptiste Magnier, Bastien Durand, Chao Liu, Dalila Goudia, Guilherme Bontorin, Jean Da Rolt, João Azevedo, Johann Lamaury, Luis Vitorio Cargnini, Nabil Zemiti, Naveed Islam, Nicolas Carlesi, Nicolas Philippe, Nicolas Riehl, Pattaraporn Warintarawej, Pawel Maciejasz, Raphael M. Brum, Renan Alves Fonseca, Roseline Bénérière, Sébastien Cotton, Sébastien Krut, Vincent Bonnet, and Zeineb Zarrouk.

My heart will always have a special place for good friends: Alfredo (compadre) Toriz Palacios, Andreea (niña) Elena Ancuta, Divine Maalouf, Giulia Toncelli, Guilherme Sartori Natal, Hai Yang, Jovana Jovic, Lama (Lamex) Mourad, Lotfi (rio de janeeeiro) Chikh, Luis Alonso (Banderas) Sánchez Secades, Michel Dominici, Michele Vanoncini, Mourad Benoussad, Pedro Moreira, Qin Zhang, and Xianbo Xiang.

It is very difficult to express in words my gratitude for these most cherished friends: Antônio P. L. Bó, Mariana C. Bernardes, Carla S. R. Aguiar, and Rogério Richa. We've been good friends for many years, and I'm proud to say that they are part of my family.

Outside the lab, dear friends have become part of my life: Gabriel, Bruna and little João Gabriel (you'll have to bring João to visit Alice!) and Luiz and Denise. I have missed my family in Brazil terribly these last three years, and I'm grateful that they were able to understand and graciously accept my absence. Their support has been so important to me.

Last, but most importantly, I would like to thank my wife Polyana and our two wonderful daughters, Alice and Helena. These three girls definitely make me feel fulfilled, and with them standing by my side, life is worth living.

CONTENTS

<i>List of Figures</i>	xi
<i>List of Tables</i>	xii
<i>Acronyms</i>	xiii
<i>Notation and conventions</i>	xiv
INTRODUCTION	1
1 STATE OF THE ART	5
1.1 Multi-arm manipulation	8
1.1.1 Augmented object and virtual linkage	9
1.1.2 Symmetric control scheme	11
1.1.3 Cooperative task-space	12
1.1.4 Synchronized control	14
1.1.5 Other representations	15
1.2 Conclusion	16
2 KINEMATIC MODELING USING DUAL QUATERNIONS	19
2.1 Mathematical background	20
2.1.1 Quaternions	21
2.1.2 Dual numbers	23
2.1.3 Dual quaternions	24
2.2 Rigid body motion	26
2.2.1 Rotations represented by quaternions	26
2.2.2 Rigid motions represented by quaternions	28
2.2.3 Rigid motions represented by dual quaternions	28
2.2.4 Decompositional multiplication	30
2.3 Robot kinematic modeling	36
2.3.1 Forward kinematic model	36
2.3.2 Dual quaternion Jacobian	38
2.4 Dual quaternions vs. homogeneous transformation matrices	41
2.4.1 Forward kinematic model	41
2.4.2 Jacobian	43
2.4.3 Decompositional multiplication	45
2.5 Conclusion	46
3 TWO-ARM MANIPULATION: THE COOPERATIVE DUAL TASK-SPACE APPROACH	49
3.1 The cooperative dual task-space: arm coordination	50
3.2 Cooperative Jacobians	52
3.2.1 Preliminaries	53
3.2.2 Relative dual quaternion Jacobian	54
3.2.3 Absolute dual quaternion Jacobian	54
3.3 Cooperative dual task-space: object manipulation	56
3.3.1 Preliminaries	56
3.3.2 Wrenches and twists in the cooperative dual task-space	60
3.3.3 Complete description of the cooperative dual task-space	62

- 3.4 Kinematic control in the cooperative dual task-space 67
- 3.5 Control primitives 72
- 3.6 Conclusion 81
- 4 TWO-ARM MOBILE MANIPULATION: THE COOPERATIVE DUAL TASK-SPACE APPROACH 83
 - 4.1 Serially coupled kinematic structure 84
 - 4.2 Extended cooperative dual task-space 86
 - 4.3 Case study: two-arm mobile manipulator 88
 - 4.4 Simulation results and discussions 90
 - 4.5 Conclusion 94
- 5 HUMAN-ROBOT COOPERATION: THE COOPERATIVE DUAL TASK-SPACE APPROACH 95
 - 5.1 Intuitive task definition 96
 - 5.2 Robot's perception of the human motion 97
 - 5.3 Control of the robot arm 97
 - 5.4 Control of the human arm 98
 - 5.4.1 Human arm control by using functional electrical stimulation 98
 - 5.4.2 Human arm control in the Cartesian space 99
 - 5.5 Experiments 101
 - 5.5.1 Water pouring 101
 - 5.5.2 Teleoperation 102
 - 5.5.3 Simultaneous handling using mirrored movements 103
 - 5.5.4 Ball in the hoop 105
 - 5.6 Conclusion 108
- 6 CONCLUSION AND PERSPECTIVES 113
- APPENDIX 115**
 - A GENERAL MATHEMATICAL FACTS, DEFINITIONS, AND PROPERTIES 117
 - A.1 Dual number properties 117
 - A.2 Quaternion properties 117
 - A.3 Properties of Hamilton operators 120
 - A.4 Dual quaternion properties 121
 - A.4.1 Unit dual quaternions and rigid motions 122
 - A.5 Properties of the generalized Jacobian 124
 - A.6 Properties of the decompositional multiplication 125
 - B DENAVIT-HARTENBERG PARAMETERS OF THE ROBOTS 127
 - C EXPERIMENTAL SETUP FOR THE HUMAN-ROBOT COLLABORATION 129
 - C.1 "Teleoperation with collaboration" experiment 129
 - C.2 "Water pouring" and "mirrored movements followed by simultaneous handling" experiments 130
 - C.3 Calibration procedure for the "ball in the hoop" experiment 131
- Bibliography* 133
- Publications* 141
- Index* 143

LIST OF FIGURES

Figure 1	First CAD design of the ASSIST robot, made by CEA.	2
Figure 2	Humanoid robots which can potentially work side-by-side with humans.	6
Figure 3	Two-arm mobile manipulators: PR2 and Twendy-one.	7
Figure 4	Justin and ARMAR-III.	8
Figure 5	Augmented object model.	10
Figure 6	Virtual linkage.	10
Figure 7	Two-arm manipulation using the symmetric control representation.	12
Figure 8	Synchronized interaction between two mobile manipulators.	15
Figure 9	Rotation represented by a quaternion.	26
Figure 10	Frame rotation represented by quaternions.	27
Figure 11	Point rotation represented by quaternions.	27
Figure 12	One rigid motion represented by quaternions.	28
Figure 13	Sequence of rigid motions represented by quaternions.	29
Figure 14	Robot hand manipulating a screwdriver.	31
Figure 15	Standard versus decompositional multiplication.	32
Figure 16	The “frame in the box” example.	35
Figure 17	Denavit-Hartenberg parameters	37
Figure 18	Cooperative dual task-space representation.	51
Figure 19	Manipulating a broom.	52
Figure 20	Wrenches and twists represented by dual quaternions.	56
Figure 21	Cooperative dual task-space: manipulation of a firmly grasped object.	61
Figure 22	Initial and final configurations for the KUKA arm	69
Figure 23	Error of each dual quaternion coefficient.	69
Figure 24	Trajectory executed by the KUKA arm.	70
Figure 25	Two KUKA LWR 4 manipulating a broom.	71
Figure 26	Coefficients of the relative dual quaternion of example 3.4	73
Figure 27	Coefficients of the absolute dual quaternion of example 3.4	74
Figure 28	Angle between the robotic hands in the task of manipulating a broom.	74
Figure 29	Relative translation between the robotic hands in the task of manipulating a broom.	75
Figure 30	Two-arm manipulation of a large box.	75
Figure 31	Manipulation of Rubik’s cube.	75
Figure 32	Opening a bottle.	76
Figure 33	Turning a driving wheel.	76
Figure 34	Positioning a screw driver.	78
Figure 35	Manipulation of a flashlight.	78
Figure 36	Two-arm manipulation: grabbing a balloon.	79

Figure 37	Two-arm manipulation: task of pouring water. 80
Figure 38	Serial kinematic system composed by several intermediate subsystems. 85
Figure 39	Serially coupled kinematic chain for a two-arm mobile manipulator. 88
Figure 40	Sequence corresponding to the reaching phase in the task of pouring water. 92
Figure 41	Task of pouring water using the whole body motion. 93
Figure 42	Examples of cooperative tasks. 97
Figure 43	Experimental setup for the human-robot collaboration. 98
Figure 44	Actuation of the human arm: positioning of the electrodes. 100
Figure 45	The human arm modeled as a one-link serial robot. 100
Figure 46	Task of pouring water. 102
Figure 47	Teleoperation with collaboration. 102
Figure 48	Sequence of human-robot collaboration (HRC) in mirror mode. 104
Figure 49	Constrained movement in the ball in the hoop experiment. 105
Figure 50	Coordinate systems in the ball in the hoop experiment. 106
Figure 51	Sequence of one trial in the ball in the hoop experiment. 109
Figure 52	Time response for the successful trial shown in figure 51. 110
Figure 53	Number of trials versus individual performance. 110
Figure 54	Impact of the cooperation in the success of the task. 110
Figure 55	Comparison of the success rate. 111
Figure 56	Plücker line. 123
Figure 57	Plücker line with respect to frames \mathcal{F}_a and \mathcal{F}_b . 124
Figure 58	Representation of the frames in the “teleoperation with collaboration” task. 130

LIST OF TABLES

Table 1	Cost comparison between homogeneous transformation matrices and dual quaternions. 43
Table 2	Summary of control primitives and correspondent task Jacobians. 78
Table 3	Extended absolute Jacobian. 90
Table 4	Definition of the task of pouring water. 91
Table 5	Denavit-Hartenberg (D-H) parameters for the human arm. 101
Table 6	Parameters of the cooperative tasks. 104
Table 7	Criteria used to evaluate the ball in the hoop task. 107
Table 8	Standard D-H parameters of the KUKA LWR 4 (Giordano, 2007). 127
Table 9	Modified D-H parameters of the Hoap-3’s arms. 127

ACRONYMS

ASIMO	Advanced Step in Innovative MObility
CAD	computer-aided design
CEA	Commissariat à l'énergie atomique et aux énergies alternatives
D-H	Denavit-Hartenberg
DLR	German Aerospace Center
DOF	degrees of freedom
FKM	forward kinematic model
FES	functional electrical stimulation
HRI	human-robot interaction
HRC	human-robot collaboration
ISIR	Institut des Systèmes Intelligents et de Robotique
LAAS	Laboratoire d'Analyse et d'Architecture des Systèmes
LIRMM	Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier
NASA	National Aeronautics and Space Administration
PRM	Probabilistic Roadmap
ROS	Robot Operating System

NOTATION AND CONVENTIONS

Lowercase plain letters represent scalars. For instance,

$$a = 1, b = -0.333.$$

The variables i, j, l, m, n always denote integers.

The hatted variables $\hat{i}, \hat{j}, \hat{k}$ denote imaginary units.

The following sets are used throughout the thesis:

- \mathbb{R} : set of real numbers
- \mathbb{H} : set of quaternions
- \mathcal{H} : set of dual quaternions

Lowercase bold letters represent column vectors or quaternions:

$$\mathbf{a} = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}, \mathbf{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}, \quad \mathbf{a}, \mathbf{b} \in \mathbb{R}^n$$

or

$$\mathbf{c} = c_1 + \hat{i}c_2 + \hat{j}c_3 + \hat{k}c_4, \quad \mathbf{c} \in \mathbb{H}.$$

Uppercase bold letters usually represent matrices:

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{1m} & \cdots & a_{mn} \end{bmatrix},$$

with the exception of \mathbf{F} and \mathbf{M} , which are column vectors representing forces and moments, respectively.

Underlined variables represent dual numbers. For instance, $\underline{a} = a_1 + \varepsilon a_2$ is a dual scalar whereas $\underline{\mathbf{h}} = \mathbf{h}_1 + \varepsilon \mathbf{h}_2$ is a dual quaternion, where ε is the dual unit.

The letter T is reserved to denote the transpose of matrices and vectors; for instance, \mathbf{A}^{T} is the transpose of \mathbf{A} .

Coordinate systems are represented by \mathcal{F}_{ref} , where *ref* can be any label representing the name of the coordinate system; for example, \mathcal{F}_1 and \mathcal{F}_{world} . If the coordinate system is

not important, usually the subscript is omitted, e.g., \mathcal{F} .

Rotation matrices representing the rotation from frame i to frame j are given by \mathbf{R}_j^i . If they are given with respect to a general reference frame or the reference frame is not defined, the superscript is omitted; for instance, \mathbf{R}_j . The same apply to vectors; that is, when the reference frame is not defined or all the quantities are related to the same reference frame, the superscript can be omitted. For instance, the velocities \mathbf{v}_1 and \mathbf{v}_2 are given with respect to the same reference frame (which is not specified), but \mathbf{v}_1^a and \mathbf{v}_2^b are given with respect to \mathcal{F}_a and \mathcal{F}_b , respectively.

The same rule applies for homogeneous transformation matrices, quaternions and dual quaternions; that is, \mathbf{H}_j^i represents the homogeneous transformation from frame i to frame j , the quaternion \mathbf{r}_j^i represents the rotation from frame i to frame j , and the dual quaternion $\underline{\mathbf{x}}_j^i$ represents the rigid motion from frame i to frame j .

Physical variables, such as position, velocity, force, and so on, are represented—unless explicitly stated otherwise—by quaternions; for example, the velocity \mathbf{v} is represented by the quaternion $\mathbf{v} = \hat{i}v_x + \hat{j}v_y + \hat{k}v_z$. Only chapter 1 does not follow this rule.

Dual positions, twists, and wrenches are represented by dual quaternions, and usually the symbols $\underline{\mathbf{x}}$, $\underline{\boldsymbol{\xi}}$, and $\underline{\mathbf{f}}$ are used to represent each one of these physical entities, respectively.

Matrices corresponding to task Jacobians are represented by \mathbf{J} . Subscripts are usually used to denote the controlled task.

\mathbf{J}^+ and \mathbf{J}^\dagger denote the Moore-Penrose pseudoinverse and the damped least-square inverse of the matrix \mathbf{J} , respectively.

\mathbf{G} is the generalized Jacobian.

Gain matrices are usually represented by \mathbf{K} , whereas scalar gains are usually represented by λ .

$\text{Re}(\mathbf{x})$ and $\text{Im}(\mathbf{x})$ denote the real and imaginary parts of the quaternion \mathbf{x} . The same applies for dual quaternions.

$\mathbf{1}_m = [\hat{i} \ \hat{j} \ \hat{k}]^T$ is the imaginary vector unit.

ε is the dual unit.

Given the dual quaternion $\underline{\mathbf{x}}$:

- $\overset{+}{\mathbf{H}}(\underline{\mathbf{x}})$ and $\overset{-}{\mathbf{H}}(\underline{\mathbf{x}})$ are the Hamilton operators of $\underline{\mathbf{x}}$.
- $\mathcal{P}(\underline{\mathbf{x}})$ and $\mathcal{D}(\underline{\mathbf{x}})$ denote the primary and secondary parts of $\underline{\mathbf{x}}$.
- $\underline{\mathbf{x}}^*$ is the conjugate of $\underline{\mathbf{x}}$.

- The operator $\text{vec } \underline{\mathbf{x}}$ performs the mapping of $\underline{\mathbf{x}}$ into \mathbb{R}^8 .

The symbol \otimes denotes the decompositional multiplication.

The symbol \times represents the cross product between pure quaternions (the ones with real parts equal to zero). For instance, $\mathbf{a} \times \mathbf{b}$, $\mathbf{a}, \mathbf{b} \in \mathbb{H}$, is the cross product between the quaternions \mathbf{a} and \mathbf{b} , which is equivalent to $\overset{\times}{\mathbf{H}}(\mathbf{a}) \text{vec } \mathbf{b}$ (see appendix A).

The dual quaternion $\underline{\mathbf{x}}_{\mathbf{b}}^{\mathbf{a}}$ raised to the n -th power is denoted by $(\underline{\mathbf{x}}_{\mathbf{b}}^{\mathbf{a}})^{\{n\}}$. The curly brackets are used to prevent mixing the power n with the superscript a .

INTRODUCTION

The robots of today are no longer confined to structured environments. As a result of fifty years of research, we are seeing increasingly more robots outside of factories, ranging from unmanned marine and aerial robots (Marani et al., 2010; Cheng et al., 2009) to robots in human environments (Kemp et al., 2007) like hospitals (Hockstein et al., 2007), and even robots capable of traveling across the desert (Thrun et al., 2006) and robots connected to the internet (Tenorth et al., 2011).

With this paradigm shift, the research in assistive and “human-centered” robotics has become more intensive, and new challenges are now being faced. For instance, although robots and humans do not usually share the same workspace in factories, assistant robots do physically interact with people, which poses some safety constraints. Also, industrial robots must execute tasks that are usually completely described. In human-robot interaction (HRI), however, the description of the task is often incomplete; for example, the order “give me the small bottle of water” implies recognizing a small bottle containing water. How is small defined in this case, and how can a bottle of water be differentiated from a bottle of tequila?

A closer analysis of this simple task can shed some light on its inherent difficulties. Indeed, the localization and tracking of even fully-described simple objects in cluttered environments require robust computer vision algorithms. Also, in order to grasp a distant object the robot must navigate without colliding with obstacles across the environment to reach the desired object. Still considering the “give me the small bottle of water” task, suppose that the object is located, grasped and brought to the person who commanded the robot. The robot then must safely give the bottle to the person. How should the bottle be handed over? When should the robot release the bottle? What if the person moves while the robot is making the transfer? What if the person says “*open the bottle*”?

Although there is still considerable work being done (and to be done) in computer vision, some good and stable software libraries are fortunately available (Marchand et al., 2005; Bradski & Kaehler, 2008). Navigation and motion algorithms are quite mature nowadays (Choset et al., 2005). Moreover, with the advent of software architectures like Robot Operating System (ROS)¹ and Orocos², the integration of navigation and motion algorithms into robotic platforms now tends to be widespread.

Daily domestic household tasks are usually easily performed by healthy youths but, despite recent advances, robots are not capable of performing them effectively (Kemp et al., 2007). The challenges of assistive robotics are even more accentuated when robots must assist the elderly or individuals with impairments. These people often have motion limitations that need to be taken into consideration at the moment of physical human-robot interaction (e.g., tremor, absence of motion, and so on).

The above examples showed that even the “simplest” of tasks can be very challenging for robots. The “give me the small bottle of water” task is usually considered simple because humans generally perform it naturally. But these tasks are challenging for robots

1. <http://www.ros.org/>

2. <http://www.orocos.org/>

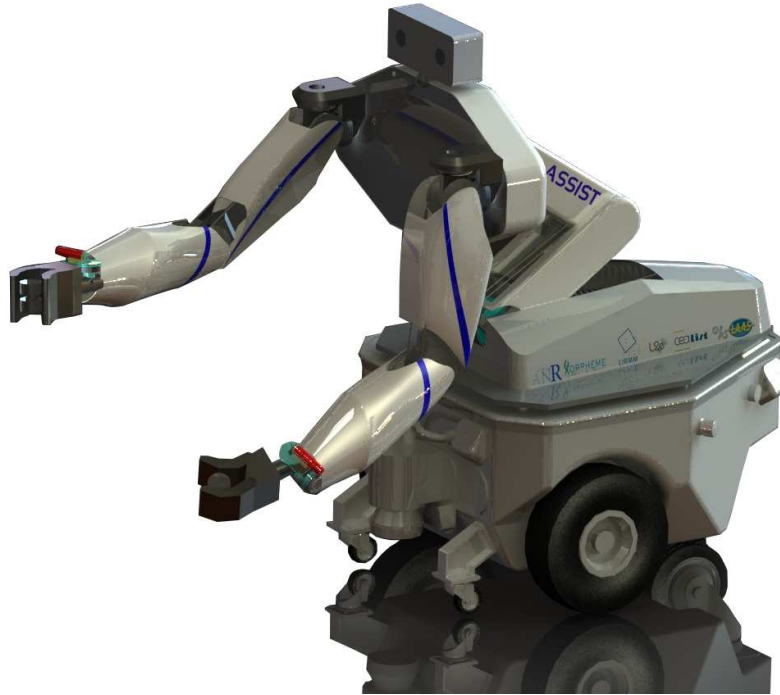


Figure 1: First CAD design of the ASSIST robot, made by CEA.

because, unlike humans, robots are not the product of a very long evolutionary process. At least for the moment, there is plenty of room for robotics research.

This thesis was developed within the context of the ASSIST project. The aim of this project is to build a two-arm mobile manipulator to assist quadriplegic individuals in their daily lives. It has five partners (Commissariat à l'énergie atomique et aux énergies alternatives (CEA), Institut des Systèmes Intelligents et de Robotique (ISIR), Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS), Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), and Propara Clinical Center) working together in order to push the frontiers of the state of the art in assistive robotics.

The principles that have guided the robot design are safety, dexterity and capabilities of human-robot interaction. The ASSIST robot, depicted in figure 1, is composed of a mobile base, two arms with seven degrees of freedom (DOF) each, and a torso with one DOF. The robot should be capable of performing simple tasks such as locating and grasping simple objects, handing over objects to a person, and opening a bottle. Although the aforementioned tasks are quite trivial for a healthy person, reliable robotic implementations of these features is still a challenge.

Because there are still many open questions in robotics and much to improve in its current techniques, it is humanly impossible to tackle them all. From the point of view of practical applications, this thesis is thus focused mainly on two-arm manipulation and its application to mobile manipulators and HRC.

Two-arm manipulation is particularly interesting from both practical and theoretical points of view. From the practical side, bi-manual tasks are constantly required over the course of a typical day, and a successful assistant robot must be skilled in these types of manipulation. From a theoretical point of view, two-arm manipulation poses several challenges. First, a suitable description for two-arm coordination should be sim-

ple enough to enable generalizations; that is, different tasks should be easily defined without any modification in the original description, and the final formalism should be easily extended to more complex robots (e.g., mobile manipulators, humanoids, and so on). Also, this description should take into account not only the bi-manual tasks executed by the robot, but also the bi-manual tasks involving different agents, such as handing over, pouring water, and so on. Because the robot must interact with humans, the two-arm coordination should also be reactive. For these reasons, the techniques developed in this thesis tend to be low-level oriented.

Last, departing from the requirements imposed by the ASSIST project, I was also interested in some fundamental aspects related to the foundations of robotics; notably, the unification of robot kinematic modeling and control by means of dual quaternions.

CONTRIBUTIONS

The contributions of this thesis are divided into three main groups:

1. The robot kinematic modeling is unified with kinematic control by means of dual quaternions, and a method for obtaining the forward kinematic model (FKM) and the differential FKM of serial robots is proposed, as well as kinematic controllers that directly use the dual quaternion as the input.
2. A generalized two-arm manipulation formalism is developed based on previous techniques from Uchiyama & Dauchez (1988), Chiacchio et al. (1996), and Cavallaro et al. (2000). The representation of two-arm manipulation is first unified by means of dual quaternions; that is, dual positions (positions and orientations), twists (linear and angular velocities), and wrenches (forces and moments) are all represented in dual quaternion space. The formalism is then extended in order to take into account any serially coupled kinematic robot; for instance, mobile manipulators.
3. Techniques for intuitive human-robot collaboration are developed, such that conceptually different tasks can be represented by the same set of equations. Tasks like pouring water, teleoperation with collaboration, simultaneous handling, and mirrored movements are all represented within the two-arm manipulation formalism. In this manner, bimanual tasks previously designed to be executed by robots can be performed cooperatively between humans and robots. In addition, a novel application of human-robot collaboration is proposed, where the robot controls not only its arm, but also the human arm by means of functional electrical stimulation (FES). This application opens a new door toward novel interactions between robots and impaired individuals (e.g., quadriplegics).

In addition to presenting the organization of the thesis, the next section highlights the contributions, chapter by chapter.

ORGANIZATION OF THE THESIS

The thesis is organized into six chapters, summarized as follows:

Chapter 1 presents some of the most recent developments in bimanual robots designed to work side-by-side with humans. Also, the mainstream two-arm manipulation techniques are revisited in order to provide the foundations on which this thesis is built.

Chapter 2 first introduces the mathematical background needed to understand the concepts proposed throughout the thesis. Readers should read this chapter carefully, as it also establishes the notation and the overall nomenclature. Also, it proposes a novel operation, the decompositional multiplication, which aims to represent rigid motions that are invariant with respect to the pose of the modified frame. The chapter further proposes robot kinematic modeling by using dual quaternions. Although robot kinematics has been extensively studied over the past forty years, this chapter presents another point of view on the subject. Last, dual quaternions and homogeneous transformation matrices are compared in terms of the number of the elementary operations required by each representation.

Chapter 3 proposes a new approach, the cooperative dual task-space, based on dual quaternions for bimanual manipulation. Inspired by some of the works reviewed in chapter 1, a complete description of two-arm coordination/manipulation is developed, by using dual quaternions, in terms of the dual positions, twists, and wrenches involved in the bimanual task. Furthermore, the chapter proposes techniques of kinematic control based entirely on the dual quaternion representation, and techniques designed for redundant robots are revisited in the context of the cooperative dual task-space. Last, control primitives are proposed in order to ease the definition of tasks.

Chapter 4 proposes a generalization of the cooperative dual task-space in order to take into account whole-body motions. A case study using a simulated two-arm mobile manipulator is presented.

Chapter 5 proposes a novel application of the cooperative dual task-space in the description and control of human-robot collaboration tasks. It shows that tasks that are usually regarded as conceptually different can be represented by the same set of equations and thus are mathematically equivalent. Last, a new approach in human-robot collaboration is proposed, where the robot controls not only its arm, but also the human arm by means of FES. This new approach can potentially be applied to novel interactions between robots and disabled people.

Chapter 6 presents the concluding remarks and perspectives for future works.

NOTE ABOUT THE LANGUAGE AND STYLE

This thesis follows the guidelines of the Chicago Manual of Style for English (The University of Chicago, 2010). This style, also adopted by the IEEE, does not condemn the use of the first person in scientific texts and thus differently somewhat from accepted style in some of the Latin languages, such as Portuguese and French. As a result, the first person singular is used in this thesis, although sparingly, to express my opinions or personal choices. On the other hand, the editorial “we” is *never* used, as the thesis was written by only one person. Therefore, whenever “I” appears in the text, the reader should read “the author of this thesis.” On the other hand, if “we” is used in the text, it will always be the inclusive “we”; that is, the readers should read “the author of this thesis and I (the reader).”

STATE OF THE ART

The dream of having robots to serve and help humans has a long history and is deeply embedded in the popular imagination. This dream is amply reflected in the large number of Hollywood movies and science fiction books on the human-robot relationship. Some of them are apocalyptic, with machines fighting against humanity, whereas others portray robots as gently helping people by means of high intelligence and motor dexterity. The common characteristic of such diametrically opposed visions is the huge capacity of robots to interact with humans and to autonomously solve tasks that should normally be solved only by intelligent beings. Although humanity has not reached this utopia, over the past forty years engineers, scientists, psychologists, mathematicians, and others have collectively been working to build robots capable of helping people in several types of tasks and fields. Today we can see, for example, rehabilitation and healthcare robots, domestic robots, robots for hazardous applications, and so on (Siciliano & Khatib, 2008).

For robots that interact closely with humans and/or in human environments, anthropomorphic structures have always had an elevated status compared with non-anthropomorphic counterparts. One reason is that “human tools match human dexterity” (Kemp et al., 2008). This means that humans design and build environments and tools suitable for human beings. Thus, one might expect that the more a robot is similar to humans, the fewer the modifications needed in the environment and/or tools in order to use the robot effectively. Furthermore, humanoids and anthropomorphic designs facilitate the interaction between an individual person and the robot, since people are used to working with other people (Kemp et al., 2008).

Several humanoid robots have already been designed to assist and interact with people. One of the most famous is Advanced Step in Innovative MObility (ASIMO), whose ultimate goal is to support human daily activities (Sakagami et al., 2002). ASIMO, shown in figure 2a, has some quite impressive abilities, being able, for example, to recognize people by means of a visual and auditory system and to interact with them by using gestures and a speech synthesis system. It can navigate in indoor environments, avoiding obstacles and using stairs, and plan its actions by using a behavior-based planning architecture (Sakagami et al., 2002).

Another robot, Robonaut (Bluethmann et al., 2003), was primarily designed to work in outer space, as it is capable of cooperating with humans in tasks such as soldering and taking electrical measurements. Although designed within a teleoperation framework, Robonaut has some degree of autonomy. As it has a vision system capable of tracking people and common tools like wrenches, it can search for a requested object by using voice recognition and give it to a human companion.

Robonaut’s second generation, Robonaut 2 (fig. 2b), is a robot with 42 DOF capable of using tools designed for humans (Diftler et al., 2011). This robot was jointly developed by National Aeronautics and Space Administration (NASA) and General Motors, being the first humanoid to be in outer space. In the robot design, human-robot interaction

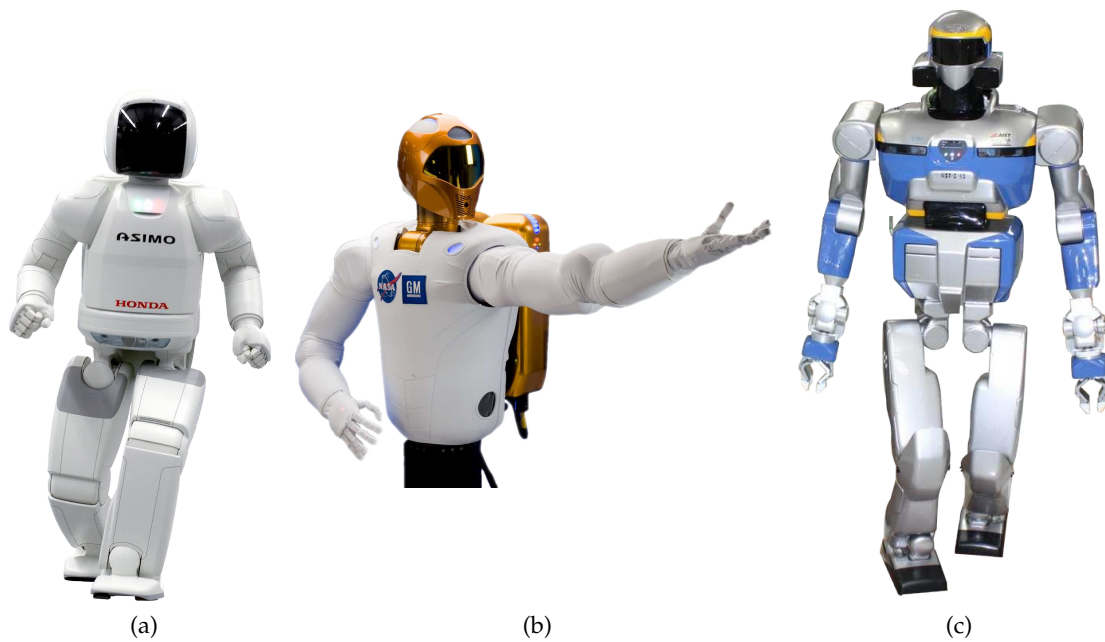


Figure 2: Humanoid robots which can potentially work side-by-side with humans: (a) ASIMO (courtesy of Honda), (b) Robonaut 2 (courtesy of NASA), and (c) HRP-2 (courtesy of AIST-CNRS).

aspects were paramount. For instance, the robot makes use of series elastic actuators, exhibiting large shock tolerance and accurate and stable force control. Furthermore, it uses an impedance-based control system that limits the force that the robot applies to the environment, thus making it suitable to work side-by-side with humans (Diftler et al., 2011).

A very popular humanoid, the HRP-2 (fig. 2c), has also been used to evaluate the capabilities of humanoids in assisting humans. For instance, Okada et al. (2005) developed a complete software system for HRP-2 and showed the impressive capabilities of this robot in a kitchen environment. The robot is capable of navigating through the environment while moving obstructing objects (e.g., chairs) whenever necessary.

The robot is equipped with a vision system, as well as several layers of motion and action planners. The motion planners take into account stability constraints, as well collision avoidance with the environment. Okada et al. (2005) performed an experiment where the robot was capable of recognizing a teapot, then poured water into a nearby cup, and finally navigated while carrying a tray with the cup on it.

One drawback of using humanoid robots is that balance control must be ensured. Although there have been considerable advances in this field, humanoids still do not have good enough balance control to interact safely with people—mainly those with impairments—in realistic scenarios. Indeed, the great majority of big humanoids are confined to laboratory environments and they are operated while attached to chain belts to avoid damage due to falling.

Other robots, mobile manipulators, have been developed to work in human environments. Willow Garage’s PR2, shown in figure 3a, integrates several software components, such as navigation, perception, manipulation, motion planning, and so on, on

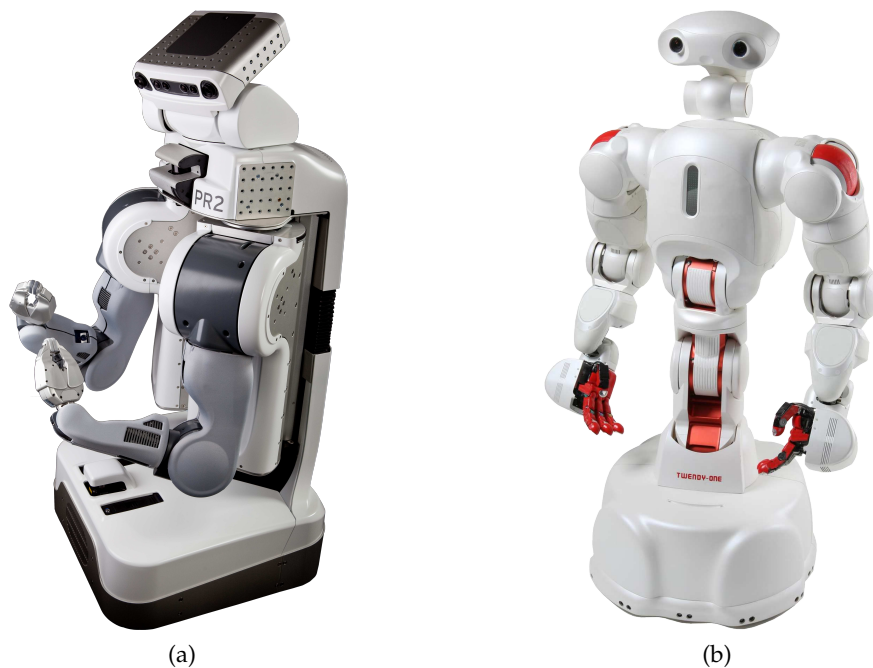


Figure 3: Two-arm mobile manipulators: (a) PR2 (Courtesy of Willow Garage); (b) Twendy-one (Courtesy of Sugano Laboratory, WASEDA University).

top of the Robot Operating System (ROS). PR2 is capable of fetching drinks, navigating and opening doors, and plugging itself into power outlets (Bohren et al., 2011). The huge complexity inherent to placing such an amount of heterogeneous components is alleviated by the use of ROS.

Specifically designed with the elderly Japanese woman in mind, Twendy-one (fig. 3b) is capable of “helping the sitting-up motion support, transferring the caretaker onto a wheelchair and manipulating kitchen tools to make the breakfast” (Iwata & Sugano, 2009). Thanks to its 4-DOF torso, Twendy-one can pick up objects from the floor. Also, the passivity-based mechanisms for the robot arms enable safe interaction with humans and the environment. Because of this passivity, the arms can absorb the external forces arising from imprecisions in position control.

Rollin’ Justin (fig. 4a), currently one of the most popular two-arm mobile manipulators, “is designed for research on sophisticated control algorithms for complex kinematic chains as well as mobile two-handed manipulation and navigation in typical human environments” (Borst et al., 2009). This robot was developed by German Aerospace Center (DLR) and can be used for safe human-robot interaction because of the impedance behavior on joint, end-effector, and object levels. Using two of DLR’s lightweight arms and an advanced computer vision system, Rollin’ Justin is capable of fine manipulation (e.g., in the task of preparing coffee (Schmidt et al., 2011)), dexterous two-arm manipulation (Ott et al., 2006), and even catching flying balls (Schmidt et al., 2011).

Another robot, ARMAR-III (fig. 4b), is also capable of performing complex tasks in household environments. For instance, ARMAR-III can recognize cups, dishes, and boxes of any kind of food, and it can autonomously load and unload a dishwasher with different kind of objects—e.g., tetra packs, bottle with tags, etc (Asfour et al., 2006).

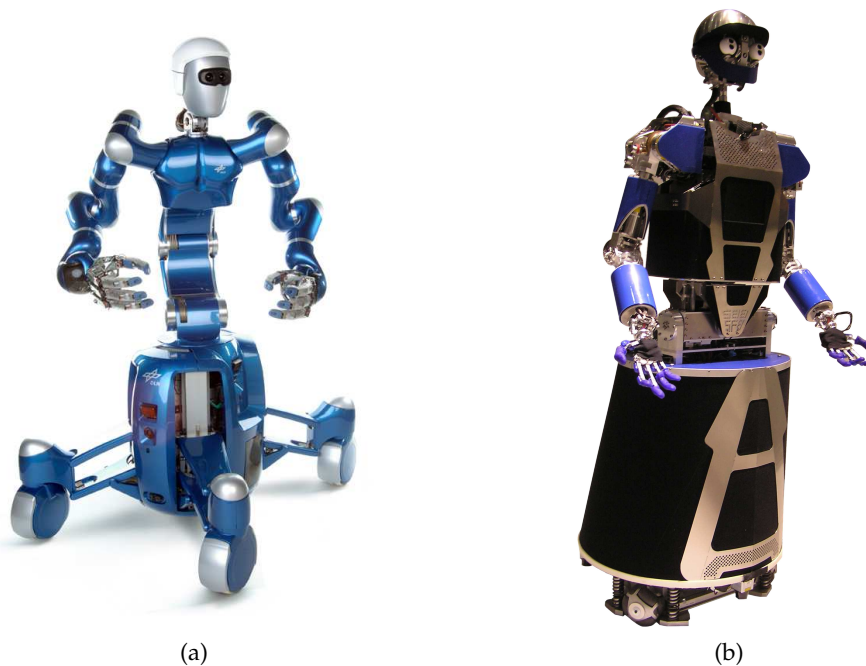


Figure 4: Two-arm mobile manipulators: (a) Justin (Courtesy of DLR); (b) ARMAR-III (courtesy of Karlsruhe Institute of Technology).

The aforementioned robot systems demonstrate an important point: in order to share the same environment with humans, robots must be adapted for human environments and for using human tools. Dexterous manipulation turns out to be crucial in these robot systems. Moreover, because two-arm manipulation offers possibilities in terms of dexterous manipulation (e.g., carrying large payloads, opening bottles), successful assistant robots must be skilled in this kind of manipulation.

The next section presents the state of the art and mainstream techniques used to represent and perform multi-arm manipulation.

1.1 MULTI-ARM MANIPULATION

In the past thirty years, considerable research has been conducted on multi-manipulator systems (Caccavale & Uchiyama, 2008). This kind of cooperative system—of which two-arm manipulation is a particular case—can be used to carry heavy payloads and perform complex assembly tasks (Caccavale & Uchiyama, 2008). However, these advantages come with the drawback of having a more complex system. For instance, multiple manipulators cause internal stresses in the manipulated object, and typically a force control scheme has to be used in order to minimize these forces.

The following sections present the main approaches proposed to tackle the problem of multi-arm manipulation.

1.1.1 Augmented object and virtual linkage

Khatib (1988) presented a control scheme using N robots with m -DOF each and rigidly connected to a common manipulated object. Khatib called the resultant system—that is, the end-effector plus the object—the augmented object, since the description of the overall system takes into account the inertial characteristics of all the effectors and the object. This augmented object is submitted to an operational¹ force \mathbf{f}_O at the operational point \mathbf{x}_O , as shown in figure 5. The operational force \mathbf{f}_O is the resultant of the contribution of each of the end-effectors' wrenches. Considering the constraints of the augmented object model, Khatib wrote the system's equations of motion analogously to his operational space formulation (Khatib, 1987), providing an elegant generalization of the operational space:

$$\mathbf{f}_O = \Lambda_s(\mathbf{x}_O) \ddot{\mathbf{x}}_O + \Pi_s(\mathbf{x}_O) [\dot{\mathbf{x}}_O \dot{\mathbf{x}}_O] + \mathbf{p}_s(\mathbf{x}_O),$$

where $\Lambda_s(\mathbf{x}_O)$ is the kinetic energy matrix, $\Pi_s(\mathbf{x}_O)$ is the matrix of the centrifugal and Coriolis forces² and $\mathbf{p}_s(\mathbf{x}_O)$ is the gravity vector. The subscript s stands for the N -effector/object system, implying that the aforementioned matrices and gravity vector take into account the object and all N -effectors. In order to control the augmented object, Khatib proposed the control structure

$$\mathbf{f}_O = \widehat{\Lambda}_s(\mathbf{x}_O) \mathbf{f}_{Od} + \widehat{\Pi}_s(\mathbf{x}_O) [\dot{\mathbf{x}}_O \dot{\mathbf{x}}_O] + \widehat{\mathbf{p}}_s(\mathbf{x}_O),$$

where $\widehat{\Lambda}_s$, $\widehat{\Pi}_s$, and $\widehat{\mathbf{p}}_s$ are the estimates of the respective matrices, and \mathbf{f}_{Od} is the desired operational force. This control structure considers the net value of the operational forces. However, to allocate the individual effector forces, Khatib imposed the condition that the forces \mathbf{f}_{O_i} produced by each effector would be aligned with \mathbf{f}_O ; that is,

$$\mathbf{f}_{O_i} = \alpha_i \mathbf{f}_O, \quad (1.1)$$

resulting in the following vector of joint forces:

$$\boldsymbol{\tau}_i = \alpha_i \mathbf{J}_i^T \mathbf{f}_{O_i}, \quad (1.2)$$

where \mathbf{J}_i is the geometric Jacobian of the i -th manipulator and α_i is chosen such that the effort is equally distributed among the manipulators. Eq. (1.1) and (1.2) show two important consequences: first, the system is not capable of handling internal forces (i.e., forces that do not produce movements, only internal stresses in the manipulated object) and, second, the system requires torque-actuated robots, which are not always available.

In order to control the internal forces and moments, Williams & Khatib (1993) proposed a physical model for these internal stresses that appear in multi-arm manipulation. This model is based on a virtual linkage between the grasping points, as shown in figure 6. Acknowledging that forces cause stress throughout the object, whereas moments cause local stress, Williams & Khatib separated the kinematic structure of the virtual linkage into two elements. The first element is a prismatic joint to represent the

1. In Khatib's nomenclature, the operational force is a wrench (the vector containing force and moment) applied at the operational point. Moreover, the vector \mathbf{x}_O representing the generalized coordinates of the operational point contains information of position and orientation.

2. In Khatib's symbolic notation, $[\dot{\mathbf{x}}\dot{\mathbf{x}}]$ stands for $\begin{bmatrix} \dot{x}_1^2 & 2\dot{x}_1\dot{x}_2 & \dots & 2\dot{x}_1\dot{x}_m & \dot{x}_2^2 & \dots & \dot{x}_m^2 \end{bmatrix}$, where $x_1 \dots x_m$ are the coordinates of the operational point.

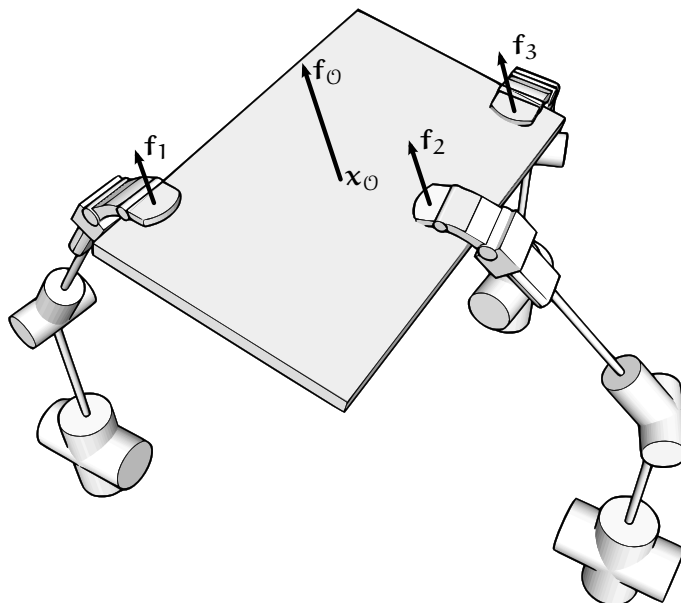


Figure 5: Augmented object model: the forces and moments f_i at the i -th end-effector are aligned with the operational force f_0 . Using only this model, the internal forces cannot be controlled.

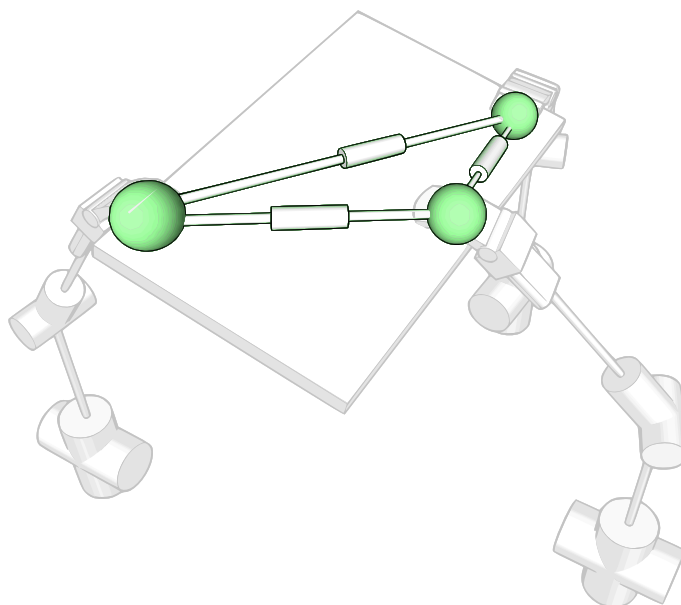


Figure 6: Virtual linkage: spherical joints at each grasp represent internal moments, whereas prismatic joints between two end-effectors represent internal forces.

internal forces due to the interaction between two arms. The second element, representing the internal moments caused by each actuator, is a spherical joint connecting the prismatic joints, as illustrated in figure 6.

The relationship expressing the resultant (\mathbf{f}_{res}) and internal (\mathbf{f}_{int}) wrenches in terms of applied forces and moments at each grasp point is given by:

$$\begin{bmatrix} \mathbf{f}_{\text{res}} \\ \mathbf{f}_{\text{int}} \end{bmatrix} = \mathbf{W} \begin{bmatrix} \mathbf{F}_G \\ \mathbf{M}_G \end{bmatrix},$$

where $\mathbf{F}_G = [\mathbf{F}_1^T \ \dots \ \mathbf{F}_N^T]^T$ and $\mathbf{M}_G = [\mathbf{M}_1^T \ \dots \ \mathbf{M}_N^T]^T$ are the vectors with the forces and moments in each grasp point and \mathbf{W} is the grasp description matrix.

In order to determine the grasp description matrix, Williams & Khatib performed a quasi-static analysis. As a consequence, the description does not take into account the object velocities and accelerations to describe the internal forces. Moreover, the authors do not consider the influence of internal moments in the resolution of the internal forces, although this approximation is valid when the internal moments are very small.

1.1.2 Symmetric control scheme

Uchiyama & Dauchez (1988) introduced the concept of a symmetric control scheme for a tightly grasped object. This concept was based on the relationship between forces and velocities at the object and the counterparts at the “virtual sticks”; that is, vectors originating from the end-effector frame and ending at the origin of the reference frame \mathcal{F}_a attached to the object, as shown in figure 7. Using a static analysis and assuming that the deformation of the object was very small, Uchiyama & Dauchez showed that the external and internal forces in the object were given by

$$\mathbf{f}_c = \mathbf{U}^{-1} \mathbf{f}_s, \quad (1.3)$$

where

$$\mathbf{f}_c = \begin{bmatrix} \mathbf{f}_a \\ \mathbf{f}_r \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} \frac{1}{2}\mathbf{I}_6 & \mathbf{I}_6 \\ \frac{1}{2}\mathbf{I}_6 & -\mathbf{I}_6 \end{bmatrix}, \quad \mathbf{f}_s = \begin{bmatrix} \mathbf{f}_{s_1} \\ \mathbf{f}_{s_2} \end{bmatrix}, \quad \mathbf{f}_{s_i} = \begin{bmatrix} \mathbf{F}_i \\ \mathbf{M}_i + \mathbf{F}_i \times \mathbf{p}_i \end{bmatrix},$$

with \mathbf{f}_{s_i} being the wrench at the tip of the virtual sticks \mathbf{p}_i caused by forces and moments, \mathbf{F}_i and \mathbf{M}_i , at each grasp point.

Based on the principle of virtual work, the authors realized that the velocities applied at the tip of the virtual sticks and the resultant velocity in the object are given by

$$\mathbf{f}_s^T \boldsymbol{\xi}_s = \mathbf{f}_c^T \boldsymbol{\xi}_c \xrightarrow{(1.3)} \boldsymbol{\xi}_c = \mathbf{U}^T \boldsymbol{\xi}_s, \quad (1.4)$$

where $\boldsymbol{\xi}_s = [\boldsymbol{\xi}_{s_1}^T \ \boldsymbol{\xi}_{s_2}^T]^T$ and $\boldsymbol{\xi}_c = [\boldsymbol{\xi}_a^T \ \boldsymbol{\xi}_r^T]^T$. The vector $\boldsymbol{\xi} = [\mathbf{v}^T \ \boldsymbol{\omega}^T]^T$ is a twist, where \mathbf{v} is the time derivative of the position vector and $\boldsymbol{\omega}$ is the angular velocity.

Finally, the position of the virtual sticks could be determined by integrating the velocities, although extra care had to be taken in the integration of the rotational components, since it did not lead to a unique representation of the orientation (Uchiyama & Dauchez, 1988). Again, the authors had to consider a very small deformation, which limited the application of the overall method. Considering this approximation, the result of the integration of $\boldsymbol{\xi}_c$ leads to

$$\boldsymbol{\chi}_c = \mathbf{U}^T \boldsymbol{\chi}_s, \quad (1.5)$$

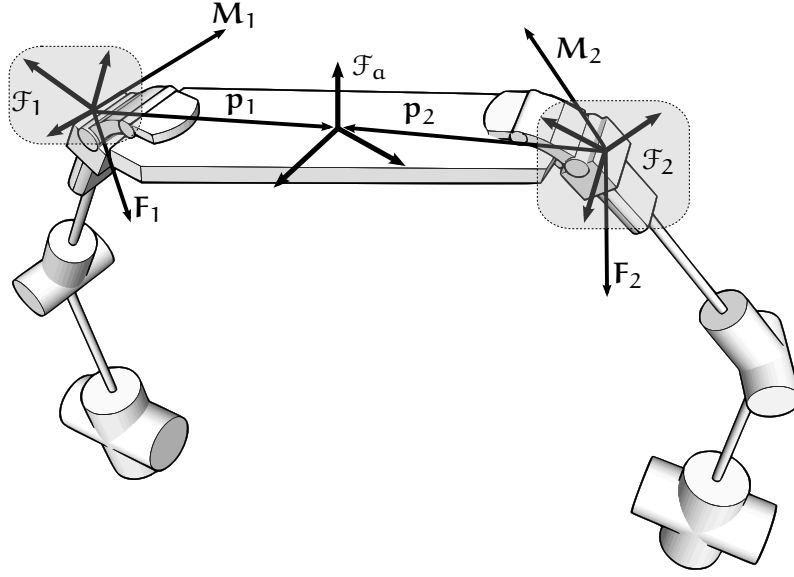


Figure 7: Two-arm manipulation using the symmetric control representation. \mathcal{F}_a is a reference frame attached to the manipulated object, \mathbf{p}_1 and \mathbf{p}_2 are the virtual sticks, and \mathbf{F}_i and \mathbf{M}_i are force and moment acting at the i -th end-effector.

where $\mathbf{x}_s = [\mathbf{x}_{s_1}^T \ \mathbf{x}_{s_2}^T]^T$ and $\mathbf{x}_c = [\mathbf{x}_a^T \ \mathbf{x}_r^T]^T$, with $\mathbf{x} = [\mathbf{p}^T \ \boldsymbol{\phi}^T]^T$. In the generalized position vector \mathbf{x} , the vector \mathbf{p} is the position and the vector $\boldsymbol{\phi}$ is the orientation. The orientation vector typically depends on the representation used for rotations (e.g., Euler angles, quaternions, etc).

Yamano et al. (2004) proposed further extensions to the symmetric control scheme to take into account flexible arms. Assuming that the manipulated object was rigid, they used (1.3)–(1.5) to represent the cooperative task and perform hybrid position/force control. These set of equations were also used to derive the equations of motion of the two-arm system, which were subsequently used to perform vibration suppression. The remarkable finding was that, although the control law had to be changed to take into account the elastic deflections of the arms, the task could be still described by (1.3)–(1.5).

1.1.3 Cooperative task-space

Chiacchio et al. (1996) realized that two-arm coordination does not always require a firmly grasped object and thus took the inverse approach of Uchiyama & Dauchez by directly defining the absolute and relative positions and orientations to represent the cooperative task. The result is a set of four variables, given by

$$\mathbf{p}_r = \mathbf{p}_2 - \mathbf{p}_1 \quad (1.6)$$

$$\mathbf{R}_r = \mathbf{R}_2^1 \quad (1.7)$$

$$\mathbf{p}_a = \frac{\mathbf{p}_1 + \mathbf{p}_2}{2} \quad (1.8)$$

$$\mathbf{R}_a = \mathbf{R}_1 \mathbf{R} \left\{ \mathbf{n}_{12}, \frac{\phi_{12}}{2} \right\}, \quad (1.9)$$

where \mathbf{R}_r is the rotation matrix that represents the relative orientation of the end-effectors; that is, the rotation which aligns \mathcal{F}_1 with \mathcal{F}_2 (see fig. 7). The rotation matrix $\mathbf{R}\{\mathbf{n}_{12}, \phi_{12}/2\}$ is defined such that

$$\mathbf{R}_2^1 = \mathbf{R}\{\mathbf{n}_{12}, \phi_{12}\},$$

where \mathbf{n}_{12} and ϕ_{12} are the unit vector and the angle that realize the rotation described by \mathbf{R}_2^1 (Chiacchio et al., 1996). Thus, \mathbf{R}_a can be regarded as “half of the rotation” needed to align \mathcal{F}_1 with \mathcal{F}_2 , expressed with respect to a fixed reference frame.

Note that (1.6)–(1.9) has geometrical meaning similar to that of (1.5). Whereas in the symmetric control scheme the relative variables give the relative pose between the tips of the virtual sticks, in the cooperative task-space the relative variables give the relative pose between the end-effectors. On the other hand, in the symmetric control scheme the absolute variables represent half of the displacement (and rotation) between the tips of the virtual sticks, whereas in the cooperative task-space the absolute variables represent half of the displacement (and rotation) between the end-effectors.

In order to apply an inverse kinematics algorithm, Chiacchio et al. (1996) obtained the relation between the absolute and relative velocities and their counterparts in the end-effectors:

$$\xi_c = \mathbf{U}^T \xi_{\text{eff}}, \quad (1.10)$$

where $\xi_c = \begin{bmatrix} \xi_a^T & \xi_r^T \end{bmatrix}^T$, $\xi_{\text{eff}} = \begin{bmatrix} \xi_1^T & \xi_2^T \end{bmatrix}^T$. Furthermore, because the authors used rotation matrices to represent orientations, they did not make any assumption of small variations in the relative rotation. Hence, (1.5) and (1.6)–(1.9) are analogous, but the latter results in an exact description, whereas the former is an approximation that takes into consideration small relative rotations. Because both describe the same phenomenon, it is clear that the representation used for the rotation (and for the rigid motion in general) plays an important role in the overall description of the cooperative task.

Chiacchio et al. (1996) showed that the coordination between the arms can be achieved with the following control law:

$$\dot{\theta} = \mathbf{J}^{-1} (\xi_{\text{od}} + \mathbf{K}\mathbf{e}), \quad (1.11)$$

where $\theta = \begin{bmatrix} \theta_1^T & \theta_2^T \end{bmatrix}^T$ is the vector containing the joint variables of the two arms, ξ_{od} is the desired twist, \mathbf{K} is a positive definite gain matrix, $\mathbf{e} = \begin{bmatrix} \mathbf{e}_a^T & \mathbf{e}_r^T \end{bmatrix}^T$ is the error encompassing both absolute and relative errors and

$$\mathbf{J} = \begin{bmatrix} \frac{1}{2}\mathbf{J}_1 & \frac{1}{2}\mathbf{J}_2 \\ -\mathbf{J}_1 & \mathbf{J}_2 \end{bmatrix},$$

with \mathbf{J}_1 and \mathbf{J}_2 being the geometric Jacobian of the arms. Although the position error can be given by a simple subtraction between the desired and actual values, Chiacchio et al. (1996) used the rotation error proposed by Luh et al. (1980), which is valid only for small errors. Consequently, the controller (1.11) relies on a trajectory generator to ensure stability. However, (1.6)–(1.9) do not impose any particular control strategy, and thus (1.11) can be changed without changing the overall description of the cooperative task.

Chiacchio et al. (1996) also had a keen insight regarding loose grasps. They found that non-tight grasps, like rolling and sliding ones, can be represented by virtual joints describing the kinematics of the contact. In this manner, the definition of the cooperative task-space variables remains the same, and the virtual joints corresponding to the contact are added to the kinematic chain of the manipulators performing the non-tight grasp.

Caccavale et al. (2000) realized the role of geometric representation in the overall description of the collaborative task and thus used quaternions to represent rotations in the cooperative task-space. Using quaternion algebra, they analyzed the equilibrium of the closed-loop system. They also performed kinetostatic filtering in order to minimize the internal forces for a firmly grasped object. Remarkably, using kinetostatic filtering, the system can limit the internal forces in steady state without force feedback.

Continuing the work on force control in the cooperative task-space, Caccavale et al. (2008) proposed an impedance control in the cooperative task-space for a particular case where the end-effectors have the same orientation. Using the quaternion to represent orientations, they proposed a geometrically consistent stiffness. Finally, using impedance control, they controlled both external and internal forces in a two-arm system firmly grasping an object while the object followed a specified trajectory.

Notably enough, Caccavale et al. (2000) and Caccavale et al. (2008) used quaternions to represent the orientations in certain convenient cases, but they frequently resorted to rotation matrices. Although mathematically consistent, the representation that they proposed lacks uniformity. This is mainly due to the fact that orientation and positions are represented using different mathematical objects.

1.1.4 Synchronized control

Sun & Mills (2002) presented a synchronization strategy for cooperative manipulators based on the general synchronization function

$$\mathbf{f}(\boldsymbol{\beta}_1(t), \boldsymbol{\beta}_2(t), \dots, \boldsymbol{\beta}_n(t)) = 0, \quad (1.12)$$

where $\boldsymbol{\beta}_i$ can be the joint position vector or the Cartesian position. Using a Taylor series expansion at the desired coordinates $\boldsymbol{\beta}_{id}$ and ignoring higher order terms, the authors found an equivalent expression for (1.12):

$$\sum_{i=1}^n \left. \frac{\partial \mathbf{f}(\boldsymbol{\beta}_1(t), \dots, \boldsymbol{\beta}_n(t))}{\partial \boldsymbol{\beta}_i} \right|_{\boldsymbol{\beta}_{id}} \mathbf{e}_i(t) = 0,$$

where $\mathbf{e}_i(t) = \boldsymbol{\beta}_{id}(t) - \boldsymbol{\beta}_i(t)$ is the position error. Defining the synchronization error as

$$\mathbf{e}_{\text{sync}} = \sum_{i=1}^n \left. \frac{\partial \mathbf{f}(\boldsymbol{\beta}_1(t), \dots, \boldsymbol{\beta}_n(t))}{\partial \boldsymbol{\beta}_i} \right|_{\boldsymbol{\beta}_{id}} \mathbf{e}_i(t)$$

and the control objective as

$$\mathbf{e}_{\text{sync}} \rightarrow \mathbf{0}, \quad (1.13)$$

they introduced an adaptive control strategy that is capable of reducing the errors in the coordination. The advantages of using a control law based on the control objective (1.13) is that an arbitrary synchronization function can be used.

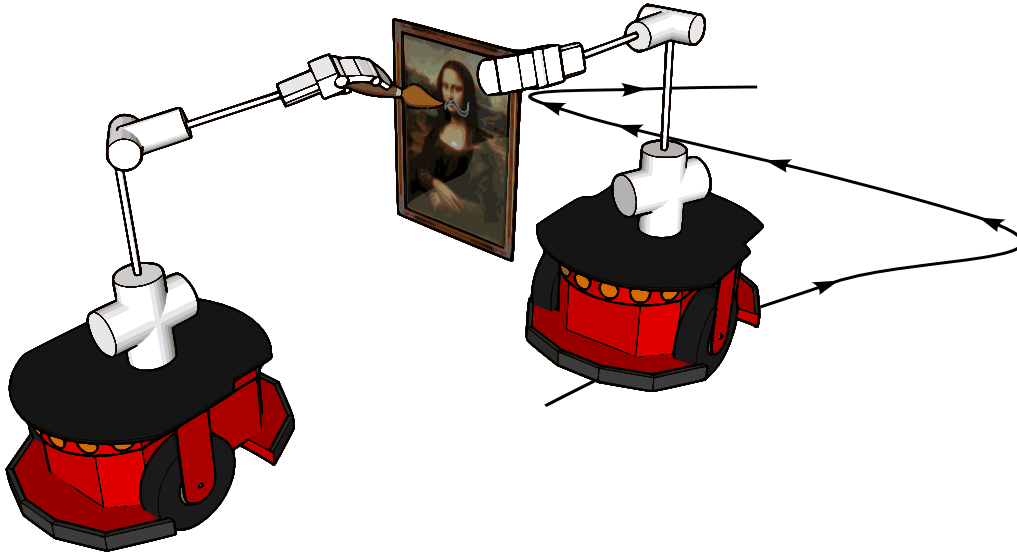


Figure 8: Synchronized interaction between two mobile manipulators. The robots have different goals: one robot must paint a picture, while the second must follow a trajectory holding a frame. The synchronization is performed by choosing an appropriate synchronization function.

All the trajectories must be generated for each manipulator, which makes sense if the robots must coordinately perform independent tasks. However, if the tasks are coupled, generating individual trajectories for each robot can be an unnecessary burden. Moreover, the control law presented by Sun & Mills (2002) does not take into account orientations of the end-effectors, which reduces the range of applicability of their method.

Nevertheless, the method is promising and a good choice of the synchronizing function could potentially lead to complex behaviors, as in the example shown in figure 8. In this scenario, the robots have different goals: one robot must paint a picture, while the second must follow a trajectory holding a frame. The tasks can be described separately, and the whole system can work together given an appropriate synchronization function that couples the two tasks together, resulting in cooperative behavior.

Rodriguez-Angeles & Nijmeijer (2004) also presented a method for synchronized control between manipulators based on a feedback control law in joint space and a set of nonlinear observers that estimate the joint velocities and accelerations. However, their method is based on the condition that all the robots must track a common desired trajectory. This restricts the applicability of the method to more general situations where the robots follow different trajectories (e.g., opening a bottle using two hands).

1.1.5 Other representations

Connolly & Pfeiffer (1994) used normalized dual quaternion interpolation to generate a path between two points respecting the kinematics constraints of the resulting closed chain mechanism. Moreover, they minimized the internal forces by means of an external hybrid position/force control scheme using an approximation for the differential of dual quaternions. However, they did not present any proof of stability for the control strategy. Also, the method does not handle the control of external forces.

Dooley & McCarthy (1993) introduced the concept of operational image space for cooperative manipulators. The operational space formulation for multiple manipulators—the augmented object model (Khatib, 1988)—was represented in a subspace of the dual quaternion space. This representation was used to perform geometric analysis of the trajectories of cooperating robots. However, the task was defined directly in the joint space, which can be quite counterintuitive.

Tinos & Terra (2002) presented an interesting strategy to control cooperative manipulators with passive joints. Assuming a rigid grasped object and then considering the kinematic constraints of the system, they partitioned the joint variables of the multi-arm system into sets of actuated and passive joints. Then, using these sets, they found the Jacobian relating the velocity of the object and the velocity of the actuated joints. Finally, using the robot dynamic model, they proposed a controller to perform both motion control and the control of internal forces taking into consideration the passive joints. However, they did not address the control of external forces.

Departing from control theory approaches, Gharbi et al. (2008) developed a motion planning algorithm based on a multi-layer Probabilistic Roadmap (PRM) to perform two-arm manipulations. The method takes into account the arm singularities, such that it automatically calculates arm reconfigurations in order to perform complex motions while avoiding obstacles. However, the method needs an accurate model of the environment to perform collision detection and it is not suitable for real-time or reactive motions. On the other hand, this kind of technique is quite useful for precomputing complex feasible motions in the workspace, but generating the trajectories in joint space. Hence, the method can be complementary to low level joint control.

1.2 CONCLUSION

This chapter started with the presentation of robots capable of interacting with humans and capable of performing two-arm manipulations. It then presented the state-of-the-art techniques suitable for performing two-arm manipulation. Remarkably, most part of these widespread techniques are now about 20 years old and they still constitute the state of the art. They aim to tackle both internal and external wrenches acting on a firmly grasped object. Whereas external wrenches can induce resultant motion in the manipulated object, internal wrenches cause internal stresses which in most cases should be minimized—although they can be exploited to deform the manipulated object.

More related to a dynamic formulation, the augmented object model—in conjunction with virtual linkage—is capable of representing and controlling the internal and external wrenches acting on an object manipulated by a multi-arm system. Since the augmented object model is an extension of Khatib's operational space formulation, control laws designed for the latter can be analogously applied to the former. However, these control laws usually require torque-actuated robots, which are not always available.

Another established technique, the symmetric formulation, can also represent the internal and external wrenches acting on a firmly grasped object. Using the concept of “virtual stick” and the principle of virtual work, it is possible to find the relation between twists in the manipulated object and twists at the tips of the virtual sticks. In

addition, this formalism provides the relation between the position/orientation at the tips of the virtual sticks and the absolute and relative positions/orientations. However, since Euler angles are used to represent orientations, only small relative rotations are allowed.

Chiacchio et al. (1996) realized that the relative and absolute positions and orientations can be directly defined without considering a manipulated object. They thus proposed the cooperative task-space, which is equivalent to the symmetric formulation if the virtual sticks are considered as being part of the kinematic chain of each manipulator. Furthermore, Chiacchio et al. used rotation matrices to represent the orientation; thus, the resultant formalism does not impose any kind of restriction regarding relative rotations.

Although traditionally treated as different frameworks, the symmetric formulation and the cooperative task-space can be considered as different points of view of the same formalism. Indeed, if a rigid object is firmly grasped, both formulations are equivalent, since the relation between the end-effectors and the tips of virtual sticks is well defined. Thus, wrenches specified at the tip of virtual sticks have a straightforward relation with wrenches applied at the end-effectors. On the other hand, if there is no object and the robot is performing a coordinated movement, both formulations will also be the equivalent if the virtual sticks reduce to a point. Due to the aforementioned equivalences between the symmetric formulation and the cooperative task-space, it makes sense to treat them as a single formalism, which in this thesis will be referred as the symmetric cooperative task-space.

The last and most recent mainstream technique presented is synchronized control. This technique is particularly attractive in situations where two or more robots perform unrelated tasks but need to be coordinated. In this situation, the problem is to define a suitable synchronization function. In terms of motion coordination between robotic arms, the symmetric cooperative task-space can be regarded as a particular case of the synchronized control. Indeed, the synchronization function can be chosen as a trajectory defined in terms of relative position/orientation.

In this thesis, the symmetric cooperative task-space is revisited in the light of a unified representation: dual quaternions. From this new point of view, wrenches, twists and positions/orientations—referred to as dual positions henceforth—are all represented by dual quaternions. Although equivalent to the preceding formulations of the symmetric cooperative task-space, the representation proposed in this thesis is more compact and can be seen as a unified treatment. Furthermore, because the representation is unified, both generalizations to mobile manipulators and human-robot collaboration are straightforward, as shown in chapters 4 and 5. Last, since the formalism imposes no restriction with respect to the low-level control laws and low-level actuation (i.e., at the joint level), unusual cases of two-arm coordination can be tackled as well. One of these cases, presented in chapter 5, is when the robot collaborates with a person but, in addition to its own arm, the robot also controls the human arm by means of FES.

Last, the next chapter provides the mathematical background—including some novel concepts—needed to understand the ideas presented throughout the thesis. One should read it carefully to become acquainted with the notation, in addition to the theoretical contributions that are presented. It also presents the kinematic modeling of serial robots

by using dual quaternions, which is a small contribution but a good exercise to establish knowledge of the concepts introduced therein.

KINEMATIC MODELING OF MANIPULATOR ROBOTS USING DUAL QUATERNIONS

Their [dual quaternions] operations seem more susceptible to meaningful geometrical interpretation than those of matrix algebra. Dual quaternions possess a high degree of flexibility, that is, one needs only to expand those terms that are necessary for a particular purpose; for the rest, one can retain them in compact form.

— An Tzu Yang (Yang, 1963)

In robotics textbooks, one of the first chapters is typically dedicated to the presentation of the theory of rigid body motion. In general, the representations of translation and rotation are introduced separately, and then they are grouped together to form the homogeneous transformation.

Typically, rotation matrices are used to represent rotation, whereas translation is represented by the Cartesian position. Grouping them together leads to homogeneous transformation matrices. Even in cases where different parametrizations of rigid motions are shown for the case of completeness, the homogeneous transformation matrices are used throughout the text. Examples of this kind of exposition are found in the textbooks of Paul (1981), Spong et al. (2006), Dombre & Khalil (2007) and Siciliano et al. (2009).

An alternative is presented by Murray et al. (1994). In their textbook, a complete presentation of robot modeling and control is made in the light of screw theory. However, this kind of presentation seems to be much more an exception than the rule.

Although homogeneous transformation matrices are quite common to represent kinematic motion, they impose some additional work to control the end-effector. More specifically, a very common technique is kinematic control in Cartesian space. Such control techniques take into consideration the relationship between the operational space variables and the joint variables. This relationship is typically given by

$$\dot{\mathbf{x}} = \mathbf{J}\dot{\boldsymbol{\theta}}, \quad (2.1)$$

where \mathbf{x} is the vector of the operational space variables and $\boldsymbol{\theta}$ is the vector of the joint variables.

The parametrization of the operational space variables is usually given by a vector and not directly by the homogeneous transformation matrix. Hence, it is necessary to choose the parametrization and also to extract these variables from the homogeneous transformation matrix that represents the end-effector pose.

Although one can consider these problems as secondary issues, the better theory is usually the one that explains more using fewer—and preferably simpler—arguments. Hence, a theory capable of eliminating intermediate steps between modeling and control would be better according to the aforementioned criterion. Moreover, from an engineering point of view, fewer intermediate steps between modeling and control can potentially reduce errors in implementation and development, leading to safer robots.

Murray et al. (1994) present a more mathematical approach to robot modeling and control based on the screw theory and matrix exponentials, whereas McCarthy (1990) presents an approach based on dual quaternions. Despite the fact that the latter is not a robotics textbook, McCarthy and his collaborators consistently developed a theory for robot modeling using dual quaternion theory (Dooley & McCarthy, 1993; Perez & McCarthy, 2004). Also, as they are more focused on mechanism analysis, they have not given a complete account of robot control.

This thesis presents another point of view on kinematic modeling and control based on dual quaternions. The choice of dual quaternions instead of homogeneous transformation matrices was made because dual quaternions provide a unified representation for robot modeling and control. Furthermore, they are more compact and, in my opinion, more intuitive to use than matrix exponentials.

Other related work using dual quaternions to model serial chain manipulators was published by Kim & Kumar (1990). They use dual quaternions to represent line transformations and apply the resultant theory to obtain the kinematic model. Their approach differs from the one presented herein, since this thesis uses point transformations. Admittedly, line transformations offer some advantages for representing screw motion and performing velocity analysis (Kim & Kumar, 1990). However, at present, point transformations seem to be more widespread in robot modeling, planning and control, making this a convenient choice for this thesis.

In the light of the previous discussion, the following section presents the general concepts regarding quaternions and dual quaternions. These concepts—which are later used to represent rigid motions—form the basis of the mathematical description used in this manuscript. Following the introduction of the mathematical concepts, the chapter presents one of its main contributions, decompositional multiplication. This operation provides a new class of decomposed motions that are invariant with respect to the configuration of the modified frame. Also, the chapter introduces an analytic formulation to find both the forward kinematic model and the Jacobian of serial robots using dual quaternions, provided that the D-H representation is used.

2.1 MATHEMATICAL BACKGROUND

This section presents the mathematical background and notation used throughout the thesis. More specifically, it presents the main concepts and operations of quaternions, dual numbers and dual quaternions. Although part of the notation is drawn from the current literature, some elements are particular to this thesis. Hence, readers are encouraged to read the whole section, even if they are already acquainted with the notions presented herein.

2.1.1 Quaternions

The quaternions were introduced by Hamilton in the nineteenth century and can be regarded as an extension of complex numbers, where the three imaginary components $\hat{i}, \hat{j}, \hat{k}$ are defined and have the following properties (Hamilton, 1844):

$$\begin{aligned} -\hat{j}\hat{i} &= \hat{i}\hat{j} = \hat{k}, & -\hat{k}\hat{j} &= \hat{j}\hat{k} = \hat{i}, & -\hat{i}\hat{k} &= \hat{k}\hat{i} = \hat{j}, \\ \hat{i}^2 &= \hat{j}^2 = \hat{k}^2 = -1. \end{aligned}$$

The quaternion \mathbf{h} was defined by Hamilton as

$$\mathbf{h} = h_1 + \hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4, \quad h_i \in \mathbb{R}, i = 1, \dots, 4.$$

Using an analogy with standard complex numbers, quaternions can be divided in real and imaginary parts. The real part of \mathbf{h} , denoted by $\text{Re}(\mathbf{h})$, is the scalar h_1 . The imaginary part, denoted by $\text{Im}(\mathbf{h})$ is the column vector with the imaginary coefficients; that is, $\text{Im}(\mathbf{h}) = [h_2 \ h_3 \ h_4]^T$. Let the imaginary vector unit be defined as $\mathbf{v}_m = [\hat{i} \ \hat{j} \ \hat{k}]^T$. Thus,

$$\mathbf{h} = \text{Re}(\mathbf{h}) + \mathbf{v}_m \cdot \text{Im}(\mathbf{h}),$$

where the \cdot is the dot product between two vectors. Observe that the previous equation is analogous to a classic complex number.

The addition/subtraction and multiplication of quaternions use the properties of the imaginary components and are given by the next two definitions.

Definition 2.1. Let $\mathbf{h} = h_1 + \hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4$ and $\mathbf{h}' = h'_1 + \hat{i}h'_2 + \hat{j}h'_3 + \hat{k}h'_4$ be two quaternions. The quaternion sum/subtraction is

$$\begin{aligned} \mathbf{h} \pm \mathbf{h}' &= h_1 \pm h'_1 + \hat{i}(h_2 \pm h'_2) + \hat{j}(h_3 \pm h'_3) + \hat{k}(h_4 \pm h'_4) \\ &= \text{Re}(\mathbf{h}) \pm \text{Re}(\mathbf{h}') + \mathbf{v}_m \cdot (\text{Im}(\mathbf{h}) \pm \text{Im}(\mathbf{h}')) \end{aligned}$$

Definition 2.2. Let $\mathbf{h} = h_1 + \hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4$ and $\mathbf{h}' = h'_1 + \hat{i}h'_2 + \hat{j}h'_3 + \hat{k}h'_4$ be two quaternions. The quaternion multiplication is

$$\begin{aligned} \mathbf{h}\mathbf{h}' &= (h_1 + \hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4)(h'_1 + \hat{i}h'_2 + \hat{j}h'_3 + \hat{k}h'_4) \\ &= (h_1h'_1 - h_2h'_2 - h_3h'_3 - h_4h'_4) + \\ &\quad \hat{i}(h_1h'_2 + h_2h'_1 + h_3h'_4 - h_4h'_3) + \\ &\quad \hat{j}(h_1h'_3 - h_2h'_4 + h_3h'_1 + h_4h'_2) + \\ &\quad \hat{k}(h_1h'_4 + h_2h'_3 - h_3h'_2 + h_4h'_1). \end{aligned} \tag{2.2}$$

The set of quaternions \mathbb{H} forms a group under quaternion multiplication (Murray et al., 1994). It is easy to show that quaternions are associative and distributive, but non-commutative. The next two definitions refer to the conjugate and norm of quaternions.

Definition 2.3. The conjugate of a quaternion $\mathbf{h} = \text{Re}(\mathbf{h}) + \mathbf{v}_m \cdot \text{Im}(\mathbf{h})$ is

$$\mathbf{h}^* = \text{Re}(\mathbf{h}) - \mathbf{v}_m \cdot \text{Im}(\mathbf{h}).$$

Definition 2.4. The norm of a quaternion \mathbf{h} is

$$\begin{aligned}\|\mathbf{h}\| &= \sqrt{\operatorname{Re}(\mathbf{h})^2 + \operatorname{Im}(\mathbf{h}) \cdot \operatorname{Im}(\mathbf{h})} \\ &= \sqrt{\mathbf{h}\mathbf{h}^*} = \sqrt{\mathbf{h}^*\mathbf{h}},\end{aligned}$$

where the second line can be verified by direct calculation.

Sometimes it is useful to perform the multiplication between matrices and quaternions. Some authors prefer to use an implicit notation for this operation; that is, they implicitly consider the parametrization of the quaternion into a vector space before doing the multiplication. However, this type of notation can lead to confusion, mainly if several complex operations are performed mixing quaternions and matrices. In the sequel, the vec operator is introduced, followed by the definition of the multiplication between four-by-four matrices and quaternions.

Definition 2.5. Given a quaternion $\mathbf{h} = \operatorname{Re}(\mathbf{h}) + \mathbf{i}_m \cdot \operatorname{Im}(\mathbf{h})$, the vec operator performs the one-by-one mapping $\operatorname{vec} : \mathbb{H} \rightarrow \mathbb{R}^4$; that is,

$$\operatorname{vec} \mathbf{h} \triangleq \begin{bmatrix} \operatorname{Re}(\mathbf{h}) \\ \operatorname{Im}(\mathbf{h}) \end{bmatrix}.$$

The inverse operation is given by $\operatorname{vec}^{-1} : \mathbb{R}^4 \rightarrow \mathbb{H}$; that is, let $\mathbf{u} = [u_1 \ u_2 \ u_3 \ u_4]^T$,

$$\begin{aligned}\mathbf{h} &= \operatorname{vec}^{-1} \mathbf{u}, \\ \operatorname{Re}(\mathbf{h}) &= u_1, \\ \operatorname{Im}(\mathbf{h}) &= [u_2 \ u_3 \ u_4]^T.\end{aligned}$$

The vec operator just takes each coefficient of the quaternion and stacks them in a vector. The inverse operation just takes a four-dimensional vector and maps its elements to the coefficients of a quaternion.

Example 2.1. Let $\mathbf{h} = a + \hat{\mathbf{i}}b + \hat{\mathbf{j}}c + \hat{\mathbf{k}}d$, then $\operatorname{vec} \mathbf{h} = [a \ b \ c \ d]^T$.

Example 2.2. Let $\mathbf{v} = [0 \ a \ 0 \ -b]^T$, then $\operatorname{vec}^{-1} \mathbf{v} = \hat{\mathbf{i}}a - \hat{\mathbf{k}}b$.

The previous definition leads to an important result in terms of quaternion multiplication, shown next.

Definition 2.6. Using (2.2), it is easy to verify by direct calculation that, for $\mathbf{h}, \mathbf{h}' \in \mathbb{H}$,

$$\operatorname{vec}(\mathbf{h}\mathbf{h}') = \overset{+}{\mathbf{H}}(\mathbf{h}) \operatorname{vec} \mathbf{h}' = \bar{\mathbf{H}}(\mathbf{h}') \operatorname{vec} \mathbf{h},$$

where

$$\overset{+}{\mathbf{H}}(\mathbf{h}) = \begin{bmatrix} h_1 & -h_2 & -h_3 & -h_4 \\ h_2 & h_1 & -h_4 & h_3 \\ h_3 & h_4 & h_1 & -h_2 \\ h_4 & -h_3 & h_2 & h_1 \end{bmatrix}, \quad \bar{\mathbf{H}}(\mathbf{h}') = \begin{bmatrix} h'_1 & -h'_2 & -h'_3 & -h'_4 \\ h'_2 & h'_1 & h'_4 & -h'_3 \\ h'_3 & -h'_4 & h'_1 & h'_2 \\ h'_4 & h'_3 & -h'_2 & h'_1 \end{bmatrix},$$

and $\overset{+}{\mathbf{H}}$ and $\bar{\mathbf{H}}$ are called Hamilton operators¹.

1. The term Hamilton operator is not commonly used, at least in the robotics literature. But since it seemed appropriate, I borrowed the term from Akyar (2008).

This latter definition states that, even though the quaternion multiplication is not commutative, the Hamilton operators commute between them. As it will become evident in the next chapters, this property is quite useful. For a more complete account of the properties of Hamilton operators, see (Chou, 1992; Akyar, 2008).

2.1.2 Dual numbers

Dual numbers were introduced by Clifford (1873), who proposed the dual unit ε to create a new algebra. In this algebra, ε is nilpotent and has the following properties:

$$\begin{aligned}\varepsilon &\neq 0, \\ \varepsilon^2 &= 0.\end{aligned}$$

Definition 2.7. For a dual number $\underline{a} = a + \varepsilon a'$, the number a is the primary part whereas a' is the dual part². Also, the primary and dual parts can be extracted by using the operators $\mathcal{P}(\underline{a})$ and $\mathcal{D}(\underline{a})$, respectively. Hence,

$$\underline{a} = \mathcal{P}(\underline{a}) + \varepsilon \mathcal{D}(\underline{a}).$$

The usual operations—sum/subtraction, multiplication—take into account the ε operator and are defined bellow.

Definition 2.8. Let \underline{a}_1 and \underline{a}_2 be dual numbers. The sum/subtraction between them is

$$\underline{a}_1 \pm \underline{a}_2 = \mathcal{P}(\underline{a}_1) \pm \mathcal{P}(\underline{a}_2) + \varepsilon(\mathcal{D}(\underline{a}_1) \pm \mathcal{D}(\underline{a}_2)).$$

Definition 2.9. Let \underline{a}_1 and \underline{a}_2 be dual numbers. The multiplication between them is

$$\begin{aligned}\underline{a}_1 \underline{a}_2 &= (\mathcal{P}(\underline{a}_1) + \varepsilon \mathcal{D}(\underline{a}_1))(\mathcal{P}(\underline{a}_2) + \varepsilon \mathcal{D}(\underline{a}_2)) \\ &= \mathcal{P}(\underline{a}_1) \mathcal{P}(\underline{a}_2) + \varepsilon(\mathcal{P}(\underline{a}_1) \mathcal{D}(\underline{a}_2) + \mathcal{D}(\underline{a}_1) \mathcal{P}(\underline{a}_2)).\end{aligned}$$

Note that the nilpotent property of the ε operator is used in the multiplication operation.

Fact 2.1. The inverse of a dual number \underline{a} is

$$\underline{a}^{-1} = \frac{1}{\mathcal{P}(\underline{a})} - \varepsilon \frac{\mathcal{D}(\underline{a})}{\mathcal{P}(\underline{a})^2}, \quad \mathcal{P}(\underline{a}) \neq 0.$$

Proof. See Appendix 2 on page 117. □

Note that the inverse can be undefined even when $\underline{a} \neq 0$ (i.e., for all cases where only the dual part is different from zero).

2. Typically, the primary and dual parts are composed of the same type of elements; namely: scalars, vectors, matrices, and quaternions.

2.1.3 Dual quaternions

Dual quaternions, also introduced by Clifford (1873), are dual numbers in which the primary and dual parts are quaternions; namely,

$$\underline{\mathbf{h}} = \mathcal{P}(\underline{\mathbf{h}}) + \varepsilon \mathcal{D}(\underline{\mathbf{h}}),$$

with $\mathcal{P}(\underline{\mathbf{h}}), \mathcal{D}(\underline{\mathbf{h}}) \in \mathbb{H}$. The set of dual quaternions is denoted by \mathcal{H} .

The real and imaginary parts of $\underline{\mathbf{h}} \in \mathcal{H}$ are a dual scalar and a dual vector, respectively:

$$\begin{aligned} \operatorname{Re}(\underline{\mathbf{h}}) &\triangleq \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) + \varepsilon \operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})), \\ \operatorname{Im}(\underline{\mathbf{h}}) &\triangleq \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}})) + \varepsilon \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}})). \end{aligned}$$

Thus, it is easy to verify (see Appendix 6 on page 121) that

$$\underline{\mathbf{h}} = \operatorname{Re}(\underline{\mathbf{h}}) + \mathbf{i}_m \cdot \operatorname{Im}(\underline{\mathbf{h}}).$$

Definition 2.10. The multiplication of dual quaternions follows the same rules as for dual numbers, but respecting the quaternion operations. For instance, let $\underline{\mathbf{h}}, \underline{\mathbf{h}}' \in \mathcal{H}$,

$$\begin{aligned} \underline{\mathbf{h}}\underline{\mathbf{h}}' &= (\mathcal{P}(\underline{\mathbf{h}}) + \varepsilon \mathcal{D}(\underline{\mathbf{h}}))(\mathcal{P}(\underline{\mathbf{h}}') + \varepsilon \mathcal{D}(\underline{\mathbf{h}}')) \\ &= \mathcal{P}(\underline{\mathbf{h}})\mathcal{P}(\underline{\mathbf{h}}') + \varepsilon(\mathcal{P}(\underline{\mathbf{h}})\mathcal{D}(\underline{\mathbf{h}}') + \mathcal{D}(\underline{\mathbf{h}})\mathcal{P}(\underline{\mathbf{h}}')). \end{aligned} \quad (2.3)$$

Definition 2.11. The conjugate of the dual quaternion $\underline{\mathbf{h}}$ is

$$\underline{\mathbf{h}}^* \triangleq \mathcal{P}(\underline{\mathbf{h}})^* + \varepsilon \mathcal{D}(\underline{\mathbf{h}})^*.$$

The vec operator and its inverse can be extended to dual quaternions, analogously to quaternions, as shown below.

Definition 2.12. Given a dual quaternion $\underline{\mathbf{h}}$, the vec operator performs the one-by-one mapping $\operatorname{vec} : \mathcal{H} \rightarrow \mathbb{R}^8$; that is,

$$\operatorname{vec} \underline{\mathbf{h}} = \begin{bmatrix} \operatorname{vec}(\mathcal{P}(\underline{\mathbf{h}})) \\ \operatorname{vec}(\mathcal{D}(\underline{\mathbf{h}})) \end{bmatrix}.$$

The inverse operation is given by $\operatorname{vec}^{-1} : \mathbb{R}^8 \rightarrow \mathcal{H}$; that is, let $\mathbf{u} = [u_1 \ \dots \ u_8]^T$,

$$\begin{aligned} \underline{\mathbf{h}} &= \operatorname{vec}^{-1} \mathbf{u}, \\ \operatorname{Re}(\underline{\mathbf{h}}) &= u_1 + \varepsilon u_5, \\ \operatorname{Im}(\underline{\mathbf{h}}) &= \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix} + \varepsilon \begin{bmatrix} u_6 \\ u_7 \\ u_8 \end{bmatrix}. \end{aligned}$$

Example 2.3. Let $\mathbf{u} = [0 \ 2 \ 0 \ 0 \ 1 \ 2 \ 3 \ 4]^T$, then $\operatorname{vec}^{-1} \mathbf{u} = \hat{\mathbf{i}}2 + \varepsilon(1 + \hat{\mathbf{i}}2 + \hat{\mathbf{j}}3 + \hat{\mathbf{k}}4)$.

Definition 2.13. Comparing (2.3) with definition 2.12, it is easy to verify by direct calculation that, for $\underline{\mathbf{h}}, \underline{\mathbf{h}}' \in \mathcal{H}$,

$$\text{vec}(\underline{\mathbf{h}}\underline{\mathbf{h}}') = \overset{+}{\mathbf{H}}(\underline{\mathbf{h}}) \text{vec} \underline{\mathbf{h}}' = \bar{\mathbf{H}}(\underline{\mathbf{h}}') \text{vec} \underline{\mathbf{h}},$$

where

$$\overset{+}{\mathbf{H}}(\underline{\mathbf{h}}) = \begin{bmatrix} \overset{+}{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{h}})) & \mathbf{0}_4 \\ \overset{+}{\mathbf{H}}(\mathcal{D}(\underline{\mathbf{h}})) & \overset{+}{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{h}})) \end{bmatrix}, \quad \bar{\mathbf{H}}(\underline{\mathbf{h}}') = \begin{bmatrix} \bar{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{h}}')) & \mathbf{0}_4 \\ \bar{\mathbf{H}}(\mathcal{D}(\underline{\mathbf{h}}')) & \bar{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{h}}')) \end{bmatrix},$$

and $\overset{+}{\mathbf{H}}$ and $\bar{\mathbf{H}}$ are the Hamilton operators extended to dual quaternions.

Definition 2.14. The norm of the dual quaternion $\underline{\mathbf{h}}$ is given by

$$\begin{aligned} \|\underline{\mathbf{h}}\| &= \underline{\mathbf{h}}\mathbf{h}^* = \mathbf{h}^*\underline{\mathbf{h}} \\ &= \|\mathcal{P}(\underline{\mathbf{h}})\|^2 + 2\varepsilon \text{vec} \mathcal{P}(\underline{\mathbf{h}}) \cdot \text{vec} \mathcal{D}(\underline{\mathbf{h}}), \end{aligned}$$

where the proof for the equation in the second line is shown in Appendix 7 on page 121.

Unit dual quaternions (dual quaternions possessing unit norm) play a very important role in the representation of rigid motions. The next three statements are related to unit dual quaternions.

Proposition 2.1. Let $\underline{\mathbf{h}} = \mathbf{r} + \varepsilon \frac{1}{2} \mathbf{p}\mathbf{r}$ be a unit dual quaternion with $\mathbf{r} = \cos\left(\frac{\phi}{2}\right) + \sin\left(\frac{\phi}{2}\right) \mathbf{n}$, $\mathbf{n} = \hat{\mathbf{i}}n_x + \hat{\mathbf{j}}n_y + \hat{\mathbf{k}}n_z$ and $\mathbf{p} = \hat{\mathbf{i}}p_x + \hat{\mathbf{j}}p_y + \hat{\mathbf{k}}p_z$. The logarithm of $\underline{\mathbf{h}}$ is

$$\log \underline{\mathbf{h}} = \frac{\phi \mathbf{n}}{2} + \varepsilon \frac{\mathbf{p}}{2}.$$

Proof. See (Han et al., 2008). □

Note that $\log \underline{\mathbf{h}} \in \mathcal{H}$ and $\text{Re}(\underline{\mathbf{h}}) = 0$.

Proposition 2.2. Let $\underline{\mathbf{g}} \in \mathcal{H}$ such that $\text{Re}(\underline{\mathbf{g}}) = 0$. The exponential of $\underline{\mathbf{g}}$ is

$$\exp \underline{\mathbf{g}} = \mathcal{P}(\exp \underline{\mathbf{g}}) + \varepsilon \mathcal{D}(\underline{\mathbf{g}}) \mathcal{P}(\exp \underline{\mathbf{g}}) \tag{2.4}$$

$$\mathcal{P}(\exp \underline{\mathbf{g}}) = \begin{cases} \cos \|\mathcal{P}(\underline{\mathbf{g}})\| + \frac{\sin \|\mathcal{P}(\underline{\mathbf{g}})\|}{\|\mathcal{P}(\underline{\mathbf{g}})\|} \mathcal{P}(\underline{\mathbf{g}}) & \text{if } \|\mathcal{P}(\underline{\mathbf{g}})\| \neq 0 \\ 1 & \text{otherwise.} \end{cases} \tag{2.5}$$

Proof. Eq. (2.4) is a direct consequence of (A.1) and (2.5) is shown by Kim et al. (1996). □

Definition 2.15. Given propositions 2.1 and 2.2, the dual quaternion $\underline{\mathbf{h}}$ raised to the t -th power is

$$\underline{\mathbf{h}}^{\{t\}} \triangleq \exp(t \log \underline{\mathbf{h}}).$$

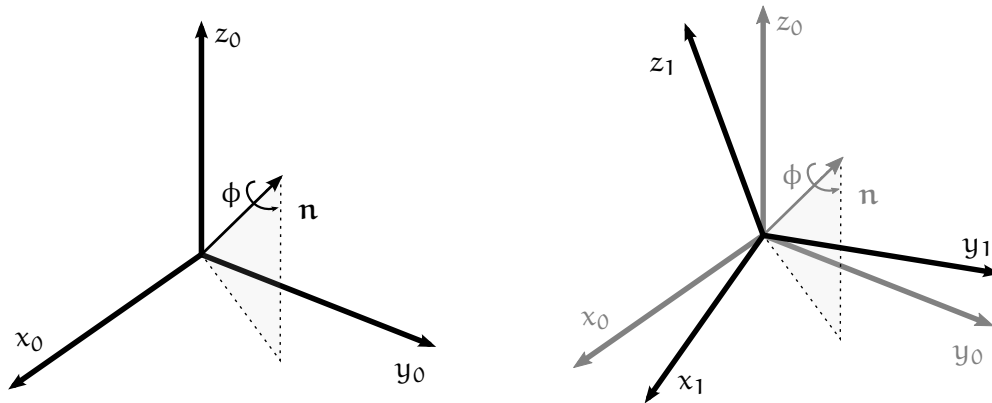


Figure 9: Rotation represented by a quaternion: rotation of ϕ around the unit vector \mathbf{n} .

2.2 RIGID BODY MOTION

This section is devoted to the representation of rigid motions using dual quaternions. In order to give a smooth introduction to the subject, quaternions are first used to represent rotations and they are then used to represent the complete rigid motion. Next, dual quaternions are introduced to provide a more compact representation for the complete rigid motion. Last, one of the main contributions of this chapter closes the section: the decompositional multiplication.

2.2.1 Rotations represented by quaternions

The unit norm quaternions can be used to represent rotations (Kuipers, 1999). For instance, a rotation ϕ around a unit norm vector $\mathbf{n} = [n_x \ n_y \ n_z]^T$ in a tridimensional Euclidean space can be given by $\mathbf{r} \in \mathbb{H}$, such that

$$\begin{aligned} \operatorname{Re}(\mathbf{r}) &= \cos\left(\frac{\phi}{2}\right), & \operatorname{Im}(\mathbf{r}) &= \sin\left(\frac{\phi}{2}\right) \mathbf{n}, \\ \mathbf{r} &= \cos\left(\frac{\phi}{2}\right) + \mathbf{r}_m \cdot \sin\left(\frac{\phi}{2}\right) \mathbf{n}. \end{aligned} \tag{2.6}$$

It is easy to verify that (2.6) is a unit quaternion. The rotation described by (2.6) is illustrated in figure 9.

Let a unit quaternion $\mathbf{r}_0^0 = 1$ represent the initial orientation of a reference frame \mathcal{F}_0 . After n rotations, the final orientation is given by $\mathbf{r}_n^0 = \mathbf{r}_1^0 \dots \mathbf{r}_n^{n-1}$. Note that the superscript and subscript represent the original and final frames, respectively.

2.2.1.1 Frame rotation

Points and translations can be represented by pure quaternions, that is, quaternions with the real part equal to zero. For example, the quaternion $\mathbf{p}_h = p_x \hat{i} + p_y \hat{j} + p_z \hat{k}$ corresponds to the vector $\mathbf{p}_v = [p_x \ p_y \ p_z]^T$. Hereafter, translations and rotations will always be defined as quaternions, unless explicitly stated.

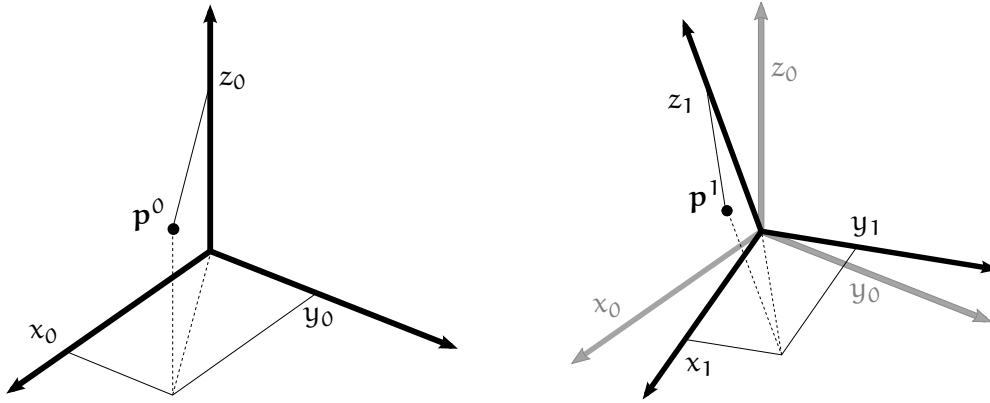


Figure 10: Frame rotation corresponding to the transformation $\mathbf{p}^1 = \mathbf{r}_1^{0*} \mathbf{p}^0 \mathbf{r}_1^0$.

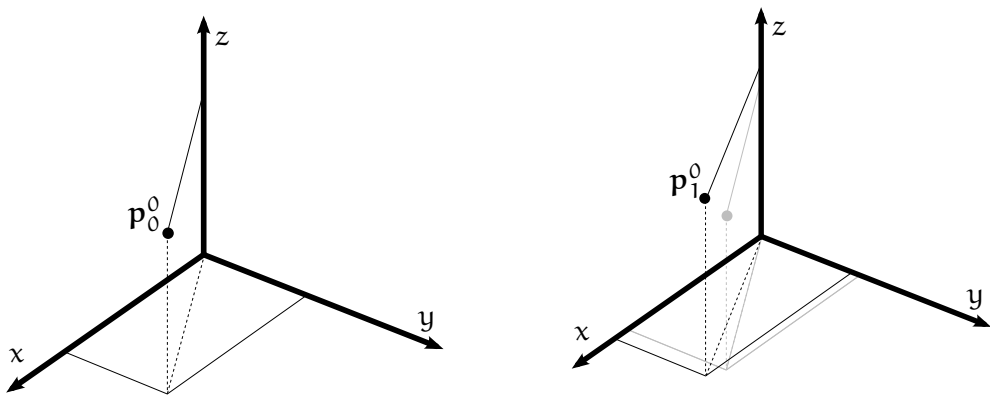


Figure 11: Point rotation corresponding to the transformation $\mathbf{p}_1^0 = \mathbf{r}_1^0 \mathbf{p}_0^0 \mathbf{r}_1^{0*}$.

Let \mathbf{p}^0 be a point with respect to \mathcal{F}_0 . The frame \mathcal{F}_1 is obtained by rotating \mathcal{F}_0 by a quaternion \mathbf{r}_1^0 . The point with respect to this new frame is given by (Kuipers, 1999)

$$\mathbf{p}^1 = \mathbf{r}_1^{0*} \mathbf{p}^0 \mathbf{r}_1^0. \quad (2.7)$$

The transformation (2.7) is illustrated in figure 10 and is called frame rotation.

2.2.1.2 Point rotation

Let \mathbf{p}_0^0 be a point with respect to \mathcal{F}_0 and consider the same quaternion \mathbf{r}_1^0 used in figure 10. The transformation

$$\mathbf{p}_1^0 = \mathbf{r}_1^0 \mathbf{p}_0^0 \mathbf{r}_1^{0*} \quad (2.8)$$

illustrated in figure 11 is called point rotation (Kuipers, 1999). Note that both points are referenced in \mathcal{F}_0 . The difference between frame rotation and point rotation is a matter of viewpoint. In the former, observers located at the point will be stationary while the frame moves. Conversely, in point rotation, observers located at the frame will be stationary while the point moves.

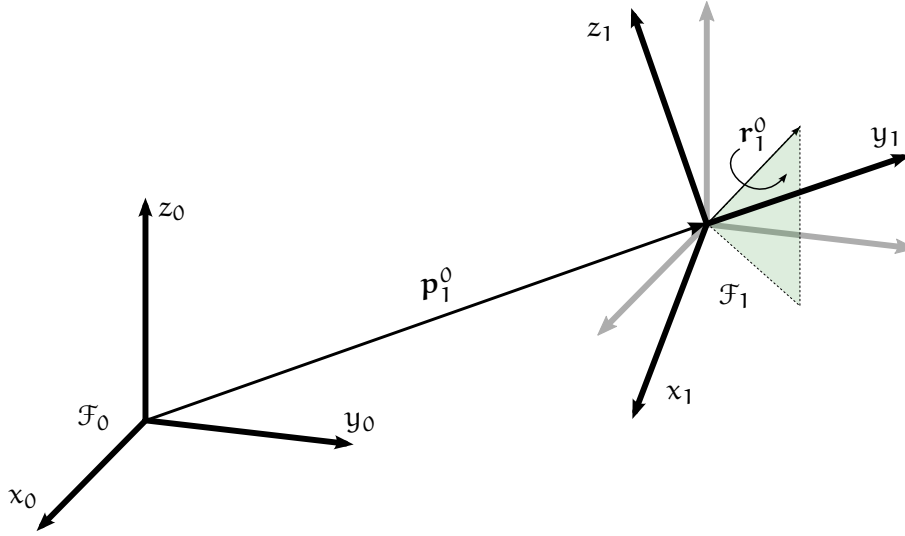


Figure 12: A rigid motion represented by quaternions: first a translation \mathbf{p}_1^0 is performed, followed by a rotation \mathbf{r}_1^0 .

2.2.2 Rigid motions represented by quaternions

The complete rigid motion between \mathcal{F}_0 and \mathcal{F}_1 can be represented by a translation \mathbf{p}_1^0 and a rotation \mathbf{r}_1^0 , as illustrated in figure 12. The sequence of rigid motions using quaternions is given by the ring operations (addition and multiplication) of \mathbb{H} . For instance, let the tuple $(\mathbf{p}_{0,1}^0, \mathbf{r}_1^0)$ represent the rigid motion from ${}^3\mathcal{F}_0$ to \mathcal{F}_1 and $(\mathbf{p}_{1,2}^1, \mathbf{r}_2^1)$ represents the rigid motion from \mathcal{F}_1 to \mathcal{F}_2 (see fig. 13). The resultant rigid motion from \mathcal{F}_0 to \mathcal{F}_2 is given by

$$(\mathbf{p}_{0,2}^0, \mathbf{r}_2^0) = (\mathbf{p}_{0,1}^0 + \mathbf{r}_1^0 \mathbf{p}_{1,2}^1 \mathbf{r}_1^{0*}, \mathbf{r}_1^0 \mathbf{r}_2^1), \quad (2.9)$$

that is, the final rotation is given by the composition between the intermediate rotations, whereas the final position takes into consideration the frame-rotation movement; that is, $\mathbf{p}_{1,2}^1$ is projected in \mathcal{F}_0 by a frame transformation $\mathbf{p}_{1,2}^0 = \mathbf{r}_1^{0*} \mathbf{p}_{1,2}^1 \mathbf{r}_1^0$ (recall that $\mathbf{r}_1^0 = \mathbf{r}_0^{1*}$), and then $\mathbf{p}_{0,1}^0 + \mathbf{p}_{1,2}^0 = \mathbf{p}_{0,2}^0$.

2.2.3 Rigid motions represented by dual quaternions

Dual quaternions have been proven to be a useful representation to describe rigid motions (Yang, 1963; Bottema & Roth, 1979; Dooley & McCarthy, 1991), because they simultaneously describe both positions and orientations (i.e., dual positions) using only eight parameters. They play the same role as the homogeneous transformation matrices, where the complete rigid motion is described by a single mathematical object. Hence, a sequence of rigid motions is represented by a sequence of dual quaternion multiplications.

3. Note that, sometimes, it is convenient in a sequence of rigid motions to represent the original and final frames in the subscript of translation quaternions; for instance, $\mathbf{p}_{0,1}^0$ is a translation from \mathcal{F}_0 to \mathcal{F}_1 , represented in \mathcal{F}_0 . On the other hand, $\mathbf{p}_{0,1}^1$ is the same translation from \mathcal{F}_0 to \mathcal{F}_1 , but represented in \mathcal{F}_1 .

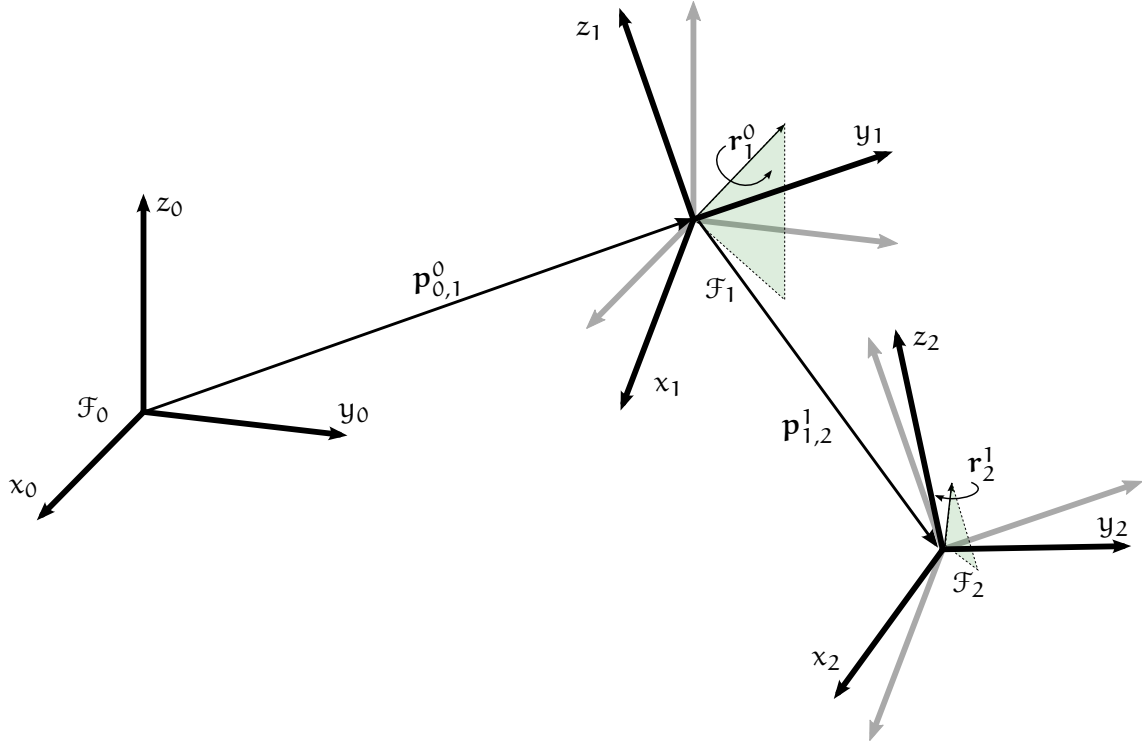


Figure 13: Sequence of rigid motions represented by quaternions.

The dual quaternion

$$\underline{\mathbf{h}}_1^0 = \mathbf{r}_1^0 + \varepsilon \frac{1}{2} \mathbf{p}_1^0 \mathbf{r}_1^0 \quad (2.10)$$

represents the rigid motion illustrated in figure 12. It is a dual number composed of a unit norm quaternion $\mathcal{P}(\underline{\mathbf{h}}_1^0) = \mathbf{r}_1^0$ representing a rotation and an unbounded norm quaternion $\mathcal{D}(\underline{\mathbf{h}}_1^0) = (1/2) \mathbf{p}_1^0 \mathbf{r}_1^0$ indirectly representing a translation. The translation can be easily retrieved using quaternion operations; that is,

$$\mathbf{p}_1^0 = 2 \mathcal{D}(\underline{\mathbf{h}}_1^0) \mathcal{P}(\underline{\mathbf{h}}_1^0)^* . \quad (2.11)$$

Eq. (2.10) is obtained by means of the representation of a Plücker line with respect to two different frames (see section A.4). One can verify that (2.10) follows two restrictions:

$$\mathcal{P}(\underline{\mathbf{h}}_1^0) \mathcal{P}(\underline{\mathbf{h}}_1^0)^* = 1, \quad (2.12)$$

$$\mathcal{P}(\underline{\mathbf{h}}_1^0)^* \mathcal{D}(\underline{\mathbf{h}}_1^0) + \mathcal{D}(\underline{\mathbf{h}}_1^0)^* \mathcal{P}(\underline{\mathbf{h}}_1^0) = 0, \quad (2.13)$$

Eq. (2.12) is due to the fact that the rotation is represented by a unit norm quaternion and (2.13) can be verified by direct calculation.

Because a sequence of rigid motions can be represented by a sequence of dual quaternion multiplications, if the rigid motion from \mathcal{F}_0 to \mathcal{F}_1 is given by $\underline{\mathbf{h}}_1^0$ and the one from \mathcal{F}_1 to \mathcal{F}_2 by $\underline{\mathbf{h}}_2^1$, the resultant motion with respect to \mathcal{F}_0 is

$$\underline{\mathbf{h}}_2^0 = \underline{\mathbf{h}}_1^0 \underline{\mathbf{h}}_2^1. \quad (2.14)$$

By expanding (2.14) and applying (2.11), it is straightforward to see the equivalence between (2.14) and (2.9).

2.2.4 Decompositional multiplication

This subsection presents one of the main contributions of this chapter: the decompositional multiplication. The main application of this new operation is in the description of movements that can modify a target frame, but using another frame as the reference for the motion. The main implication of this new operation is that, for an observer at a fixed reference frame, the resultant motion is invariant with respect to the pose of the target frame. Some examples and illustrations are shown along with the mathematical definitions to provide some intuition into the formal development.

The motivation for the use of this operator is as follows: when representing a sequence of rigid motions by a sequence of dual quaternion multiplications—e.g., $\underline{\mathbf{h}}_n^0 = \underline{\mathbf{h}}_1^0 \underline{\mathbf{h}}_2^1 \dots \underline{\mathbf{h}}_n^{n-1}$ —an intermediate transformation is always given with respect to the previous frame. For example, in the transformation $\underline{\mathbf{h}}_i^{i-1}$, \mathcal{F}_{i-1} is used as the reference coordinate system for the transformation. However, in some situations, it can be useful to perform the same transformation $\underline{\mathbf{h}}_i^{i-1}$ to the frame \mathcal{F}_{i-1} , but using another coordinate system as the reference.

One common situation is shown in figure 14: although the robot hands have different poses, they both must perform the movement prescribed from the point of view of the screw; that is, a clockwise motion. However, if the movement is considered with respect to the robot hand, as in the work of Lozano-Perez et al. (1984), the task must take into account a priori which axis of the robot hand has to be aligned with the rotation axis of the screw. On the other hand, Kröger et al. (2004) showed that if some arbitrary reference frame is used, several subsequent transformations must be performed in order to represent the task in the hand frame so that the inverse kinematic model can be calculated. Also, they relied on trajectory planning to generate motion for the rotational components.

As will be shown next, these limitations are no longer a problem if decompositional multiplication is used, because this operation can modify the frame attached to the robot hand, but using an arbitrary frame as the reference frame for the movement. Also, since the decompositional multiplication is a mathematical operation, there are some intrinsic properties—associativity, existence of the inverse operation, etc—that could be exploited in the description of robot tasks.

Given the previous motivation, in the sequel, definition 2.16 introduces the $\overset{+}{\mathbf{t}}$ operator which represents the translational component of a rigid motion. Then, definition 2.17 introduces the decompositional multiplication, which aims to achieve invariant motions with respect to the pose of the modified frame.

Definition 2.16. Let $\underline{\mathbf{h}} \in \mathcal{H}$, the operator $\overset{+}{\mathbf{t}}(\underline{\mathbf{h}})$ is given by

$$\overset{+}{\mathbf{t}}(\underline{\mathbf{h}}) \triangleq \underline{\mathbf{h}} \mathcal{P}(\underline{\mathbf{h}})^*.$$

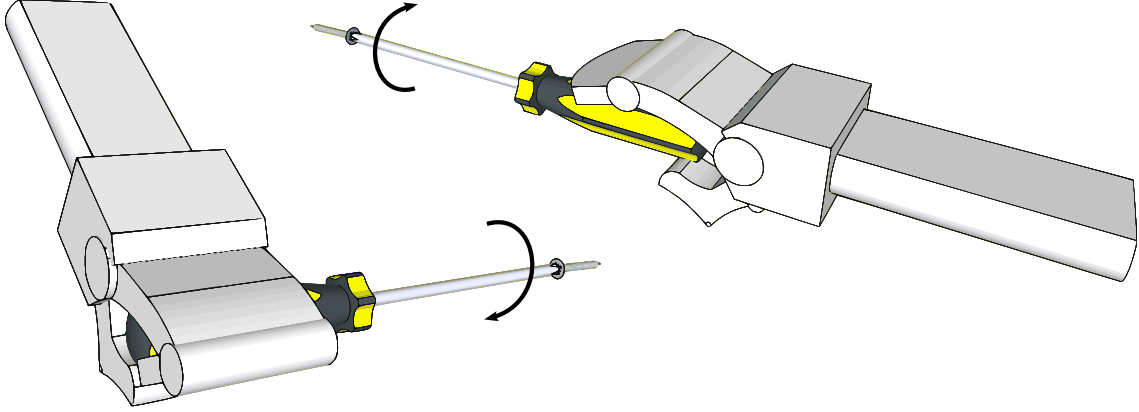


Figure 14: Robot hand manipulating a screwdriver: different hand poses, but the same description for the movement (i.e., from the point of view of the screw, the screwdriver must be rotated clockwise).

The $\overset{+}{\mathbf{t}}$ operator, when applied to the unit quaternion $\underline{\mathbf{h}}$, extracts its translation. For example, given $\underline{\mathbf{h}} = \mathbf{r} + \varepsilon \frac{1}{2} \mathbf{p} \mathbf{r}$,

$$\begin{aligned} \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}) &= \underline{\mathbf{h}} \mathcal{P}(\underline{\mathbf{h}})^* \\ &= (\mathbf{r} + \varepsilon \frac{1}{2} \mathbf{p} \mathbf{r}) \mathbf{r}^* \\ &= 1 + \varepsilon \frac{1}{2} \mathbf{p}. \end{aligned}$$

Example 2.4. Let $\underline{\mathbf{h}}_1^0$ and $\underline{\mathbf{h}}_2^1$ represent the rigid motion from \mathcal{F}_0 to \mathcal{F}_1 and \mathcal{F}_1 to \mathcal{F}_2 , respectively. The resultant transformation caused by the standard multiplication is

$$\begin{aligned} \underline{\mathbf{h}}_2^0 &= \underline{\mathbf{h}}_1^0 \underline{\mathbf{h}}_2^1 \\ &= \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_1^0) \mathcal{P}(\underline{\mathbf{h}}_1^0) \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_2^1) \mathcal{P}(\underline{\mathbf{h}}_2^1). \end{aligned}$$

Without loss of generality, let $\overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_1^0) = 1 + \varepsilon (\hat{\mathbf{j}} \mathbf{a}_1) / 2$ and $\mathcal{P}(\underline{\mathbf{h}}_1^0) = \cos(\pi/4) + \hat{\mathbf{k}} \sin(\pi/4)$; that is, $\underline{\mathbf{h}}_1^0$ corresponds to a translation \mathbf{a}_1 along the y axis followed by a rotation of $\pi/2$ around the z axis. Also, consider $\overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_2^1) = 1 + \varepsilon (\hat{\mathbf{j}} \mathbf{a}_2) / 2$ and $\mathcal{P}(\underline{\mathbf{h}}_2^1) = \cos(\pi/4) + \hat{\mathbf{i}} \sin(\pi/4)$; that is, $\underline{\mathbf{h}}_2^1$ corresponds to a translation \mathbf{a}_2 along the y axis and a rotation of $\pi/2$ around the x axis. The resultant transformation $\underline{\mathbf{h}}_2^0$ is illustrated in figure 15a.

Definition 2.17. The decompositional dual quaternion multiplication, represented by \otimes , is a binary operation⁴ that has precedence over the standard dual quaternion multiplication and is given by

$$\underline{\mathbf{h}}_1^0 \otimes \underline{\mathbf{h}}_2^1 \triangleq \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_2^1) \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_1^0) \mathcal{P}(\underline{\mathbf{h}}_2^1) \mathcal{P}(\underline{\mathbf{h}}_1^0). \quad (2.15)$$

4. Note that binary operation here stands for the 2-ary operation (Bronshtein et al., 2007, p. 298); that is, an operation between two operands.

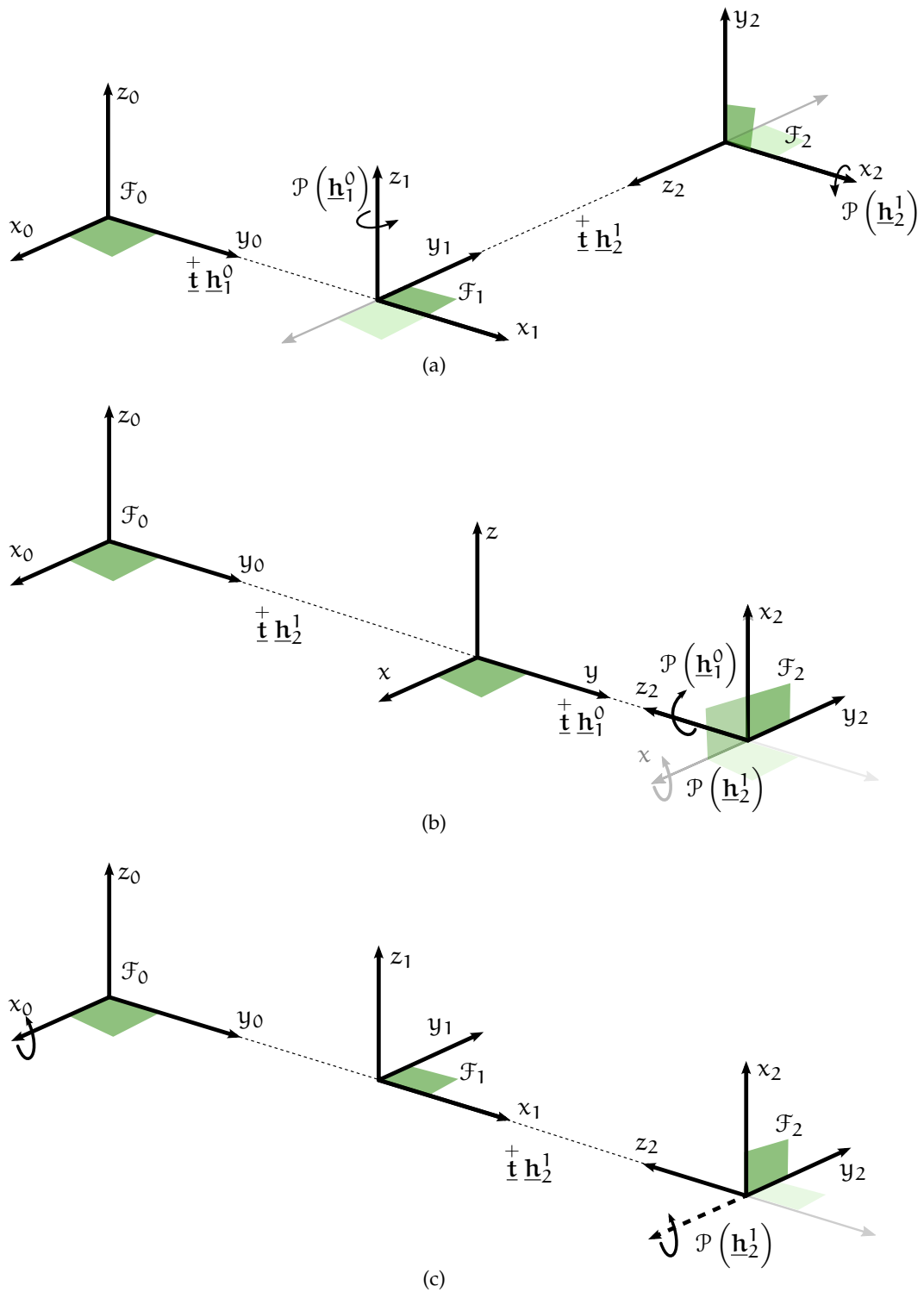


Figure 15: Standard (a) versus decompositional multiplication (b and c). The transformation \underline{h}_1^0 corresponds to a translation along the y axis followed by a rotation of $\pi/2$ around the z axis. The transformation \underline{h}_2^1 corresponds to a translation along the y axis and a rotation of $\pi/2$ around the x axis (see examples 2.4, 2.5, and 2.6). Note that in the standard multiplication \underline{h}_2^1 is given with respect to \mathcal{F}_1 . In the decompositional multiplication, \underline{h}_2^1 transforms \mathcal{F}_1 but uses \mathcal{F}_0 as the reference frame.

Example 2.5. Consider $\underline{\mathbf{h}}_1^0$ and $\underline{\mathbf{h}}_2^1$ from example 2.4. The final transformation under the dual quaternion multiplication,

$$\begin{aligned}\underline{\mathbf{h}}_2^0 &= \underline{\mathbf{h}}_1^0 \otimes \underline{\mathbf{h}}_2^1 \\ &= \underline{\mathbf{t}}(\underline{\mathbf{h}}_2^1) \underline{\mathbf{t}}(\underline{\mathbf{h}}_1^0) \mathcal{P}(\underline{\mathbf{h}}_2^1) \mathcal{P}(\underline{\mathbf{h}}_1^0),\end{aligned}$$

is illustrated in figure 15b.

To clarify definition 2.17 and example 2.5, recall that the dual quaternion $\underline{\mathbf{h}}_1^0$ represents the initial pose of \mathcal{F}_1 —which is the target frame to be modified—with respect to the base frame \mathcal{F}_0 . Also, the dual quaternion $\underline{\mathbf{h}}_2^1$ modifies \mathcal{F}_1 under the decompositional multiplication. Because this operation decomposes the movement, the movement is better seen as a sequence of decomposed movements starting from the reference frame; that is, an observer located at the base frame \mathcal{F}_0 will first see the translations corresponding to the transformation $\underline{\mathbf{t}}(\underline{\mathbf{h}}_2^1)$ and the one corresponding to the original translation $\underline{\mathbf{t}}(\underline{\mathbf{h}}_1^0)$. The effect is a final translation with respect to the base frame. Then, the observer will see the rotation due to the transformation $\mathcal{P}(\underline{\mathbf{h}}_2^1)$, which is also given with respect to a frame aligned with the base frame; until this moment, the base frame was used as the reference for the movement. Finally, the rotation $\mathcal{P}(\underline{\mathbf{h}}_1^0)$ of the original pose is applied, but in a scenario that has already been modified by the transformation $\underline{\mathbf{h}}_2^1$.

Example 2.6. From the previous discussion, since \mathcal{F}_1 is given by the transformation $\underline{\mathbf{h}}_1^0$ and $\underline{\mathbf{h}}_2^0 = \underline{\mathbf{h}}_1^0 \otimes \underline{\mathbf{h}}_2^1$, then it means that $\underline{\mathbf{h}}_2^1$ transforms \mathcal{F}_1 , but uses \mathcal{F}_0 as the reference frame for the transformation. Another way of seeing this comes from definition 2.17, where

$$\begin{aligned}\underline{\mathbf{h}}_1^0 \otimes \underline{\mathbf{h}}_2^1 &= \underline{\mathbf{t}}(\underline{\mathbf{h}}_2^1) \underline{\mathbf{t}}(\underline{\mathbf{h}}_1^0) \mathcal{P}(\underline{\mathbf{h}}_2^1) \mathcal{P}(\underline{\mathbf{h}}_1^0) \\ &= \underline{\mathbf{t}}(\underline{\mathbf{h}}_1^0) \underline{\mathbf{t}}(\underline{\mathbf{h}}_2^1) \mathcal{P}(\underline{\mathbf{h}}_2^1) \mathcal{P}(\underline{\mathbf{h}}_1^0) \\ &= \underline{\mathbf{t}}(\underline{\mathbf{h}}_1^0) \underline{\mathbf{h}}_2^1 \mathcal{P}(\underline{\mathbf{h}}_1^0).\end{aligned}\tag{2.16}$$

Eq. 2.16 shows that a frame at the same position as \mathcal{F}_1 , but aligned with \mathcal{F}_0 , is transformed by $\underline{\mathbf{h}}_2^1$. Since $\underline{\mathbf{h}}_2^1$ is applied to a frame aligned with \mathcal{F}_0 , the initial orientation of \mathcal{F}_1 is not taken into account in the transformation. Hence, $\underline{\mathbf{h}}_2^1$ takes into account only the orientation of the reference frame. Finally, the rotation $\mathcal{P}(\underline{\mathbf{h}}_1^0)$ of \mathcal{F}_1 is applied in a frame already modified by $\underline{\mathbf{h}}_2^1$. In terms of final result, \mathcal{F}_1 will be translated and rotated by a movement defined at \mathcal{F}_0 , which is completely independent of the configuration of \mathcal{F}_1 . This intuitive interpretation is illustrated in figure 15c.

Proposition 2.3. *Given a frame \mathcal{F}_0 , a sequence of decompositional multiplications will always result in a decomposed movement, that is*

$$\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2 \otimes \dots \otimes \underline{\mathbf{h}}_n = \underline{\mathbf{t}}(\underline{\mathbf{h}}_n) \dots \underline{\mathbf{t}}(\underline{\mathbf{h}}_1) \mathcal{P}(\underline{\mathbf{h}}_n) \dots \mathcal{P}(\underline{\mathbf{h}}_1)$$

Proof. The proof is straightforward and uses the definition of decompositional multiplication:

$$\underline{\mathbf{h}}_i \otimes \underline{\mathbf{h}}_{i+1} = \underline{\mathbf{t}}(\underline{\mathbf{h}}_{i+1}) \underline{\mathbf{t}}(\underline{\mathbf{h}}_i) \mathcal{P}(\underline{\mathbf{h}}_{i+1}) \mathcal{P}(\underline{\mathbf{h}}_i).$$

Since $\mathcal{P}(\underline{\mathbf{h}}_i \otimes \underline{\mathbf{h}}_{i+1}) = \mathcal{P}(\underline{\mathbf{h}}_{i+1}) \mathcal{P}(\underline{\mathbf{h}}_i)$ and $\underline{\mathbf{t}}^+(\underline{\mathbf{h}}_i \otimes \underline{\mathbf{h}}_{i+1}) = \underline{\mathbf{t}}^+(\underline{\mathbf{h}}_{i+1}) \underline{\mathbf{t}}^+(\underline{\mathbf{h}}_i)$, using the associative property (see Fact A.11 on page 125) leads to

$$\begin{aligned} \underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2 \otimes \underline{\mathbf{h}}_3 \otimes \dots \otimes \underline{\mathbf{h}}_n &= \left((\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2) \otimes \underline{\mathbf{h}}_3 \otimes \dots \right) \otimes \underline{\mathbf{h}}_n \\ &= \left(\left(\underline{\mathbf{t}}^+(\underline{\mathbf{h}}_3) \underline{\mathbf{t}}^+(\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2) \mathcal{P}(\underline{\mathbf{h}}_3) \mathcal{P}(\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2) \right) \otimes \dots \right) \otimes \underline{\mathbf{h}}_n \\ &= \left(\left(\underline{\mathbf{t}}^+(\underline{\mathbf{h}}_3) \underline{\mathbf{t}}^+(\underline{\mathbf{h}}_2) \underline{\mathbf{t}}^+(\underline{\mathbf{h}}_1) \mathcal{P}(\underline{\mathbf{h}}_3) \mathcal{P}(\underline{\mathbf{h}}_2) \mathcal{P}(\underline{\mathbf{h}}_1) \right) \otimes \dots \right) \otimes \underline{\mathbf{h}}_n \\ &\vdots \\ &= \underline{\mathbf{t}}^+(\underline{\mathbf{h}}_n) \dots \underline{\mathbf{t}}^+(\underline{\mathbf{h}}_1) \mathcal{P}(\underline{\mathbf{h}}_n) \dots \mathcal{P}(\underline{\mathbf{h}}_1) \end{aligned}$$

□

The previous proposition has an important consequence which is exemplified in figure 16.

Finally, the last proposition related to the decompositional multiplication is the inverse operation, which is necessary to retrieve the inverse motion caused by a decomposed movement.

Proposition 2.4. *Let $\underline{\mathbf{h}}$ be a unit dual quaternion. The dual quaternion inverse, under decompositional multiplication, is the dual quaternion $\underline{\mathbf{h}}^\dagger$ such that*

$$\underline{\mathbf{h}}^\dagger = \underline{\mathbf{t}}_{inv}^{+*} \underline{\mathbf{h}}^*,$$

with $\underline{\mathbf{t}}_{inv}^+ = \underline{\mathbf{t}}^+(\underline{\mathbf{h}}^*) \underline{\mathbf{t}}^+(\underline{\mathbf{h}})$.

Proof. By definition, the inverse of the unit dual quaternion under the decompositional multiplication satisfies

$$\underline{\mathbf{h}}^\dagger \otimes \underline{\mathbf{h}} = 1 = \underline{\mathbf{h}} \otimes \underline{\mathbf{h}}^\dagger. \quad (2.17)$$

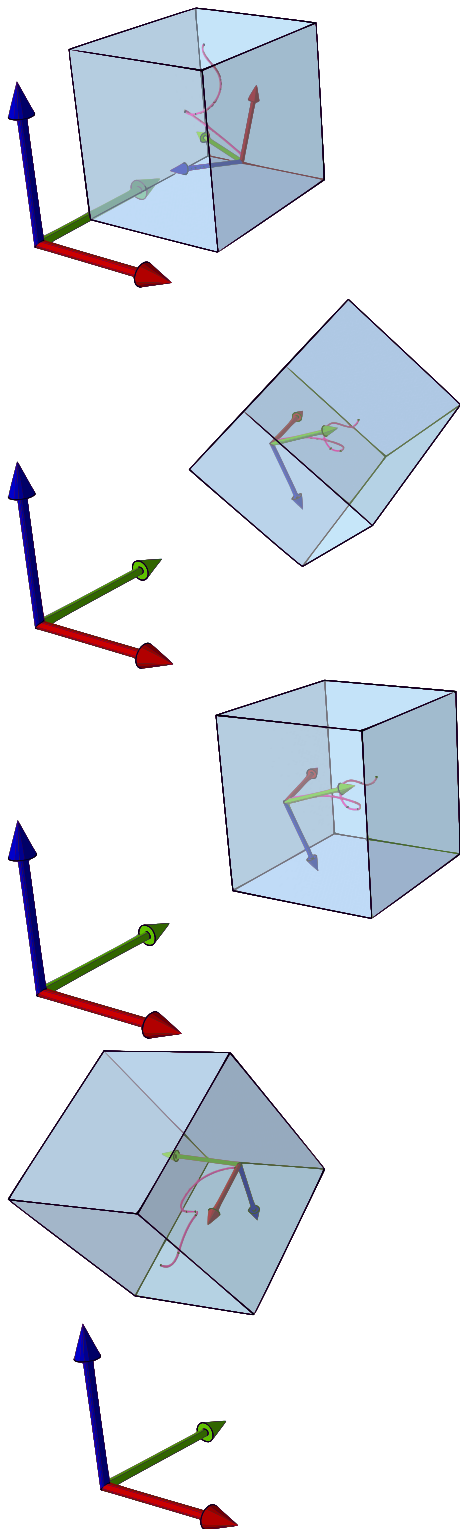
By right multiplying—using the decompositional operation—the first equality of (2.17) by $\underline{\mathbf{h}}^*$, one obtains

$$\begin{aligned} \underline{\mathbf{h}}^* &= \underline{\mathbf{h}}^\dagger \otimes \underline{\mathbf{h}} \otimes \underline{\mathbf{h}}^* \\ &= \underline{\mathbf{h}}^\dagger \otimes \left(\underline{\mathbf{t}}^+(\underline{\mathbf{h}}^*) \underline{\mathbf{t}}^+(\underline{\mathbf{h}}) \mathcal{P}(\underline{\mathbf{h}})^* \mathcal{P}(\underline{\mathbf{h}}) \right) \\ &= \underline{\mathbf{h}}^\dagger \otimes \underbrace{\left(\underline{\mathbf{t}}^+(\underline{\mathbf{h}}^*) \underline{\mathbf{t}}^+(\underline{\mathbf{h}}) \right)}_{\underline{\mathbf{t}}_{inv}^+}. \end{aligned} \quad (2.18)$$

Using fact A.13, then (2.18) becomes

$$\begin{aligned} \underline{\mathbf{t}}_{inv}^+ \underline{\mathbf{h}}^\dagger &= \underline{\mathbf{h}}^* \\ \underline{\mathbf{h}}^\dagger &= \underline{\mathbf{t}}_{inv}^{+*} \underline{\mathbf{h}}^*. \end{aligned}$$

□



Initial pose. Note that before applying the decompositional multiplication, the box is always aligned with the base frame.

The first decompositional multiplication is applied. It translates and rotates the box with respect to the base frame. The internal frame (and any trajectory described in it) will change according to the movement of the box, because they are rigidly attached to each other.

Before the second decompositional multiplication, the frame is put inside a second box (observe that at the beginning of the transformation the box is aligned with the base frame).

Final pose after the second decompositional multiplication. No matter what the initial configuration of the inside frame was, the rigid motion is always specified by the movement of the box with respect to the base frame. However, because the inside frame is rigidly attached to the box, the former will also be transformed.

Figure 16: Invariance obtained by using decompositional multiplication: the “frame in the box” example. There are two frames: a reference frame and a frame inside a virtual box. The inside frame is rigidly attached to the box. An external observer placed at the base frame is interested in what happens with the box and does not observe the internal frame. In this example, two decompositional multiplications are applied.

Corollary 2.1. *Decompositional multiplication can also be achieved by using homogeneous transformation matrices. Consider \mathbf{T}_1 and \mathbf{T}_2 two homogeneous transformation matrices; that is,*

$$\mathbf{T}_i = \begin{bmatrix} \mathbf{R}_i & \mathbf{p}_i \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix},$$

where $\mathbf{R}_i \in \text{SO}(3)$ and $\mathbf{p}_i \in \mathbb{R}^3$. The decompositional multiplication is

$$\begin{aligned} \mathbf{T}_1 \otimes \mathbf{T}_2 &\triangleq \begin{bmatrix} \mathbf{I}_3 & \mathbf{p}_2 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_3 & \mathbf{p}_1 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_2 & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{R}_2 \mathbf{R}_1 & \mathbf{p}_2 + \mathbf{p}_1 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}. \end{aligned}$$

Proof. In order to show the validity of the previous equation, it suffices to show its equivalence to $\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2 = \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_2) \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_1) \mathcal{P}(\underline{\mathbf{h}}_2) \mathcal{P}(\underline{\mathbf{h}}_1)$. Let the operands be dual quaternions such that $\underline{\mathbf{h}}_i = \mathbf{r}_i + \varepsilon(1/2)\mathbf{p}_i\mathbf{r}_i$, with $\mathbf{p}_i, \mathbf{r}_i \in \mathbb{H}$; thus,

$$\begin{aligned} \underline{\mathbf{h}}_3 &= \underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2 = \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_2) \overset{+}{\mathbf{t}}(\underline{\mathbf{h}}_1) \mathcal{P}(\underline{\mathbf{h}}_2) \mathcal{P}(\underline{\mathbf{h}}_1) \\ &= \mathbf{r}_2 \mathbf{r}_1 + \varepsilon \left(\frac{\mathbf{p}_1 + \mathbf{p}_2}{2} \right) \mathbf{r}_2 \mathbf{r}_1. \end{aligned}$$

Since $\mathcal{P}(\underline{\mathbf{h}}_3) = \mathbf{r}_2 \mathbf{r}_1$ corresponds to the final rotation, the equivalence with the final rotation matrix $\mathbf{R}_2 \mathbf{R}_1$ becomes evident. Furthermore, since the final translation is given by (2.11)—that is, $\mathbf{p}_3 = 2 \mathcal{D}(\underline{\mathbf{h}}_3) \mathcal{P}(\underline{\mathbf{h}}_3)^* = \mathbf{p}_1 + \mathbf{p}_2$ —the complete equivalence between the decompositional multiplication by using dual quaternions and the decompositional multiplication by using homogeneous transformation matrices is obtained. \square

2.3 ROBOT KINEMATIC MODELING

This section presents a method to obtain both the FKM and the Jacobian by using dual quaternions. First, the general case is presented, and next the modeling is done for cases when the D-H notation (both standard and modified) is used. Because a similar set of equations can be found when using homogeneous transformation matrices (e.g., Spong et al., 2006; McCarthy, 1990), the section is closed with a comparison—in terms of the number of calculations—between the models obtained using these two parametrizations.

2.3.1 Forward kinematic model

The FKM provides the relation between the positions of the robot joints and the pose of the end-effector. When dual quaternions are used, the FKM is given by

$$\underline{\mathbf{x}}_E = \underline{\mathbf{f}}(\boldsymbol{\theta}), \quad (2.19)$$

where $\underline{\mathbf{x}}_E$ is the dual position representing the pose of the end-effector, $\boldsymbol{\theta}$ is the vector of joint positions, and $\underline{\mathbf{f}}: \mathbb{R}^n \rightarrow \mathcal{H}$.

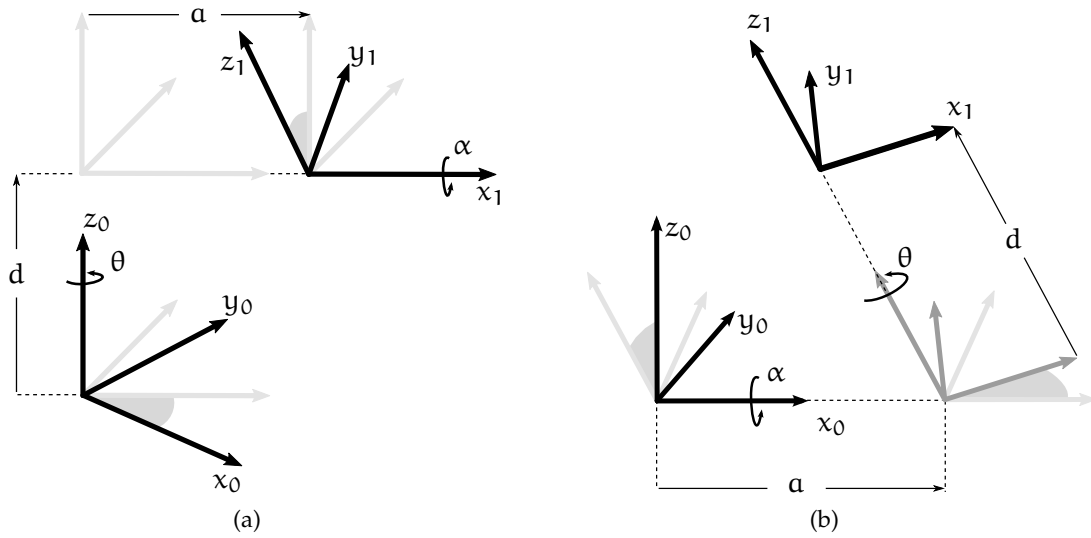


Figure 17: Denavit-Hartenberg parameters: (a) standard and (b) modified conventions.

Consider a serial robot with n links. Each intermediate link is represented by an intermediate transformation \underline{x}_i^{i-1} , relating the configuration of link i with respect to the previous one in the chain. Eq. (2.19) becomes

$$\underline{x}_E = \underline{x}_1^0 \underline{x}_2^1 \dots \underline{x}_n^{n-1}. \quad (2.20)$$

Typically, a serial robot is modeled by using the D-H convention (Spong et al., 2006; Dombre & Khalil, 2007). Given a link of the serial robot, the D-H convention uses four parameters to describe the pose of the link with respect to the previous one in the kinematic chain.

Figure 17a shows these parameters for the standard D-H convention (Spong et al., 2006): first a rotation θ is performed around the z -axis, followed by a translation d along the z -axis; then, a translation a is performed along the x -axis, followed by a rotation α around the x -axis.

Figure 17b shows the transformations described by the modified D-H parameters (Dombre & Khalil, 2007): first a rotation α is performed around the x -axis, followed by a translation a along the x -axis; then, a rotation θ is done around the z -axis, followed by a translation d along the z -axis.

The representation of the D-H in dual quaternion space is straightforward and consists in multiplying the four dual quaternions corresponding to each transformation. Thus,

$$\underline{x}_{DH} = \mathbf{r}_{z,\theta} \underline{\mathbf{p}}_{z,d} \underline{\mathbf{p}}_{x,a} \mathbf{r}_{x,\alpha}, \quad (2.21)$$

where $\mathbf{r}_{z,\theta}$ represents a pure rotation of θ around the z -axis (and analogously for $\mathbf{r}_{x,\alpha}$) and $\underline{\mathbf{p}}_{x,a}$ represents a pure translation of a along the x -axis (analogously for $\underline{\mathbf{p}}_{z,d}$). More

specifically, $\underline{\mathbf{p}}_{x,a} = 1 + \varepsilon (a/2) \hat{\mathbf{i}}$ and $\underline{\mathbf{p}}_{z,d} = 1 + \varepsilon (d/2) \hat{\mathbf{k}}$. Expanding the quaternion multiplications leads to

$$\begin{aligned} \mathcal{P}(\underline{\mathbf{x}}_{\text{DH}}) &= h_{\text{dh}1} + \hat{\mathbf{i}}h_{\text{dh}2} + \hat{\mathbf{j}}h_{\text{dh}3} + \hat{\mathbf{k}}h_{\text{dh}4} \\ \mathcal{D}(\underline{\mathbf{x}}_{\text{DH}}) &= -\frac{(dh_{\text{dh}4} + ah_{\text{dh}2})}{2} + \hat{\mathbf{i}}\frac{(-dh_{\text{dh}3} + ah_{\text{dh}1})}{2} \\ &\quad + \hat{\mathbf{j}}\frac{(dh_{\text{dh}2} + ah_{\text{dh}4})}{2} + \hat{\mathbf{k}}\frac{(dh_{\text{dh}1} - ah_{\text{dh}3})}{2}. \end{aligned} \quad (2.22)$$

with

$$\begin{aligned} h_{\text{dh}1} &= \cos\left(\frac{\theta}{2}\right) \cos\left(\frac{\alpha}{2}\right), & h_{\text{dh}2} &= \cos\left(\frac{\theta}{2}\right) \sin\left(\frac{\alpha}{2}\right), \\ h_{\text{dh}3} &= \sin\left(\frac{\theta}{2}\right) \sin\left(\frac{\alpha}{2}\right), & h_{\text{dh}4} &= \sin\left(\frac{\theta}{2}\right) \cos\left(\frac{\alpha}{2}\right). \end{aligned}$$

The derivation of the modified D-H parameters follows the same procedure. Since the sequence of transformations depicted in figure 17b is described as the following equation

$$\underline{\mathbf{x}}_{\text{mDH}} = \mathbf{r}_{x,\alpha} \underline{\mathbf{p}}_{x,a} \mathbf{r}_{z,\theta} \underline{\mathbf{p}}_{z,d}'$$

the expansion of the right-side leads to

$$\begin{aligned} \mathcal{P}(\underline{\mathbf{x}}_{\text{mDH}}) &= h_{\text{dh}1} + \hat{\mathbf{i}}h_{\text{dh}2} - \hat{\mathbf{j}}h_{\text{dh}3} + \hat{\mathbf{k}}h_{\text{dh}4} \\ \mathcal{D}(\underline{\mathbf{x}}_{\text{mDH}}) &= -\frac{(dh_{\text{dh}4} + ah_{\text{dh}2})}{2} + \hat{\mathbf{i}}\frac{(-dh_{\text{dh}3} + ah_{\text{dh}1})}{2} \\ &\quad - \hat{\mathbf{j}}\frac{(dh_{\text{dh}2} + ah_{\text{dh}4})}{2} + \hat{\mathbf{k}}\frac{(dh_{\text{dh}1} - ah_{\text{dh}3})}{2}. \end{aligned} \quad (2.23)$$

2.3.2 Dual quaternion Jacobian

The dual quaternion Jacobian is the matrix that satisfies

$$\text{vec } \dot{\underline{\mathbf{x}}}_E = \mathbf{J}_{\underline{\mathbf{x}}} \dot{\boldsymbol{\theta}}, \quad (2.24)$$

in which $\dot{\underline{\mathbf{x}}}_E$ is the first derivative of the dual quaternion representing the end-effector pose and $\dot{\boldsymbol{\theta}}$ is the joint velocity vector.

The next subsections show a method to find the dual quaternion Jacobian for the general case, as well for cases when both standard and modified D-H are used. The equations developed will be useful in later chapters, where Jacobian-based controllers are used to generate robot motion.

2.3.2.1 General case

Take the first derivative of (2.20) to obtain

$$\dot{\underline{\mathbf{x}}}_E = \sum_{i=0}^{n-1} \underline{\mathbf{x}}_i^0 \dot{\underline{\mathbf{x}}}_{i+1}^i \underline{\mathbf{x}}_n^{i+1}. \quad (2.25)$$

Let $\underline{\omega}_{i+1}^i$ be a dual quaternion which satisfies $\dot{\underline{x}}_{i+1}^i = (1/2) \underline{\omega}_{i+1}^i \underline{x}_{i+1}^i$; thus,

$$\begin{aligned}\dot{\underline{x}}_E &= \sum_{i=0}^{n-1} \underline{x}_i^0 \left(\frac{1}{2} \underline{\omega}_{i+1}^i \underline{x}_{i+1}^i \right) \underline{x}_n^{i+1} \\ &= \frac{1}{2} \sum_{i=0}^{n-1} \underline{x}_i^0 \underline{\omega}_{i+1}^i \underline{x}_n^i \\ &= \frac{1}{2} \sum_{i=0}^{n-1} \underline{x}_i^0 \underline{\omega}_{i+1}^i (\underline{x}_i^0)^* \underline{x}_E,\end{aligned}\quad (2.26)$$

Because \underline{x}_{i+1}^i is a function of θ_{i+1} , consequently

$$\begin{aligned}\underline{\omega}_{i+1}^i &= 2 \dot{\underline{x}}_{i+1}^i (\underline{x}_{i+1}^i)^* \\ &= 2 \frac{\partial \underline{x}_{i+1}^i}{\partial \theta_{i+1}} (\underline{x}_{i+1}^i)^* \dot{\theta}_{i+1} \\ &= \underline{w}_{i+1}^i \dot{\theta}_{i+1}.\end{aligned}\quad (2.27)$$

Let

$$\begin{aligned}\underline{z}_i \dot{\theta}_{i+1} &= \frac{1}{2} \underline{x}_i^0 \underline{\omega}_{i+1}^i (\underline{x}_i^0)^* \\ \underline{z}_i &= \frac{1}{2} \underline{x}_i^0 \underline{w}_{i+1}^i (\underline{x}_i^0)^*,\end{aligned}\quad (2.28)$$

then (2.26) becomes

$$\dot{\underline{x}}_E = \sum_{i=0}^{n-1} \underline{z}_i \underline{x}_E \dot{\theta}_{i+1}.\quad (2.29)$$

Recall that $\text{vec } \dot{\underline{x}}_E = \mathbf{J}_x \dot{\theta}_{i+1}$ and let $\mathbf{J}_x = [\mathbf{j}_1 \ \cdots \ \mathbf{j}_n]$, with \mathbf{j}_i being column vectors. Inspecting (2.29), it turns out that

$$\mathbf{j}_{i+1} = \text{vec} (\underline{z}_i \underline{x}_E), \quad i = 0, \dots, n-1,\quad (2.30)$$

where \underline{x}_E is the end-effector dual position, which is found by using (2.20); that is, the equation of the FKM.

2.3.2.2 Standard Denavit-Hartenberg

The general solution (2.30) depends on \underline{z}_i , which is a variable that depends on the parametrization used to represent the robot. In the following, the parameter \underline{z}_i is calculated for the case when the standard D-H convention is used.

Eq. (2.28) shows that \underline{w}_{i+1}^i must be calculated in order to find \underline{z}_i . Using (2.27), first $\dot{\underline{x}}_{i+1}^i$ is calculated from (2.22):

$$\begin{aligned}\dot{\underline{x}}_{i+1}^i &= \frac{d}{dt} \underline{x}_{i+1}^i \\ &= \mathcal{P} (\dot{\underline{x}}_{i+1}^i) + \varepsilon \mathcal{D} (\dot{\underline{x}}_{i+1}^i), \\ \mathcal{P} (\dot{\underline{x}}_{i+1}^i) &= \frac{\dot{\theta}_{i+1}}{2} (-h_{dh4} - \hat{i} h_{dh3} + \hat{j} h_{dh2} + \hat{k} h_{dh1}), \\ \mathcal{D} (\dot{\underline{x}}_{i+1}^i) &= \frac{\dot{\theta}_{i+1}}{4} \left((-dh_{dh1} + ah_{dh3}) + \hat{i} (-dh_{dh2} - ah_{dh4}) \right. \\ &\quad \left. + \hat{j} (-dh_{dh3} + ah_{dh1}) + \hat{k} (-dh_{dh4} - ah_{dh2}) \right),\end{aligned}\quad (2.31)$$

with $\underline{\mathbf{x}}_{i+1}^i$ given by (2.22). Then

$$\begin{aligned}\underline{\mathbf{w}}_{i+1}^i &= 2 \frac{\partial \underline{\mathbf{x}}_{i+1}^i}{\partial \theta_{i+1}} (\underline{\mathbf{x}}_{i+1}^i)^* \\ &= \hat{\mathbf{k}}.\end{aligned}\quad (2.32)$$

Replace (2.32) in (2.28) to obtain

$$\underline{\mathbf{z}}_i = \frac{1}{2} \underline{\mathbf{x}}_i^0 \hat{\mathbf{k}} (\underline{\mathbf{x}}_i^0)^* . \quad (2.33)$$

Resolving (2.33) explicitly,

$$\begin{aligned}\mathcal{P}(\underline{\mathbf{z}}_i) &= \hat{\mathbf{i}}(h_{i_2} h_{i_4} + h_{i_1} h_{i_3}) \\ &\quad + \hat{\mathbf{j}}(h_{i_3} h_{i_4} - h_{i_1} h_{i_2}) \\ &\quad + \hat{\mathbf{k}} \left(\frac{h_{i_4}^2 - h_{i_3}^2 - h_{i_2}^2 + h_{i_1}^2}{2} \right),\end{aligned}\quad (2.34)$$

$$\begin{aligned}\mathcal{D}(\underline{\mathbf{z}}_i) &= \hat{\mathbf{i}}(h_{i_2} h_{i_8} + h_{i_6} h_{i_4} + h_{i_1} q_{i_7} + h_{i_5} h_{i_3}) \\ &\quad + \hat{\mathbf{j}}(h_{i_3} h_{i_8} + h_{i_7} h_{i_4} - h_{i_1} h_{i_6} - h_{i_5} h_{i_2}) \\ &\quad + \hat{\mathbf{k}}(h_{i_4} h_{i_8} - h_{i_3} h_{i_7} - h_{i_2} h_{i_6} + h_{i_1} h_{i_5}),\end{aligned}$$

where $h_{i_1} \dots h_{i_8}$ are the coefficients of $\underline{\mathbf{x}}_i^0$.

2.3.2.3 Modified Denavit-Hartenberg

To calculate the dual quaternion Jacobian using the modified D-H parameters, $\underline{\mathbf{x}}_{i+1}^i$ is calculated considering (2.23); thus, $\dot{\underline{\mathbf{x}}}_{i+1}^i$ is

$$\begin{aligned}\dot{\underline{\mathbf{x}}}_{i+1}^i &= \frac{d}{dt} \underline{\mathbf{x}}_{i+1}^i \\ &= \mathcal{P}(\dot{\underline{\mathbf{x}}}_{i+1}^i) + \varepsilon \mathcal{D}(\dot{\underline{\mathbf{x}}}_{i+1}^i), \\ \mathcal{P}(\dot{\underline{\mathbf{x}}}_{i+1}^i) &= \frac{\dot{\theta}_{i+1}}{2} (-h_{dh4} - \hat{\mathbf{i}}h_{dh3} - \hat{\mathbf{j}}h_{dh2} + \hat{\mathbf{k}}h_{dh1}), \\ \mathcal{D}(\dot{\underline{\mathbf{x}}}_{i+1}^i) &= \frac{\dot{\theta}_{i+1}}{4} \left((-dh_{dh1} + ah_{dh3}) + \hat{\mathbf{i}}(-dh_{dh2} - ah_{dh4}) \right. \\ &\quad \left. + \hat{\mathbf{j}}(dh_{dh3} - ah_{dh1}) + \hat{\mathbf{k}}(-dh_{dh4} - ah_{dh2}) \right).\end{aligned}\quad (2.35)$$

Substitute both (2.23) and (2.35) in (2.27). After some calculations, one obtain

$$\underline{\mathbf{w}}_{i+1}^i = -\hat{\mathbf{j}} \sin \alpha_{i+1} + \hat{\mathbf{k}} \cos \alpha_{i+1} - \varepsilon a_{i+1} (\hat{\mathbf{j}} \cos \alpha_{i+1} + \hat{\mathbf{k}} \sin \alpha_{i+1}).$$

The general procedure to calculate the dual quaternion Jacobian convention is summarized in algorithm 2.1. The calculations in lines 4 and 6 depend on the set of parameters used, as shown in the last two subsections.

Algorithm 2.1 Calculation of the dual quaternion Jacobian.

```

1: Calculate  $\underline{x}_E$ 
2:  $\underline{x} \leftarrow 1$ 
3: for  $i = 0$  to  $n - 1$  do
4:   Calculate  $\underline{z}$ 
5:    $\underline{j}_{i+1} \leftarrow \text{vec}(\underline{z}\underline{x}_E)$ 
6:    $\underline{x} \leftarrow \underline{x}\underline{x}_{i+1}^i$ 
7: end for

```

2.4 COMPARISON BETWEEN DUAL QUATERNIONS AND HOMOGENEOUS TRANSFORMATION MATRICES

The comparison is made in terms of the number of elementary operations involved in each method; that is, the number of trigonometric functions, additions/subtractions, and multiplications. To make the calculation clearer, a new nomenclature is used. Given a specified operation, its cost is denoted by the tuple

$$\text{cost}(\text{operation}) = \left\{ \underset{\text{total}}{\text{cost}}(\text{operation}), \underset{\text{mult}}{\text{cost}}(\text{operation}), \underset{\text{add}}{\text{cost}}(\text{operation}) \right\}.$$

For example, the cost of the multiplication of dual quaternions is

$$\underset{\text{total}}{\text{cost}}(\text{mult. of DQ}) = \left\{ \underset{\text{trig}}{\text{cost}}(\text{mult. of DQ}), \underset{\text{mult}}{\text{cost}}(\text{mult. of DQ}), \underset{\text{add}}{\text{cost}}(\text{mult. of DQ}) \right\}.$$

If an element is a function of other variables, the cost for calculating this element is given separately. For example, assume that both \underline{x}_1 and \underline{x}_2 are calculated according to (2.22); thus,

$$\underset{\text{total}}{\text{cost}}(\underline{x}_1 \underline{x}_2) = 2 \underset{\text{total}}{\text{cost}}(\text{Eq. (2.22)}) + \underset{\text{total}}{\text{cost}}(\text{mult. of DQ}).$$

The costs of other operations are defined analogously, as are the costs of the homogeneous transformation matrices.

The next subsections compare the cost of calculating the forward kinematic model, Jacobian, and decompositional multiplication for both methods; namely, using dual quaternions and using homogeneous transformation matrices.

2.4.1 Forward kinematic model

In the comparison of the FKM, the standard D-H convention is used.

2.4.1.1 Homogeneous transformation matrices

Given a serial robot with n links, the end-effector's pose is given by

$$\mathbf{T}_E = \mathbf{T}_{DH_1} \mathbf{T}_{DH_2} \cdots \mathbf{T}_{DH_n},$$

where \mathbf{T}_{DH_i} is the i -th homogeneous transformation matrix representing the transformation from the link $i - 1$ to the link i . Each \mathbf{T}_{DH_i} is given by (Spong et al., 2006)

$$\mathbf{T}_{\text{DHi}} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.36)$$

The cost of calculating \mathbf{T}_{DHi} is

$$\begin{aligned} \text{cost}_{\text{total}}(\mathbf{T}_{\text{DHi}}) &= \left\{ \text{cost}_{\text{trig}}(\mathbf{T}_{\text{DHi}}), \text{cost}_{\text{mult}}(\mathbf{T}_{\text{DHi}}), \text{cost}_{\text{add}}(\mathbf{T}_{\text{DHi}}) \right\} \\ &= \{4, 6, 0\}. \end{aligned}$$

The cost of calculating a multiplication between two homogeneous transformation matrices is

$$\begin{aligned} \text{cost}_{\text{total}}(\text{mult. of HTM}) &= \left\{ \text{cost}_{\text{trig}}(\text{mult. of HTM}), \text{cost}_{\text{mult}}(\text{mult. of HTM}), \text{cost}_{\text{add}}(\text{mult. of HTM}) \right\} \\ &= \{0, 64, 48\}. \end{aligned}$$

Finally,

$$\begin{aligned} \text{cost}_{\text{total}}(\mathbf{T}_{\mathcal{E}}) &= n \text{cost}_{\text{total}}(\mathbf{T}_{\text{DHi}}) + (n-1) \text{cost}_{\text{total}}(\text{mult. of HTM}) \\ &= \{4n, 6n, 0\} + \{0, (n-1)64, (n-1)48\} \\ &= \{4n, 70n - 64, 48n - 48\}. \end{aligned}$$

2.4.1.2 Dual quaternions

Considering that the standard D-H convention is used, the cost of calculating each $\underline{\mathbf{x}}_{\text{DHi}}$ of (2.20) is

$$\begin{aligned} \text{cost}_{\text{total}}(\underline{\mathbf{x}}_{\text{DHi}}) &= \left\{ \text{cost}_{\text{trig}}(\underline{\mathbf{x}}_{\text{DHi}}), \text{cost}_{\text{mult}}(\underline{\mathbf{x}}_{\text{DHi}}), \text{cost}_{\text{add}}(\underline{\mathbf{x}}_{\text{DHi}}) \right\} \\ &= \{4, 12, 4\}. \end{aligned}$$

The cost of multiplying two dual quaternions, that is, the cost of the following operation

$$\underline{\mathbf{x}}_1 \underline{\mathbf{x}}_2 = \mathcal{P}(\underline{\mathbf{x}}_1) \mathcal{P}(\underline{\mathbf{x}}_2) + \varepsilon (\mathcal{P}(\underline{\mathbf{x}}_1) \mathcal{D}(\underline{\mathbf{x}}_2) + \mathcal{D}(\underline{\mathbf{x}}_1) \mathcal{P}(\underline{\mathbf{x}}_2)),$$

is

$$\text{cost}_{\text{total}}(\text{mult. of DQ}) = 3 \text{cost}_{\text{total}}(\text{mult. of quat.}) + \text{cost}_{\text{total}}(\text{add. of quat.}).$$

From (2.2), it follows that

$$\begin{aligned} \text{cost}_{\text{total}}(\text{mult. of DQ}) &= 3\{0, 16, 12\} + \{0, 0, 4\} \\ &= \{0, 48, 40\}. \end{aligned}$$

Table 1: Cost comparison between homogeneous transformation matrices and dual quaternions for an n -DOF serial robot.

	Trigonometric functions	Multiplications	Additions / Subtractions
Cost of calculating the forward kinematic model			
Homogeneous transformation matrices	$4n$	$70n - 64$	$48n - 48$
Dual quaternions	$4n$	$60n - 48$	$44n - 40$
Cost of calculating the Jacobian			
Homogeneous transformation matrices	$8n$	$155n - 64$	$108n - 48$
Dual quaternions	$8n$	$189n - 48$	$142n - 40$
Cost of calculating the decompositional multiplication			
Homogeneous transformation matrices	0	27	21
Dual quaternions	0	64	52

The calculation of the FKM using dual quaternions is given by 2.20; that is, $n - 1$ dual quaternion multiplications; hence,

$$\begin{aligned}
 \text{cost}_{\text{total}}(\underline{x}_E) &= (n - 1)\text{cost}_{\text{total}}(\text{mult. of DQ}) + n\text{cost}_{\text{total}}(\underline{x}_{\text{DH}_i}) \\
 &= \{0, 48n - 48, 40n - 40\} + \{4n, 12n, 4n\} \\
 &= \{4n, 60n - 48, 44n - 40\}.
 \end{aligned}$$

The comparison between the number of operations for each method is summarized in table 1.

In the calculation of the forward kinematic model, dual quaternions cost less than homogeneous transformation matrices. However, the differences are not significant and all the costs are linear in the number of degrees of freedom for both methods.

2.4.2 Jacobian

In the comparison of the calculations for the Jacobians, the standard D-H convention is used. Also, it is assumed that the robots only have rotational joints—the class of robots used in this thesis. The calculations are based on algorithms 2.1 and 2.2. The latter is based on the equations found in standard textbooks (e.g., Spong et al., 2006; McCarthy, 1990).

Algorithm 2.2 Calculation of the geometric Jacobian: standard D-H convention and revolute joints.

- 1: Calculate $\mathbf{T}_{\mathcal{E}}$, with $\mathbf{T}_{\mathcal{E}} = \begin{bmatrix} \mathbf{R}_{\mathcal{E}} & \mathbf{p}_{\mathcal{E}} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$
 - 2: $\mathbf{T} \leftarrow 1$
 - 3: $\mathbf{z} \leftarrow \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$
 - 4: **for** $i = 1$ to n **do**
 - 5: $\mathbf{j}_{\omega_i} \leftarrow \mathbf{z}$
 - 6: $\mathbf{j}_{\mathbf{v}_i} \leftarrow \mathbf{z} \times (\mathbf{p}_{\mathcal{E}} - \mathbf{p})$, where \mathbf{p} is extracted from \mathbf{T}
 - 7: $\mathbf{T} \leftarrow \mathbf{T} \cdot \mathbf{T}_{\text{DHi}}$, with $\mathbf{T}_{\text{DHi}} = \begin{bmatrix} \mathbf{R}_i & \mathbf{p}_i \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$
 - 8: $\mathbf{z} \leftarrow \mathbf{R} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, where \mathbf{R} is extracted from \mathbf{T}
 - 9: **end for**
-

2.4.2.1 Homogeneous transformation matrices

The cost of calculating the geometric Jacobian described in algorithm 2.2 is

$$\begin{aligned} \text{cost}_{\text{total}}(\mathbf{J}) = \text{cost}_{\text{total}}(\mathbf{T}_{\mathcal{E}}) + n \left(\text{cost}_{\text{total}}(\mathbf{p}_{\mathcal{E}} - \mathbf{p}) + \text{cost}_{\text{total}}(\times) + \text{cost}_{\text{total}}(\mathbf{T}_{\text{DHi}}) \right. \\ \left. + \text{cost}_{\text{total}}(\text{mult. of HTM}) + \text{cost}_{\text{total}}\left(\mathbf{R} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T\right) \right) \end{aligned}$$

where $\text{cost}_{\text{total}}(\times)$ is the cost of the cross product. Since

$$\begin{aligned} \text{cost}_{\text{total}}(\mathbf{p}_{\mathcal{E}} - \mathbf{p}) &= \{0, 0, 3\} \\ \text{cost}_{\text{total}}(\times) &= \{0, 6, 3\} \\ \text{cost}_{\text{total}}\left(\mathbf{R} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T\right) &= \{0, 9, 6\}, \end{aligned}$$

then

$$\begin{aligned} \text{cost}_{\text{total}}(\mathbf{J}) &= \{4n, 70n - 64, 48n - 48\} + n (\{0, 0, 3\} + \{0, 6, 3\} + \{4, 6, 0\} + \{0, 64, 48\} + \{0, 9, 6\}) \\ &= \{8n, 155n - 64, 108n - 48\} \end{aligned}$$

2.4.2.2 Dual quaternions

The cost of calculating the analytic Jacobian, using the dual quaternion representation, based on algorithm 2.1, is

$$\text{cost}_{\text{total}}(\mathbf{J}_{\underline{\mathbf{x}}_{\mathcal{E}}}) = \text{cost}_{\text{total}}(\underline{\mathbf{x}}_{\mathcal{E}}) + n \left(\text{cost}_{\text{total}}(\underline{\mathbf{z}}) + \text{cost}_{\text{total}}(\underline{\mathbf{x}}_{\text{DHi}}) + 2\text{cost}_{\text{total}}(\text{mult. of DQ}) \right).$$

Since $\underline{\mathbf{z}}$ is calculated using (2.34), its cost is

$$\begin{aligned} \text{cost}_{\text{total}}(\underline{\mathbf{z}}) &= \left\{ \text{cost}_{\text{trig}}(\underline{\mathbf{z}}), \text{cost}_{\text{mult}}(\underline{\mathbf{z}}), \text{cost}_{\text{add}}(\underline{\mathbf{z}}) \right\} \\ &= \{0, 21, 14\}. \end{aligned}$$

Hence,

$$\begin{aligned} \text{cost}_{\text{total}}(\mathbf{J}_{\underline{x}_E}) &= \{4n, 60n - 48, 44n - 40\} + n(\{0, 21, 14\} + \{4, 12, 4\} + 2\{0, 48, 40\}) \\ &= \{8n, 189n - 48, 142 - 40\} \end{aligned}$$

The results are summarized in table 1.

In the calculation of the Jacobians, dual quaternions cost more than homogeneous transformation matrices. This is not surprising, since the geometric Jacobian is a $6 \times n$ matrix, whereas the dual quaternion Jacobian is a $8 \times n$ matrix. Again, the differences are not significant and all costs are linear in the number of degrees of freedom for both methods.

2.4.3 Decompositional multiplication

Recall from corollary 2.1 that the decompositional multiplication between two homogeneous transformation matrices $\mathbf{T}_1, \mathbf{T}_2 \in SE(3)$ is

$$\mathbf{T}_1 \otimes \mathbf{T}_2 = \begin{bmatrix} \mathbf{R}_2 \mathbf{R}_1 & \mathbf{p}_2 + \mathbf{p}_1 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}.$$

Hence, the cost of calculating a decompositional multiplication by using homogeneous transformation matrices is

$$\begin{aligned} \text{cost}_{\text{total}}(\mathbf{T}_1 \otimes \mathbf{T}_2) &= \left\{ \text{cost}_{\text{trig}}(\mathbf{T}_1 \otimes \mathbf{T}_2), \text{cost}_{\text{mult}}(\mathbf{T}_1 \otimes \mathbf{T}_2), \text{cost}_{\text{add}}(\mathbf{T}_1 \otimes \mathbf{T}_2) \right\} \\ &= \text{cost}_{\text{total}}(\mathbf{R}_2 \mathbf{R}_1) + \text{cost}_{\text{total}}(\mathbf{p}_2 + \mathbf{p}_1) \\ &= \{0, 27, 21\}. \end{aligned}$$

For the dual quaternions $\underline{\mathbf{h}}_1, \underline{\mathbf{h}}_2 \in \mathcal{H}$, with $\underline{\mathbf{h}}_i = \mathcal{P}(\underline{\mathbf{h}}_i) + \varepsilon \mathcal{D}(\underline{\mathbf{h}}_i)$, decompositional multiplication leads to

$$\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2 = \mathcal{P}(\underline{\mathbf{h}}_2) \mathcal{P}(\underline{\mathbf{h}}_1) + \varepsilon (\mathcal{D}(\underline{\mathbf{h}}_1) \mathcal{P}(\underline{\mathbf{h}}_2^*) \mathcal{P}(\underline{\mathbf{h}}_2) + \mathcal{D}(\underline{\mathbf{h}}_2)) \mathcal{P}(\underline{\mathbf{h}}_1),$$

which costs

$$\begin{aligned} \text{cost}_{\text{total}}(\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2) &= \left\{ \text{cost}_{\text{total}}(\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2), \text{cost}_{\text{mult}}(\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2), \text{cost}_{\text{add}}(\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2) \right\} \\ &= 4 \text{cost}_{\text{total}}(\text{mult. of quat.}) + \text{cost}_{\text{total}}(\text{add. of quat.}) \\ &= 4\{0, 16, 12\} + \{0, 0, 4\} \\ &= \{0, 64, 52\}. \end{aligned}$$

Remarkably, for the decompositional multiplication, the cost of the homogeneous transformation matrices is less than half of the cost of dual quaternions. This is due to the fact that, in homogeneous transformation matrices, the translation vector is already separated from the rotation matrix. On the other hand, the rotation and translation quaternions are combined in the dual part of dual quaternions and need to be separated when the decompositional multiplication is performed. The results are summarized in table 1.

2.5 CONCLUSION

Robot modeling using dual quaternions is not particularly new. In fact, Yang laid the foundation for using dual quaternion coordinates in robot modeling in 1963. In his work, Yang dealt with both the kinematic and static aspects arising in spatial mechanisms, although he did not apply dual quaternions directly to robots—these were quite rare at the time. Almost three decades later, McCarthy also contributed to the description of spatial mechanisms using dual quaternions (McCarthy, 1990), and he extensively used matricial notation to represent dual quaternions. Afterwards, McCarthy and collaborators worked specifically in robot modeling. Their works encompassed the modeling of robot dynamics (Dooley & McCarthy, 1991) and the transcription of the robot model in dual quaternion coordinates into Khatib’s operational space framework (Khatib, 1987), leading to what they called “operational image space” (Dooley & McCarthy, 1993). More recently, Perez & McCarthy (2004) have used dual quaternions to synthesize constrained robotic systems.

In addition to the modeling of mechanisms by means of dual quaternions, all the aforementioned works have something else in common. They are focused on robot modeling, but without any concern for how the robot will be controlled. Because the relationship between the model and control is not clear and straightforward, the consequence is that the robot is most often described with one set of mathematical tools (i.e., dual quaternions or, still more common, Cartesian coordinates and quaternions), but then the representation is converted to more common tools (e.g., homogeneous transformation matrices) in order to use standard robot control theory. Whenever less common representations are used for robot kinematic modeling, this separation between robot control and modeling is given, in part because, although controlling the position is straightforward, controlling the orientation is representation-dependent (Campa & de la Torre, 2009).

The first part of this chapter presented the mathematical foundations that are used throughout the thesis. First, a brief introduction showed the motivations for using dual quaternions as the main mathematical object to model serial robots. Next, the chapter presented the general operations of quaternions, dual numbers and dual quaternions. Although part of the notation was drawn from the literature, some elements are not common, but useful to avoid ambiguity in the mathematical description. For instance, quaternions are considered in terms of real and imaginary parts, analogously to the representation of standard imaginary numbers; on the other hand, the separation between the scalar and vector parts—which is due to Hamilton—was banished to avoid the slightest abuse/misunderstanding of the notation (including my own). In this manner, whenever quaternions or dual quaternions are mixed with matrices, the *vec* operator must be used to ensure clarity about what operation is involved.

After the initial mathematical description, the chapter showed how complete rigid motions can be compactly represented using dual quaternions. Moreover, a new operation—one of the main contributions of this chapter—was introduced; namely, the *decompositional multiplication*. This new operation enables the descriptions of rigid motions that modify a target frame, but that are invariant with respect to the pose of this target frame. The main implication is that one can modify a target frame by conveniently using another reference frame to describe the movement. Besides the mathematical description,

some examples were given to illustrate the decomposed movements provided by the decompositional multiplication. Because decompositional multiplication is a mathematical operation, some useful properties can be inferred; for example, the existence of an inverse operation and the guarantee that a sequence of decompositional multiplications will always result in a decomposed motion. Moreover, as shown in example 2.1, the decompositional multiplication is not an operation restricted to dual quaternions. Instead, as a general concept it can be applied to homogeneous transformation matrices as well.

The second part of the chapter presented another contribution: the kinematic modeling of serial robots using dual quaternions. This modeling was explicitly derived for special cases when the Denavit-Hartenberg parameters (standard and modified) are used. Both forward kinematic model and dual quaternion Jacobian were derived for serial robots with rotational joints, which are the main class of robots used in this thesis. However, the method can trivially take into account prismatic joints. Also, parametrizations different from the D-H parameters can be used to represent the FKM; moreover, it should not be difficult to find the analytic Jacobian for these different parametrizations—provided that the resultant FKM is differentiable in the set of chosen parameters.

Last, a comparison was made between the models obtained in dual quaternion space with the ones obtained using homogeneous transformation matrices. The criterion of comparison was the number of calculations required by each method. Regarding the FKM, dual quaternions require fewer multiplications and fewer additions/subtractions. When the Jacobian matrix is considered, the geometric Jacobian requires fewer multiplications and fewer additions/subtractions than the dual quaternion Jacobian. This is not surprising, since the former is a $6 \times n$ matrix, whereas the latter is an $8 \times n$ matrix. Furthermore, for the decompositional multiplication, the cost of the homogeneous transformation matrices is less than half the cost of dual quaternions. This is due to the fact that, in homogeneous transformation matrices, the translation vector is already separated from the rotation matrix. On the other hand, the rotation and translation quaternions are combined in the dual part of dual quaternions and need to be separated when the decompositional multiplication is performed. Despite some differences in terms of the number of elementary operations required by each representation, both representations have linear cost with respect to the number of DOF. As a consequence, these small differences are negligible in practice.

The difference—in terms of the number of calculations—between the dual quaternion representation and the homogeneous transformation matrices is not sensitive in the calculation of either the FKM (where dual quaternions perform better) or the Jacobian or decompositional multiplication (where dual quaternions perform worse). However, dual quaternions provide some attractive properties not found in homogeneous transformation matrices: the commutativity of the Hamilton operators and, as will become evident in the next chapter, the possibility of using the same set of variables to represent the forward kinematics and perform robot control. More specifically, the Hamilton operators are used in the next chapter to easily derive new task Jacobians of the constrained two-arm manipulation task. Finally, the unified representation will bring closer modeling and control, making the resultant theory more compact.

TWO-ARM MANIPULATION: THE COOPERATIVE DUAL TASK-SPACE APPROACH

Two-arm manipulation has been intensively studied over the last thirty years. One of the reasons is the wide range of potential applications for robots equipped with two arms; for instance, they can be used to carry heavy payloads and perform complex assembly tasks (Caccavale & Uchiyama, 2008). Moreover, robots can better perceive the manipulated object by using two arms; for example, one arm can change the pose of the manipulated object to improve feature extraction for a computer vision system, while the other arm performs the manipulation itself, simplifying both perception and control (Edsinger & Kemp, 2008). Also, even if the world is unstructured and only partially modeled, usually the robot is sufficiently modeled and structured. Thus, two-arm manipulation can bring the object to a controlled environment, and the robot can ensure that the object is in a favorable configuration to minimize, for example, the interaction forces (Edsinger & Kemp, 2008).

One of the most well-known frameworks to tackle two-arm manipulations is the symmetric cooperative task-space (Uchiyama & Dauchez, 1988; Chiacchio et al., 1996; Caccavale et al., 2000). In this formalism, whenever an object is firmly grasped, the manipulation task is described in terms of internal and external wrenches¹. On the other hand, if the robot has two-arm coordination but is not necessarily grasping an object, or the manipulated object is flexible, the manipulation task is better described in terms of relative pose and absolute pose (or, alternatively, in terms of relative and absolute twists²). Whereas relative pose describes the configuration between the arms, absolute pose describes the pose of a frame located between the two arms (in the case of manipulation, this frame is typically placed on the object).

This chapter first describes the cooperative dual task-space, which is a new point of view on the symmetric cooperative task-space. More specifically, dual quaternions are used to represent all the elements of the manipulation task; namely, dual positions, twists and wrenches. Such a point of view has some advantages. First, the final description is compact and unified. Furthermore, dual quaternions can directly serve as the input of kinematic controllers, without any intermediate parametrization. Last, but not least, the final representation completely agrees with others related to the symmetric cooperative task-space. Thus, the cooperative dual task-space inherits all the advantages of its former counterparts.

The second part of the chapter is dedicated to kinematic control in the cooperative dual task-space. First, a controller based on dual quaternion feedback is introduced along with some examples of one-arm control. Then, control primitives are derived with the goal of simplifying the definition of the task. Last, the chapter presents simulations—using the Robotics Toolbox (Corke, 1996)—of two-arm kinematic control for two KUKA

1. Wrenches simultaneously represent force and moment.

2. Twists simultaneously represent linear and angular velocities.

LWR 4 robots and also experiments of two-arm kinematic control for the Fujitsu Hoap-3 humanoid robot.

3.1 THE COOPERATIVE DUAL TASK-SPACE: ARM COORDINATION

Consider a two-arm robot where $\underline{\mathbf{x}}_{E_1}$ and $\underline{\mathbf{x}}_{E_2}$ represent—with respect to a common coordinate system—the dual position of left and right end-effectors, respectively, as illustrated in figure 18. The relative dual position determines the configuration of the left end-effector with respect to the right one, whereas the absolute dual position corresponds to a frame located between the end-effectors. The formal definition is given in the following.

Definition 3.1. The relative and absolute dual positions can be defined as

$$\underline{\mathbf{x}}_r^{E_2} \triangleq \underline{\mathbf{x}}_{E_2}^* \underline{\mathbf{x}}_{E_1} \quad (3.1)$$

$$\underline{\mathbf{x}}_a \triangleq \underline{\mathbf{x}}_{E_2} \underline{\mathbf{x}}_{r/2}^{E_2}, \quad (3.2)$$

where $\underline{\mathbf{x}}_{E_2}^*$ is the conjugate of $\underline{\mathbf{x}}_{E_2}$ and $\underline{\mathbf{x}}_{r/2}^{E_2} = \left(\underline{\mathbf{x}}_r^{E_2}\right)^{\{1/2\}}$ is the transformation that corresponds to half of the angle ϕ_r around the axis $\mathbf{n}_r^{E_2} = \hat{\mathbf{i}}n_x + \hat{\mathbf{j}}n_y + \hat{\mathbf{k}}n_z$ of the quaternion $\mathcal{P}\left(\underline{\mathbf{x}}_r^{E_2}\right)$ and half of the translation between the two arms. The dual quaternions $\underline{\mathbf{x}}_a$ and $\underline{\mathbf{x}}_r^{E_2}$ are called *cooperative variables*.

Whereas $\underline{\mathbf{x}}_r$ represents the pose of the first hand with respect to the second one, the geometrical meaning of $\left(\underline{\mathbf{x}}_r^{E_2}\right)^{\{1/2\}}$ can be inferred as follows. First, recall from definition 2.15 that

$$\left(\underline{\mathbf{x}}_r^{E_2}\right)^{\{1/2\}} = \exp\left(\frac{1}{2} \log\left(\underline{\mathbf{x}}_r^{E_2}\right)\right). \quad (3.3)$$

Then, recall that $\underline{\mathbf{x}}_r^{E_2}$ can be written as

$$\begin{aligned} \underline{\mathbf{x}}_r^{E_2} &= \mathcal{P}\left(\underline{\mathbf{x}}_r^{E_2}\right) + \varepsilon \mathcal{D}\left(\underline{\mathbf{x}}_r^{E_2}\right), \\ \mathcal{P}\left(\underline{\mathbf{x}}_r^{E_2}\right) &= \cos\left(\frac{\phi_r}{2}\right) + \sin\left(\frac{\phi_r}{2}\right) \mathbf{n}_r^{E_2} = \mathbf{r}_r^{E_2}, \\ \mathcal{D}\left(\underline{\mathbf{x}}_r^{E_2}\right) &= \frac{1}{2} \mathbf{p}_r^{E_2} \mathbf{r}_r^{E_2}, \end{aligned} \quad (3.4)$$

and let $\underline{\mathbf{x}}_{r/2}^{E_2} = \left(\underline{\mathbf{x}}_r^{E_2}\right)^{\{1/2\}}$ such that $\underline{\mathbf{x}}_{r/2}^{E_2} = \mathbf{r}_{r/2}^{E_2} + \varepsilon(1/2)\mathbf{p}_{r/2}^{E_2}\mathbf{r}_{r/2}^{E_2}$. Because $\|\mathbf{n}_r^{E_2}\| = 1$, and using propositions 2.1 and 2.2, then (3.3) becomes

$$\begin{aligned} \left(\underline{\mathbf{x}}_r^{E_2}\right)^{\{1/2\}} &= \exp\left(\frac{\phi_r}{4} \mathbf{n}_r^{E_2} + \varepsilon \frac{\mathbf{p}_r^{E_2}}{4}\right), \\ \mathcal{P}\left(\left(\underline{\mathbf{x}}_r^{E_2}\right)^{\{1/2\}}\right) &= \mathbf{r}_{r/2}^{E_2} \\ &= \cos\left(\frac{\phi_r}{4}\right) + \sin\left(\frac{\phi_r}{4}\right) \mathbf{n}_r^{E_2}, \\ \mathcal{D}\left(\left(\underline{\mathbf{x}}_r^{E_2}\right)^{\{1/2\}}\right) &= \frac{\mathbf{p}_r^{E_2}}{4} \mathbf{r}_{r/2}^{E_2}. \end{aligned} \quad (3.5)$$

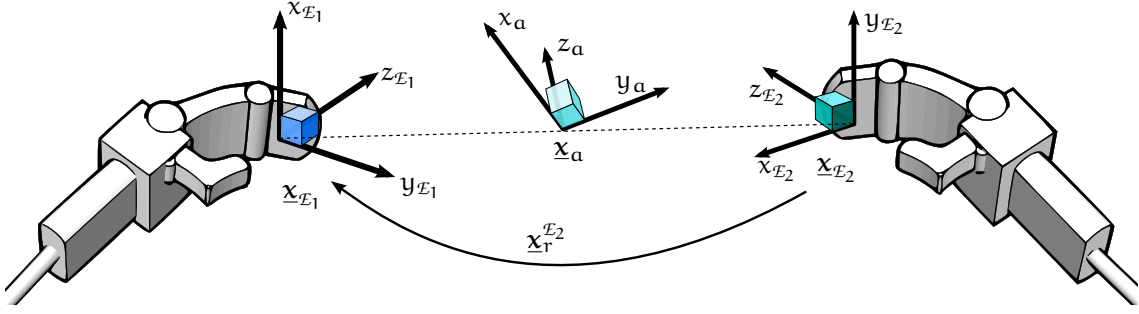


Figure 18: Cooperative dual task-space representation: the absolute and relative dual positions \underline{x}_a and \underline{x}_r completely describe the manipulation task.

Finally, comparing (3.4) with (3.5) leads to

$$\begin{aligned} \mathbf{p}_{r/2}^{E_2} &= \frac{\mathbf{p}_r^{E_2}}{2}, \\ \phi_{r/2} &= \frac{\phi_r}{2}, \\ \mathbf{n}_{r/2}^{E_2} &= \mathbf{n}_r^{E_2}. \end{aligned} \quad (3.6)$$

The relative and absolute dual positions, $\underline{x}_r^{E_2}$ and \underline{x}_a , respectively, are shown in figure 18. The next example shows how a task can be defined in the cooperative dual task-space.

Example 3.1. Consider the task of manipulating a broom, as illustrated in figure 19. Assuming that the robot has already grasped the broom, the task consists of maintaining the relative dual position constant and changing the orientation of the broom. If the robot is mechanically compliant or the broom is not too rigid, the internal forces can be considered negligible if the control of the relative dual position is sufficiently good. In this manner, consider $\underline{x}_{r_0}^{E_2}$ and \underline{x}_{a_0} the relative and absolute dual positions calculated once the robot has grasped the broom with both hands. Because in this example the relative dual position must be kept constant, the desired relative dual position is set to

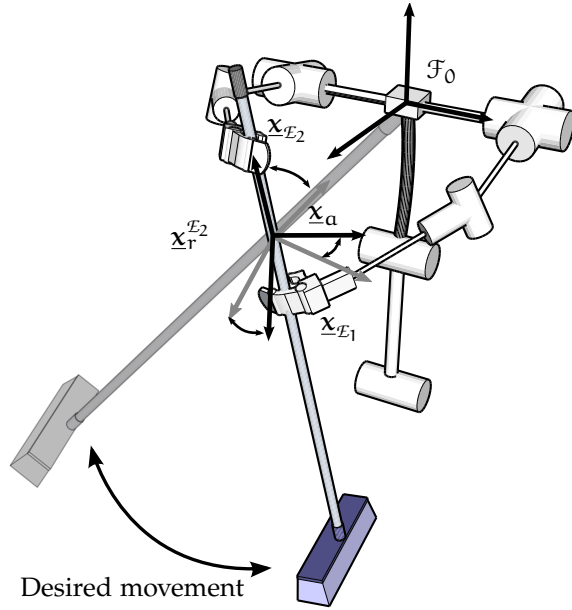
$$\underline{x}_{r_d}^{E_2} = \underline{x}_{r_0}^{E_2}. \quad (3.7)$$

On the other hand, because the absolute dual position depends on the current configuration of both hands, it is not convenient to define the task directly in the absolute frame (the frame described by \underline{x}_a). Otherwise, it would be necessary to redefine the task depending on the grasp. Rather, it is more appropriate to define the task with respect to a more convenient frame, like the one located at the torso, \mathcal{F}_0 .

Let the desired motions $\underline{x}_{a_{d1}}$ and $\underline{x}_{a_{d2}}$, in terms of the absolute dual position, consist of sweeping back and forth. From the initial configuration \underline{x}_{a_0} , a motion $\underline{x}_{\text{motion}}$ with respect to the torso can be predefined such that the desired absolute dual position is given by

$$\underline{x}_{a_{d1}} = \underline{x}_{a_0} \otimes \underline{x}_{\text{motion}} \quad (3.8)$$

Figure 19: Manipulating a broom. Considering a firm grasp, the relative dual position should be constant during the whole manipulation, whereas the movement of the broom can be better described in the frame located at the torso.



for the sweeping movement in one direction. When the motion finishes, the inverse motion is performed; that is,

$$\begin{aligned} \underline{x}_{a,d2} &= \underline{x}_{a_0} \otimes \underline{x}_{\text{motion}} \otimes \underline{x}_{\text{motion}}^\dagger \\ &= \underline{x}_{a_0}. \end{aligned} \quad (3.9)$$

From the previous example, it turns out that the definition of the task is independent of how the broom is grasped or the convention adopted to assign the coordinate system at the end-effectors. This is due to the fact that the decompositional multiplication transforms the absolute dual position, but it uses the coordinate system located at the torso as the reference for the motion.

3.2 COOPERATIVE JACOBIANS

Analogously to the previous chapter, where forward and velocity kinematics were derived for single manipulators (see 2.3), this section shows how the velocity of joint variables affect the velocity of the cooperative variables. More specifically, the goal is to find the Jacobian matrices $\mathbf{J}_{\underline{x}_a}$ and $\mathbf{J}_{\underline{x}_r}$ satisfying the relations

$$\text{vec } \dot{\underline{x}}_a = \mathbf{J}_{\underline{x}_a} \dot{\boldsymbol{\theta}}_R, \quad (3.10)$$

$$\text{vec } \dot{\underline{x}}_r = \mathbf{J}_{\underline{x}_r} \dot{\boldsymbol{\theta}}_R, \quad (3.11)$$

where $\boldsymbol{\theta}_R = [\boldsymbol{\theta}_1^T \quad \boldsymbol{\theta}_2^T]^T$ is the stacked vector of joint variables corresponding to the two-arm system. However, before the derivation of (3.10)–(3.11), some preliminary concepts must be introduced.

3.2.1 Preliminaries

The Jacobian $\underline{J}_{\underline{x}}$ relating the derivative of joint vector $\dot{\theta}$ with the derivative of the dual quaternion $\dot{\underline{x}} = \underline{f}(\theta)$; that is,

$$\text{vec } \dot{\underline{x}} = \underline{J}_{\underline{x}} \dot{\theta},$$

can be decomposed as

$$\underline{J}_{\underline{x}} \triangleq \begin{bmatrix} \underline{J}^{\mathcal{P}(\underline{x})} \\ \underline{J}^{\mathcal{D}(\underline{x})} \end{bmatrix},$$

such that

$$\begin{aligned} \text{vec } (\mathcal{P}(\dot{\underline{x}})) &= \underline{J}^{\mathcal{P}(\underline{x})} \dot{\theta}, \\ \text{vec } (\mathcal{D}(\dot{\underline{x}})) &= \underline{J}^{\mathcal{D}(\underline{x})} \dot{\theta}. \end{aligned} \quad (3.12)$$

Lemma 3.1. *If $\text{vec } \dot{\underline{x}} = \underline{J}_{\underline{x}} \dot{\theta}$, then*

$$\text{vec } \dot{\underline{x}}^* = \underline{J}_{\underline{x}}^* \dot{\theta},$$

where

$$\begin{aligned} \underline{J}_{\underline{x}}^* &= \text{diag}(1, -1, -1, -1, 1, -1, -1, -1) \underline{J}_{\underline{x}} \\ &= \text{diag}(\underline{C}_4, \underline{C}_4) \underline{J}_{\underline{x}} \\ &= \underline{C}_8 \underline{J}_{\underline{x}}; \end{aligned} \quad (3.13)$$

that is, the matrix $\underline{J}_{\underline{x}}^*$ corresponds to the dual quaternion Jacobian premultiplied by a diagonal matrix in order to relate the derivative of the joint variable vector with the derivative of the conjugate dual quaternion.

Proof. Let the matrix \underline{C}_4 be

$$\underline{C}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix},$$

it is straightforward to verify by direct calculation that

$$\text{vec } \dot{\underline{x}}^* = \begin{bmatrix} \underline{C}_4 & \mathbf{0}_4 \\ \mathbf{0}_4 & \underline{C}_4 \end{bmatrix} \text{vec } \dot{\underline{x}}.$$

Taking the first derivative of the previous equation leads to

$$\begin{aligned} \text{vec } \dot{\underline{x}}^* &= \begin{bmatrix} \underline{C}_4 & \mathbf{0}_4 \\ \mathbf{0}_4 & \underline{C}_4 \end{bmatrix} \begin{bmatrix} \underline{J}^{\mathcal{P}(\underline{x})} \\ \underline{J}^{\mathcal{D}(\underline{x})} \end{bmatrix} \dot{\theta} \\ &= \underline{C}_8 \begin{bmatrix} \underline{J}^{\mathcal{P}(\underline{x})} \\ \underline{J}^{\mathcal{D}(\underline{x})} \end{bmatrix} \dot{\theta} \\ &= \underline{J}_{\underline{x}}^* \dot{\theta}. \end{aligned}$$

□

Lemma 3.2. Consider a unit dual quaternion $\underline{\mathbf{x}}$ with $\text{vec } \dot{\underline{\mathbf{x}}} = \mathbf{J}_{\underline{\mathbf{x}}} \dot{\boldsymbol{\theta}}$. Let \mathbf{p} be the translation quaternion related to $\underline{\mathbf{x}}$, then

$$\text{vec } \dot{\mathbf{p}} = \mathbf{J}_{\mathbf{p}} \dot{\boldsymbol{\theta}},$$

with

$$\mathbf{J}_{\mathbf{p}} = 2\bar{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{x}}^*)) \mathbf{J}_{\mathcal{D}(\underline{\mathbf{x}})} + 2\mathbf{H}^+(\mathcal{D}(\underline{\mathbf{x}})) \mathbf{J}_{\mathcal{P}(\underline{\mathbf{x}})}^*, \quad (3.14)$$

Proof. Recall that $\mathbf{p} = 2\mathcal{D}(\underline{\mathbf{x}})\mathcal{P}(\underline{\mathbf{x}}^*)$; thus

$$\begin{aligned} \dot{\mathbf{p}} &= 2\mathcal{D}(\dot{\underline{\mathbf{x}}})\mathcal{P}(\underline{\mathbf{x}}^*) + 2\mathcal{D}(\underline{\mathbf{x}})\mathcal{P}(\dot{\underline{\mathbf{x}}^*}), \\ \text{vec } \dot{\mathbf{p}} &= 2\bar{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{x}}^*)) \text{vec}(\mathcal{D}(\dot{\underline{\mathbf{x}}})) + 2\mathbf{H}^+(\mathcal{D}(\underline{\mathbf{x}})) \text{vec}(\mathcal{P}(\dot{\underline{\mathbf{x}}^*})) \\ &= \left(2\bar{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{x}}^*)) \mathbf{J}_{\mathcal{D}(\underline{\mathbf{x}})} + 2\mathbf{H}^+(\mathcal{D}(\underline{\mathbf{x}})) \mathbf{J}_{\mathcal{P}(\underline{\mathbf{x}})}^* \right) \dot{\boldsymbol{\theta}} \\ &= \mathbf{J}_{\mathbf{p}} \dot{\boldsymbol{\theta}}. \end{aligned}$$

□

3.2.2 Relative dual quaternion Jacobian

As shown in (3.11), the relative dual quaternion Jacobian satisfies the relation

$$\text{vec } \dot{\underline{\mathbf{x}}}_r^{E_2} = \mathbf{J}_{\underline{\mathbf{x}}_r^{E_2}} \dot{\boldsymbol{\theta}}_R,$$

where $\boldsymbol{\theta}_R = [\boldsymbol{\theta}_1 \quad \boldsymbol{\theta}_2]^T$ is the vector composed by the joint variables of the first and second arms.

Take the first derivative of (3.1) and use the commutativity property of the Hamilton operators to obtain

$$\begin{aligned} \dot{\underline{\mathbf{x}}}_r^{E_2} &= \dot{\underline{\mathbf{x}}}_2^* \underline{\mathbf{x}}_1 + \underline{\mathbf{x}}_2^* \dot{\underline{\mathbf{x}}}_1 \\ \text{vec } \dot{\underline{\mathbf{x}}}_r^{E_2} &= \bar{\mathbf{H}}(\underline{\mathbf{x}}_1) \text{vec } \dot{\underline{\mathbf{x}}}_2^* + \mathbf{H}^+(\underline{\mathbf{x}}_2^*) \text{vec } \dot{\underline{\mathbf{x}}}_1 \\ &= \bar{\mathbf{H}}(\underline{\mathbf{x}}_1) \mathbf{J}_{\underline{\mathbf{x}}_2}^* \dot{\boldsymbol{\theta}}_2 + \mathbf{H}^+(\underline{\mathbf{x}}_2^*) \mathbf{J}_{\underline{\mathbf{x}}_1} \dot{\boldsymbol{\theta}}_1 \\ &= \underbrace{\begin{bmatrix} \mathbf{H}^+(\underline{\mathbf{x}}_2^*) \mathbf{J}_{\underline{\mathbf{x}}_1} & \bar{\mathbf{H}}(\underline{\mathbf{x}}_1) \mathbf{J}_{\underline{\mathbf{x}}_2}^* \end{bmatrix}}_{\mathbf{J}_{\underline{\mathbf{x}}_r^{E_2}}} \underbrace{\begin{bmatrix} \dot{\boldsymbol{\theta}}_1 & \dot{\boldsymbol{\theta}}_2 \end{bmatrix}}_{\dot{\boldsymbol{\theta}}_R}^T. \end{aligned} \quad (3.15)$$

3.2.3 Absolute dual quaternion Jacobian

Analogously to the relative dual quaternion Jacobian, the absolute dual quaternion Jacobian satisfies the relation

$$\text{vec } \dot{\underline{\mathbf{x}}}_a = \mathbf{J}_{\underline{\mathbf{x}}_a} \dot{\boldsymbol{\theta}}_R.$$

Take the first derivative of (3.2), then apply the Hamilton operators to obtain

$$\begin{aligned} \dot{\underline{\mathbf{x}}}_a &= \dot{\underline{\mathbf{x}}}_2 \underline{\mathbf{x}}_{r/2}^{E_2} + \underline{\mathbf{x}}_2 \dot{\underline{\mathbf{x}}}_{r/2}^{E_2}, \\ \text{vec } \dot{\underline{\mathbf{x}}}_a &= \bar{\mathbf{H}}(\underline{\mathbf{x}}_{r/2}^{E_2}) \text{vec } \dot{\underline{\mathbf{x}}}_2 + \mathbf{H}^+(\underline{\mathbf{x}}_2) \text{vec } \dot{\underline{\mathbf{x}}}_{r/2}^{E_2}. \end{aligned} \quad (3.16)$$

Using (3.5) and (3.6), $\dot{\underline{\mathbf{x}}}_{r/2}^{E_2}$ is expanded

$$\begin{aligned}\dot{\underline{\mathbf{x}}}_{r/2}^{E_2} &= \dot{\mathbf{r}}_{r/2}^{E_2} + \varepsilon \frac{1}{2} \left(\dot{\mathbf{p}}_{r/2}^{E_2} \mathbf{r}_{r/2}^{E_2} + \mathbf{p}_{r/2}^{E_2} \dot{\mathbf{r}}_{r/2}^{E_2} \right), \\ \text{vec } \dot{\underline{\mathbf{x}}}_{r/2}^{E_2} &= \text{vec } \dot{\mathbf{r}}_{r/2}^{E_2} + \varepsilon \frac{1}{4} \left(\bar{\mathbf{H}} \left(\mathbf{r}_{r/2}^{E_2} \right) \text{vec } \dot{\mathbf{p}}_{r/2}^{E_2} + \mathbf{H} \left(\mathbf{p}_{r/2}^{E_2} \right) \text{vec } \dot{\mathbf{r}}_{r/2}^{E_2} \right).\end{aligned}\quad (3.17)$$

In the previous equation, the term $\dot{\mathbf{p}}_r$ is given by (3.14) and $\dot{\mathbf{r}}_{r/2}^{E_2}$ still remains to be found. Recall from (3.5) that

$$\mathbf{r}_{r/2}^{E_2} = \cos \left(\frac{\phi_r}{4} \right) + \sin \left(\frac{\phi_r}{4} \right) \mathbf{n}_r^{E_2}.$$

Using the quaternion propagation equation (A.7) for both $\dot{\mathbf{r}}_r^{E_2}$ and $\dot{\mathbf{r}}_{r/2}^{E_2}$, and realizing that $\boldsymbol{\omega}_r^{E_2} = 2\boldsymbol{\omega}_{r/2}^{E_2}$ (see (3.47))

$$\begin{aligned}\boldsymbol{\omega}_{r/2}^{E_2} &= 2\dot{\mathbf{r}}_{r/2}^{E_2} \mathbf{r}_{r/2}^{E_2*} \\ \boldsymbol{\omega}_r^{E_2} &= 2\dot{\mathbf{r}}_r^{E_2} \mathbf{r}_r^{E_2*}\end{aligned}\quad (3.18)$$

$$= 4\dot{\mathbf{r}}_{r/2}^{E_2} \mathbf{r}_{r/2}^{E_2*}.\quad (3.19)$$

Since $\mathbf{r}_r^{E_2} = \mathbf{r}_{r/2}^{E_2} \mathbf{r}_{r/2}^{E_2}$ (fact A.7), one obtains

$$\begin{aligned}4\dot{\mathbf{r}}_{r/2}^{E_2} \mathbf{r}_{r/2}^{E_2*} &= 2\dot{\mathbf{r}}_r^{E_2} \mathbf{r}_r^{E_2*} \\ \dot{\mathbf{r}}_{r/2}^{E_2} &= \frac{1}{2} \dot{\mathbf{r}}_r^{E_2} \mathbf{r}_{r/2}^{E_2*} \\ \text{vec } \dot{\mathbf{r}}_{r/2}^{E_2} &= \frac{1}{2} \bar{\mathbf{H}} \left(\mathbf{r}_{r/2}^{E_2*} \right) \text{vec } \dot{\mathbf{r}}_r^{E_2}.\end{aligned}\quad (3.20)$$

Using (3.12), replace (3.20) and (3.14) in (3.17) to obtain

$$\begin{aligned}\text{vec } \mathcal{P} \left(\dot{\underline{\mathbf{x}}}_{r/2}^{E_2} \right) &= \frac{1}{2} \bar{\mathbf{H}} \left(\mathbf{r}_{r/2}^{E_2*} \right) \underbrace{\mathbf{J}_{\mathcal{P}} \left(\underline{\mathbf{x}}_{r/2}^{E_2} \right)}_{\mathbf{J}_{\mathcal{P}} \left(\underline{\mathbf{x}}_{r/2}^{E_2} \right)} \dot{\boldsymbol{\theta}}_R, \\ \text{vec } \mathcal{D} \left(\dot{\underline{\mathbf{x}}}_{r/2}^{E_2} \right) &= \frac{1}{4} \left(\bar{\mathbf{H}} \left(\mathbf{r}_{r/2}^{E_2} \right) \mathbf{J}_{\mathbf{p}_r} + \mathbf{H} \left(\mathbf{p}_{r/2}^{E_2} \right) \mathbf{J}_{\mathcal{P}} \left(\underline{\mathbf{x}}_{r/2}^{E_2} \right) \right) \dot{\boldsymbol{\theta}}_R.\end{aligned}\quad (3.21)$$

Substituting (3.21) in (3.16) leads to

$$\text{vec } \dot{\underline{\mathbf{x}}}_a = \underbrace{\left(\bar{\mathbf{H}} \left(\underline{\mathbf{x}}_{r/2}^{E_2} \right) \mathbf{J}_{\mathbf{x}_{2\text{ext}}} + \mathbf{H} \left(\underline{\mathbf{x}}_2 \right) \mathbf{J}_{\underline{\mathbf{x}}_{r/2}^{E_2}} \right)}_{\mathbf{J}_{\underline{\mathbf{x}}_a}} \dot{\boldsymbol{\theta}}_R,\quad (3.22)$$

where $\mathbf{J}_{\mathbf{x}_{2\text{ext}}} = \begin{bmatrix} \mathbf{0}_{8 \times \dim \boldsymbol{\theta}_1} & \mathbf{J}_{\mathbf{x}_2} \end{bmatrix}$ and $\dim \boldsymbol{\theta}_1$ corresponds to the dimension of the vector $\boldsymbol{\theta}_1$.

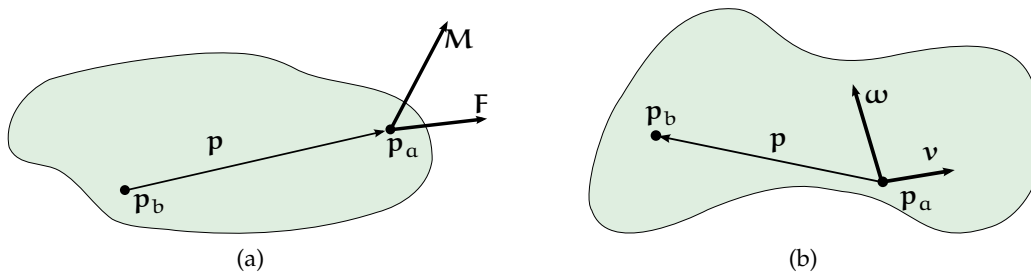


Figure 20: Wrenches and twists represented by dual quaternions: (a) in the wrench, the primary part corresponds to the force, whereas the dual part corresponds to the moment; (b) in the twist, the primary part corresponds to the linear velocity, whereas the dual part corresponds to the angular velocity.

3.3 COOPERATIVE DUAL TASK-SPACE: OBJECT MANIPULATION

Whenever an object is manipulated, wrenches appear in the object. In this manner, the cooperative dual task-space can also be described by internal and external wrenches, as already shown by Uchiyama & Dauchez (1988). Here, the set of equations describing the cooperative dual task-space are analogous to the one obtained by Uchiyama & Dauchez (1988), but with the difference that the representation is unified by means of dual quaternions; hence, wrenches must be represented by dual quaternions. Also, in order to find the equivalence between the equations derived herein and the ones found in preceding works (Uchiyama & Dauchez, 1988; Chiacchio et al., 1996; Caccavale et al., 2000), some preliminary concepts must be revised, such as generalized speeds and generalized forces.

3.3.1 Preliminaries

In his seminal work, Yang (1963) showed that forces and moments can be represented in the dual space by a dual vector. Indeed, forces and moments can be regarded as the components of a Plücker line (see section A.4.1). Since a dual vector is a particular case of a dual quaternion—that is, the former is a dual quaternion with the real part equal to zero—dual forces can also be represented by dual quaternions. Consider the rigid body shown in 20a. Let the force and moment acting at the point \mathbf{p}_a be $\mathbf{F} = \hat{i}F_x + \hat{j}F_y + \hat{k}F_z$ and $\mathbf{M} = \hat{i}M_x + \hat{j}M_y + \hat{k}M_z$, respectively. The wrench at the point \mathbf{p}_a is represented by

$$\underline{\mathbf{f}}_a = \mathbf{F} + \varepsilon \mathbf{M}.$$

If the reference point is now shifted from \mathbf{p}_a to \mathbf{p}_b , the wrench with respect to this new reference is

$$\underline{\mathbf{f}}_b = \mathbf{F} + \varepsilon (\mathbf{M} + \mathbf{p} \times \mathbf{F}),$$

where $\mathbf{p} = \mathbf{p}_a - \mathbf{p}_b$.

Analogously, twists are also Plücker lines and can be represented by dual quaternions as well. Consider the rigid body shown in figure 20b. Let the linear and angular veloci-

ties seen at \mathbf{p}_a be given by $\mathbf{v} = \hat{i}v_x + \hat{j}v_y + \hat{k}v_z$ and $\boldsymbol{\omega} = \hat{i}\omega_x + \hat{j}\omega_y + \hat{k}\omega_z$, respectively. The twist at point \mathbf{p}_a is

$$\underline{\xi}_a = \mathbf{v} + \varepsilon\boldsymbol{\omega}.$$

If a point \mathbf{p}_b is considered as the new reference point, such that $\mathbf{p} = \mathbf{p}_a - \mathbf{p}_b$, then

$$\underline{\xi}_b = (\mathbf{v} + \boldsymbol{\omega} \times \mathbf{p}) + \varepsilon\boldsymbol{\omega}.$$

The next two propositions use the notion of the generalized Jacobian, which plays an important role in the equivalence between the dual quaternion derivative and twists, and between wrenches and generalized forces. Also, the generalized Jacobian gives the relation between the dual quaternion Jacobian and the widely used geometrical Jacobian.

Proposition 3.1. *Let the dual quaternion $\underline{\mathbf{x}}_E$ represent the pose of the end-effector with $\text{vec } \underline{\mathbf{x}}_E = [h_1 \ \dots \ h_8]^\top$. If h_i , $i = 1, \dots, 8$, are defined as generalized coordinates, then*

$$\text{vec } \underline{\xi}_E = \mathbf{G} \text{vec } \dot{\underline{\mathbf{x}}}_E,$$

in which $\underline{\xi}_E = \mathbf{v}_E + \varepsilon\boldsymbol{\omega}_E$ is the twist dual quaternion and \mathbf{G} is the generalized Jacobian relating the twist to the dual quaternion derivative of the end-effector.

Proof. Let the coefficients h_1, \dots, h_8 form the generalized coordinates of the end-effector and $\dot{h}_1, \dots, \dot{h}_8$ form a set of generalized speeds. The following equalities hold (see Kane et al., 1983, p. 87 and Dooley & McCarthy, 1991):

$$\text{vec } \mathbf{v}_E = \sum_{i=1}^8 \boldsymbol{\Psi}_i \dot{h}_i, \quad (3.23)$$

$$\text{vec } \boldsymbol{\omega}_E = \sum_{i=1}^8 \boldsymbol{\Omega}_i \dot{h}_i, \quad (3.24)$$

in which $\boldsymbol{\Psi}_i, \boldsymbol{\Omega}_i \in \mathbb{R}^4$ are the partial linear velocity and partial angular velocity, respectively.

Let $\boldsymbol{\Psi} = [\boldsymbol{\Psi}_1 \ \dots \ \boldsymbol{\Psi}_8]$ and $\boldsymbol{\Omega} = [\boldsymbol{\Omega}_1 \ \dots \ \boldsymbol{\Omega}_8]$; thus,

$$\text{vec } \mathbf{v}_E = \boldsymbol{\Psi} \text{vec } \dot{\underline{\mathbf{x}}}_E,$$

$$\text{vec } \boldsymbol{\omega}_E = \boldsymbol{\Omega} \text{vec } \dot{\underline{\mathbf{x}}}_E.$$

Recall from (2.11) that the translation can be found by using $\mathbf{p}_E = 2 \mathcal{D}(\underline{\mathbf{x}}_E) \mathcal{P}(\underline{\mathbf{x}}_E^*)$. Hence

$$\begin{aligned} \mathbf{v}_E = \dot{\mathbf{p}}_E &= 2 \mathcal{D}(\underline{\mathbf{x}}_E) \frac{\mathcal{P}(\underline{\mathbf{x}}_E^*)}{dt} + 2 \frac{d \mathcal{D}(\underline{\mathbf{x}}_E)}{dt} \mathcal{P}(\underline{\mathbf{x}}_E^*) \\ &= 2 \overset{+}{\mathbf{H}}(\mathcal{D}(\underline{\mathbf{x}}_E)) \frac{\mathcal{P}(\underline{\mathbf{x}}_E^*)}{dt} + 2 \overset{-}{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{x}}_E^*)) \frac{d \mathcal{D}(\underline{\mathbf{x}}_E)}{dt}. \end{aligned} \quad (3.25)$$

Solving the term after the equality of (3.25) and comparing with (3.23), $\boldsymbol{\Psi}$ is found by inspection

$$\Psi = 2 \left[\begin{array}{cccc|cccc} h_5 & h_6 & h_7 & h_8 & h_1 & h_2 & h_3 & h_4 \\ h_6 & -h_5 & h_8 & -h_7 & -h_2 & h_1 & -h_4 & h_3 \\ h_7 & -h_8 & -h_5 & h_6 & -h_3 & h_4 & h_1 & -h_2 \\ h_8 & h_7 & -h_6 & -h_5 & -h_4 & -h_3 & h_2 & h_1 \end{array} \right]. \quad (3.26)$$

To find Ω , first recall the quaternion propagation equation (A.7)

$$\dot{\mathbf{r}}_E = \frac{1}{2} \boldsymbol{\omega}_E \mathbf{r}_E.$$

Thus,

$$\begin{aligned} \Omega \operatorname{vec} \dot{\underline{\mathbf{x}}}_E &= \operatorname{vec} (2\dot{\mathbf{r}}_E \mathbf{r}_E^*) \\ \Omega \begin{bmatrix} \operatorname{vec} \mathcal{P}(\dot{\underline{\mathbf{x}}}_E) \\ \operatorname{vec} \mathcal{D}(\dot{\underline{\mathbf{x}}}_E) \end{bmatrix} &= 2\bar{\mathbf{H}}(\mathbf{r}_E^*) \operatorname{vec} \dot{\mathbf{r}}_E \end{aligned}$$

and, since $\mathcal{P}(\dot{\underline{\mathbf{x}}}_E) = \dot{\mathbf{r}}_E$,

$$\Omega \begin{bmatrix} \operatorname{vec} \mathcal{P}(\dot{\underline{\mathbf{x}}}_E) \\ \operatorname{vec} \mathcal{D}(\dot{\underline{\mathbf{x}}}_E) \end{bmatrix} = \begin{bmatrix} 2\bar{\mathbf{H}}(\mathbf{r}_E^*) & \mathbf{0}_4 \end{bmatrix} \begin{bmatrix} \operatorname{vec} \mathcal{P}(\dot{\underline{\mathbf{x}}}_E) \\ \operatorname{vec} \mathcal{D}(\dot{\underline{\mathbf{x}}}_E) \end{bmatrix}.$$

Finally,

$$\begin{aligned} \Omega &= \begin{bmatrix} 2\bar{\mathbf{H}}(\mathbf{r}_E^*) & \mathbf{0}_4 \end{bmatrix} \\ &= 2 \left[\begin{array}{cccc|cccc} h_1 & h_2 & h_3 & h_4 & 0 & 0 & 0 & 0 \\ -h_2 & h_1 & -h_4 & h_3 & 0 & 0 & 0 & 0 \\ -h_3 & h_4 & h_1 & -h_2 & 0 & 0 & 0 & 0 \\ -h_4 & -h_3 & h_2 & h_1 & 0 & 0 & 0 & 0 \end{array} \right]. \end{aligned} \quad (3.27)$$

Let

$$\mathbf{G} = \begin{bmatrix} \Psi \\ \Omega \end{bmatrix}, \quad (3.28)$$

where $\mathbf{G} \in \mathbb{R}^{8 \times 8}$. Using (3.23)–(3.24) concludes the proof. That is,

$$\operatorname{vec} \underline{\boldsymbol{\xi}}_E = \mathbf{G} \operatorname{vec} \dot{\underline{\mathbf{x}}}_E.$$

□

Proposition 3.2. *Let the dual quaternion $\underline{\mathbf{x}}_E$ represent the pose of the end-effector with $\operatorname{vec} \underline{\mathbf{x}}_E = [h_1 \ \dots \ h_8]^\top$. If h_i , $i = 1, \dots, 8$, are defined as generalized coordinates, then*

$$\operatorname{vec} \underline{\boldsymbol{\Gamma}}_E = \mathbf{G}^\top \operatorname{vec} \underline{\mathbf{f}}_E,$$

in which $\underline{\mathbf{f}}_E = \mathbf{F}_E + \varepsilon \mathbf{M}_E$ is the wrench dual quaternion at the end-effector, $\underline{\boldsymbol{\Gamma}}_E$ is the generalized wrench and \mathbf{G} is the generalized Jacobian.

Proof. Recall from proposition 3.1 that

$$\mathbf{G} = \begin{bmatrix} \boldsymbol{\Psi} \\ \boldsymbol{\Omega} \end{bmatrix},$$

with $\boldsymbol{\Psi} = [\boldsymbol{\Psi}_1 \ \dots \ \boldsymbol{\Psi}_8]$ and $\boldsymbol{\Omega} = [\boldsymbol{\Omega}_1 \ \dots \ \boldsymbol{\Omega}_8]$. Let the generalized wrench be the dual quaternion $\underline{\Gamma}_E$ such that $\text{vec } \underline{\Gamma}_E = [\Gamma_1 \ \dots \ \Gamma_8]^\top$. The generalized active forces Γ_i associated with each generalized coordinate h_i are calculated according to (Kane et al., 1983; Dooley & McCarthy, 1991)

$$\Gamma_i = \text{vec } \mathbf{F}_E \cdot \boldsymbol{\Psi}_i + \text{vec } \mathbf{M}_E \cdot \boldsymbol{\Omega}_i.$$

Let Γ_i be a coefficient of the dual quaternion $\underline{\Gamma}_E$; thus,

$$\begin{aligned} \text{vec } \underline{\Gamma}_E &= \boldsymbol{\Psi}^\top \text{vec } \mathbf{F}_E + \boldsymbol{\Omega}^\top \text{vec } \mathbf{M}_E \\ &= \mathbf{G}^\top \text{vec } \underline{\mathbf{f}}_E \end{aligned}$$

□

Corollary 3.1. *Let the geometrical Jacobian be the matrix $\mathbf{J}_{\underline{\xi}_E}$ that satisfies*

$$\text{vec } \underline{\xi}_E = \mathbf{J}_{\underline{\xi}_E} \dot{\boldsymbol{\theta}},$$

where $\text{vec } \underline{\xi}_E$ is the twist at the end-effector and $\boldsymbol{\theta}$ is the vector of joint variables; thus,

$$\mathbf{J}_{\underline{\xi}_E} = \mathbf{G} \mathbf{J}_{\underline{\mathbf{x}}},$$

where $\mathbf{J}_{\underline{\mathbf{x}}}$ is the dual quaternion Jacobian.

Proof. From (2.24), $\text{vec } \dot{\underline{\mathbf{x}}}_E = \mathbf{J}_{\underline{\mathbf{x}}} \dot{\boldsymbol{\theta}}$. From proposition 3.1, $\text{vec } \underline{\xi}_E = \mathbf{G} \text{vec } \dot{\underline{\mathbf{x}}}_E$; thus,

$$\begin{aligned} \text{vec } \underline{\xi}_E &= \mathbf{G} \text{vec } \dot{\underline{\mathbf{x}}}_E \\ &= \mathbf{G} \mathbf{J}_{\underline{\mathbf{x}}} \dot{\boldsymbol{\theta}} \\ &= \mathbf{J}_{\underline{\xi}_E} \dot{\boldsymbol{\theta}}, \end{aligned}$$

hence $\mathbf{J}_{\underline{\xi}_E} = \mathbf{G} \mathbf{J}_{\underline{\mathbf{x}}}$, concluding the proof. □

This preliminary part introduced three important concepts that are summarized as follows: first, wrenches and twists can be represented by dual quaternions; for example,

$$\begin{aligned} \underline{\mathbf{f}} &= \mathbf{F} + \varepsilon \mathbf{M}, \\ \underline{\boldsymbol{\xi}} &= \mathbf{v} + \varepsilon \boldsymbol{\omega}. \end{aligned}$$

Second, using the generalized Jacobian \mathbf{G} , the following relations were derived in propositions 3.1 and 3.2:

$$\begin{aligned} \text{vec } \underline{\xi}_E &= \mathbf{G} \text{vec } \dot{\underline{\mathbf{x}}}_E, \\ \text{vec } \underline{\Gamma}_E &= \mathbf{G}^\top \text{vec } \underline{\mathbf{f}}_E, \end{aligned}$$

where $\underline{\Gamma}_E$ are the generalized forces.

Last, corollary 3.1 showed that the geometric Jacobian $\mathbf{J}_{\underline{x}_e}$ and the dual quaternion Jacobian $\mathbf{J}_{\underline{x}}$ are related by

$$\mathbf{J}_{\underline{x}_e} = \mathbf{G}\mathbf{J}_{\underline{x}}.$$

Using these three concepts, the next section represents wrenches and twists in the dual cooperative task-space for the case of a firmly grasped object, and then the complete description of two-arm manipulation—in terms of dual positions, wrenches, and twists—is presented with respect to both the manipulated object and the end-effectors.

3.3.2 Wrenches and twists in the cooperative dual task-space

Whenever an object is firmly grasped by two end-effectors, external and internal wrenches can appear in the object. Assume a reference frame \mathcal{F}_a ; the wrench at \mathcal{F}_a , due to the i -th end-effector, is given by (Uchiyama & Dauchez, 1988)

$$\underline{\mathbf{f}}_{s_i} = \mathbf{F}_{E_i} + \varepsilon (\mathbf{M}_{E_i} + \mathbf{p}_i \times \mathbf{F}_{E_i}), \quad (3.29)$$

where \mathbf{p}_i is commonly called “virtual sticks” (see fig. 21).

Lemma 3.3. *Consider a reference frame \mathcal{F}_a attached to an object that is firmly grasped by a two-arm system. The external and internal wrenches, $\underline{\mathbf{f}}_a$ and $\underline{\mathbf{f}}_r$, caused by the end-effectors, are*

$$\underline{\mathbf{f}}_a = \underline{\mathbf{f}}_{s_1} + \underline{\mathbf{f}}_{s_2} \quad (3.30)$$

$$\underline{\mathbf{f}}_{s_r} = \frac{1}{2} (\underline{\mathbf{f}}_{s_1} - \underline{\mathbf{f}}_{s_2}), \quad (3.31)$$

where $\underline{\mathbf{f}}_{s_1}$ and $\underline{\mathbf{f}}_{s_2}$ are given by the wrenches at the tips of the virtual sticks \mathbf{p}_1 and \mathbf{p}_2 , respectively.

Proof. Eq. (3.30) follows directly from the fact that the resultant wrench at \mathcal{F}_a is the sum of all wrenches of the system applied to this point. However, $\underline{\mathbf{f}}_{s_1}$ and $\underline{\mathbf{f}}_{s_2}$ cannot be uniquely determined by using only (3.30), because there are the two variables $\underline{\mathbf{f}}_{s_1}$ and $\underline{\mathbf{f}}_{s_2}$ to be determined, but just one equation. Hence, one more equation is needed to fully specify the wrenches in each arm. Indeed, given a force $\underline{\mathbf{f}}_a$, there are infinite solutions for $\underline{\mathbf{f}}_{s_1}$ and $\underline{\mathbf{f}}_{s_2}$, leading to the conclusion that wrenches not contributing to the movement of the frame \mathcal{F}_a are contributing to internal wrenches. Because (3.30) is a linear mapping between the wrenches applied at the end-effector and the external wrench, one can show that the internal wrench acts in the nullspace of this linear mapping (Uchiyama & Dauchez, 1988).

From the previous discussion, one more equation expressing the internal wrench must be defined to uniquely determine $\underline{\mathbf{f}}_{s_1}$ and $\underline{\mathbf{f}}_{s_2}$. Hence,

$$\underline{\mathbf{f}}_{s_r} = \alpha \underline{\mathbf{f}}_{s_1} + \beta \underline{\mathbf{f}}_{s_2}, \quad (3.32)$$

where α, β are real parameters. Because the choice of α and β is arbitrary, a simple one is

$$\alpha = -\beta. \quad (3.33)$$

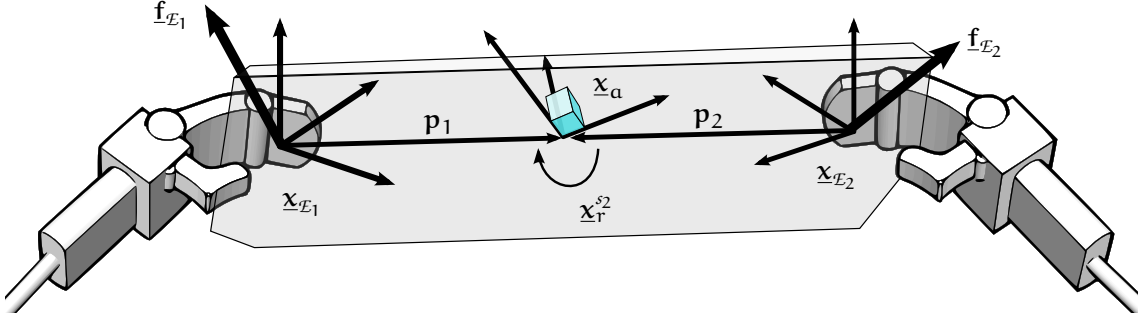


Figure 21: Cooperative dual task-space: manipulation of a firmly grasped object. Note that the relative dual position can also be given by considering the tips of the virtual sticks.

In this manner, when $\underline{f}_{s_1} = \underline{f}_{s_2} \neq 0$, there is no internal stress in the object and \underline{f}_{s_r} equals zero. The choice $\alpha = -\beta = 1/2$ leads to a result analogous to the one obtained by Uchiyama & Dauchez (1988):

$$\underline{f}_{s_r} = \frac{1}{2} (\underline{f}_{s_1} - \underline{f}_{s_2}). \quad (3.34)$$

□

Lemma 3.4. Consider a two-arm system and let $\underline{\xi}_{s_1}$ and $\underline{\xi}_{s_2}$ be the twist at the tip of each virtual stick. The absolute and relative twists are given by

$$\underline{\xi}_a = \frac{\underline{\xi}_{s_1} + \underline{\xi}_{s_2}}{2} \quad (3.35)$$

$$\underline{\xi}_{s_r} = \underline{\xi}_{s_1} - \underline{\xi}_{s_2}. \quad (3.36)$$

Proof. Using the vec operator, the proof is identical to the one of Uchiyama & Dauchez (1988) and is given as follows. Let the cooperative wrench vector and the stacked effector-wrench vector be

$$\underline{f}_c = \begin{bmatrix} \text{vec } \underline{f}_a \\ \text{vec } \underline{f}_{s_r} \end{bmatrix} \text{ and } \underline{f}_s = \begin{bmatrix} \text{vec } \underline{f}_{s_1} \\ \text{vec } \underline{f}_{s_2} \end{bmatrix},$$

respectively. Then

$$\begin{aligned} \underline{f}_c &= \begin{bmatrix} \mathbf{I}_8 & \mathbf{I}_8 \\ \frac{1}{2}\mathbf{I}_8 & -\frac{1}{2}\mathbf{I}_8 \end{bmatrix} \underline{f}_s \\ &= \mathbf{U}^{-1} \underline{f}_s, \end{aligned}$$

where

$$\mathbf{U} = \begin{bmatrix} \frac{1}{2}\mathbf{I}_8 & \mathbf{I}_8 \\ \frac{1}{2}\mathbf{I}_8 & -\mathbf{I}_8 \end{bmatrix}.$$

Repeating the procedure with the absolute and relative twists leads to

$$\underline{\xi}_c = \begin{bmatrix} \text{vec } \underline{\xi}_a \\ \text{vec } \underline{\xi}_{s_r} \end{bmatrix} \text{ and } \underline{\xi}_s = \begin{bmatrix} \text{vec } \underline{\xi}_{s_1} \\ \text{vec } \underline{\xi}_{s_2} \end{bmatrix}.$$

Analogously,

$$\mathbf{x}_c = \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_a \\ \text{vec } \underline{\mathbf{x}}_{s_r} \end{bmatrix} \text{ and } \mathbf{x}_s = \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_{s_1} \\ \text{vec } \underline{\mathbf{x}}_{s_2} \end{bmatrix}.$$

Using the principle of virtual work,

$$\begin{aligned} \delta \mathbf{x}_s^T \mathbf{f}_s &= \delta \mathbf{x}_c^T \mathbf{f}_c \\ \lim_{\delta t \rightarrow 0} \frac{\delta \mathbf{x}_s^T}{\delta t} \mathbf{f}_s &= \lim_{\delta t \rightarrow 0} \frac{\delta \mathbf{x}_c^T}{\delta t} \mathbf{f}_c \\ \underline{\boldsymbol{\xi}}_s^T \mathbf{f}_s &= \underline{\boldsymbol{\xi}}_c^T \mathbf{f}_c \\ \left(\underline{\boldsymbol{\xi}}_s^T \mathbf{U} - \underline{\boldsymbol{\xi}}_c^T \right) \mathbf{f}_c &= 0 \end{aligned}$$

leads to $\underline{\boldsymbol{\xi}}_c = \mathbf{U}^T \underline{\boldsymbol{\xi}}_s$; that is,

$$\begin{aligned} \text{vec } \underline{\boldsymbol{\xi}}_a &= \frac{\text{vec } \underline{\boldsymbol{\xi}}_{s_1} + \text{vec } \underline{\boldsymbol{\xi}}_{s_2}}{2} \\ \text{vec } \underline{\boldsymbol{\xi}}_r &= \text{vec } \underline{\boldsymbol{\xi}}_{s_1} - \text{vec } \underline{\boldsymbol{\xi}}_{s_2}, \end{aligned}$$

which implies (3.35) and (3.36), respectively. \square

The external and internal wrenches, given by (3.30)–(3.31), and the absolute and relative twists, given by (3.35)–(3.36), are used next to describe the cooperative variables at both object and end-effector.

3.3.3 Complete description of the cooperative dual task-space

Whenever there is a coordination task with no objects involved, dual positions or twists are better suited to define the task. In this case, the notion of virtual sticks can be dropped, and the cooperative variables can be given by considering the end-effectors' poses. On the other hand, if a rigid object is firmly grasped, wrenches can also be used to describe the manipulation task. Thus, in order to maintain consistency between wrenches, twists and dual positions, the relations between all these physical quantities must be established.

Theorem 3.1. *Whenever a rigid object is firmly grasped, the cooperative dual task-space is described, at the tips of the virtual sticks, by the following set of equations*

$$\begin{aligned} \mathbf{x}_c &= \mathbf{U}_x \mathbf{x}_s \\ \underline{\boldsymbol{\xi}}_c &= \mathbf{U}_\xi \underline{\boldsymbol{\xi}}_s \\ \mathbf{f}_c &= \mathbf{U}_f \mathbf{f}_s, \end{aligned}$$

where

$$\begin{aligned} \mathbf{x}_c &= \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_a \\ \text{vec } \underline{\mathbf{x}}_{s_r} \end{bmatrix}, & \underline{\boldsymbol{\xi}}_c &= \begin{bmatrix} \text{vec } \underline{\boldsymbol{\xi}}_a \\ \text{vec } \underline{\boldsymbol{\xi}}_{s_r} \end{bmatrix}, & \mathbf{f}_c &= \begin{bmatrix} \text{vec } \underline{\mathbf{f}}_a \\ \text{vec } \underline{\mathbf{f}}_{s_r} \end{bmatrix} \\ \mathbf{x}_s &= \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_{s_1} \\ \text{vec } \underline{\mathbf{x}}_{s_2} \end{bmatrix}, & \underline{\boldsymbol{\xi}}_s &= \begin{bmatrix} \text{vec } \underline{\boldsymbol{\xi}}_{s_1} \\ \text{vec } \underline{\boldsymbol{\xi}}_{s_2} \end{bmatrix}, & \mathbf{f}_s &= \begin{bmatrix} \text{vec } \underline{\mathbf{f}}_{s_1} \\ \text{vec } \underline{\mathbf{f}}_{s_2} \end{bmatrix} \end{aligned}$$

and

$$\mathbf{u}_x = \begin{bmatrix} \mathbf{0}_8 & \bar{\mathbf{H}} \left((\underline{\mathbf{x}}_{s_1}^{s_2})^{\{1/2\}} \right) \\ \mathbf{H}^+ (\underline{\mathbf{x}}_{s_2}^*) & \mathbf{0}_8 \end{bmatrix}, \quad \mathbf{u}_\xi = \begin{bmatrix} \frac{1}{2} \mathbf{I}_8 & \frac{1}{2} \mathbf{I}_8 \\ \mathbf{I}_8 & -\mathbf{I}_8 \end{bmatrix}, \quad \mathbf{u}_f = \begin{bmatrix} \mathbf{I}_8 & \mathbf{I}_8 \\ \frac{1}{2} \mathbf{I}_8 & -\frac{1}{2} \mathbf{I}_8 \end{bmatrix}.$$

Proof. The relation between the vector \mathbf{f}_c of cooperative wrenches and the vector ξ_c of cooperative twists results directly from lemmas (3.3) and (3.4). It remains to find the relation between the vector of cooperative twists and the vector \mathbf{x}_c of cooperative dual positions. It is sufficient to show that the derivative of absolute and relative positions leads directly to (3.35) and (3.36).

The proof is divided into two parts: first, the relation between the linear velocities is established; then, the relation between the angular velocities finishes the proof.

First part: linear velocities

Recall that the dual part of (3.1) is

$$\begin{aligned} \mathcal{D}(\underline{\mathbf{x}}_{s_r}^{s_2}) &= \mathcal{P}(\underline{\mathbf{x}}_{s_2}^*) \mathcal{D}(\underline{\mathbf{x}}_{s_1}) + \mathcal{D}(\underline{\mathbf{x}}_{s_2}^*) \mathcal{P}(\underline{\mathbf{x}}_{s_1}). \\ &= \frac{1}{2} \mathbf{p}_{s_r}^{s_2} \mathbf{r}_{s_1}^{s_2}. \end{aligned}$$

Because $\underline{\mathbf{x}}_r$ is the transformation from $\underline{\mathbf{x}}_{s_2}$ to $\underline{\mathbf{x}}_{s_1}$, the second arm is the reference frame for the transformation; thus,

$$\begin{aligned} \frac{1}{2} \mathbf{p}_{s_r}^{s_2} \mathbf{r}_{s_2}^* \mathbf{r}_{s_1} &= \mathbf{r}_{s_2}^* \frac{1}{2} \mathbf{p}_{s_1} \mathbf{r}_{s_1} - \frac{1}{2} \mathbf{r}_{s_2}^* \mathbf{p}_{s_2} \mathbf{r}_{s_1} \\ \mathbf{p}_{s_r}^{s_2} &= \mathbf{r}_{s_2}^* (\mathbf{p}_{s_1} - \mathbf{p}_{s_2}) \mathbf{r}_{s_2} \\ &= \mathbf{p}_{s_1}^{s_2} - \mathbf{p}_{s_2}^{s_2}, \end{aligned}$$

which leads to

$$\mathbf{p}_{s_r} = \mathbf{p}_{s_1} - \mathbf{p}_{s_2}. \quad (3.37)$$

Finally, taking the first derivative of (3.37) leads to the primary part of (3.36).

The dual part is proved analogously. From (3.2), and observing that $\mathbf{p}_{s_{r/2}}^{s_2} = \mathbf{p}_{s_r}^{s_2}/2$,

$$\begin{aligned} \frac{1}{2} \mathbf{p}_a \mathbf{r}_a &= \frac{1}{2} \mathbf{p}_{s_2} \mathbf{r}_{s_2} \mathbf{r}_{s_{r/2}}^{s_2} + \mathbf{r}_{s_2} \frac{1}{2} \mathbf{p}_{s_{r/2}}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2} \\ \mathbf{p}_a &= \mathbf{p}_{s_2} + \frac{1}{2} \mathbf{r}_{s_2} \mathbf{p}_r^{s_2} \mathbf{r}_{s_2}^* \\ &= \mathbf{p}_{s_2} + \frac{1}{2} \mathbf{p}_{s_r}. \end{aligned}$$

Using 3.37, one obtains

$$\mathbf{p}_a = \frac{\mathbf{p}_{s_1} + \mathbf{p}_{s_2}}{2}. \quad (3.38)$$

The first derivative of (3.38) equals the primary part of (3.35).

Second part: angular velocities

Consider the relative dual position given with respect to the tips of the virtual sticks. Thus, from (3.1),

$$\mathbf{r}_{s_1} = \mathbf{r}_{s_2} \mathbf{r}_{s_r}^{s_2}.$$

Using the quaternion propagation equation (A.7), the first derivative of \mathbf{r}_{s_1} is

$$\dot{\mathbf{r}}_{s_1} = \dot{\mathbf{r}}_{s_2} \mathbf{r}_{s_r}^{s_2} + \mathbf{r}_{s_2} \dot{\mathbf{r}}_{s_r}^{s_2} \quad (3.39)$$

$$\boldsymbol{\omega}_{s_1} \mathbf{r}_{s_1} = \boldsymbol{\omega}_{s_2} \mathbf{r}_{s_2} \mathbf{r}_{s_r}^{s_2} + \mathbf{r}_{s_2} \boldsymbol{\omega}_{s_r}^{s_2} \mathbf{r}_{s_r}^{s_2} \quad (3.40)$$

$$\boldsymbol{\omega}_{s_1} = \boldsymbol{\omega}_{s_2} + \mathbf{r}_{s_2} \boldsymbol{\omega}_{s_r}^{s_2} \mathbf{r}_{s_2}^* \quad (3.41)$$

$$\boldsymbol{\omega}_{s_1} - \boldsymbol{\omega}_{s_2} = \boldsymbol{\omega}_{s_r}, \quad (3.42)$$

which is the dual part of (3.36). In an analogous way, the first derivative of the primary part of (3.2) is

$$\dot{\mathbf{r}}_a = \dot{\mathbf{r}}_{s_2} \mathbf{r}_{s_{r/2}}^{s_2} + \mathbf{r}_{s_2} \dot{\mathbf{r}}_{s_{r/2}}^{s_2} \quad (3.43)$$

$$\boldsymbol{\omega}_a = \boldsymbol{\omega}_{s_2} + \mathbf{r}_{s_2} \boldsymbol{\omega}_{s_{r/2}}^{s_2} \mathbf{r}_{s_2}^* \quad (3.44)$$

$$\boldsymbol{\omega}_{s_{r/2}} = \boldsymbol{\omega}_a - \boldsymbol{\omega}_{s_2}. \quad (3.45)$$

Note that (3.45) introduces the intermediate variable $\boldsymbol{\omega}_{s_{r/2}}$. This term can be found if one considers \mathbf{r}_r as a two-step transformation; that is,

$$\mathbf{r}_{s_r}^{s_2} = \mathbf{r}_{s_{r/2}}^{s_2} \mathbf{r}_{s_1}^{s_{r/2}}, \quad \mathbf{r}_{s_{r/2}}^{s_2} = \mathbf{r}_{s_1}^{s_{r/2}}, \quad (3.46)$$

where the second equation holds because $\mathbf{r}_{s_{r/2}}^{s_2} = \mathbf{r}_{s_1}^{s_{r/2}} = (\mathbf{r}_r^{s_2})^{\{1/2\}}$; that is, both represent the same rotation angle around the same rotation axis. This latter fact also implies $\boldsymbol{\omega}_{s_{r/2}}^{s_2} = \boldsymbol{\omega}_{s_1}^{s_{r/2}}$. The first derivative of (3.46) leads to

$$\begin{aligned} \dot{\mathbf{r}}_{s_r}^{s_2} &= \dot{\mathbf{r}}_{s_{r/2}}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2} + \mathbf{r}_{s_{r/2}}^{s_2} \dot{\mathbf{r}}_{s_{r/2}}^{s_2} \\ \frac{1}{2} \boldsymbol{\omega}_{s_r}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2} &= \frac{1}{2} \boldsymbol{\omega}_{s_{r/2}}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2} + \mathbf{r}_{s_{r/2}}^{s_2} \frac{1}{2} \boldsymbol{\omega}_{s_{r/2}}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2} \\ \boldsymbol{\omega}_{s_r}^{s_2} &= \boldsymbol{\omega}_{s_{r/2}}^{s_2} + \mathbf{r}_{s_{r/2}}^{s_2} \boldsymbol{\omega}_{s_{r/2}}^{s_2} \mathbf{r}_{s_{r/2}}^{s_2*}. \end{aligned}$$

Since the rotation axis of $\mathbf{r}_{s_{r/2}}^{s_2}$ and $\boldsymbol{\omega}_{s_{r/2}}^{s_2}$ have the same direction, Fact A.6 holds, leading to

$$\boldsymbol{\omega}_{s_r}^{s_2} = \boldsymbol{\omega}_{s_{r/2}}^{s_2} + \boldsymbol{\omega}_{s_{r/2}}^{s_2}, \quad (3.47)$$

which implies

$$\boldsymbol{\omega}_a = \frac{\boldsymbol{\omega}_{s_1} + \boldsymbol{\omega}_{s_2}}{2}, \quad (3.48)$$

completing the proof. \square

Corollary 3.2. *Considering a firmly grasped object and using generalized coordinates, speeds, and forces, the cooperative dual task-space is described by the following set of equations*

$$\mathbf{x}_c = \mathbf{U}_x \mathbf{x}_s$$

$$\dot{\mathbf{x}}_c = \mathbf{U}_{\dot{x}} \dot{\mathbf{x}}_s$$

$$\boldsymbol{\Gamma}_c = \mathbf{U}_\Gamma \boldsymbol{\Gamma}_s,$$

where

$$\begin{aligned} \mathbf{x}_c &= \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_a \\ \text{vec } \underline{\mathbf{x}}_r^{s_2} \end{bmatrix}, & \dot{\mathbf{x}}_c &= \begin{bmatrix} \text{vec } \dot{\underline{\mathbf{x}}}_a \\ \text{vec } \dot{\underline{\mathbf{x}}}_{s_r} \end{bmatrix}, & \Gamma_c &= \begin{bmatrix} \Gamma_a \\ \Gamma_{s_r} \end{bmatrix} \\ \mathbf{x}_s &= \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_{s_1} \\ \text{vec } \underline{\mathbf{x}}_{s_2} \end{bmatrix}, & \dot{\mathbf{x}}_s &= \begin{bmatrix} \text{vec } \dot{\underline{\mathbf{x}}}_{s_1} \\ \text{vec } \dot{\underline{\mathbf{x}}}_{s_2} \end{bmatrix}, & \Gamma_s &= \begin{bmatrix} \Gamma_{s_1} \\ \Gamma_{s_2} \end{bmatrix} \end{aligned}$$

and

$$\begin{aligned} \mathbf{U}_x &= \begin{bmatrix} \mathbf{0}_8 & \bar{\mathbf{H}} \left((\underline{\mathbf{x}}_{s_1}^{s_2})^{\{1/2\}} \right) \\ \mathbf{H}^+ (\underline{\mathbf{x}}_{s_2}^*) & \mathbf{0}_8 \end{bmatrix}, & \mathbf{U}_{\dot{\mathbf{x}}} &= \begin{bmatrix} \frac{1}{2} \mathbf{G}_{\underline{\mathbf{x}}_a}^{-1} \mathbf{G}_{s_1} & \frac{1}{2} \mathbf{G}_{\underline{\mathbf{x}}_a}^{-1} \mathbf{G}_{s_2} \\ \mathbf{G}_{\underline{\mathbf{x}}_{s_r}}^{-1} \mathbf{G}_{s_1} & -\mathbf{G}_{\underline{\mathbf{x}}_{s_r}}^{-1} \mathbf{G}_{s_2} \end{bmatrix}, \\ \mathbf{U}_\Gamma &= \begin{bmatrix} \mathbf{G}_{\underline{\mathbf{x}}_a}^\top \mathbf{G}_{s_1}^{-\top} & \mathbf{G}_{\underline{\mathbf{x}}_a}^\top \mathbf{G}_{s_2}^{-\top} \\ \frac{1}{2} \mathbf{G}_{\underline{\mathbf{x}}_{s_r}}^\top \mathbf{G}_{s_1}^{-\top} & -\frac{1}{2} \mathbf{G}_{\underline{\mathbf{x}}_{s_r}}^\top \mathbf{G}_{s_2}^{-\top} \end{bmatrix}. \end{aligned}$$

Proof. In order to find \mathbf{U}_x , recall that

$$\begin{aligned} \underline{\mathbf{x}}_{s_r}^{s_2} &= \underline{\mathbf{x}}_{s_2}^* \underline{\mathbf{x}}_{s_1}, \\ \underline{\mathbf{x}}_a &= \underline{\mathbf{x}}_{s_2} \underline{\mathbf{x}}_{s_r/2}^{s_2}. \end{aligned}$$

Thus, $\text{vec } \underline{\mathbf{x}}_{s_r}^{s_2} = \mathbf{H}^+ (\underline{\mathbf{x}}_{s_2}^*) \text{vec } \underline{\mathbf{x}}_1$ and, since $\underline{\mathbf{x}}_{s_r/2}^{s_2} = (\underline{\mathbf{x}}_{s_1}^{s_2})^{\{1/2\}}$, then $\underline{\mathbf{x}}_a = \bar{\mathbf{H}} (\underline{\mathbf{x}}_{s_r/2}^{s_2}) \text{vec } \underline{\mathbf{x}}_{s_2}$, from which \mathbf{U}_x follows directly. The matrices $\mathbf{U}_{\dot{\mathbf{x}}}$ and \mathbf{U}_Γ are obtained directly from (3.35)–(3.36), where the relation $\text{vec } \underline{\xi}_i = \mathbf{G}_i \text{vec } \underline{\mathbf{x}}_i$ is used. Furthermore, the matrices $\mathbf{U}_{\dot{\mathbf{x}}}$ and \mathbf{U}_Γ satisfy the relation

$$\mathbf{U}_{\dot{\mathbf{x}}}^\top = \mathbf{U}_\Gamma^{-1}.$$

□

Remark 3.1. The relation $\mathbf{U}_{\dot{\mathbf{x}}}^\top = \mathbf{U}_\Gamma^{-1}$ also comes from the fact that the generalized speeds and generalized forces satisfy the principle of virtual work; that is,

$$\begin{aligned} \dot{\mathbf{x}}_s^\top \Gamma_s &= \dot{\mathbf{x}}_c^\top \Gamma_c \\ &= \dot{\mathbf{x}}_s^\top \mathbf{U}_{\dot{\mathbf{x}}}^\top \Gamma_c, \\ \Gamma_s &= \mathbf{U}_{\dot{\mathbf{x}}}^\top \Gamma_c \\ &= \mathbf{U}_\Gamma^{-1} \Gamma_c. \end{aligned}$$

Remark 3.2. Using the commutative property of the Hamilton operators, it is easy to verify that the matrix \mathbf{U}_x can also be written as

$$\mathbf{U}_x = \begin{bmatrix} \bar{\mathbf{H}} \left((\underline{\mathbf{x}}_{s_2}^{s_1})^{\{1/2\}} \right) & \mathbf{0}_8 \\ \mathbf{0}_8 & \bar{\mathbf{H}} (\underline{\mathbf{x}}_{s_1}) \mathbf{C}_8 \end{bmatrix}.$$

Theorem 3.2. *Considering a firmly grasped rigid object, the cooperative variables, perceived at the end effectors, are given by the following set of equations*

$$\begin{aligned} \mathbf{x}_E &= \mathbf{V}_x^{-1} \mathbf{x}_s \\ \underline{\xi}_E &= \mathbf{V}_\xi^{-1} \underline{\xi}_s \\ \mathbf{f}_E &= \mathbf{V}_f^{-1} \mathbf{f}_s, \end{aligned}$$

where

$$\begin{aligned} \mathbf{x}_{\mathcal{E}} &= \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_{\mathcal{E}_1} \\ \text{vec } \underline{\mathbf{x}}_{\mathcal{E}_2} \end{bmatrix}, & \underline{\boldsymbol{\xi}}_{\mathcal{E}} &= \begin{bmatrix} \text{vec } \underline{\boldsymbol{\xi}}_{\mathcal{E}_1} \\ \text{vec } \underline{\boldsymbol{\xi}}_{\mathcal{E}_2} \end{bmatrix}, & \mathbf{f}_{\mathcal{E}} &= \begin{bmatrix} \text{vec } \underline{\mathbf{f}}_{\mathcal{E}_1} \\ \text{vec } \underline{\mathbf{f}}_{\mathcal{E}_2} \end{bmatrix} \\ \mathbf{x}_{\mathcal{S}} &= \begin{bmatrix} \text{vec } \underline{\mathbf{x}}_{\mathcal{S}_1} \\ \text{vec } \underline{\mathbf{x}}_{\mathcal{S}_2} \end{bmatrix}, & \underline{\boldsymbol{\xi}}_{\mathcal{S}} &= \begin{bmatrix} \text{vec } \underline{\boldsymbol{\xi}}_{\mathcal{S}_1} \\ \text{vec } \underline{\boldsymbol{\xi}}_{\mathcal{S}_2} \end{bmatrix}, & \mathbf{f}_{\mathcal{S}} &= \begin{bmatrix} \text{vec } \underline{\mathbf{f}}_{\mathcal{S}_1} \\ \text{vec } \underline{\mathbf{f}}_{\mathcal{S}_2} \end{bmatrix} \end{aligned}$$

and

$$\mathbf{V}_{\mathbf{x}} = \begin{bmatrix} \bar{\mathbf{H}}(\underline{\mathbf{p}}_{\mathcal{E}_1, \alpha}^{\mathcal{E}_1}) & \mathbf{0}_8 \\ \mathbf{0}_8 & \bar{\mathbf{H}}(\underline{\mathbf{p}}_{\mathcal{E}_2, \alpha}^{\mathcal{E}_2}) \end{bmatrix}, \quad \mathbf{V}_{\underline{\boldsymbol{\xi}}} = \begin{bmatrix} \mathbf{V}_{\underline{\boldsymbol{\xi}}_1} & \mathbf{0}_8 \\ \mathbf{0}_8 & \mathbf{V}_{\underline{\boldsymbol{\xi}}_2} \end{bmatrix}, \quad \mathbf{V}_{\mathbf{f}} = \mathbf{V}_{\underline{\boldsymbol{\xi}}}^{\top}, \quad (3.49)$$

with

$$\mathbf{V}_{\underline{\boldsymbol{\xi}}_i} = \begin{bmatrix} \mathbf{I}_4 & \overset{\times}{\mathbf{H}}(-\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}) \\ \mathbf{0}_4 & \mathbf{I}_4 \end{bmatrix}, \quad \underline{\mathbf{p}}_{\mathcal{E}_i, \alpha} = \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\mathcal{E}_i})^* \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\alpha}), \quad (3.50)$$

$$\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}^{\mathcal{E}_i} = \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\mathcal{E}_i}^* \underline{\mathbf{x}}_{\alpha}), \quad \underline{\mathbf{p}}_{\mathcal{E}_i, \alpha} = 2 \mathcal{D}(\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}). \quad (3.51)$$

Proof. If the object is firmly grasped and the assumption of no deformation holds, the tips of the virtual sticks are always placed at the origin of the absolute frame \mathcal{F}_{α} . Furthermore, the virtual stick $\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}$ is invariant with respect to the frame \mathcal{F}_i . Let $\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}$ be

$$\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha} = 1 + \varepsilon \frac{\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}}{2},$$

then, by using (2.11), $\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha} = 2 \mathcal{D}(\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha})$. Furthermore, since the tips of the virtual sticks are always placed at the origin of the absolute frame, the following holds

$$\begin{aligned} \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\mathcal{E}_i}) \overset{+}{\mathbf{t}}(\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}) &= \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\alpha}) \\ \overset{+}{\mathbf{t}}(\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}) &= \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\alpha}) \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\mathcal{E}_i})^*. \end{aligned}$$

Since the $\overset{+}{\mathbf{t}}$ operator is commutative with itself and $\mathcal{P}(\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha})$ equals zero, thus $\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha} = \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\mathcal{E}_i})^* \overset{+}{\mathbf{t}}(\underline{\mathbf{x}}_{\alpha})$.

Recall that $\underline{\boldsymbol{\xi}}_{\mathcal{E}_i} = \mathbf{v}_{\mathcal{E}_i} + \varepsilon \boldsymbol{\omega}_{\mathcal{E}_i}$ and $\underline{\boldsymbol{\xi}}_{\mathcal{S}_i} = (\mathbf{v}_{\mathcal{S}_i} + \boldsymbol{\omega}_{\mathcal{E}_i} \times \underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}) + \varepsilon \boldsymbol{\omega}_{\mathcal{E}_i}$. Applying the vec operator leads to

$$\begin{aligned} \text{vec } \underline{\boldsymbol{\xi}}_{\mathcal{S}_i} &= \begin{bmatrix} \text{vec}(\mathbf{v}_{\mathcal{E}_i} + \boldsymbol{\omega}_{\mathcal{E}_i} \times \underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}) \\ \text{vec } \boldsymbol{\omega}_{\mathcal{E}_i} \end{bmatrix} \\ &= \begin{bmatrix} \text{vec } \mathbf{v}_{\mathcal{E}_i} + \overset{\times}{\mathbf{H}}(-\underline{\mathbf{p}}_{\mathcal{E}_i, \alpha}) \boldsymbol{\omega}_{\mathcal{E}_i} \\ \text{vec } \boldsymbol{\omega}_{\mathcal{E}_i} \end{bmatrix} \\ &= \mathbf{V}_{\underline{\boldsymbol{\xi}}_i} \text{vec } \underline{\boldsymbol{\xi}}_{\mathcal{E}_i}. \end{aligned}$$

The matrix \mathbf{V}_ξ can be verified by a simple inspection and \mathbf{V}_f can be found by using direct analysis, analogously to the one just presented, or by the principle of virtual work. Finally, the matrix \mathbf{V}_x follows from the fact that there is no deformation, and the relation $\underline{x}_{s_i} = \underline{x}_{E_i} \mathbf{p}_{E_i, a}^{E_i}$ is always valid during the manipulation. Using the Hamilton operator leads to

$$\text{vec } \underline{x}_{s_i} = \bar{\mathbf{H}} \left(\mathbf{p}_{E_i, a}^{E_i} \right) \text{vec } \underline{x}_{E_i},$$

from which the matrix \mathbf{V}_x is easily found. \square

In summary, the cooperative dual task-space can be described in two different but equivalent ways. It can be represented at the tips of the virtual sticks (theorem 3.1) or at the end-effectors (theorem 3.2). Typically, the cooperative variables are defined at the object (i.e., at the tips of the virtual sticks) but are controlled with respect to the end-effector; hence, both representations have practical use.

3.4 KINEMATIC CONTROL IN THE COOPERATIVE DUAL TASK-SPACE

The kinematic control of robotic manipulators has been widely studied in robotics, and this topic is covered in most robotics textbooks (see, for instance, Spong et al. (2006); Siciliano & Khatib (2008); Siciliano et al. (2009)). Usually, there are control methods in joint-space and task-space—the latter can be defined in terms of the end-effector dual position. Although control in joint-space coordinates is simpler if each joint is considered separately, the task is usually defined in the task-space and thus task-space control can be more suitable for complex tasks, as Whitney (1969) observed more than forty years ago.

Kinematic controllers do not take into account the non-linear and coupling effects of the robot dynamic model, but they are conceptually simple and can be easily implemented in position-actuated robots.

Consider \mathbf{x}_d and \mathbf{x}_m the desired and measured values for the task-space variables, respectively. Siciliano et al. (2009) showed that the following control law is asymptotically stable, assuming no representation singularities³:

$$\dot{\boldsymbol{\theta}} = \mathbf{J}_A^+ (\dot{\mathbf{x}}_d + \mathbf{K} \mathbf{e}), \quad (3.52)$$

where \mathbf{J}_A^+ is the pseudo-inverse of the analytical Jacobian, \mathbf{K} is a positive definite gain matrix and $\mathbf{e} = \mathbf{x}_d - \mathbf{x}_m$. The problem of the previous control law is exactly the assumption of absence of representation singularities, which can be a problem for the orientation part of the rigid motion. Since widely used parametrizations for orientations (e.g., Euler angles) suffer from representation singularities, a common approach is to resort to the geometric Jacobian. Also, even when such non-singular representations are used, as quaternions or rotation matrices, the geometric Jacobian is often used. This happens because the geometric Jacobian can be easily obtained and the relation between the quaternion (or rotation matrix) derivative and the angular velocity is also obtained

3. As defined by Siciliano et al. (2009), and assuming that the orientation is represented by a vector \mathbf{r} , singularities in the representation mean that there exist angular velocities which cannot be expressed by means of $\dot{\mathbf{r}}$. This is the case of Euler angles, for example.

in a straightforward manner—in the case of quaternions, the quaternion propagation equation (A.7) can be used to obtain this relation.

In this thesis, a more unified approach is proposed. Since the robot can be completely modeled by dual quaternions, as shown in chapter (2), the control law (3.52) can be written as⁴

$$\dot{\theta} = J_{\underline{x}}^+ (\text{vec } \dot{\underline{x}}_d + \mathbf{K}\mathbf{e}), \quad (3.53)$$

where $J_{\underline{x}}^+$ is the pseudo-inverse of the dual quaternion Jacobian, \underline{x}_d and \underline{x}_m are the desired and measured dual positions, and $\mathbf{e} = \text{vec}(\underline{x}_d - \underline{x}_m)$. In this manner, the same parametrization used to model the robot is used to control it, without any problem of representation singularities.

For simplicity, the desired generalized speed $\dot{\underline{x}}_d$ of the control law (3.53) can be set to zero, so the problem reduces to a set-point tracking. Considering a discrete controller, 3.53 becomes

$$\frac{\theta_k - \theta_{k-1}}{T} = J_{\underline{x}}^+ \mathbf{K}\mathbf{e}, \quad (3.54)$$

where T is the sampling period. Since both sampling time and matrix gain are constant, 3.54 reduces to

$$\theta_k = \theta_{k-1} + J_{\underline{x}}^+ \mathbf{K}\mathbf{e}, \quad (3.55)$$

because the matrix gain can be tuned to take into account the sampling time.

Example 3.2. Consider one KUKA LWR 4 model with standard D-H parameters given by table 8. The FKM and dual quaternion Jacobian are obtained by using the modeling equations introduced in the previous chapter. More specifically, the FKM is obtained by using (2.20) with (2.22), whereas the dual quaternion Jacobian is obtained by using (2.30) with (2.34). The gain matrix was set as

$$\mathbf{K} = \lambda \mathbf{I}_8,$$

which is equivalent to having a scalar gain.

For the initial and final configurations of figure 22, the error between the desired and measured dual quaternions is shown in figure 23 for different gain values. Although larger gain values imply faster convergence, they must be upper-bounded in discrete systems to ensure stability (Siciliano et al., 2009). The trajectories in Cartesian space corresponding to each gain are shown in figure 24. Despite the fact that, in this example, the trajectories tended to be shorter for larger gain values, no straightforward conclusion can be drawn about the correlation between gain value and trajectory length. If, on the one hand, although this analysis would be simpler (but not necessarily simple) if only translations were considered, rotations, on the other hand, impose some difficulties in the definition of length (Kuffner, 2004).

The advantage of using the dual quaternion kinematic control law (3.53) is that it can be readily applied to the cooperative variables; that is, it can be applied to a stacked vector containing the absolute dual position \underline{x}_a and the relative dual position $\underline{x}_r^{E_2}$. For

4. This control law was investigated in collaboration with Hoang-Lan Pham (Pham et al., 2010). Whereas the use of dual quaternions to model and control the robot in a unified manner was a result of my work—under the supervision of Prof. Philippe Fraisse—H-L. Pham showed the stability for this control law and extensively tested it on an Adept Viper s850 robot.

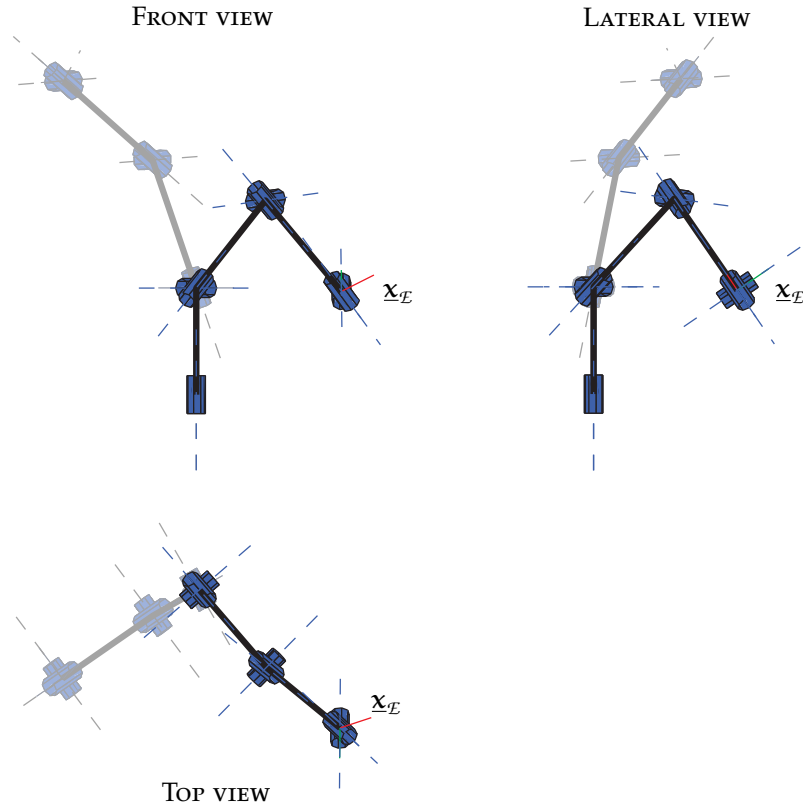


Figure 22: Initial and final configurations for the KUKA arm of example 3.2.

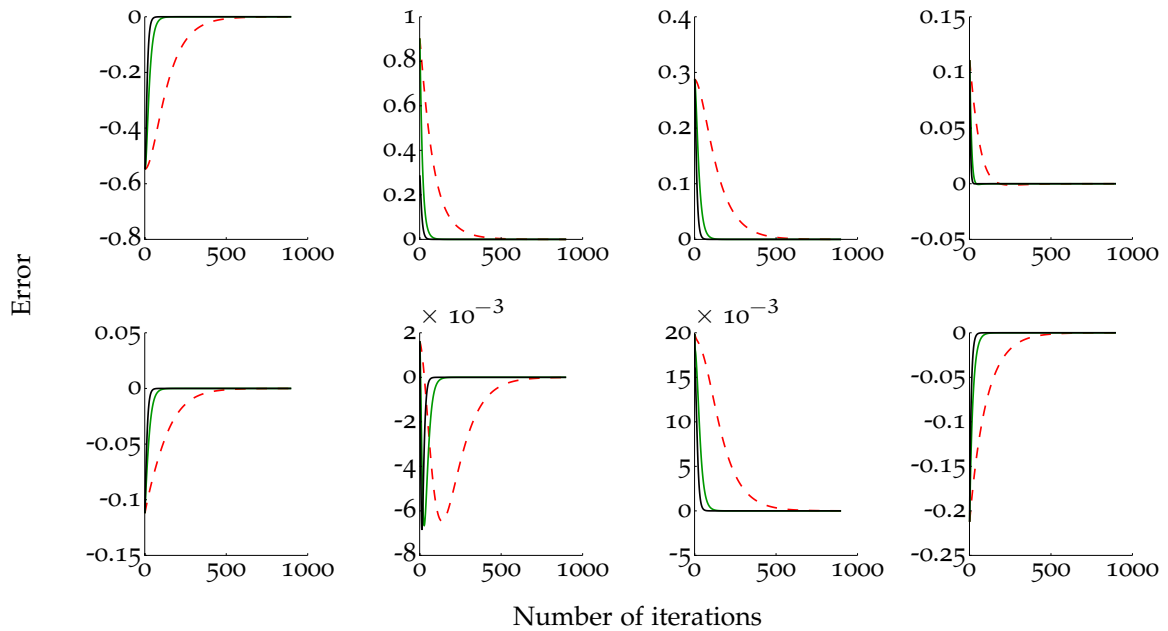


Figure 23: Error of each dual quaternion coefficient of example 3.2. The first coefficient is shown in the top left plot, and the others follow in clockwise direction. The dashed red, solid green, and solid black curves correspond to gains $\lambda = 0.01$, $\lambda = 0.05$, and $\lambda = 0.1$, respectively.

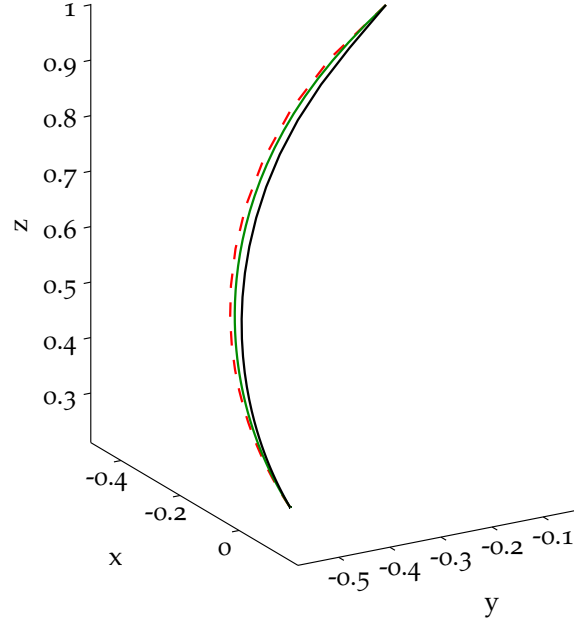


Figure 24: Trajectory executed by the KUKA arm of example 3.2. The dashed red, solid green, and solid black curves correspond to gains $\lambda = 0.01$, $\lambda = 0.05$, and $\lambda = 0.1$, respectively.

example, using an augmented Jacobian, the cooperative variables can be controlled according to

$$\boldsymbol{\theta}_{R,k} = \boldsymbol{\theta}_{R,k-1} + \mathbf{J}_{\text{task}}^+ \lambda \mathbf{e}_{\text{task}}, \quad (3.56)$$

where

$$\mathbf{J}_{\text{task}} = \begin{bmatrix} \mathbf{J}_{\underline{\mathbf{x}}_a} \\ \mathbf{J}_{\underline{\mathbf{x}}_r^{E_2}} \end{bmatrix}, \quad \mathbf{e}_{\text{task}} = \begin{bmatrix} \text{vec}(\underline{\mathbf{x}}_{a,d} - \underline{\mathbf{x}}_{a,m}) \\ \text{vec}(\underline{\mathbf{x}}_{r,d}^{E_2} - \underline{\mathbf{x}}_{r,m}^{E_2}) \end{bmatrix}.$$

The control law (3.56) considers that both variables have the same priority; that is, the control law will minimize the error \mathbf{e}_{task} , taking into account both cooperative variables at the same time. In some situations, however, one cooperative variable can have higher priority and should not be disturbed by the second one. In this case, the second cooperative variable should be controlled in the nullspace of the first one. Consider the task Jacobian $\mathbf{J}_{\text{task}i}$ and error $\mathbf{e}_{\text{task}i}$ associated with each cooperative variable i . The control law (3.56) becomes (Ligeois, 1977; Siciliano & Slotine, 1991)

$$\boldsymbol{\theta}_{R,k} = \boldsymbol{\theta}_{R,k-1} + \mathbf{J}_{\text{task}1}^+ \lambda_1 \mathbf{e}_{\text{task}1} + \mathbf{P}_{\text{task}1} \mathbf{J}_{\text{task}2}^+ \lambda_2 \mathbf{e}_{\text{task}2}, \quad (3.57)$$

where $\mathbf{P}_{\text{task}1} = (\mathbf{I} - \mathbf{J}_{\text{task}1}^+ \mathbf{J}_{\text{task}1})$ is a projector onto the nullspace of $\mathbf{J}_{\text{task}1}$, and the strictly positive scalars λ_1 and λ_2 are gains associated with each task. The advantage of this control law is that the second task will be executed without disturbing the first one.

Example 3.3. Let a task be defined by the cooperative variables $\underline{\mathbf{x}}_a$ and $\underline{\mathbf{x}}_r^{E_2}$, where the relative dual position has highest priority. The prioritized control law is given by

$$\boldsymbol{\theta}_{R,k} = \boldsymbol{\theta}_{R,k-1} + \mathbf{J}_{\underline{\mathbf{x}}_r^{E_2}}^+ \lambda_r (\underline{\mathbf{x}}_{r,d}^{E_2} - \underline{\mathbf{x}}_{r,m}^{E_2}) + (\mathbf{I} - \mathbf{J}_{\underline{\mathbf{x}}_r^{E_2}}^+ \mathbf{J}_{\underline{\mathbf{x}}_r^{E_2}}) \mathbf{J}_{\underline{\mathbf{x}}_a}^+ \lambda_a (\underline{\mathbf{x}}_{a,d} - \underline{\mathbf{x}}_{a,m}),$$

where λ_r and λ_a are scalar gains associated with the relative and absolute dual positions, and the subscripts d and m stand for the desired and measured values, respectively.

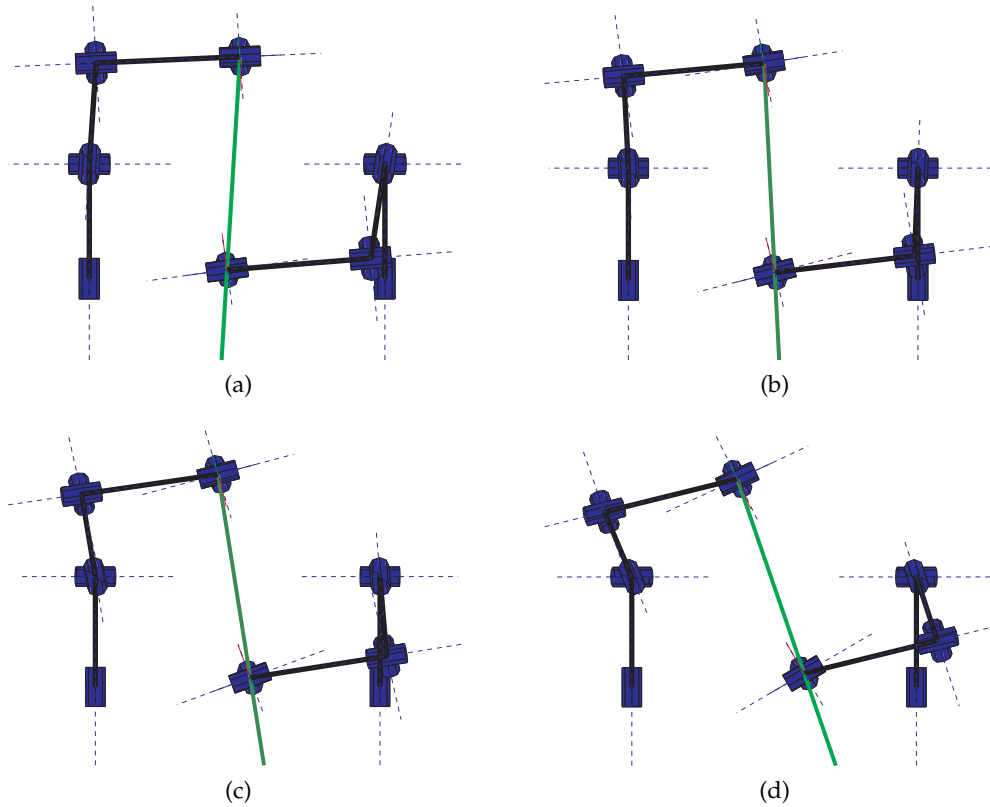


Figure 25: Sequence (a)–(d) showing two KUKA LWR 4 manipulating a broom in one direction.

Example 3.4. Recall the task of manipulating a broom of example 3.1. The relative dual position must be kept constant and the absolute dual position must change according to $\underline{x}_{\text{motion}}$. Assuming the initial configuration shown in figure 25, let the required motion be a rotation of $\pi/8$ rad around the axis pointing out from the torso. If, for example, this axis is the y -axis, then

$$\underline{x}_{\text{motion}} = \cos\left(\frac{\pi}{16}\right) + \hat{j} \sin\left(\frac{\pi}{16}\right).$$

The final configuration is shown in figure 25.

Example 3.5. Both control laws (3.56) and 3.57 can be used to achieve this motion. However, in the prioritized control law, one must choose which task has the highest priority. If the robot is not equipped with force sensors, the relative dual position should have the highest priority, because deviations in the relative dual position will cause corresponding internal wrenches that can damage the robot or the object.

Whereas the reference signal for the relative dual position is constant, the reference signal for the absolute dual position corresponds to a complete movement cycle; that is, \underline{x}_{a_0} to $\underline{x}_{a_0} \otimes \underline{x}_{\text{motion}}$ followed by $\underline{x}_{a_0} \otimes \underline{x}_{\text{motion}}$ to \underline{x}_{a_0} .

For the same scalar gains in both control laws ($\lambda = 0.5$), the coefficients of the relative dual position are shown in figure 26. The reference signal (*dashed red curve*) is constant during the whole movement cycle. However, the measured signals of the augmented Jacobian and prioritized control laws show transients corresponding to a change in the reference of the absolute dual position. As expected, the prioritized control law provides

fewer deviations from the constant relative dual position reference signal, because in this control law the relative dual position has higher priority and should not be disturbed by the second task. On the other hand, the convergence of the absolute dual position is slower in the prioritized scheme, as shown in figure 27. This is a trade-off that depends on the task requirements.

Still considering example 3.4, in terms of physical quantities, the transients of the relative dual position are shown in figures 28 and 29. For all physical quantities, the prioritized control law led to smaller transients, corresponding exactly to the behavior already shown in figure 26. With respect to the orientation angle, the angle between the arms deviates about 0.4 degree from the desired value in one of the transients, when the augmented Jacobian control law is considered. On the other hand, when the prioritized control law is used, this angle deviates up to about 0.025 degree from the desired value. In the case of a real setup—taking into consideration only the angle between the hands—both controllers would probably suffice to manipulate the broom, since the broom's stick is normally made of wood or plastic. These materials are relatively soft, and such small deviations probably would not be an ultimate concern.

If the relative translation is considered, the situation changes drastically. In the augmented Jacobian control law, although the peaks in the y - and z -axes correspond to deviations of less than one millimeter, in the x -axis there is one peak of a large deviation of 2.7 cm, which could cause a considerable internal wrench, although in a very short time. On the other hand, if the prioritized control law is considered, the deviations from the desired position are quite small; the largest case is about one millimeter in the x -axis, which could still be accepted for this kind of manipulation, even without force control.

It is important to underline, however, that force control should be used in more realistic scenarios, because inaccurate models and other disturbances can lead to high internal wrenches, capable of causing damage to the object or the robot.

3.5 CONTROL PRIMITIVES

In order to have a more flexible formalism and ease the definition of cooperative tasks, the goal of this section is to develop control primitives that can be used—standalone or combined—as the input of generic kinematic controllers. Recall the previous section, where the cooperative variables were used to coordinate the arms in the task of manipulating a broom. Two strategies were adopted: in the first one, the cooperative variables were controlled with the same priority; in an alternative strategy, they were controlled in a prioritized scheme, and the relative dual position had higher priority than the absolute dual position. Sometimes, however, the task can be more easily defined in terms of different primitives; that is, instead of describing it by relative and absolute dual positions, other primitives like distances, translations, or only rotations could be sufficient for the description. For instance, consider the task of carrying a large box, as illustrated

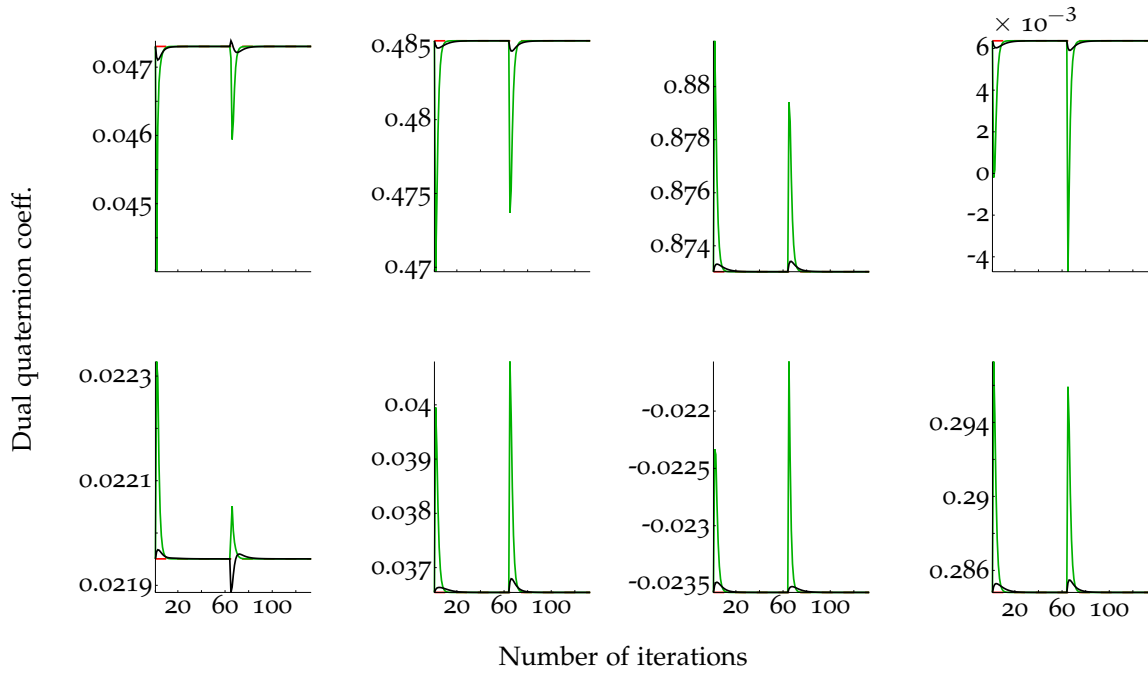


Figure 26: Coefficients of the relative dual quaternion of example 3.4. The reference signal (*dashed red*) is constant along the whole movement cycle. The measured signals of the augmented Jacobian control law (*solid green*) and the prioritized control law (*solid black*) show a transient corresponding to a change in the reference of the absolute dual position.

in figure 30. In order to drop the box, it suffices to augment the relative distance between the hands. If the task consists of rotating one face of a Rubik's cube (fig. 31), the relative dual position should be controlled. To open a bottle, on the other hand, not only the relative dual position should be controlled, but also the absolute orientation to avoid spilling the bottle's contents. Finally, some tasks require both relative and absolute dual positions to be controlled, like the manipulation of a steering wheel (fig. 33).

The main point of the previous discussion is that different tasks require different primitives. Hence, in order to have a more flexible formalism, the low level control laws should not be highly dependent on the type of primitive being controlled. In this way, higher level decision layers could choose the most appropriate primitive set to be controlled for a specific task. Furthermore, such primitives should have appealing geometrical meaning in order to be easily defined by a human designer. Finally, if only a minimal set is controlled, some DOF can be freed, if possible, to control secondary tasks.

The general control law (which can be directly extended to a prioritized framework (Siciliano & Slotine, 1991)) is

$$\dot{\theta} = \mathbf{J}_{\text{task}}^+ \mathbf{K} (\mathbf{u}_d - \mathbf{u}_m), \quad (3.58)$$

where θ is the vector of joint variables, $\mathbf{J}_{\text{task}}^+$ is the generalized pseudo-inverse corresponding to the task Jacobian (which varies according to the control primitives), \mathbf{K} is a positive definite gain matrix, and \mathbf{u}_d , \mathbf{u}_m are the desired and measured primitives, respectively.

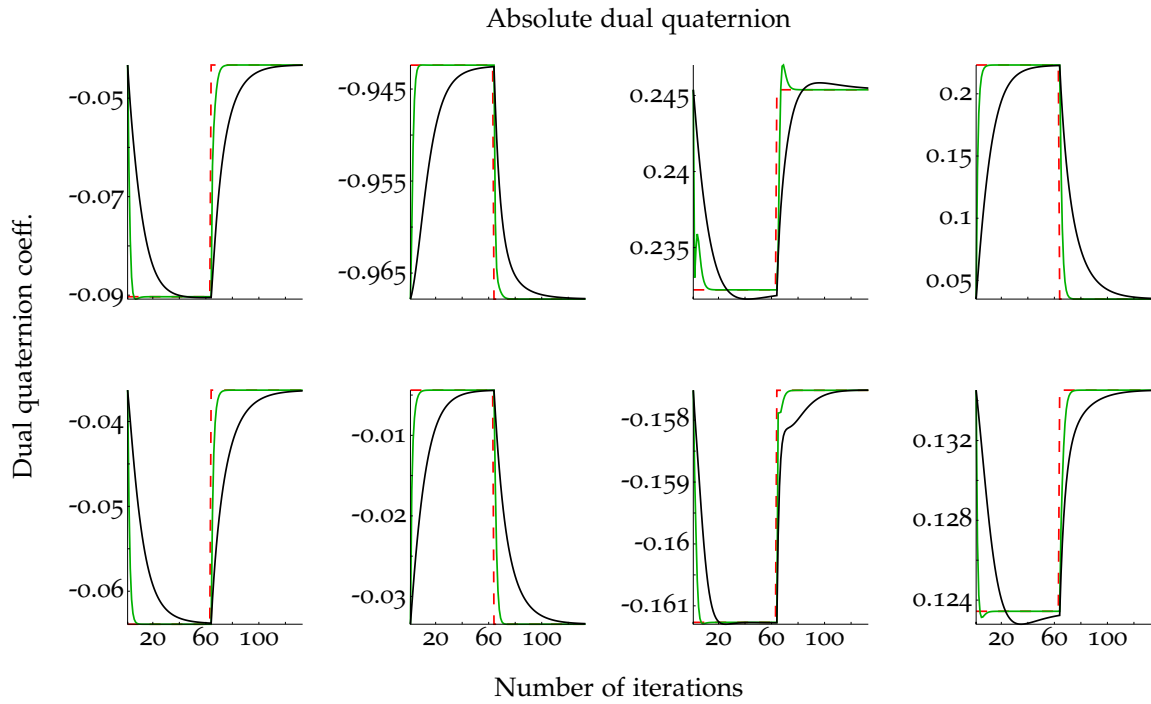


Figure 27: Coefficients of the absolute dual quaternion of example 3.4. The reference signal is represented by the *dashed red* curve, whereas the measured signals provided by the augmented Jacobian control law and the prioritized control law are represented by *solid green* and *solid black* curves, respectively.

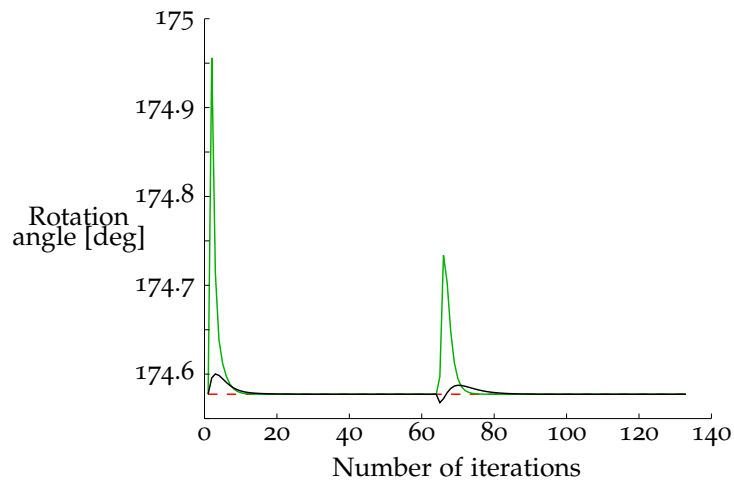


Figure 28: Angle between the robotic hands in the task of manipulating a broom (see example 3.4). The reference signal is represented by the *dashed-red* curve, whereas the measured signals provided by the augmented Jacobian control law and the prioritized control law are represented by *solid green* and *solid black* curves, respectively.

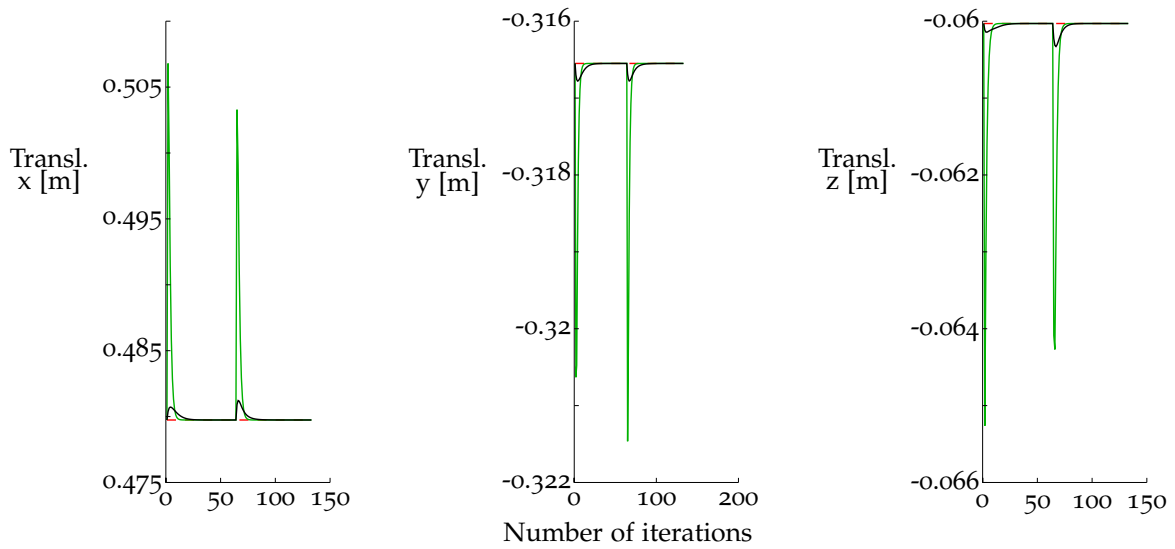


Figure 29: Relative translation between the robotic hands in the task of manipulating a broom (see example 3.4). The reference signal is represented by the *dashed red* curve, whereas the measured signals provided by the augmented Jacobian control law and the prioritized control law are represented by *solid green* and *solid black* curves, respectively.

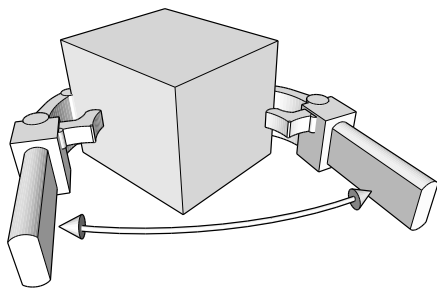


Figure 30: Two-arm manipulation of a large box: in order to drop the box, it suffices to augment the distance between the hands.

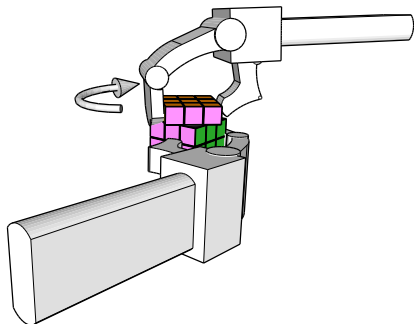


Figure 31: Manipulation of Rubik's cube. Only the relative rotation need to be controlled.

Figure 32: Opening a bottle. Not only the relative position should be controlled, but also the absolute orientation to avoid spilling the bottle's contents.

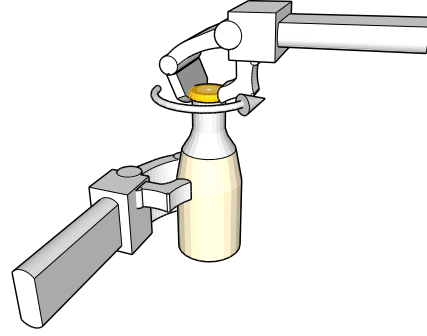
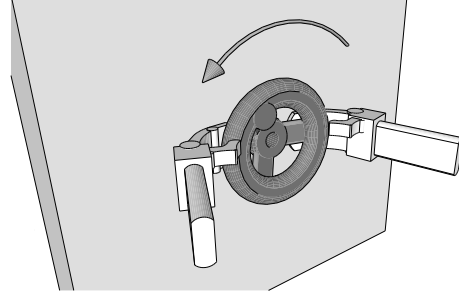


Figure 33: Turning a steering wheel. All cooperative variables must be controlled.



In the cooperative dual task-space, the following basic primitives can be used: dual position, orientation, position, and distance. In this case, the goal is to find a task Jacobian J_{task} relating the first time derivative of each primitive with the first time derivative of the joint variables. Excepting the Jacobian relating the distance with the joint variables, the rest were derived in the last chapter and preceding sections.

Proposition 3.3. Let \mathbf{p} be a translation quaternion such that

$$\text{vec } \dot{\mathbf{p}} = J_{\mathbf{p}} \dot{\boldsymbol{\theta}}.$$

The distance Jacobian; that is, the Jacobian that satisfies

$$\dot{c} = J_{\mathbf{d}} \dot{\boldsymbol{\theta}},$$

where the scalar function $c = f(\mathbf{p})$ is given by

$$J_{\mathbf{d}} = \nabla c \cdot J_{\mathbf{p}},$$

and ∇c is the gradient of c .

Proof. From lemma 3.2, the position is a function of the vector of joint variables; that is, $\mathbf{p} = \mathbf{g}(\boldsymbol{\theta})$. Hence, $c = (f \circ \mathbf{g})(\boldsymbol{\theta})$. Taking the first derivative and applying the chain rule gives

$$\begin{aligned} \dot{c} &= \frac{\partial c}{\partial \text{vec } \mathbf{p}} \cdot \frac{\partial \text{vec } \mathbf{p}}{\partial \boldsymbol{\theta}} \cdot \frac{d\boldsymbol{\theta}}{dt} \\ &= \nabla c J_{\mathbf{p}} \dot{\boldsymbol{\theta}} \\ &= J_{\mathbf{d}} \dot{\boldsymbol{\theta}}. \end{aligned}$$

□

From the previous proposition, the problem is finding a suitable scalar function $c = f(\mathbf{p})$. The most obvious choice would be $c \triangleq \|\mathbf{p}\|$, but the gradient of this function is singular for $\mathbf{p} = \mathbf{0}$, because

$$\begin{aligned}\nabla c &= \nabla (p_x^2 + p_y^2 + p_z^2)^{1/2} \\ &= \frac{(\text{vec } \mathbf{p})^\top}{(p_x^2 + p_y^2 + p_z^2)^{1/2}}.\end{aligned}$$

Thus, a simple choice is $c \triangleq \|\mathbf{p}\|^2$, such that

$$\dot{c} = \underbrace{2(\text{vec } \mathbf{p})^\top}_{J_d} \mathbf{J}_p \dot{\boldsymbol{\theta}}. \quad (3.59)$$

Remark 3.3. The Jacobian of each primitive is a function of the Jacobian corresponding to the dual position. More specifically, for a dual quaternion $\underline{\mathbf{x}} = \underline{\mathbf{f}}(\boldsymbol{\theta})$, the dual quaternion Jacobian satisfies the relation

$$\text{vec } \dot{\underline{\mathbf{x}}} = \mathbf{J}_x \dot{\boldsymbol{\theta}}.$$

The orientation Jacobian corresponds to the four upper rows of \mathbf{J}_x . The translation Jacobian satisfies the relation

$$\text{vec } \dot{\mathbf{p}} = \mathbf{J}_p \dot{\boldsymbol{\theta}},$$

where \mathbf{J}_p is given by (3.14), which is obtained from the dual quaternion Jacobian. Finally, as shown by (3.59), the distance Jacobian is a function of the translation Jacobian.

Remark 3.4. From remark 3.3, all primitives are functions of dual positions. Hence, these primitives can be used in one arm control, and also in two-arm coordination. For example, to control the dual position $\underline{\mathbf{x}}_E$ of a manipulator's end-effector, the dual quaternion Jacobian of the manipulator is used. If the position \mathbf{p}_E or distance $c = \|\mathbf{p}_E\|^2$ is controlled, it will be with respect to the reference frame and all related Jacobians will be functions of the dual quaternion Jacobian of that manipulator. On the other hand, the relative dual quaternion Jacobian is used to control the relative dual position in a two-arm robot. In this case, the control of the relative position $\mathbf{p}_r^{E_2}$ or relative distance $c = \|\mathbf{p}_r^{E_2}\|^2$ will imply the control of the position or distance of one arm with respect to the other. Also, the respective task Jacobians will be functions of the relative dual quaternion Jacobian of the two-arm system.

The primitives are summarized in table 2.

Example 3.6. Consider the task of positioning a screwdriver. The tip of the screwdriver must coincide with the head of the screw. Thus, the position must be specified. Furthermore, since the screwdriver must be aligned with the screw, orientation must also be specified. Hence, there is a need for controlling the dual position, as illustrated in figure 34.

Example 3.7. Consider the manipulation of a flashlight illustrated in figure 35. Typically, the position of the flashlight is not as important as the orientation, and thus orientation control suffices to direct the light to the desired spot.

Table 2: Summary of control primitives and correspondent task Jacobians.

	Control primitive \mathbf{u}	Task Jacobian \mathbf{J}_{task}
Dual position	$\underline{\mathbf{x}} = \mathbf{r} + \varepsilon (1/2) \mathbf{p} \mathbf{r}$	$\mathbf{J}_{\underline{\mathbf{x}}}$
Orientation	\mathbf{r}	$\mathbf{J}_{\mathcal{P}(\underline{\mathbf{x}})}$
Cartesian position	\mathbf{p}	$2\bar{\mathbf{H}}(\mathcal{P}(\underline{\mathbf{x}}^*)) \mathbf{J}_{\mathcal{D}(\underline{\mathbf{x}})} + 2\bar{\mathbf{H}}^+(\mathcal{D}(\underline{\mathbf{x}})) \mathbf{J}_{\mathcal{P}(\underline{\mathbf{x}})}^*$
Distance	$\ \mathbf{p}\ ^2$	$2(\text{vec } \mathbf{p})^T \mathbf{J}_{\mathbf{p}}$

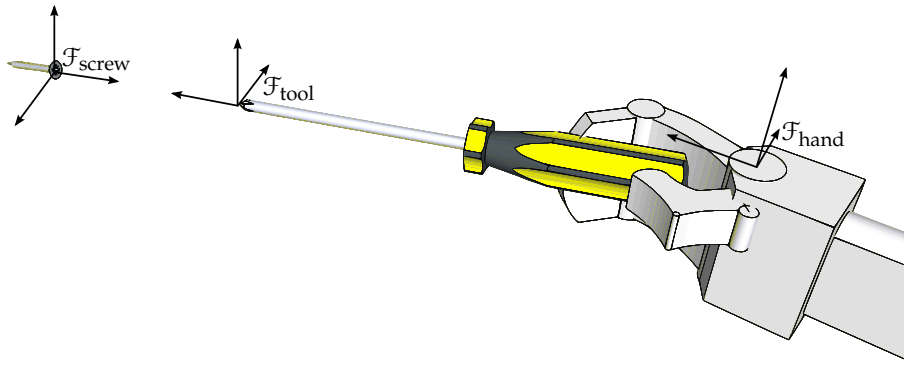


Figure 34: Positioning a screwdriver: both position and orientation must be controlled.

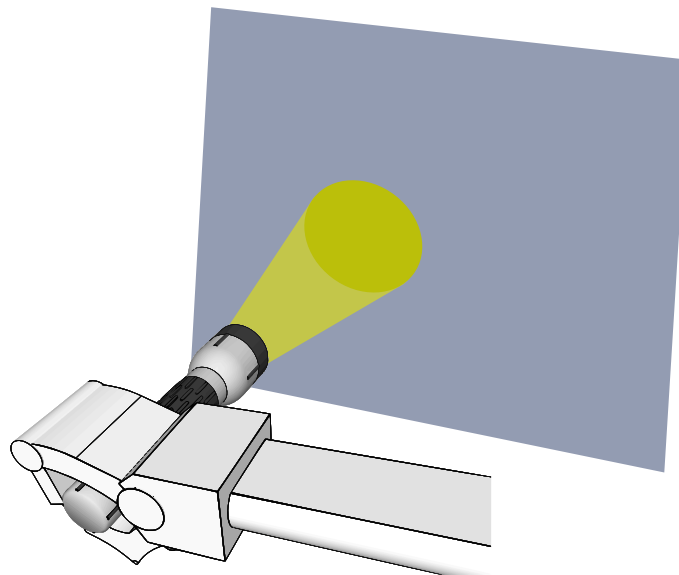


Figure 35: Manipulation of a flashlight. It is necessary to control only the orientation.

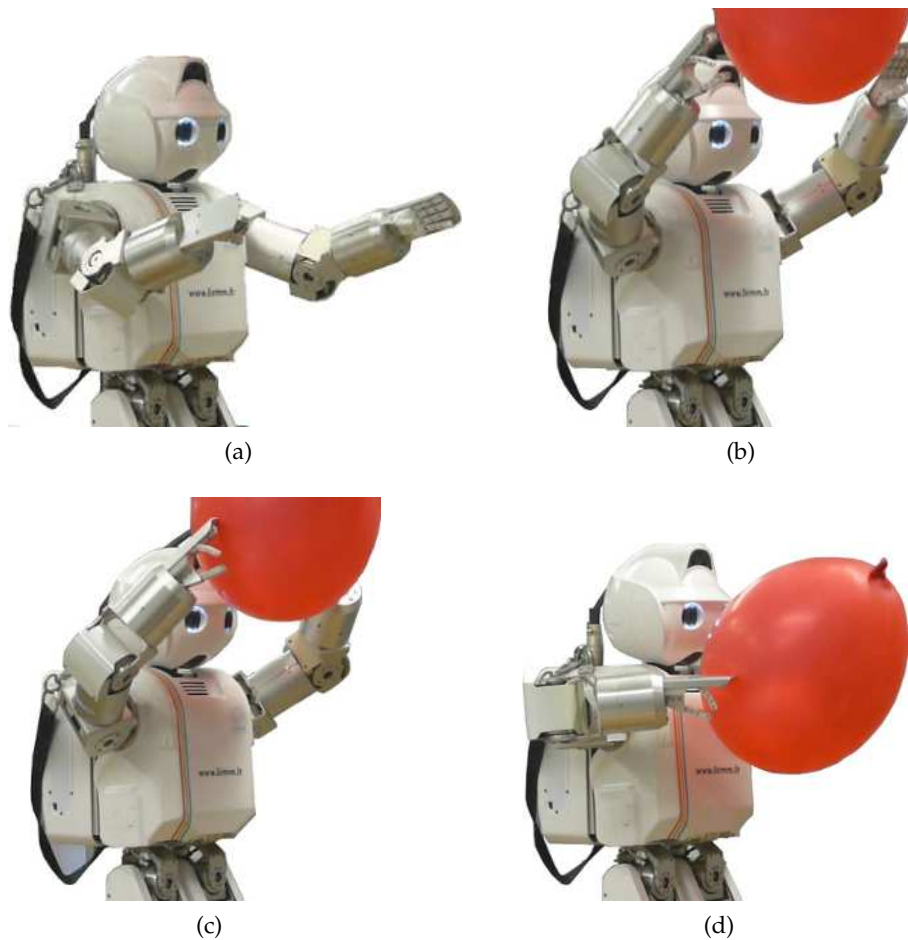


Figure 36: Grabbing a balloon. (a) Initial configuration. (b) Full dual position control: rough reaching phase. (c) Relative distance control. (d) Full dual position control: final configuration.

The following examples show how the control primitives can be used to perform more complex tasks. These examples were implemented in the Hoap-3 humanoid robot by using the general control law (3.58).

Example 3.8. Consider the task of grabbing a balloon. The task can be subdivided in the following manner. Given the initial configuration shown in figure 36a, first both cooperative variables must be controlled, as the current coordinate system of the balloon is used as the reference for the desired absolute dual position, and the relative dual position must be roughly specified so that the robot can reach the balloon with a suitable pose. The final configuration after this rough reaching phase is illustrated in figure 36b. The relative distance is then controlled and the robot closes the arms and holds the balloon, as illustrated in figure 36c. Finally, in order to move the object while holding it, the relative dual position is maintained constant and the absolute dual position of the object is changed, with the final result illustrated in figure 36d.

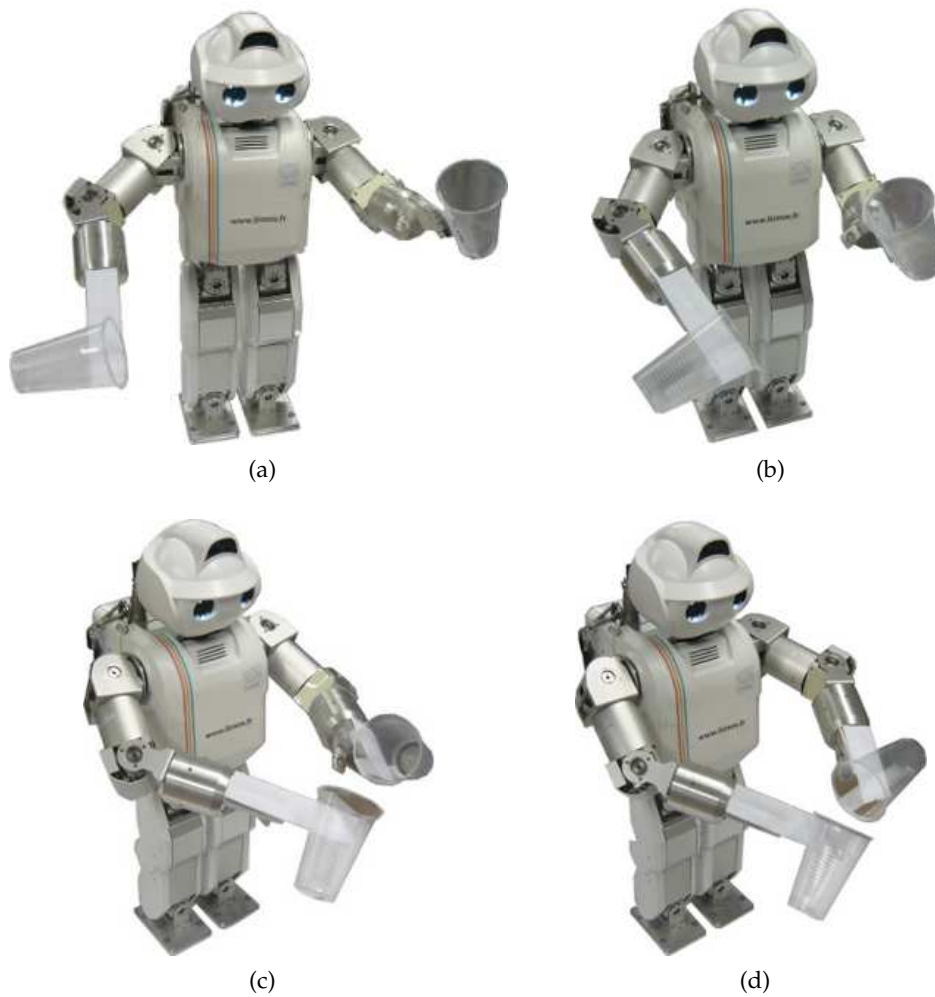


Figure 37: Pouring water. (a) Initial configuration of the hands: start of the relative Cartesian position control. (b) Relative Cartesian position control: intermediate configuration. (c) Start of the relative dual position control. (d) End configuration of the relative dual position control.

Example 3.9. Consider the task of pouring water. In order to define this task, the absolute position can be neglected, since the only condition for an effective coordination is that it must occur inside the workspace of the two arms. Considering the initial configuration of figure 37a, the relative position is controlled to put one hand closer to the other, as shown in figures 37b and 37c. Next, the relative dual position is controlled to maintain the distance between the hands constant and to change only their relative orientation, as shown in figure 37d.

In examples 3.8 and 3.9, the tasks were defined by hand. Because the control primitives were used, these definitions were more intuitive. For example, consider the example where the robot must close the arms to grab the balloon. Instead of thinking about the position of one hand with respect to the other, it is easier to close the arms taking into consideration the distance between them. Hence, the relative distance should be

controlled, not the relative position. Also, because the final orientation of the hands does not matter for this task, the relative orientation is not specified either.

Another advantage of using control primitives is that they provide some flexibility to higher level planners and controllers. For example, recall the example of pouring water. In order to bring one hand closer to the other, only the relative position was controlled. Although a prioritized strategy was not implemented in the example, controlling only the relative position freed some DOF that could be used for secondary tasks. On the other hand, the lack of control in the orientation could be quite dangerous to the robot, since the water could spill, but the robot is not waterproof. In this case, both relative and absolute orientations should have been controlled to ensure safety in the execution of the task. The problem is that nine-DOF are required to control both relative dual position and absolute orientation, but the robot has only eight-DOF. A possible solution, not implemented in example 3.9, would be to implement a higher level system that could observe the task and change it accordingly to ensure a safe execution. In this manner, only the relative position would be initially controlled, but if the robot realized that water could spill, the orientations could be controlled—or the task could be halted—in order to ensure safety.

3.6 CONCLUSION

This chapter presented a new point of view in the symmetric cooperative task-space. This formalism was revisited and then rewritten in terms of dual quaternions, which enabled a unified treatment of the cooperative variables; namely, dual positions, twists, and wrenches. All these quantities were expressed both at the tips of the so-called “virtual sticks,” and at the end-effectors. The solution for the case of two-arm manipulation of a firmly grasped object was shown, as well for the case of two-arm coordination. The resulting theory agrees with the previous formulations of the symmetric cooperative task-space (Uchiyama & Dauchez, 1988; Chiacchio et al., 1996; Caccavale et al., 2000).

Besides the representation in terms of dual positions, twists and wrenches, the cooperative dual task-space was also represented by an alternative formulation by using the concept of generalized speeds and generalized forces, and these quantities were related to twists and wrenches by means of the generalized Jacobian. Although out of the scope of this thesis, one could investigate six-axis impedance controllers based entirely on dual positions and their generalized counterparts, potentially leading to geometric consistent controllers. A good starting point for such development is presented by Caccavale et al. (1999, 2008).

Because dual quaternions do not exhibit representation singularities, stable control laws were developed by using the dual quaternion Jacobian and dual quaternions were used directly as the input for the kinematic controllers. However, alternatives for this approach could be developed by using nonlinear techniques like, for example, logarithmic feedback (Han et al., 2008). Furthermore, the differential kinematics for the two-arm system was derived in terms of the cooperative Jacobians. In this manner, the control laws used for one-arm control were directly applied to control the cooperative variables. One drawback of these control laws, which is intrinsic to any control law based on matrix inversions, is that high velocities appear whenever the Jacobian loses rank—the

well-known kinematic singularities. However, well-known techniques can be used to mitigate this effect as, for example, the use of damped least-square inverses (Wampler, 1986; Chiaverini, 1997). Another important drawback of the kinematic control techniques proposed in this thesis is that the dynamics of the two-arm system are neglected.

A set of control primitives and the respective Jacobians were developed in order to simplify the definition of two-arm manipulation tasks. As the analytical Jacobians were used with representations that do not suffer from singularities, all primitives could be controlled by using the same control law. This can be useful if higher level controllers and planners are used, because they can change the controlled primitives according to the task requirements (Mansard & Chaumette, 2007).

Simulations were performed by using the model of two KUKA-LWR arms, whereas experiments were performed on a real Hoap-3 robot to validate the proposed techniques. Despite the fact that wrenches and twists were completely modeled, force control was out of the scope of this thesis and hence postponed to future works. However, since the representation introduced in this chapter agrees with the literature, the state-of-the-art force controllers applied to previous formulations of the symmetric cooperative task-space should be easily adapted to the cooperative dual task-space.

Also as future work, the equivalences between synchronized control and the cooperative dual task-space should be investigated more thoroughly. This would result in a still more general formalism capable of describing rich manipulation tasks.

The next two chapters present two extensions of the cooperative dual task-space. The first is related to whole-body manipulation applied to mobile manipulators, and the second extension is applied to human-robot collaboration.

TWO-ARM MOBILE MANIPULATION: THE COOPERATIVE DUAL TASK-SPACE APPROACH

Over the past few years, the scientific community has seen a sudden development of several branches of robotics. Robots are no longer constrained to structured environments and they are now being used almost everywhere, from outer space and underwater to air and off-road environments. One should expect that robots will soon have the useful capabilities of interacting with humans and within common human environments.

One of the robotics fields that has received particular attention is assistive robotics. In addition to satisfying the usual dream of having a personal maid for washing the dishes or cleaning the house, assistant robots can be useful in helping the elderly or people with disabilities. However, this implies an extra challenge, since the safety level must be increased for the human-robot interaction. Furthermore, evidence indicates that the human-robot interaction is improved when the robot has an anthropomorphic appearance (Kemp et al., 2008). Consequently, humanoids and anthropomorphic mobile manipulators have been used to this end, leading to three particularly great challenges: simultaneous control of several degrees of freedom (DOF), two-arm manipulation, and balance control in humanoids.

Whereas balance control can be avoided when using a mobile manipulator, a high number of DOF have to be controlled in order to accomplish complex tasks, and thus several frameworks have been developed in the last thirty years to tackle redundant systems. Starting with the works of Liegeois (1977), several authors have investigated the problem of prioritized task control to deal with highly redundant systems (Nakamura et al., 1987; Siciliano & Slotine, 1991; Chiaverini, 1997; Kanoun, 2009; Mansard, 2006). More focused on whole-body control for humanoid robots, Sentis (2007) developed a framework based on the operational space that takes into account the balance and contact stability constraints to synthesize dynamic behaviors.

In order to develop useful robotic assistants, whole-body motion turns out to be of central importance in manipulation tasks. Indeed, assistant robots as well as human-robot interaction require complex movements that are human-centered, and a high number of these human-centered movements are related to manipulation. Therefore, robotic assistants should be designed to perform movements that are “manipulation-centered.” For example, if the assistant robot is helping a person to carry a large box to another room, the robot will have to displace itself while holding the box.

Several researchers have achieved expressive results in whole-body motion for mobile manipulators. Earlier works from Carriker et al. (1989) considered the task of motion planning as an optimization problem. However, they did not develop the coordination between the manipulator and the mobile base. Indeed, once the mobile base delivered the manipulator to the desired task-space, the latter would behave as if it were stationary, and hence the mobile base would not be used.

Seraji (1993) proposed a simple kinematic controller based on the damped least-squares inversion method to coordinate the movement of a one-arm mobile manipulator. The forward kinematic model (FKM) was described in terms of the joint variables of both manipulator and mobile base, and the differential FKM was obtained by grouping the respective Jacobians into an augmented Jacobian matrix.

Yamamoto & Yun (1994) designed the coordination of a one-arm mobile manipulator, where the manipulator was considered as a passive device without dynamics, and the mobile base was controlled in order to maximize the manipulator manipulability.

Perrier et al. (1997, 1998); Perrier (1998) developed a method to achieve the coordinated behavior of a mobile manipulator. Based on the system variables, they found the equations corresponding to elementary displacements for the robot and then applied the linearized version of these equations to build elementary displacement operators. These operators were represented both in homogeneous transformation matrices and dual quaternions, but they concluded that both would lead to acceptable results. However, they found that dual quaternions were less sensitive with respect to the changing of units.

Khatib and collaborators proposed a decentralized scheme for the coordination of two mobile manipulators, and the cooperative manipulation was based on the augmented object and virtual linkage models (Khatib et al., 1996; Khatib, 1999; Khatib et al., 1999).

Andaluz et al. (2010) considered a one-arm mobile manipulator, and the whole-body coordination was achieved by means of a cascaded system. This system was composed of a kinematic controller that provided the velocity reference to a dynamic controller, and the latter used this reference to compensate for the dynamics of the system.

This chapter focuses in two-arm mobile manipulation; that is, tasks that require two arms and that are executed by mobile manipulators. This development is not only a natural extension of the techniques presented in the previous chapters, but also a complement to the techniques presented in the literature that consider mainly one-arm mobile manipulators. The first part of the chapter is devoted to the representation of whole-body manipulation, where the control inputs for body motion are the primitives related to the geometry of the task. Furthermore, although the development is biased toward whole-body manipulations involving the coordination between two arms, whole-body manipulations using a single arm can be regarded as a particular case. The second part presents a case study of a two-arm mobile manipulator with a nonholonomic base in a simple but illustrative task of pouring water for a moving person.

4.1 SERIALY COUPLED KINEMATIC STRUCTURE

Let us consider systems that can be described by rigid motions; for example, industrial manipulator robots, mobile bases, free flying robots, and so on. Let θ be the state vector of the system, and the final pose be described by $\underline{x} = \underline{f}(\theta)$. For example, consider a manipulator where θ corresponds to the joint variables, whereas in the case of a mobile robot the vector space is typically composed of the Cartesian coordinates x, y and the heading angle ϕ of the mobile robot; hence, $\theta = \{x, y, \phi\}$ (Fukao et al., 2000).

Consider different kinematic systems—each one described by a function $\underline{x}_i = \underline{f}_i(\theta_i)$ —which are coupled serially, as illustrated in figure 38. Changes in the state of previous

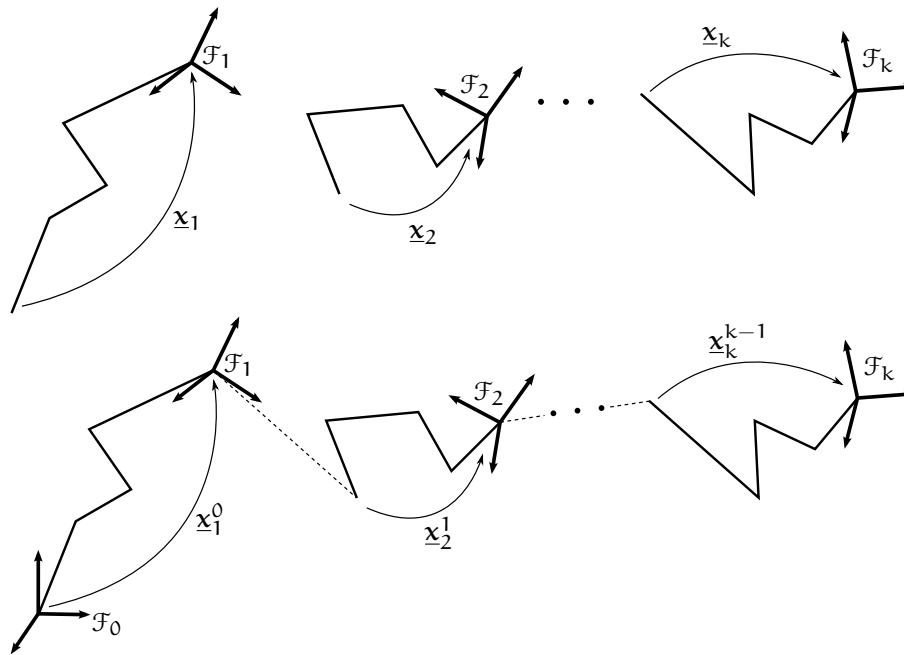


Figure 38: In the *top figure*, each kinematic subsystem is described by an intermediate rigid motion. When these subsystems are serially coupled (*bottom*), the rigid motion of one kinematic subsystem is given with respect to the previous one in the chain.

kinematic systems cause variations in all the rest of the chain; hence, the variation of the last frame in the chain will be a function of the variation of all previous frames. This idea is well known in the field of robot modeling, because a serial manipulator is composed of several rigid links attached serially, and each link is a kinematic structure.

Example 4.1. Consider a serial arm fixed on top of a mobile base. The configuration of the end-effector with respect to a global frame will be a function of both the position of the arm's joints and the configuration of the mobile base.

Frequently, the different parts of a complex robotic system are modeled separately. For example, when modeling a humanoid robot, one can divide it into legs, torso, arms, and head. After each part is modeled separately, they can be represented with respect to a common frame, and thus the complete model is obtained by considering the interaction of all individual parts.

The following corollary uses an argument similar to that of section 2.3.2 on page 38 to provide a systematic and direct methodology to serially couple separated kinematic models. In the example of a manipulator on top of a mobile base, this integration will enable the description of the mobile manipulator as a sole entity. Furthermore, even if more—and more complex—systems are added serially, they can be added to the final description by using the same methodology.

Corollary 4.1. Let a serial kinematic system be composed of a set of k coupled subsystems, each one described by the rigid transformations $\underline{x}_1, \underline{x}_2, \dots, \underline{x}_k$. Considering that \underline{x}_k^0 corresponds to

the transformation of the last frame \mathcal{F}_k with respect to the base frame \mathcal{F}_0 , the first derivative is given by

$$\text{vec } \dot{\underline{\mathbf{x}}}_k^0 = \sum_{i=0}^{k-1} \overset{+}{\mathbf{H}}(\underline{\mathbf{x}}_i^0) \bar{\mathbf{H}}(\underline{\mathbf{x}}_k^{i+1}) \text{vec } \dot{\underline{\mathbf{x}}}_{i+1}^i,$$

Proof. From (2.25),

$$\dot{\underline{\mathbf{x}}}_k^0 = \sum_{i=0}^{k-1} \underline{\mathbf{x}}_i^0 \dot{\underline{\mathbf{x}}}_{i+1}^i \underline{\mathbf{x}}_k^{i+1}.$$

Applying the Hamilton operators leads directly to

$$\text{vec } \dot{\underline{\mathbf{x}}}_k^0 = \sum_{i=0}^{k-1} \overset{+}{\mathbf{H}}(\underline{\mathbf{x}}_i^0) \bar{\mathbf{H}}(\underline{\mathbf{x}}_k^{i+1}) \text{vec } \dot{\underline{\mathbf{x}}}_{i+1}^i,$$

since $\overset{+}{\mathbf{H}}(\underline{\mathbf{x}}_i^i) = \bar{\mathbf{H}}(\underline{\mathbf{x}}_i^i) = \mathbf{I}_8$. □

Using corollary 4.1, the cooperative dual task-space can be generalized in order to take into account different kinematic structures attached to the two-arm system, such as a mobile base.

4.2 EXTENDED COOPERATIVE DUAL TASK-SPACE

Consider the situation where an assistant robot must pour a glass of water and hand it over to a person nearby. This task can be described in terms of the cooperative dual task-space variables, since the relative dual position $\underline{\mathbf{x}}_r$ represents the geometric relation between the arms—hence the relation between the glass and the bottle¹—and the absolute dual position $\underline{\mathbf{x}}_a$ describes the dual position of the two-arm system (or bottle-glass ensemble) with respect to a reference coordinate system. However, in the original formulation of the cooperative dual task-space, only the DOF corresponding to the two arms can be used to control $\underline{\mathbf{x}}_r$ and $\underline{\mathbf{x}}_a$. As a consequence, if the robot must pour the water into the glass and then hand it over to a person standing outside the robot's workspace the task cannot be accomplished, because the person is not reachable. On the other hand, if the arms are part of a humanoid or a mobile manipulator, the robot can use its legs or its mobile base to reach the person and hand over the glass of water. More specifically, the motion controllers should take into account all the available DOF in order to perform the task. Furthermore, whenever the person moves, the robot should be able to follow her without redefining the task; that is, the manipulation should be interactive by means of reactive motions.

Using the reasoning about serially coupled kinematic chains presented in the previous section, the cooperative dual task-space is redefined to take into account all the available DOF.

1. This holds under the assumption that the glass and the bottle are stationary with respect to the arms, or that there is a perception system capable of estimating the relation between the grasped objects and the respective end-effectors.

Definition 4.1. Assuming that the two-arm subsystem is the last one in the serial kinematic system, the extended cooperative dual task-space is described by

$$\underline{\mathbf{x}}_r^{E_2} = \underline{\mathbf{x}}_{E_2}^* \underline{\mathbf{x}}_{E_1}, \quad (4.1)$$

$$\underline{\mathbf{x}}_a^0 = \underline{\mathbf{x}}_{k-1}^0 \underline{\mathbf{x}}_{E_2} \underline{\mathbf{x}}_{r/2}^{E_2}, \quad (4.2)$$

where $\underline{\mathbf{x}}_{r/2}^{E_2} = \left(\underline{\mathbf{x}}_r^{E_2} \right)^{\{1/2\}}$, $\underline{\mathbf{x}}_a^0$ is the absolute dual position with respect to a reference frame \mathcal{F}_0 and $\underline{\mathbf{x}}_{k-1}^0$ is the rigid transformation describing the motion from the reference frame \mathcal{F}_0 to the original base frame of the two-arm system.

In order to better understand definition 4.1, it is first important to note that the relation between the arms is not affected by the motion of previous subsystems; hence, the relative dual position definition is exactly the same as definition 3.1. On the other hand, the absolute dual position changes according to the motions of the previous kinematic subsystems. Thus, the transformation $\underline{\mathbf{x}}_{k-1}^0$ is added to the previous definition of the absolute dual position. Figure 39 illustrates a typical situation where this extension is necessary, since the two-arm system is attached to another serial mechanism, the torso, and both are on a mobile base. Consequently, as the absolute dual position is a function of the configuration of the mobile base, torso, and the two arms, more DOF are available to execute the task.

Using corollary 4.1, the absolute dual position and its derivative can easily be expressed with respect to any previous frame in the kinematic chain. The absolute dual position derivative of the whole body is

$$\text{vec } \dot{\underline{\mathbf{x}}}_a^0 = \mathbf{J}_{\underline{\mathbf{x}}_a^0} \dot{\boldsymbol{\theta}}_{wb}, \quad (4.3)$$

where $\boldsymbol{\theta}_{wb}$ is the vector composed of the state variables of the whole body.

The explicit value of $\mathbf{J}_{\underline{\mathbf{x}}_a^0}$ is given as follows. From corollary 4.1,

$$\text{vec } \dot{\underline{\mathbf{x}}}_k^0 = \sum_{i=0}^{k-1} \mathbf{H}(\underline{\mathbf{x}}_i^0) \bar{\mathbf{H}}(\underline{\mathbf{x}}_k^{i+1}) \text{vec } \dot{\underline{\mathbf{x}}}_{i+1}^i, \quad (4.4)$$

$$= \sum_{i=0}^{k-1} \mathbf{L}_{i+1} \dot{\boldsymbol{\theta}}_{i+1}, \quad (4.5)$$

with

$$\text{vec } \dot{\underline{\mathbf{x}}}_{i+1}^i = \mathbf{J}_{\underline{\mathbf{x}}_{i+1}^i} \dot{\boldsymbol{\theta}}_{i+1}, \quad (4.6)$$

$$\mathbf{L}_{i+1} = \mathbf{H}(\underline{\mathbf{x}}_i^0) \bar{\mathbf{H}}(\underline{\mathbf{x}}_k^{i+1}) \mathbf{J}_{\underline{\mathbf{x}}_{i+1}^i}. \quad (4.7)$$

Group the matrices \mathbf{L}_{i+1} together and stack the vectors corresponding to the state variables of each subsystem

$$\mathbf{J}_{\underline{\mathbf{x}}_k^0} = [\mathbf{L}_1 \dots \mathbf{L}_k] \quad (4.8)$$

and

$$\boldsymbol{\theta} = \left[\boldsymbol{\theta}_1^T \dots \boldsymbol{\theta}_k^T \right]^T; \quad (4.9)$$

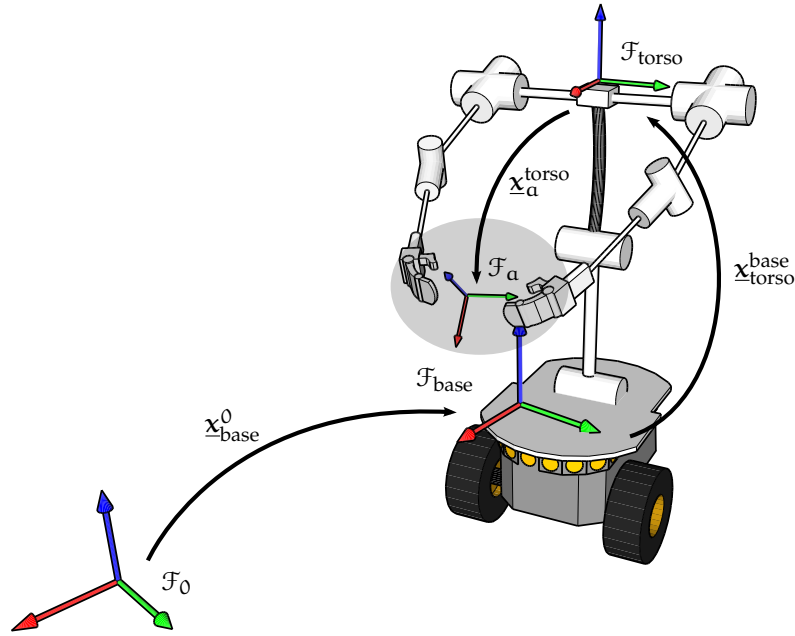


Figure 39: Serially coupled kinematic chain for a two-arm mobile manipulator.

thus,

$$\text{vec } \dot{\underline{x}}_k^0 = \mathbf{J}_{\underline{x}_k^0} \dot{\boldsymbol{\theta}}.$$

Since the absolute frame is the last one in the serially coupled kinematic system, $\mathbf{J}_{\underline{q}_a^0} = \mathbf{J}_{\underline{q}_k^0}$ and $\boldsymbol{\theta}_{wb} = \boldsymbol{\theta}$. It is important to note that the i -th kinematic subsystem is described by a forward kinematic model and a Jacobian—that is, by \underline{x}_i and \mathbf{J}_i . This means that, given the current configuration of the previous elements in the kinematic chain, \mathbf{L}_i encompasses the effect of the Jacobian \mathbf{J}_i in the absolute dual position \underline{x}_a^0 .

4.3 CASE STUDY: TWO-ARM MOBILE MANIPULATOR

Consider the two-arm mobile manipulator illustrated in figure 39. This robot consists of three subsystems: two four-DOF arms², a torso with two DOF, and a differential-drive mobile base. Each one of these subsystems can be modeled separately and the resulting models can be coupled together by using corollary 4.1.

In order to derive the kinematic model of a nonholonomic mobile base in the plane, first consider a holonomic mobile robot. It can be parameterized by $\boldsymbol{\theta}_{hol} = [x \ y \ \phi]^T$, where x, y are the Cartesian coordinates and ϕ is the heading angle. Using quaternion representation, the position is given by $\mathbf{p}_{hol} = \hat{i}x + \hat{j}y$ and the orientation by $\mathbf{r}_{hol} = \cos(\phi/2) + \hat{k} \sin(\phi/2)$. The resulting unit dual quaternion is

$$\underline{\mathbf{x}}_{hol} = \mathbf{r}_{hol} + \varepsilon \frac{1}{2} \mathbf{p}_{hol} \mathbf{r}_{hol}$$

2. Incidentally, the two-arm model is inherited from the Hoap-3 humanoid.

and, in expanded form,

$$\begin{aligned} \underline{x}_{\text{hol}} &= \cos\left(\frac{\phi}{2}\right) \hat{k} + \sin\left(\frac{\phi}{2}\right) \hat{i} \\ &+ \varepsilon \frac{1}{2} \left[\hat{i} \left(x \cos\left(\frac{\phi}{2}\right) + y \sin\left(\frac{\phi}{2}\right) \right) + \hat{j} \left(-x \sin\left(\frac{\phi}{2}\right) + y \cos\left(\frac{\phi}{2}\right) \right) \right] \end{aligned} \quad (4.10)$$

The first derivative of (4.10) provides the relation between the dual quaternion derivative and the parameters of the mobile robot; that is,

$$\text{vec } \dot{\underline{x}}_{\text{hol}} = \mathbf{J}_{\text{hol}} \dot{\underline{\theta}}_{\text{hol}}, \quad (4.11)$$

where

$$\mathbf{J}_{\text{hol}} = \begin{bmatrix} 0 & 0 & j_{13} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & j_{43} \\ 0 & 0 & 0 \\ j_{61} & j_{62} & j_{63} \\ j_{71} & j_{72} & j_{73} \\ 0 & 0 & 0 \end{bmatrix}, \text{ with}$$

$$\begin{aligned} j_{13} &= -j_{62} = j_{71} = -\frac{1}{2} \sin\left(\frac{\phi}{2}\right) \\ j_{43} &= j_{61} = j_{72} = \frac{1}{2} \cos\left(\frac{\phi}{2}\right) \\ j_{63} &= \frac{1}{4} \left[-x \sin\left(\frac{\phi}{2}\right) + y \cos\left(\frac{\phi}{2}\right) \right] \\ j_{73} &= \frac{1}{4} \left[-x \cos\left(\frac{\phi}{2}\right) - y \sin\left(\frac{\phi}{2}\right) \right] \end{aligned}$$

Eq. (4.11) does not take into account the nonholonomic restrictions of the differential-drive robot. The following relation must be used to enforce the nonholonomic constraints (Fukao et al., 2000):

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{r}{2} \cos \phi & \frac{r}{2} \cos \phi \\ \frac{r}{2} \sin \phi & \frac{r}{2} \sin \phi \\ \frac{r}{2b} & -\frac{r}{2b} \end{bmatrix}}_{\mathbf{J}_{\text{restrictions}}} \underbrace{\begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix}}_{\dot{\underline{\theta}}_{\text{wheels}}},$$

where ω_r and ω_l are the angular velocities of the right and left wheels, respectively. Using (4.11), the differential forward kinematics equation of the nonholonomic base is

$$\begin{aligned} \text{vec } \dot{\underline{x}}_{\text{nonh}} &= \mathbf{J}_{\text{hol}} \mathbf{J}_{\text{restrictions}} \dot{\underline{\theta}}_{\text{wheels}} \\ &= \mathbf{J}_{\text{nonh}} \dot{\underline{\theta}}_{\text{wheels}}. \end{aligned} \quad (4.12)$$

Recall the serially coupled kinematic chain of the two-arm mobile manipulator depicted in figure 39. The sequence of rigid motions describing the chain is given by the transformation $\underline{x}_1^0 = \underline{x}_{\text{base}}^0$ between a fixed reference frame and the mobile base, followed by the transformation $\underline{x}_2^1 = \underline{x}_{\text{torso}}^{\text{base}}$ between the mobile base and the torso's end-effector. Last, the transformation $\underline{x}_a^2 = \underline{x}_a^{\text{torso}}$ provides the dual position of the absolute frame with respect to the torso's end-effector. The intermediate and extended Jacobians are summarized in table 3.

Table 3: Extended absolute Jacobian for the robot of figure 39.

i	\mathbf{L}_{i+1}	$\mathbf{H}^+(\underline{\mathbf{x}}_i^0)$	$\mathbf{H}^-(\underline{\mathbf{x}}_a^{i+1})$	$\mathbf{J}_{\underline{\mathbf{x}}_{i+1}^i}$	$\boldsymbol{\theta}_{i+1}$
0	\mathbf{L}_1	\mathbf{I}_8	$\mathbf{H}^+(\underline{\mathbf{x}}_a^1)$	\mathbf{J}_{nonh}	$\boldsymbol{\theta}_{\text{wheels}} = [\theta_r \ \theta_l]^\top$
1	\mathbf{L}_2	$\mathbf{H}^+(\underline{\mathbf{x}}_1^0)$	$\mathbf{H}^+(\underline{\mathbf{x}}_a^2)$	$\mathbf{J}_{\text{torso}}$	$\boldsymbol{\theta}_{\text{torso}}$
2	\mathbf{L}_3	$\mathbf{H}^+(\underline{\mathbf{x}}_2^0)$	\mathbf{I}_8	$\mathbf{J}_{\underline{\mathbf{x}}_a}$	$\boldsymbol{\theta}_{\text{arms}} = [\boldsymbol{\theta}_{\text{arm1}}^\top \ \boldsymbol{\theta}_{\text{arm2}}^\top]^\top$

$$\mathbf{J}_{\underline{\mathbf{x}}_a^0} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{L}_2 & \mathbf{L}_3 \end{bmatrix}, \quad \dot{\boldsymbol{\theta}}_{\text{wb}} = \begin{bmatrix} \dot{\boldsymbol{\theta}}_{\text{wheels}}^\top & \dot{\boldsymbol{\theta}}_{\text{torso}}^\top & \dot{\boldsymbol{\theta}}_{\text{arms}}^\top \end{bmatrix}^\top$$

Because the task Jacobian corresponding to the whole body is available, the robot can be controlled by using a control law analogous to (3.58); that is,

$$\boldsymbol{\theta}_{\text{R},k} = \boldsymbol{\theta}_{\text{R},k-1} + \mathbf{J}_{\text{task}}^\dagger \mathbf{K}(\mathbf{u}_d - \mathbf{u}_m), \quad (4.13)$$

where $\boldsymbol{\theta}_{\text{R},k}$ and $\boldsymbol{\theta}_{\text{R},k-1}$ are the current and last values of the state variables, respectively, $\mathbf{J}_{\text{task}}^\dagger$ is the damped least-squares inverse of the Jacobian associated with the task (Chilverini, 1997), and \mathbf{u}_d and \mathbf{u}_m are the desired and measured values of the controlled primitives. The damped least-squares inversion method was chosen because this matrix operation is robust against kinematic singularities appearing in highly redundant systems, which is the case of a mobile manipulator.

Since the mobile base has some nonholonomic constraints, the control law (4.13) has local convergence and will inevitably fail whenever the desired reference is not feasible or is too far from the current configuration. One solution is to perform trajectory generation or apply nonlinear control techniques (Morin & Samson, 2008). Since both solutions are out of the scope of this thesis, the task simulated in the next section was designed taking this limitation into consideration.

4.4 SIMULATION RESULTS AND DISCUSSIONS

In order to illustrate the applicability of the extended cooperative dual task-space, this section presents a simulation of the task of pouring water. In this simulation, the robot must reach a location given with respect to a person and it has to pour the water. The robot is handling both the bottle and the glass, and the task is divided into two parts: the reaching and filling-the-glass subtasks.

In the reaching subtask, the robot must reach the person while one hand must draw closer to the other. The task is defined in the cooperative dual task-space, and the relative variables are used to control the coordination between the hands, whereas the absolute variables are used to drive the robot toward the human. The control of the Cartesian position is used to control both the relative and absolute variables, because the exact orientation of the two variables is not crucial at this point—although they should be monitored in order to avoid spilling the water.

Table 4: Definition of the task of pouring water.

Subtask	\mathbf{u}_d	\mathbf{J}_{task}	$\dot{\boldsymbol{\theta}}_R$
Reaching	$\begin{bmatrix} \text{vec } \mathbf{p}_a^0 \\ \text{vec } \mathbf{p}_r^{E_2} \end{bmatrix}$	$\begin{bmatrix} \mathbf{J}_{\mathbf{p}_a^0} & \mathbf{0}_{4 \times 2} \\ \mathbf{0}_{4 \times 2} & \mathbf{J}_{\mathbf{p}_r^{E_2}} \end{bmatrix}$	$\begin{bmatrix} \dot{\boldsymbol{\theta}}_{\text{wheels}} \\ \dot{\boldsymbol{\theta}}_{\text{torso}} \\ \dot{\boldsymbol{\theta}}_{\text{arms}} \end{bmatrix}$
Filling-the-glass	$\text{vec } \mathbf{x}_r^{E_2}$	$\mathbf{J}_{\mathbf{x}_r}$	$\dot{\boldsymbol{\theta}}_{\text{arms}}$

The sequence for the reaching phase is shown, in top view, in figure 40. The robot must reach the person such that the absolute frame (the frame represented by \mathbf{x}_a^0) is inside the magenta circle³, while it must also bring one hand closer to the other with the purpose of saving time. Note that the task can be accomplished only if the robot uses the mobile base.

When the robot places the absolute dual position into the desired spot, the person starts to move. Since the robot must reach a location *with respect to the person*, the robot naturally follows her, without the need to modify the task; thus, the movement is naturally reactive.

The absolute and relative position responses are illustrated in figure 41. They converge to the desired input signal, even when the person moves, as shown by the shaded area in figure 41a. It is important to note that, if additional tasks must be executed with different priorities, the primitives of the extended cooperative dual task-space can be naturally used in a prioritized task control framework (e.g., Nakamura et al., 1987; Siciliano & Slotine, 1991; Mansard, 2006; Kanoun, 2009).

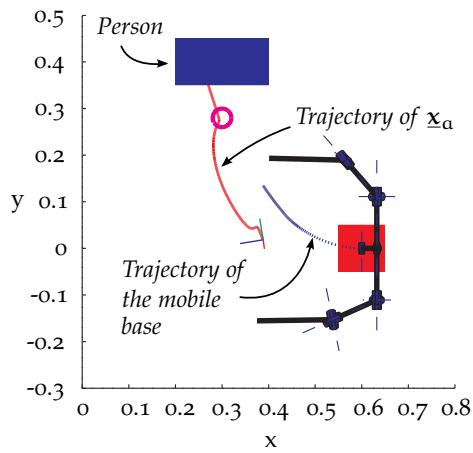
Once the person stops, the robot executes the filling-the-glass subtask. Since the relative orientation is fundamental to the successful execution of the task, as well as the relative position between the hands, the control of the relative dual position is performed. Note that the definition for this subtask is identical to the one presented in example 3.9 on page 79, as well as the simulation results, since the task involves only arm coordination. Thus, the simulation results are not presented here for brevity.

The complete task is summarized in table 4. Observe that in the reaching subtask both relative and absolute positions are controlled. Whereas to control the former only the DOF corresponding to the arms are required, in order to control the latter all available DOF are used. As a consequence, the task Jacobian must be completed with zeros, because $\mathbf{J}_{\mathbf{p}_a^0}$ is larger than $\mathbf{J}_{\mathbf{p}_r^{E_2}}$. On the other hand, the second subtask depends only on the geometrical relationship between the arms, and thus only the joints corresponding to the arms are used.

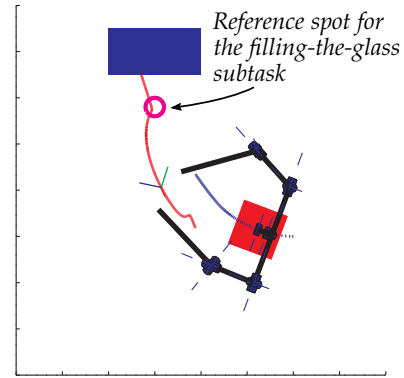
The advantage of using the extended cooperative dual task-space is that the tasks are described by the same set of primitives, independently of the number of serially coupled kinematic systems. Consequently, the primitives associated with a task will be the same, whether the robot is a two-arm manipulator or a complete mobile manipulator.

Example 4.2. Although in the previous discussion both arms were used, it is important to note that tasks involving just one arm can be treated as a particular case. Consider, for

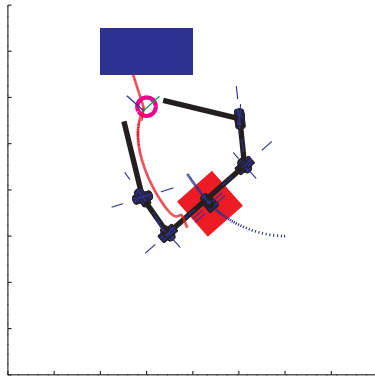
3. The strategy to define the desired location with respect to the person is out of the scope of this thesis. A good starting point for the interested reader is the work of Sisbot et al. (2010).



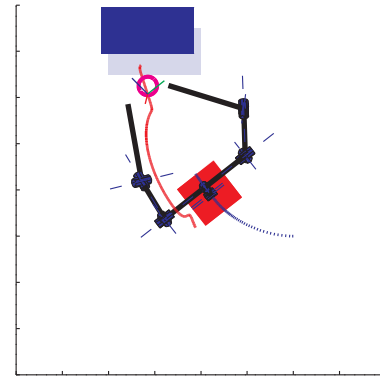
(a) Initial configuration (top view).



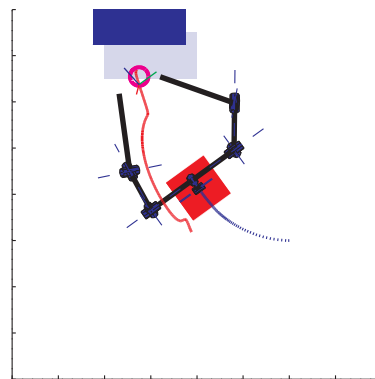
(b) The robot closes its hands while it approaches the person (represented by the blue rectangle).



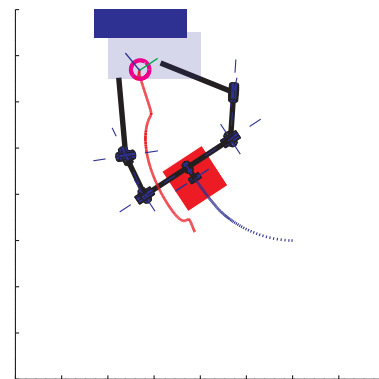
(c) The robot arrives at the right spot (magenta circle), but the person starts to move.



(d) Since the robot's reference is a specific spot (magenta circle) with respect to the person, the robot moves according to the person's movement.

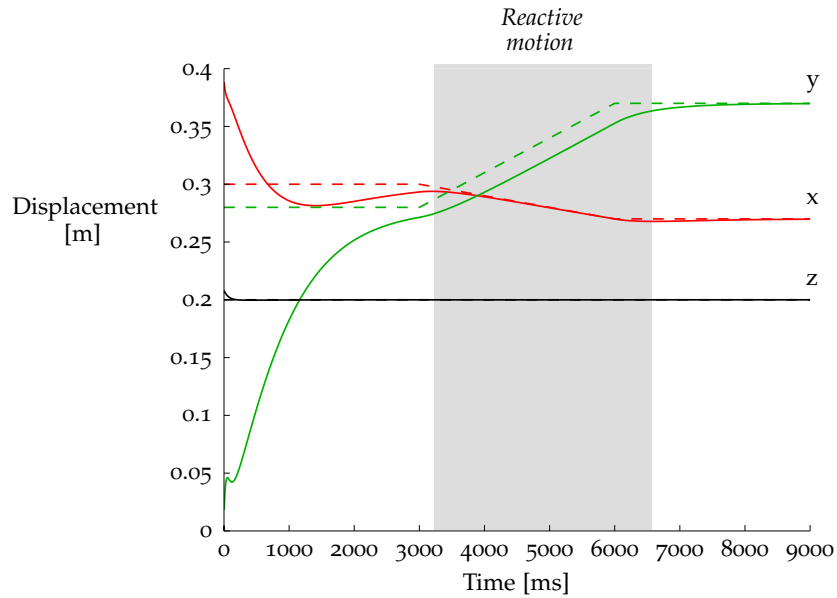


(e) The robot continues to perform the reactive behavior, following the person without the need of redefining the task.

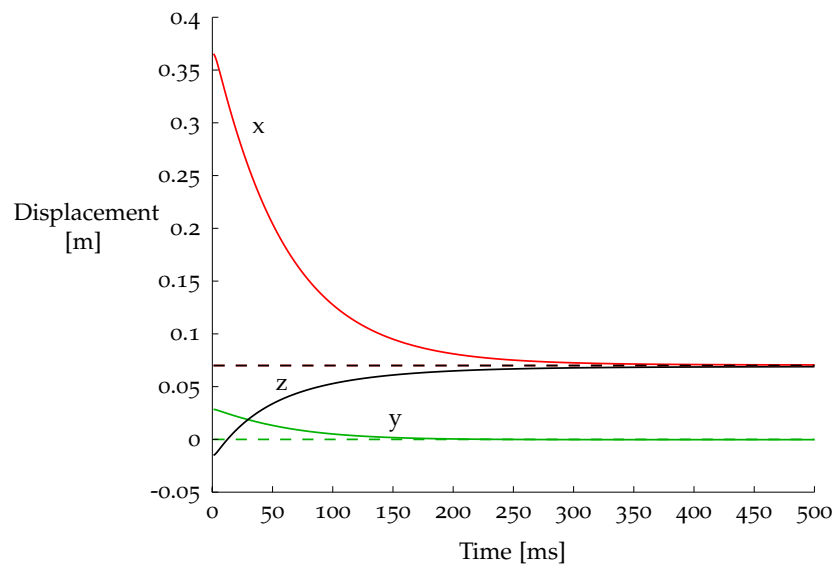


(f) The person stops and the robot arrives at the final spot, ready to pour the water.

Figure 40: Sequence corresponding to the reaching phase in the task of pouring water. The reactive motion can be observed in (d)–(f).



(a) Absolute Cartesian position control. The *shaded area* shows the reactive motion due to the person's movement.



(b) Relative position control.

Figure 41: Water pouring: the robot reaches a predefined spot and simultaneously approaches one hand next to the other. The references are represented by *dash-dotted lines* and the responses by *solid lines*.

example, the task of handing over a glass of water. Assuming that the glass is grasped by the left hand, the task parameters could be defined as

$$\begin{aligned} \mathbf{u}_d &= \text{vec}(\mathbf{x}_{\text{torso}}^0 \mathbf{x}_{\mathcal{E}_1}^{\text{torso}}) \\ \mathbf{J}_{\text{task}} &= \mathbf{J}_{\mathbf{x}_{\mathcal{E}_1}^0} \\ \dot{\boldsymbol{\theta}}_R &= \left[\dot{\boldsymbol{\theta}}_{\text{wheels}}^T \quad \dot{\boldsymbol{\theta}}_{\text{torso}}^T \quad \dot{\boldsymbol{\theta}}_{\text{arm1}}^T \right]^T, \end{aligned} \quad (4.14)$$

where the extended Jacobian $\mathbf{J}_{\mathbf{x}_{\mathcal{E}_1}^0}$ is obtained by using (4.8); that is, by considering that the frame attached to the left hand is the last one in the serially coupled kinematic chain.

4.5 CONCLUSION

This chapter presented an extension of the cooperative dual task-space that takes into account the whole-body motion in two-arm manipulations. As in the original formulation presented in chapter 3, the manipulation is described in terms of the relative and absolute variables. The notion of serially coupled kinematic systems, represented by dual quaternions, was introduced in order to describe the absolute variables in terms of all DOF present in the complete kinematic system.

A case study was performed by applying the extended formalism to a two-arm mobile manipulator composed of a nonholonomic base, a two-DOF torso and two four-DOF arms. The mobile base was modeled by using a dual quaternion description that takes into account the nonholonomic restrictions. The proposed framework was validated by using a simulated scenario for the task of pouring water, and the aspects of reactive motions were taken into account.

All the control strategies developed in chapter 3 can be applied in this extended representation. Furthermore, one-arm manipulations and the related control strategies are a particular case of the manipulation framework. Despite the fact that a simple control law was applied, prioritized schemes suitable to control redundant robots (e.g., Siciliano & Slotine, 1991; Mansard, 2006; Kanoun, 2009) can be naturally used with the extended cooperative dual task-space.

Previous works done at Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM) (Perrier et al., 1997, 1998; Perrier, 1998) proposed motion control techniques for one-arm mobile manipulators based on dual quaternions. However, these techniques were very specific for the robot that they used, and hence the formulation cannot be easily generalized to more arbitrary mobile manipulators. On the other hand, the representation proposed in this chapter is general enough to describe any kind of practical serially coupled robot, and the modeling can be performed systematically. Moreover, the representation naturally takes into account the two-arm coordination—a feature often overlooked in the literature of mobile manipulators.

HUMAN-ROBOT COOPERATION: THE COOPERATIVE DUAL TASK-SPACE APPROACH

Cooperation is related to the action of working and/or acting together toward a common goal. Two or more entities cooperate when there is substantial gain in executing the task in this manner, instead of doing it alone. The interaction of humans and robots is particularly appealing, and its development may lead to novel and unforeseen applications, although new challenges need to be faced. In this area, the literature that explores the social, cognitive, and behavioral aspects of human-robot interaction is extensive (Breazeal et al., 2008). In this chapter, however, the main interest is the collaborative aspects arising from the physical interaction between humans and robots.

A convenient classification of possible cooperative tasks is based on the level of control assumed by each one of the collaborators. In one important class of applications, the human holds complete control of the task execution. The examples in this class range from teleoperated robots to devices controlled by different forms of force control (Evrard et al., 2009), where the person plays the role of master while the robot plays the role of slave. At the opposite extreme are the less common applications where, once the task is triggered by the human, the robot controls the pace at which the task is performed (Soyama et al., 2004). Finally, there are also applications where the task control is shared between the agents involved in the cooperation, as often occurs in rehabilitation robotics (Krebs et al., 1998).

In all the aforementioned applications, the cooperative task must be properly defined in order to be provided to the system controller. However, although simple verbal task definition is often enough for human understanding, finding precise mathematical task descriptions for robot control may be a complex procedure. Some works therefore focus on the execution of only specific cooperative tasks; for instance, handing over (Huber et al., 2008; Sisbot et al., 2010), crank-rotation (Ueha et al., 2009), or carrying a large object in an indoor environment while avoiding obstacles (Adams et al., 1995).

This chapter considers the problem of mathematically describing a large set of tasks and explores an original intuitive approach to human-robot interaction. The idea is to define the cooperative tasks using the relative poses between the human's hand and the robot's hand. Several cooperative manipulation tasks are used to show how simple task descriptions may produce significant and complex cooperation.

In the first three tasks investigated—*pouring water*, *teleoperation with collaboration*, and *simultaneous handling with mirrored movements*—the person is completely autonomous, whereas the robot can be regarded as an autonomous entity or a slave, depending on the task. On the other hand, in the fourth task investigated—*ball in the hoop*—the robot is not only an autonomous entity, but it also controls the human arm by means of

functional electrical stimulation (FES)¹. All tasks are defined and controlled within the cooperative dual task-space framework.

The next section proposes human-robot interaction as an application of the cooperative dual task-space. Then, sections 5.3–4 apply the methods proposed in chapters 2 and 3 to control the arms of both human and robot, and the FES control method proposed by Bó (2010) is briefly described. The last two sections of the chapter are devoted to the experimental validation of the proposed methods and to the conclusions.

5.1 INTUITIVE TASK DEFINITION

A high number of physical human-robot interactions may be described by the relative pose between the human hand and the robot hand. For some tasks, this description is not sufficient, since, for instance, objects must be picked up before actual manipulation. However, even in this condition, the cooperative task itself may be described in terms of relative motions. Moreover, it may be much easier to describe the relative poses than the evolution of both absolute poses in time. Based on this intuition, the cooperative dual task-space can be applied to the problem of human-robot collaboration (HRC).

In order to better illustrate the sort of cooperative tasks dealt with in this chapter, consider figure 42. The task of pouring water can be completely defined by the geometric relation between the hands, and thus the control of the relative dual position suffices to accomplish the task, because no contact is involved. Whenever handing over an object, the phase where the hands are moving toward each other can also be defined geometrically and is also executed by a position controller. Turning a crank, on the other hand, is normally defined by means of the forces involved in the interaction, because small disturbances in the positioning can lead to large interaction forces. In human-robot interactions, however, the humans can use their own force feedback to manage the interaction forces.

For most of the collaborative tasks similar to the ones illustrated in figure 42, the collaborators must adapt their motions in reaction to the other's in order to obtain effective cooperation. In this context, the cooperative dual task-space can be used to describe the collaborative task, and the kinematic control laws proposed in chapter 3 can be used to perform the collaborative behavior.

This thesis considers human-robot cooperation as a particular case of the cooperative dual task-space representation. The arms of both robot and human are considered as manipulators sharing the same workspace and their end-effector configurations are represented by the dual quaternions $\underline{\mathbf{x}}_R$ and $\underline{\mathbf{x}}_H$, respectively. The task, represented by the relative configuration between the arms, is given by

$$\underline{\mathbf{x}}_{\text{task}} = \underline{\mathbf{x}}_R^* \underline{\mathbf{x}}_H. \quad (5.1)$$

1. It is important to underline that the work described in this chapter was developed in collaboration with Dr. Antônio Padilha Lanari Bó, a former PhD student at LIRMM (Adorno et al., 2011b,a). Dr. Bó participated in the design and execution of all HRC experiments, and was also responsible for the FES control of the human arm.

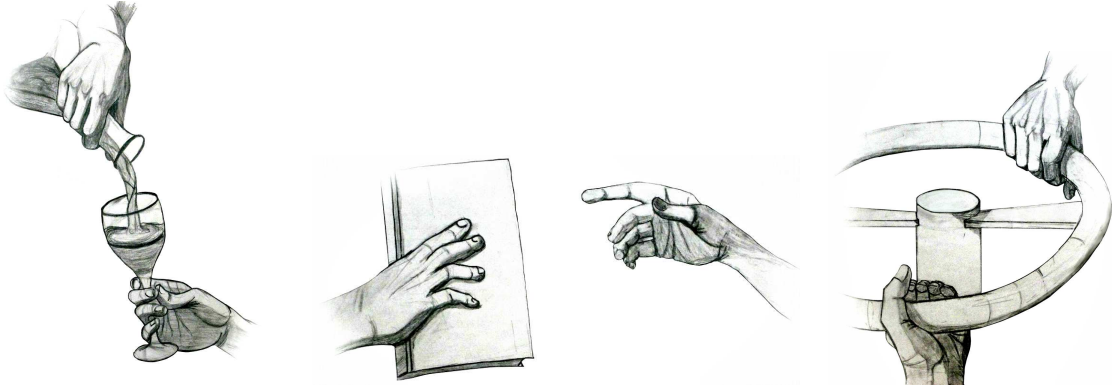


Figure 42: Examples of cooperative tasks executed by two different collaborators. From *left to right*: pouring water, handing over, and crank rotation.

5.2 ROBOT'S PERCEPTION OF THE HUMAN MOTION

Because the task is defined by the relative configuration between the hands, their poses must be estimated. As shown in chapter 2, the pose of the robot end-effector is given by the FKM but, on the other hand, the estimation of the human joint angles is not a trivial task—even if the FKM of the human arm is available. This problem can be solved if a motion tracker system is used to obtain the pose of the human hand with respect to a known reference frame. Such a setup is adopted in this thesis and is shown in figure 43.

The problem of this setup is that the robot hand is related to \mathcal{F}_T , whereas the pose of the human hand is related to \mathcal{F}_M , the coordinate system of the motion tracker. However, (5.1) requires that the poses of both the human hand and the robot hand be expressed with respect to a common frame, which was chosen as the robot torso. For the purposes of this chapter, all the reasoning is done by assuming that the pose of the human hand is already given with respect to the common frame, and the methodology to obtain such pose is presented in appendix C.

5.3 CONTROL OF THE ROBOT ARM

Recall from chapter 2 that the differential FKM of the robot arm is given by

$$\text{vec } \dot{\mathbf{x}}_R = \mathbf{J}_R \dot{\boldsymbol{\theta}}_R,$$

where $\boldsymbol{\theta}_R$ is the vector of joint variables of the robot arm, and \mathbf{J}_R is the dual quaternion Jacobian. The control law (3.55) can be applied,

$$\boldsymbol{\theta}_{R,k} = \boldsymbol{\theta}_{R,k-1} + \mathbf{J}_R^+ \mathbf{K} \text{vec} \left(\underline{\mathbf{x}}_{H_m}^T \mathbf{x}_{\text{task}}^* - \underline{\mathbf{x}}_{R_m}^T \right), \quad (5.2)$$

where \mathbf{K} is a positive definite gain matrix, and $\underline{\mathbf{x}}_{H_m}^T$ and $\underline{\mathbf{x}}_{R_m}^T$ are the current poses of the human hand and the robot hands, respectively.

It is important to underline that the desired reference $\underline{\mathbf{x}}_{H_m}^T \mathbf{x}_{\text{task}}^*$ for the robot hand depends on the measured pose of the human hand. Because there is no feedforward term

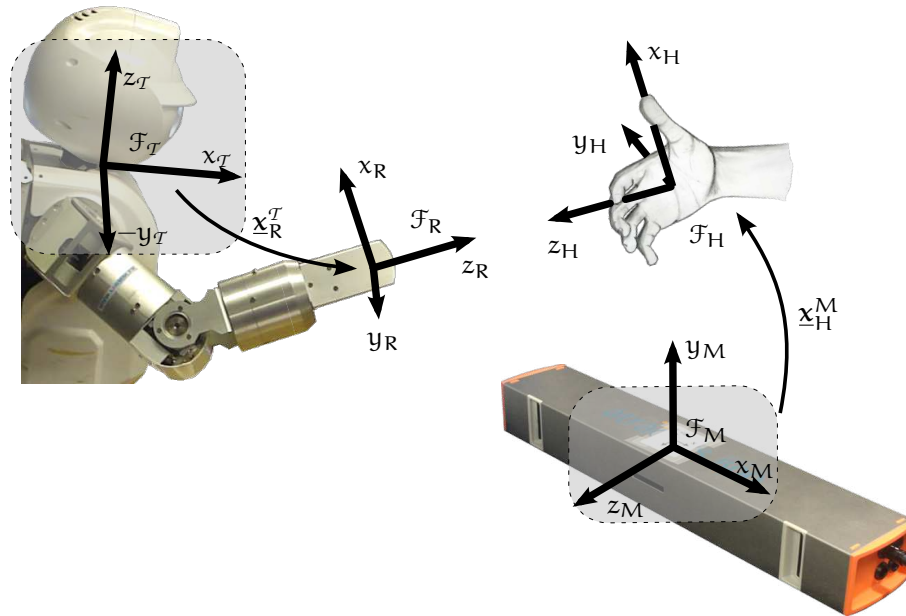


Figure 43: Experimental setup for the human-robot collaboration.

in (5.2), the robot will not be able to track the human hand in case of fast motions. However, this control law is sufficient for the purposes of this chapter, as the experiments only require relatively slow motions.

5.4 CONTROL OF THE HUMAN ARM

In order to control the human arm, two requirements must be satisfied. In the low level—that is, joint level—the muscles must receive appropriate control signals in order to move the correspondent joints. This can be accomplished by FES and is explained in the next section. In a higher level, the arm can be considered as a manipulator in order to perform kinematic control, and thus the techniques developed in chapter 3 can be applied.

5.4.1 Human arm control by using functional electrical stimulation

Functional electrical stimulation is the application of electrical pulses to the neural pathways or directly to the muscles, with the purpose of providing functional benefit to the patient, typically by restoring lost or impaired neuromuscular functions. In order to meet this goal, FES systems attempt to mimic neural excitation, which is based on trains of action potentials. Hence, FES signals are based on trains of impulses and are applied on the body by means of electrodes.

Because muscles are natural actuators in the human body, FES can be used to induce limb motion in impaired patients (e.g., quadriplegics) in order to improve their interaction with assistant robots, thus providing them more autonomy in their daily lives.

In spite of its great potential for many applications, the use of FES to control muscle action is still a major challenge (Lynch & Popovic, 2008), particularly when superficial

electrodes are used. Some of the difficulties involved are related to the diffusion of the stimulation to other muscles and the inherent complexity of muscle dynamics. For this reason, in the experiment presented in this chapter, the FES-induced human arm motion was very simple, limited to the control of the elbow angle and the opening of the hand. In order to control the elbow, only the *Biceps Brachii* was stimulated, which means that the FES system was not able to extend the elbow. In order to open the hand, the *Extensor Digitorum Communis* was activated. The placement of the electrodes is shown in figure 44.

For both stimulated muscles, an initialization procedure was required in order to set the appropriate FES parameters for a particular subject. In this procedure, the stimulation frequency, intensity and pulse width were chosen based on the obtained response and on the subject's subjective evaluation. The first two parameters were kept fixed during the experiment, while the stimulation pulse width was used to control the joint motion. In order to standardize the controller for different subjects, the control signal was first normalized based on the pulse width range defined by the subject.

Concerning the controller designed to control the elbow position, a simple PI controller with anti-windup was used. The latter feature was used due to the actuator saturation with respect to physiological limits and to provide comfort to the subject. The controller gains were set in order to minimize possible overshoots, as the elbow extension was not actuated. This FES controller was implemented by Dr. Antônio Padilha Lanari Bó, and is thoroughly described in his PhD thesis (Bó, 2010).

5.4.2 Human arm control in the Cartesian space

In order to control the human arm in the Cartesian space, the arm is considered as a serial manipulator with an end-effector pose given by \underline{x}_H and the vector of actuated joints given by θ_H .

Due to the restrictions imposed by the implementation of the FES controller, only the biceps was controlled in a continuous range, whereas the opening of the hand was controlled by an on/off controller. Hence, it was more appropriate to model the human arm as a one-DOF serial robot with the rotation axis located at the elbow and the end-effector located at the wrist, as illustrated in figure 45. The corresponding Denavit-Hartenberg (D-H) parameters are shown in table 5, where the parameter a was loosely defined.

The differential FKM of the one-link human arm is given by

$$\text{vec } \dot{\underline{x}}_{\text{wrist}}^{\text{elbow}} = \mathbf{J}_{\text{wrist}}^{\text{elbow}} \dot{\theta}_H. \quad (5.3)$$

However, there is a displacement between the wrist and the hand, given by $\underline{x}_H^{\text{wrist}}$ (see fig. 49 on page 105); thus, the human hand given with respect to the robot torso is

$$\underline{x}_H^T = \underline{x}_{\text{elbow}}^T \underline{x}_{\text{wrist}}^{\text{elbow}} \underline{x}_H^{\text{wrist}}.$$

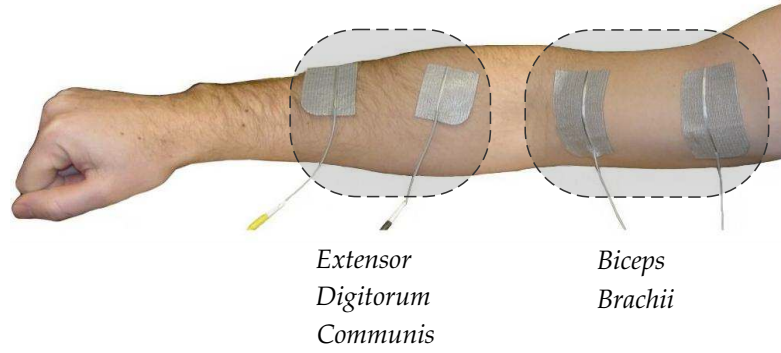


Figure 44: Actuation of the human arm: positioning of the electrodes.

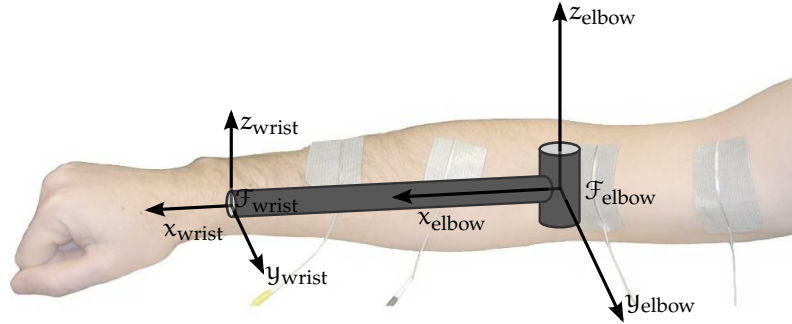


Figure 45: The human arm modeled as a one-link serial robot.

Assuming that $\underline{x}_{\text{elbow}}^T$ and $\underline{x}_H^{\text{wrist}}$ were constant during the experiment,

$$\begin{aligned}
 \dot{\underline{x}}_H^T &= \underline{x}_{\text{elbow}}^T \dot{\underline{x}}_{\text{wrist}}^{\text{elbow}} \underline{x}_H^{\text{wrist}} \\
 \text{vec } \dot{\underline{x}}_H^T &= \overset{+}{\mathbf{H}}(\underline{x}_{\text{elbow}}^T) \overset{-}{\mathbf{H}}(\underline{x}_H^{\text{wrist}}) \text{vec } \dot{\underline{x}}_{\text{wrist}}^{\text{elbow}} \\
 &= \underbrace{\overset{+}{\mathbf{H}}(\underline{x}_{\text{elbow}}^T) \overset{-}{\mathbf{H}}(\underline{x}_H^{\text{wrist}}) \mathbf{J}_{\text{wrist}}^{\text{elbow}}}_{\mathbf{J}_H^T} \dot{\theta}_H.
 \end{aligned} \tag{5.4}$$

The following control law based on (5.4),

$$\dot{\theta}_H = \mathbf{J}_H^T \mathbf{K} \text{vec}(\underline{x}_{R_m}^T \underline{x}_{\text{task}} - \underline{x}_{H_m}^T),$$

would be underactuated because the control of the dual position would require six DOF, but only one DOF could be actuated. In order to tackle this problem, only the translation part of \underline{x}_H^T was controlled, leading to the following control law:

$$\dot{\theta}_H = \mathbf{J}_{\mathbf{p}_H^T}^+ \mathbf{K} \text{vec}(\mathbf{p}_{H_d}^T - \mathbf{p}_{H_m}^T), \tag{5.5}$$

where \mathbf{K} is a positive definite gain matrix, $\mathbf{J}_{\mathbf{p}_H^T}$ is given by (3.14), and $\mathbf{p}_{H_d}^T$ is extracted from

$$\underline{x}_{H_d}^T = \underline{x}_{R_m}^T \underline{x}_{\text{task}} \tag{5.6}$$

by using (2.11). The velocity $\dot{\theta}_H$ is then provided to the joint-level PI controller with anti-windup described in section 5.4.1.

Table 5: D-H parameters for the human arm.

link	θ	d	a	α
1	0	0	0.3	0

5.5 EXPERIMENTS

Four experiments were performed in order to validate the HRC. In the first three (sections 5.5.1–3), the purpose was to illustrate the general use of the cooperative dual task-space in the description of HRC tasks. Three tasks were performed, where only the robot arm was controlled: *water pouring*, *teleoperation with collaboration*, and *mirrored movements followed by simultaneous handling* of the same object.

The fourth experiment, called *ball in the hoop* (section 5.5.4), was an attempt to verify two hypotheses: first, robots can control the human arm in order to achieve a desired coordination; second, good coordination results in a higher success rate when compared with just the performance of the best collaborator.

Whereas the first hypothesis was postulated with robotic assistance to people with impairments in mind, the second one was based on the common sense that effective collaboration can be achieved even if the collaborators do not have good enough individual performance.

The robot used in all experiments was a Hoap-3 humanoid robot, which is 60 cm tall, weighs about 8 kg, and has 28 DOF, with each arm having four DOF. Only healthy subjects participated in the experiments, and both the robot and the person used the right arm for the cooperative tasks, as shown in figure 43, which also illustrates the main coordinate frames used in the first three experiments.

In order to capture the human motion, an optical system was used. This system, called Easytrack 500, is composed of linear cameras and can track and estimate the pose of a marker equipped with infrared LEDs. Every 100 ms, the device provides a good estimation of the marker’s pose with respect to its reference frame, \mathcal{F}_M . In the first three experiments, the marker was placed on the subject’s hand or wrist, as shown in figure 46 and figure 47, defining the frame \mathcal{F}_H .

5.5.1 Water pouring

In the task of pouring water, the robot had to pour the water while the subject held the glass. Due to the robot’s physical limitations, a plastic cup was attached directly onto the back of the robot hand. The task $\underline{x}_{\text{task}}$ was defined as follows: using \mathcal{F}_R as reference, \mathbf{r}_{task} was defined as a rotation of π around x_R —enforcing a face-to-face cooperation—followed by a rotation of $\pi/4$ around z_R . This last rotation was defined to put the robot hand in a pose suitable for pouring water. On the other hand, the translation was chosen by considering the size of the plastic cups and the placement of the cup on the back of the robot hand. The parameters of the task are summarized in table 6 on page 104.

The subject was told to place his hand in different arbitrary spots, all of them feasible. The robot successfully poured the water and was capable of tracking the subject’s hand whenever the latter moved. Some of the spots are shown in figure 46.

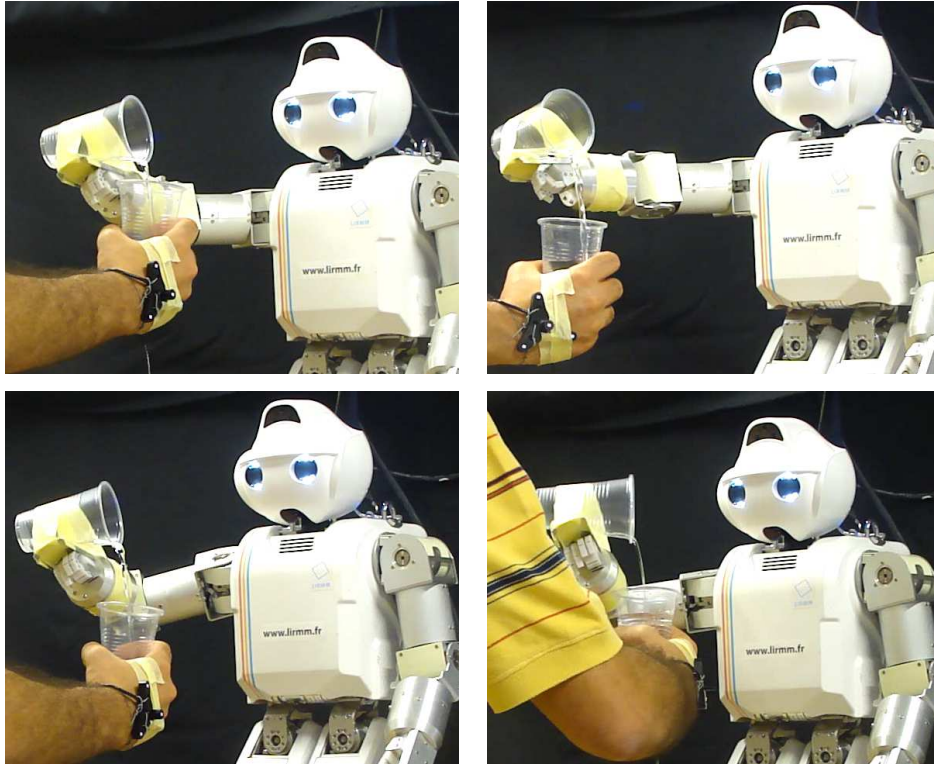


Figure 46: Task of pouring water.

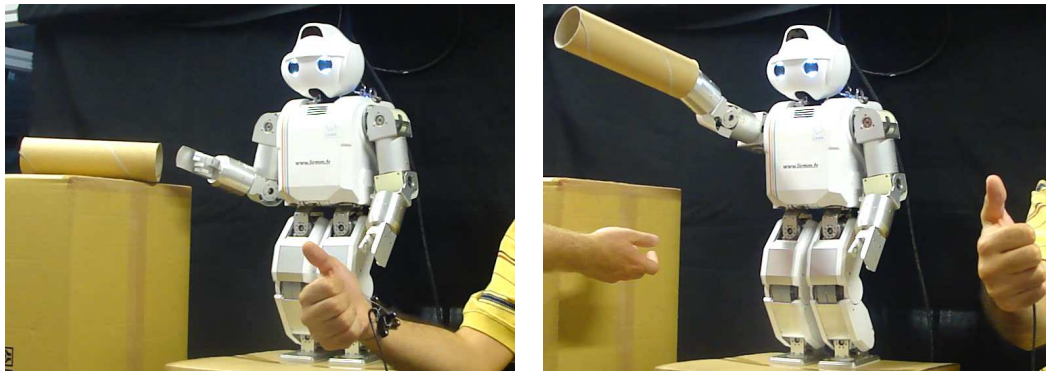


Figure 47: Teleoperation with collaboration: reaching the pipe (*left*) and handing it to another person (*right*).

5.5.2 Teleoperation

Instead of proposing a new method for teleoperation, which has already been extensively investigated in the literature, the purpose of this experiment was to show that, in terms of mathematical description, the teleoperation task can be described within the cooperative dual task-space framework. Because the robot must mimic the teleoperator, the parameter of the teleoperation task is $\underline{x}_{\text{task}} = 1$.

In the experiment, the cooperation happened when a second person, the collaborator, interacted with the robot. Figure 47 shows a sequence of teleoperation with cooperation. Using the robot, the teleoperator could grab the pipe and hand it to the collaborator.

5.5.3 Simultaneous handling using mirrored movements

As the name suggests, mirrored movements are characterized by motions analogous to the ones that one can see in front of a mirror. For instance, consider two agents, the first called active and the other called reactive. While the active agent performs a motion, the reactive agent will play the role of the mirror.

In HRC, mirrored movements can be used such that the robot acts as a “mirror” of the person until the object of interest is grasped. In this manner, the person can use mirrored movements to drive the robot hand to a good grasping position.

In order to illustrate this idea, consider the following situation. The person must grab an object but needs help to accomplish the task, because the object is too heavy or too big. This task was already extensively investigated in terms of force control (e.g., Stasse et al., 2009), where the robot follows the person by means of force compliance. However, before the activation of the force controllers, the robot should reach the object. If the robot mirrors the person, the person can drive the robot hand to the right grasp position while positioning his/her own hand.

In order to define the task, consider the initial pose of the human hand as shown in figure 48a; that is,

$$\mathbf{x}_{H_0}^T = \mathbf{x}_{R_0}^T \mathbf{r}_{x,\pi},$$

where $\mathbf{x}_{R_0}^T$ is the initial pose of the robot hand and $\mathbf{r}_{x,\pi}$ is a rotation of π rad around x . The initial position $\mathbf{p}_{H_0}^T$ is extracted by using (2.11), and the desired relative translation is calculated according to the new position \mathbf{p}_H^T of the human hand; that is,

$$\mathbf{p}_{\text{task}} = \hat{k} [2 (\mathbf{x}_H^T - \mathbf{x}_{H_0}^T)]. \quad (5.7)$$

First note that \mathcal{F}_T and \mathcal{F}_R are not aligned (see fig. 43), and thus the x axis of \mathcal{F}_T is used as the reference for the z axis of the robot hand. Eq. (5.7) means that, whenever the subject moves his/her hand vertically or horizontally, the robot will try to place its hand at the same place as the subject’s hand. However, when the subject moves the hand backward, the robot will also move its hand backward. Conversely, when the subject moves the hand forward, the robot will move its hand forward. Furthermore, differently from the two previous tasks, the parameters of this one is variable.

Figures 48a–c show the sequence obtained from a manipulation task using mirrored movements. First, the system was initialized in the face-to-face configuration. Then, the operator directed his hand toward one extremity of the pipe while simultaneously controlling the robot’s hand. Differently from a simple teleoperation, in this type of task the operator must take into account the poses of both his/her hand and the robot’s in order to cooperatively grasp the pipe.

After the pipe was handled, the current value of \mathbf{x}_{task} was stored and from that moment the stored value was used as a constant reference. In this new task the robot had to maintain the same relative configuration. The remarkable finding is that, even if the object was rigid and the robot was position controlled—and thus very stiff—the operator could still drive the pipe and was followed by the robot. This was due to the fact that the human arm has more DOF than the robot’s, and hence the operator could exploit his redundancy to maintain the pipe in the same orientation, but changing the orientation of his/her wrist. Consequently, the robot followed the operator’s movement. Moreover, the forces involved in such cooperation are measured by the person’s force

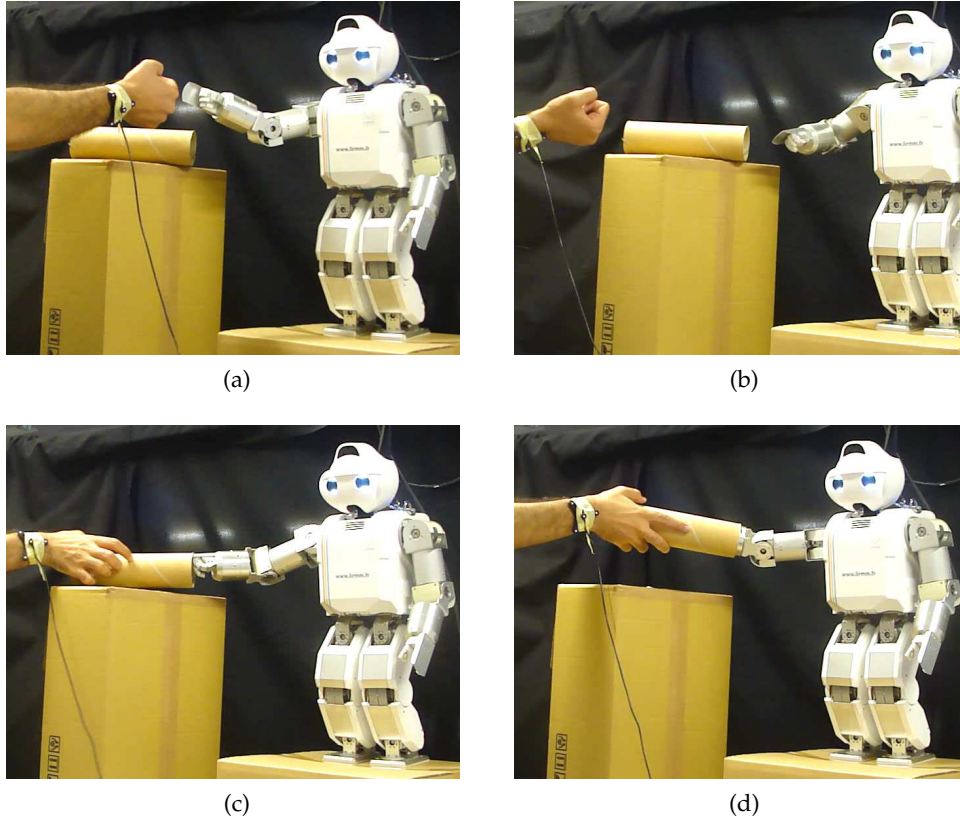


Figure 48: Mirror mode: the person controls the robot in a collaboration-like fashion. (a) The system is initialized in the face-to-face configuration; (b) the subject's hand is moved backward. The robot mirrors the movement; (c) the subject drive his hand toward the object, and the robot symmetrically follows the movement; (d) once the object is grasped, the current $\underline{x}_{\text{task}}$ is stored and remains constant. Since the subject's arm has more DOF than the robot's, the subject can still move the object, followed by the robot.

Table 6: Parameters of the cooperative tasks in the HRC experiments.

Task	$\underline{x}_{\text{task}}$
Water pouring	$\mathbf{p}_{\text{task}} = \hat{i}0.01 - \hat{j}0.05$ $\mathbf{r}_{\text{task}} = \mathbf{r}_{x_R, \pi} \mathbf{r}_{z_R, -\pi/4}$
Teleoperation	1
Mirrored movement	$\mathbf{p}_{\text{task}} = \hat{k} \left[2 \left(\mathbf{x}_H^T - \mathbf{x}_{H_0}^T \right) \right]$ $\mathbf{r}_{\text{task}} = \mathbf{r}_{x_R, \pi}$

feedback mechanism, and even if the robot was not capable of performing force control, the human was. Hence, in simultaneous handling tasks, the *human* ultimately takes into account the forces involved in the interaction.

The parameters of the cooperative tasks described in these first three experiments are summarized in table 6.

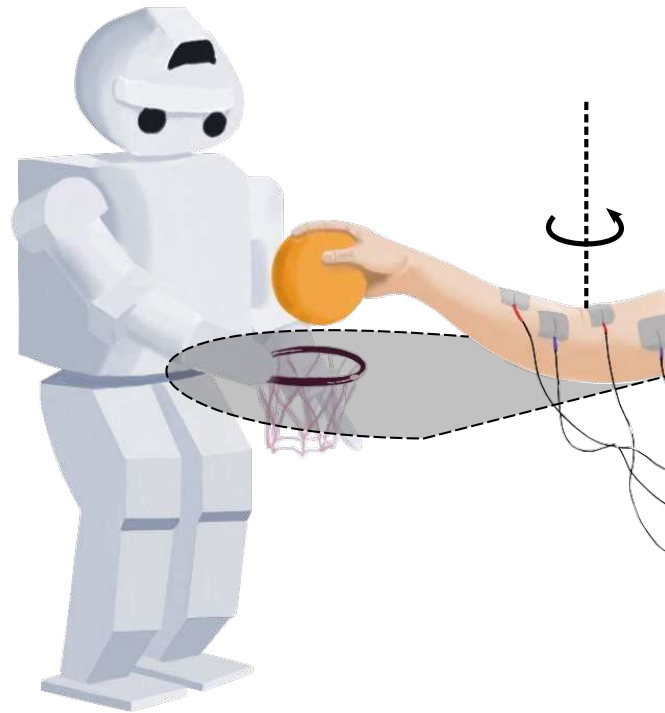


Figure 49: In the ball in the hoop experiment, the human arm was actuated by FES and was constrained to move on a plane.

5.5.4 *Ball in the hoop*

In this last experiment, the goal was to assess two hypotheses: first, robots can control a human arm in order to achieve a desired coordination; second, good coordination implies a higher success rate when compared with the performance of only the best collaborator.

In order to verify these hypotheses, a simple task was defined such that the robot had to hold a miniature basketball hoop and the person had to hold a ball. The human arm was controlled by means of FES and both arms had to be coordinated in order to drop the ball inside the hoop. To simplify the experimental setup and avoid the requirements of precise multi-joint FES control, the human arm was constrained to move only in one plane, as depicted in figure 49.

Five healthy volunteers participated in the experiment, three males and two females. They were positioned as shown in figure 51a on page 109, with the arms positioned over a surface parallel to the robot's floor. Furthermore, the subjects were blindfolded in order to prevent visual feedback and thus to prevent any attempt to accomplish the task by natural actuation; that is, not by means of the FES. In addition, they were asked not to move their bodies and to avoid the voluntary control of their arms—although they could not completely control their biceps, in principle they could disturb the task by controlling other muscles involved in the movement of the arm. Another requirement imposed on the subjects was to maintain the same configuration between the wrist and the hand with the purpose of maintaining $\underline{x}_H^{\text{wrist}}$ constant, such that (5.4) could be valid. Last, differently from the previous experiments, \mathcal{F}_H was placed on the ball, as illustrated in figure 50.

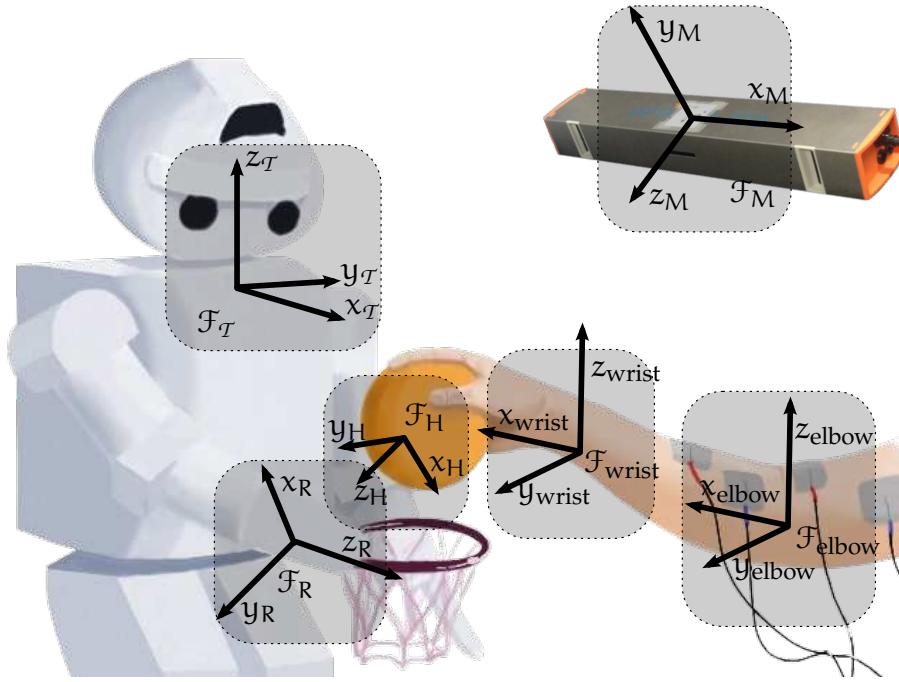


Figure 50: Coordinate systems in the ball in the hoop experiment.

In addition to the materials used for the first set of experiments, a 8-channel stimulator—called Prostim—designed jointly by the LIRMM and Neuromedics, was used to perform FES control.

The task was divided into “reaching” and “dropping the ball” subtasks. First, the reaching task had to be performed by aligning the hands along a vertical line, under the assumption that the human hand was higher than the robot hand. Once the hands aligned, the human hand had to be opened in order to drop the ball inside the hoop.

Using $\underline{x}_{\text{task}}$ obtained from the calibration process presented in section C.3 of appendix C², the subtasks are defined as follows, starting from the “reaching” subtask. The reference $\mathbf{p}_{H_d}^T$ is calculated from (5.6), and the control law (5.5) is executed in order to bring the human arm into the robot’s workspace—typically in a predefined region known to be suitable to accomplish the cooperation task—e.g., the center of the robot’s workspace. Then, θ_H is calculated from (5.5) and provided to the low level PI controllers, and the latter is responsible to control the human elbow joint. Whenever the human hand enters in a region defined by

$$\|\mathbf{p}_{H_d}^T - \mathbf{p}_{H_m}^T\| < d_{\text{coll}}, \quad (5.8)$$

the robot starts to move its arm by using (5.2). The scalar d_{coll} was used to prevent the robot from trying to cooperate when the human hand was still far from the cooperation region, and the value of d_{coll} was determined empirically.

The “dropping the ball” subtask is executed when the following condition is satisfied:

$$\|\text{vec}(\underline{\mathbf{x}}_{R_m}^{T*} \underline{\mathbf{x}}_{H_m}^T - \underline{\mathbf{x}}_{\text{task}})\| < \delta_{\text{stop}},$$

2. Departing from the previous experiments where the parameters of the task were defined by hand, in this experiment the parameters were defined by means of a simple calibration process. Because the method to define the task $\underline{x}_{\text{task}}$ is not relevant to the following discussion, they are omitted in this chapter.

Algorithm 5.1 Ball in the hoop task.

```

1: while  $\left\| \text{vec} \left( \underline{x}_{R_m}^T \underline{x}_{H_m}^T - \underline{x}_{\text{task}} \right) \right\| > \delta_{\text{stop}}$  do
2:    $\dot{\theta}_H \leftarrow \mathbf{J}_{\mathbf{p}_H}^+ \mathbf{K} \text{vec} \left( \mathbf{p}_{H_d}^T - \mathbf{p}_{H_m}^T \right)$ 
3:   Pass  $\dot{\theta}_{H_d}$  to the human elbow's PI controller with anti-windup.
4:   if  $\left\| \mathbf{p}_{H_d}^T - \mathbf{p}_{H_m}^T \right\| < d_{\text{coll}}$  then
       $\theta_{R,k} = \theta_{R,k-1} + \mathbf{J}_R^+ \mathbf{K} \text{vec} \left( \underline{x}_{H_m}^T \underline{x}_{\text{task}}^* - \underline{x}_{R_m}^T \right)$ 
5:   end if
6: end while
7: Send the FES signal to open the human hand.

```

Table 7: Criteria used to evaluate the ball in the hoop task.

Task accomplished	If the ball hits the border or enters the hoop
Robot fulfills its role	If the robot hand reaches the human hand in a suitable pose
Human has good performance	If the human arm enters and stays in the cooperation zone
Effective cooperation	If the robot fulfills its role and the human has good performance

where δ_{stop} is a convenient stop criterion for the “reaching” subtask. Once the error in the task coordination is sufficiently small, the robot executes the “dropping the ball” subtask; that is, the robot sends the FES signal to open the human hand.

The cooperative task control algorithm is summarized in algorithm 5.1.

5.5.4.1 *Experimental results*

Because the ball in the hoop experiment was not designed with the purpose of obtaining precision—but rather effective cooperation—the following metrics are used. A task is considered as accomplished whenever the ball hits the border or enters into the hoop, meaning that good coordination is achieved. The robot fulfills its role if the robot hand reaches the human hand in a pose suitable for the success of the task. The human’s performance—that is, the result of the human arm control—is considered “good” whenever the human arm enters and stays in the cooperation zone defined by (5.8); that is, in the robot’s workspace. When both agents have good performance—that is, the robot fulfills its role and the human arm enters and stays in the cooperation zone—the task is considered as an effective cooperation. The criteria are summarized in table 7. Next, a detailed analysis of the experiments performed with five healthy blindfolded subjects is presented.

Figure 51 shows a successful cooperation sequence corresponding to the time responses shown in figure 52. The human arm entered the cooperation zone in about one second, but the “dropping the ball” subtask could not be initiated because the robot was adjusting the pose of its arm. Once their relative configuration was achieved, between four and five seconds, the FES signal was sent to the human forearm in order to

open the hand. Because the agents cooperated, complete success was obtained. For this particular subject, the cooperation workspace (the workspace where the cooperation is feasible) was given roughly when the human arm joint was between 30–50°. Because the robot used the information of the marker’s position to calculate the angle of the human elbow joint, after the ball dropped the system would stop measuring the angle of the human arm.

Figure 53 shows the individual performances in the cooperative task. With the exception of subject 2, overall the robot had better performance, in terms of collaboration, than the FES-controlled human arm. This was expected since the human arm was actuated in only one direction and also there are limitations on joint control by using FES. Therefore, whenever the human arm left the cooperation zone, it was impossible to return. On the other hand, the robot’s performance would have been better if the ball’s orientation—i.e., the orientation of the marker attached to the ball—had not been considered in the task, or if the robot had had more DOF.

Figure 54 shows the impact of the effective cooperation in the accomplishment of the task. Whenever there is effective cooperation, the number of accomplished tasks tends to be close to the number of effective co-operations. This was also expected, because when both human and robot achieve good coordination, the error in the task execution tends to be smaller.

Last, figure 55 shows the comparison of the success rate (rate of accomplished tasks) based on two variables. First, the effective collaboration is considered; that is, for each subject the number of accomplished tasks is divided by the number of effective co-operations:

$$\text{success rate} = \frac{\text{number of accomplished tasks}}{\text{number of effective cooperations}}.$$

Second, the performance of the best agent is considered; that is, for each subject the number of accomplished tasks is divided by the best case shown in figure 53:

$$\text{success rate} = \frac{\text{Number of accomplished tasks}}{\max(N_r, N_H)},$$

where N_r is the number of times that the robot fulfilled its role, and N_H is the number of times that the human had good performance.

The remarkable finding is that the success rate was higher whenever there was an effective collaboration in comparison with the situation where one agent performed very well but the other one did not necessarily also have a good performance. This suggests that, instead of having one agent be extremely good in order to compensate the other’s error, the success rate is higher when both agents are “good enough.”

5.6 CONCLUSION

This chapter was devoted to the application of the cooperative dual task-space techniques in the HRC problem. Using relative variables, it was shown that different tasks can be represented and controlled by using the same set of equations. The robot’s limited dimensions, as well as its restricted motions, posed a challenge to the design of

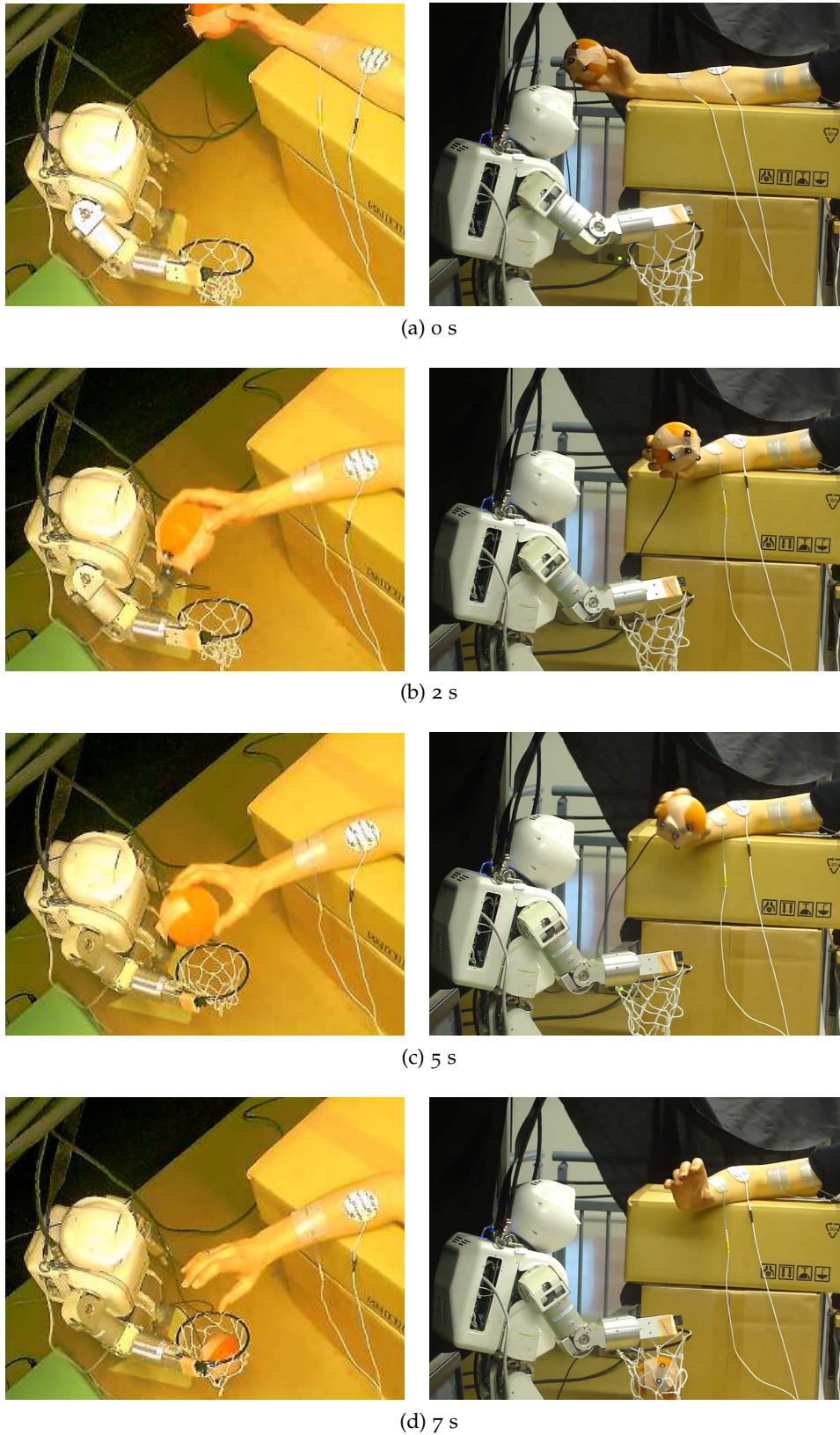


Figure 51: Top and lateral views (*left* and *right* columns, respectively) corresponding to a sequence of one trial in the ball in the hoop experiment for subject 3.

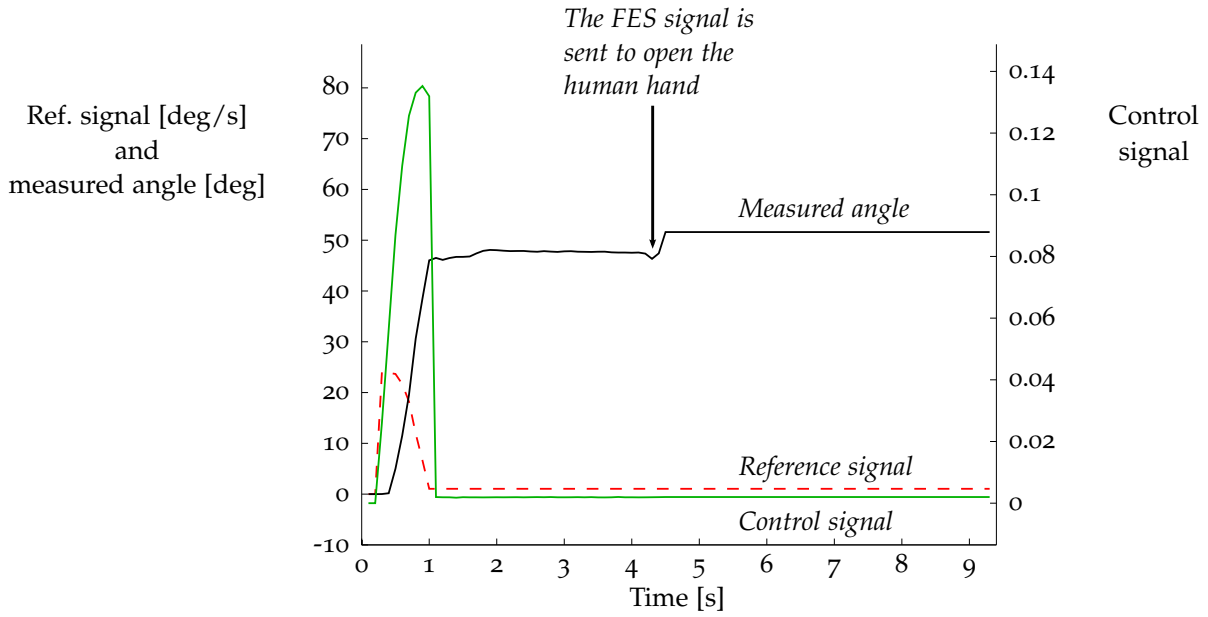


Figure 52: Time response for the successful trial shown in figure 51.

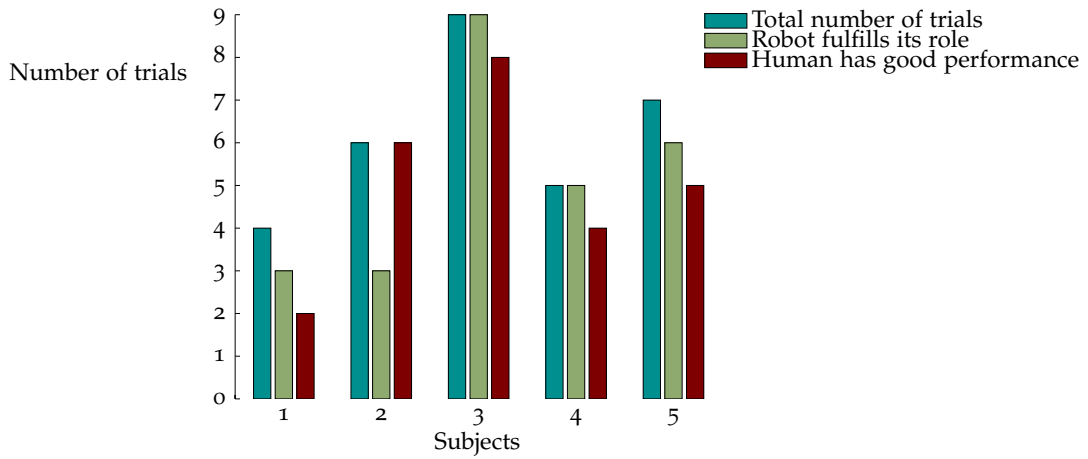


Figure 53: Number of trials versus individual performance.

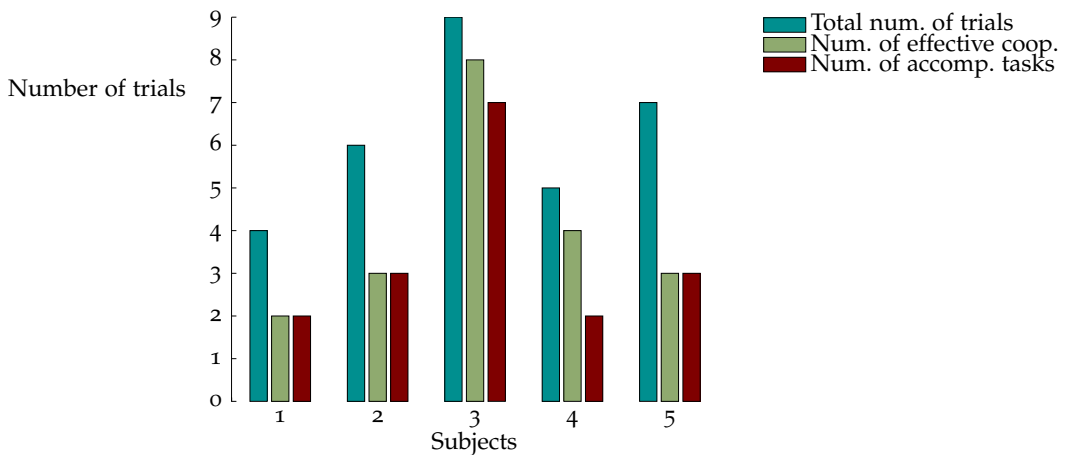


Figure 54: Impact of the cooperation in the success of the task.

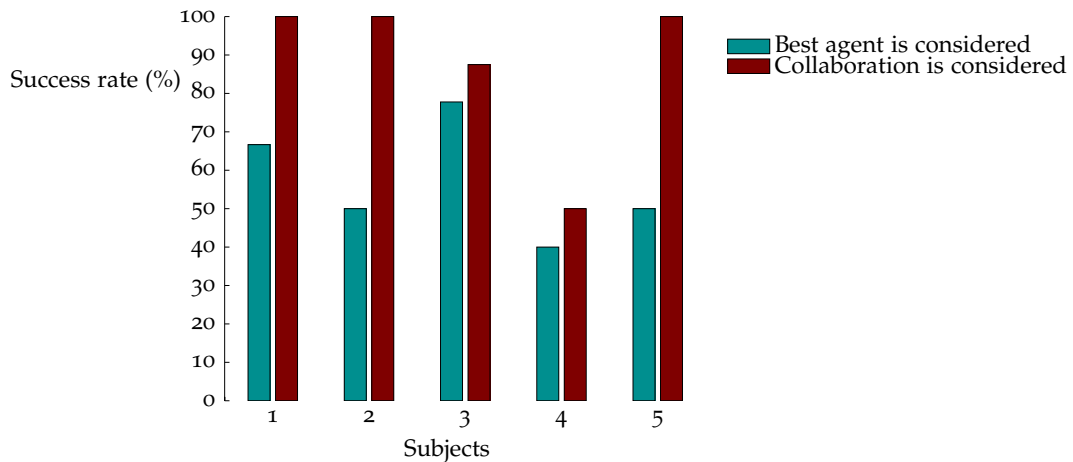


Figure 55: Comparison of the success rate.

illustrative cooperative tasks, but the platform provided enough versatility to validate the ideas proposed in this work within the chosen set of experiments.

One interesting finding was deduced from the *pouring water* experiment. In this kind of task, even if the robot is controlled as a slave, or a “follower”, humans have the tendency to regard it as an autonomous entity and they ultimately cooperate with the robot. Since the latter has several mechanical constraints, sometimes the task can be accomplished only if the person helps the robot. For instance, being aware of these limitations, people will tend to place the hand where the robot will effectively accomplish the task (or at least where they *believe* that the robot can accomplish it).

In the *teleoperation* task, the robot can be regarded as an extension of the operator’s body, since the person regards the robot as separate from his/her own body but uses the robot to perform the same actions as if he/she were physically placed at the robot’s location. In such tasks, the collaboration occurs when there is a third agent, the collaborator, that physically interacts with the robot. Because the person who controls the robot, the operator, is not present at the same environment where the robot and the collaborator are, the robot acts as an interactive interface between operator and collaborator. However, an effective collaboration will occur only if both collaborator and operator respect the robot’s constraints and limitations.

In the execution of *mirrored movements followed by simultaneous handling*, the robot cannot be regarded as either an autonomous entity or an extension of the human body. As the operator is directly controlling the robot, its status is not of an autonomous entity. On the other hand, the idea of “body extension” cannot be easily applied either, because the robot is mirroring the person instead of mimicking her. Hence, the person has to take into account the simultaneous aspect of the whole task—that is, controlling the robot while controlling her own body. Furthermore, even if the robot is not equipped with force control, in human-robot simultaneous handling, *there is* force control thanks to the natural force feedback mechanism present in the human body. In this kind of interaction, the system should be analyzed from a holistic point of view, where every agent provides something beneficial, which thus characterizes the collaboration. However, most tasks involving forces will still require that the robot be equipped with force controllers. Nevertheless, mirrored movements are useful because the human can use

the relation between his/her hand and the robot hand to position them at the desired location prior to simultaneous handling.

Despite the fact that the aforementioned tasks are conceptually different, mathematically they can be described in the same way; that is, by means of the relative configuration between the robot hand and human hand, and thus by using the cooperative variables.

The last experiment, the *ball in the hoop*, was performed by using a novel approach in HRC. In this approach, in addition to controlling its own arm, the robot controls the human arm by means of functional electrical stimulation. The experiment confirmed the hypotheses that the robot can control the human arm in order to achieve a desired coordination, and that good coordination leads to a higher success rate than just the performance of the best agent. Thus, the main conclusion is that two collaborators performing just “good enough” leads to a higher success rate than one very good agent interacting with a bad one. This has an important implication in assistive robotics, mainly when the robot must assist people with impairments who cannot control their upper limbs: the FES control of the human arm does not need to be precise, just good enough to bring the human arm into the robot’s workspace.

An interesting observation is that the addition of human arm control does not change the definition of the task. In fact, whenever the task is represented by means of the cooperative variables, there is an invariance with respect to which agents are controlled and even with respect to the controlled DOF. This invariance can be useful when developing general assistant robots, because the task is defined only once and then the robot can interact with several other agents, regardless of whether agents are other robots or people. In the case of healthy people, the robot would control only itself. In the case of people suffering from motor disabilities (e.g., quadriplegia), the robot could control the human arm if otherwise it would be impossible to accomplish the task—for example, when the robot cannot reach the human hand without controlling the human arm. Of course, this does not mean that the robot would control the subjects’ arms against their will. The robot would perform only the low-level control of the human arm, whereas the person would ultimately provide high-level commands (e.g., voice commands) to change or abort the task.

Although not explored in this thesis, the proposed HRC formalism is suitable for the integration with whole-body motion frameworks (e.g., Sentis, 2007; Mansard, 2006; Kannon, 2009), where the cooperative task would be first defined by using (5.1), and then the robot would continuously perform a whole-body motion in order to achieve the desired relative configuration between the human hand and the robot hand.

CONCLUSION AND PERSPECTIVES

This thesis has focused on the study of two-arm manipulation from a kinematic point of view, and both theoretical and practical aspects have been investigated. The research was motivated by applications in assistive and human-centered robotics, where two-arm manipulations are paramount. Nevertheless, an urge for unification has been present in every single aspect of the theory developed and, as a result, well-known results about robot kinematic modeling and control have inevitably been revisited. More specifically, dual quaternions have been the common tool used to perform kinematic modeling and control of serial robots, resulting in a compact theory.

Based on previous approaches (Uchiyama & Dauchez, 1988; Chiacchio et al., 1996; Caccavale et al., 2000), the formalism of two-arm manipulation has also been unified in the light of dual quaternions, resulting in the cooperative dual task-space; that is, dual positions (positions and orientations), twists (linear and angular velocities), and wrenches (forces and moments) have all been represented in dual quaternion space, and the resultant theory agrees with the approaches on which this thesis was based.

Thanks to this consolidation in the overall representation, generalizations of two-arm manipulation have been developed; namely, the representation of whole-body motion involving two-arm manipulations and human-robot collaboration. One of the main applications resulting from the latter case has been in human-robot collaboration, with the robot controlling the human arm by means of functional electrical stimulation. Although the performed experiment provides proof of the concept, it has also opened a new door toward handicapped-centered robots.

Substantial effort has been made to provide a rigorous mathematical presentation, both to ensure consistency and to provide a means for comparison with other representations. Illustrative examples and case studies have also been provided, such that the task of understanding the concepts proposed in the thesis could—hopefully—be alleviated.

Although the proposed techniques have been unified at the kinematic level, all dynamics have been neglected in this thesis. Further works will thus need to concentrate on the dynamic aspects within the cooperative dual task-space. A starting point is the operational image space (Dooley & McCarthy, 1993), which uses dual quaternions to represent Khatib's operational space (Khatib, 1987). This further unification could be advantageous, as the techniques designed for the symmetric cooperative task-space (Chiacchio et al., 1996; Caccavale et al., 2000; Uchiyama & Dauchez, 1988) and the augmented object/virtual linkage (Khatib, 1988; Williams & Khatib, 1993) could both be applied in a single formalism.

Some insights have been provided about the use of prioritized control within the cooperative dual task-space. More specifically, a simulated task was evaluated wherein two KUKA LWR 4 manipulated a broom, and the relative variables were controlled with higher priority than the absolute variables. The results of this simulation showed that such scheme could minimize internal forces, assuming perfect modeling, when

compared with the situation where the cooperative variables have the same priority. Nonetheless, further studies will be conducted in the context of prioritized position/force control schemes in order to apply them to the cooperative variables. A starting point for such an investigation would be the works of Siciliano & Slotine (1991), Chiaverini (1997), Mansard & Chaumette (2007), Caccavale et al. (2008), Mansard & Chaumette (2009), and Mansard et al. (2009).

The experiments of human-humanoid collaboration have shown that much work remains to be done. First, the robot needs to take into account whole-body movements in order to maintain the relative configuration of its hand and the human hand. Because kinematic control is not applicable to this case, dynamic whole-body control should be performed to take into account balance control and other possible constraints. For this reason, Sentis's dynamic whole-body motion techniques (Sentis, 2007) will be merged with the cooperative dual task-space.

Last, the cooperative dual task-space has proven its applicability in robotic two-arm manipulation and also in coordination between the human arm and robot arm. Further works will exploit *human two-arm coordination*; that is, the FES control of the human arm within the cooperative dual task-space framework. Such techniques could be applied to restore motion in quadriplegic patients, with the ultimate goal of two-arm manipulation, helping them to obtain some autonomy in their daily lives. In addition, hemiplegic patients could also benefit from therapies based on mirrored movements. In such therapies, it is believed that stroke patients can enhance the motor rehabilitation of their upper limbs when they perform mirrored movements (Lum et al., 2002). In some research on robotic rehabilitation, the patient uses the less affected arm to perform reference motions, while a manipulator robot coupled to the more affected arm uses these references to help the patient in the execution of mirrored movements (Lum et al., 2002). Using the techniques proposed in this thesis, the robot could instead be replaced by functional electrical stimulation, potentially leading to novel therapies.

APPENDIX

GENERAL MATHEMATICAL FACTS, DEFINITIONS, AND PROPERTIES

A.1 DUAL NUMBER PROPERTIES

Property 1 (Taylor expansion). *The Taylor expansion of the function $f(\underline{x}) = f(\mathcal{P}(\underline{x}) + \varepsilon \mathcal{D}(\underline{x}))$ at the point $\underline{x}_0 = \mathcal{P}(\underline{x})$ is (Gu & Luh, 1987):*

$$\begin{aligned} f(\underline{x}) &= \sum_{n=0}^{\infty} \frac{f^{(n)}(\underline{x}_0)}{n!} (\underline{x} - \underline{x}_0)^n \\ &= f(\mathcal{P}(\underline{x})) + \varepsilon \mathcal{D}(\underline{x}) f'(\mathcal{P}(\underline{x})) + (\varepsilon \mathcal{D}(\underline{x}))^2 \frac{f''(\mathcal{P}(\underline{x}))}{2} + \dots \\ &= f(\mathcal{P}(\underline{x})) + \varepsilon \mathcal{D}(\underline{x}) f'(\mathcal{P}(\underline{x})), \end{aligned} \tag{A.1}$$

where $f^{(n)}(\underline{x})$ denotes the n -th derivative of the function at $\mathcal{P}(\underline{x})$.

Property 2 (Dual number inverse (fact 2.1)).

The inverse of the dual number \underline{a} is

$$\underline{a}^{-1} \stackrel{(A.1)}{=} \frac{1}{\mathcal{P}(\underline{a})} - \varepsilon \frac{\mathcal{D}(\underline{a})}{\mathcal{P}(\underline{a})^2}, \quad \mathcal{P}(\underline{a}) \neq 0.$$

Proof. Let $f(\underline{x}) = \underline{x}^{-1}$. Since $f(\mathcal{P}(\underline{x})) = \frac{1}{\mathcal{P}(\underline{x})}$ and $f'(\mathcal{P}(\underline{x})) = -\frac{1}{\mathcal{P}(\underline{x})^2}$, using (A.1) with $\underline{x} = \underline{a}$ leads directly to the expression of \underline{a}^{-1} . \square

A.2 QUATERNION PROPERTIES

Let $\mathbf{h}_1, \mathbf{h}_2 \in \mathbb{H}$, the following facts hold:

Fact A.1. $\text{Re}(\mathbf{h}_1 \mathbf{h}_2) = \text{vec } \mathbf{h}_1^* \cdot \text{vec } \mathbf{h}_2$

Proof. Considering (2.2), this fact is verified by direct calculation. \square

Fact A.2. Let \mathbf{h} be a quaternion with unit norm; hence $\text{Re}(\dot{\mathbf{h}}\mathbf{h}^*) = \text{Re}(\mathbf{h}\dot{\mathbf{h}}^*) = 0$.

Proof. Take the first derivative of $\mathbf{h}\mathbf{h}^* = 1$

$$\begin{aligned} \dot{\mathbf{h}}\mathbf{h}^* + \mathbf{h}\dot{\mathbf{h}}^* &= \text{Re}(\dot{\mathbf{h}}\mathbf{h}^*) + \text{Re}(\mathbf{h}\dot{\mathbf{h}}^*) \\ &= \text{vec } \dot{\mathbf{h}}^* \cdot \text{vec } \mathbf{h}^* + \text{vec } \mathbf{h}^* \cdot \text{vec } \dot{\mathbf{h}}^* \\ &= 2 \text{Re}(\dot{\mathbf{h}}\mathbf{h}^*) = 2 \text{Re}(\mathbf{h}\dot{\mathbf{h}}^*) = 0. \end{aligned}$$

\square

Fact A.3. Let \mathbf{h} be a unit quaternion with the real part equal to zero; then $\mathbf{h}\mathbf{h} = -1$.

Proof. From definition 2.4, $\|\mathbf{h}\| = \sqrt{\mathbf{h}^*\mathbf{h}}$. Since \mathbf{h} has unit norm, $\mathbf{h}^*\mathbf{h} = 1$. Also, as $\text{Re}(\mathbf{h}) = 0$, the conjugate is $\mathbf{h}^* = -\mathbf{h}$; thus, $-\mathbf{h}\mathbf{h} = 1$, concluding the proof. \square

Fact A.4. If $\text{Re}(\mathbf{h}_1) = \text{Re}(\mathbf{h}_2) = 0$, then

$$\frac{\mathbf{h}_1\mathbf{h}_2 - \mathbf{h}_2\mathbf{h}_1}{2} = \mathbf{i}_m \cdot (\text{Im}(\mathbf{h}_1) \times \text{Im}(\mathbf{h}_2)) \\ \triangleq \mathbf{h}_1 \times \mathbf{h}_2.$$

Proof. Let $\mathbf{h}_1 = \hat{\mathbf{i}}a_1 + \hat{\mathbf{j}}b_1 + \hat{\mathbf{k}}c_1$ and $\mathbf{h}_2 = \hat{\mathbf{i}}a_2 + \hat{\mathbf{j}}b_2 + \hat{\mathbf{k}}c_2$; thus, by direct calculation,

$$\text{Im}(\mathbf{h}_1) \times \text{Im}(\mathbf{h}_2) = \begin{bmatrix} b_1c_2 - c_1b_2 \\ c_1a_2 - c_2a_1 \\ a_1b_2 - a_2b_1 \end{bmatrix}$$

and

$$\mathbf{h}_1\mathbf{h}_2 = -(a_1a_2 + b_1b_2 + c_1c_2) + \mathbf{i}_m \cdot (\text{Im}(\mathbf{h}_1) \times \text{Im}(\mathbf{h}_2)), \\ \mathbf{h}_2\mathbf{h}_1 = -(a_1a_2 + b_1b_2 + c_1c_2) - \mathbf{i}_m \cdot (\text{Im}(\mathbf{h}_1) \times \text{Im}(\mathbf{h}_2)),$$

which leads directly to

$$\frac{\mathbf{h}_1\mathbf{h}_2 - \mathbf{h}_2\mathbf{h}_1}{2} = \mathbf{i}_m \cdot (\text{Im}(\mathbf{h}_1) \times \text{Im}(\mathbf{h}_2)).$$

\square

Remark A.1. Fact A.4 is the equivalent of the cross product in quaternion space.

Definition A.1 (cross product). Let the operator $\overset{\times}{\mathbf{H}}(\mathbf{h})$ be

$$\overset{\times}{\mathbf{H}}(\mathbf{h}) \triangleq \frac{\overset{+}{\mathbf{H}}(\mathbf{h}) - \overset{-}{\mathbf{H}}(\mathbf{h})}{2}.$$

The equivalent of the vector cross product between two quaternions, \mathbf{h}_1 and \mathbf{h}_2 , with $\text{Re}(\mathbf{h}_1) = \text{Re}(\mathbf{h}_2) = 0$, by using the Hamilton operators, is the operation $\overset{\times}{\mathbf{H}}(\mathbf{h}_1) \text{vec } \mathbf{h}_2$; that is,

$$\overset{\times}{\mathbf{H}}(\mathbf{h}_1) \text{vec } \mathbf{h}_2 = \begin{bmatrix} b_1c_2 - c_1b_2 \\ c_1a_2 - c_2a_1 \\ a_1b_2 - a_2b_1 \end{bmatrix}.$$

Remark A.2. Definition A.1 follows directly from Fact A.4.

Remark A.3. Explicitly solving $\overset{\times}{\mathbf{H}}(\mathbf{h})$ for $\mathbf{h} = \hat{\mathbf{i}}a + \hat{\mathbf{j}}b + \hat{\mathbf{k}}c$ leads to

$$\overset{\times}{\mathbf{H}}(\mathbf{h}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -c & b \\ 0 & c & 0 & -a \\ 0 & -b & a & 0 \end{bmatrix}.$$

Fact A.5. Let $\mathbf{h}_i = \hat{\mathbf{i}}a_i + \hat{\mathbf{j}}b_i + \hat{\mathbf{k}}c_i$ and consider Remark A.3, then the following equations are easily verified

$$\begin{aligned}\overset{\times}{\mathbf{H}}(\alpha\mathbf{h}_i) &= \alpha\overset{\times}{\mathbf{H}}(\mathbf{h}_i) \\ \overset{\times}{\mathbf{H}}(\mathbf{h}_1 + \mathbf{h}_2) &= \overset{\times}{\mathbf{H}}(\mathbf{h}_1) + \overset{\times}{\mathbf{H}}(\mathbf{h}_2).\end{aligned}\tag{A.2}$$

Fact A.6. Consider the quaternion $\mathbf{r} = \cos\phi + \mathbf{n}\sin\phi$, with $\mathbf{n} = \hat{\mathbf{i}}n_x + \hat{\mathbf{j}}n_y + \hat{\mathbf{k}}n_z$ and $\|\mathbf{n}\| = 1$. Let $\boldsymbol{\omega} = \lambda\mathbf{n}$, then

$$\mathbf{r}\boldsymbol{\omega}\mathbf{r}^* = \mathbf{r}^*\boldsymbol{\omega}\mathbf{r} = \boldsymbol{\omega}.$$

Proof. The proof is given by direct calculation. Recall from fact A.3 that $\mathbf{n}\mathbf{n} = -1$. Expanding the left term, one obtains

$$\begin{aligned}\mathbf{r}\boldsymbol{\omega}\mathbf{r}^* &= (\cos\phi + \mathbf{n}\sin\phi)\lambda\mathbf{n}(\cos\phi - \mathbf{n}\sin\phi) \\ &= \lambda(\mathbf{n}\cos\phi - \sin\phi)(\cos\phi - \mathbf{n}\sin\phi) \\ &= \lambda(\mathbf{n}\cos^2\phi + \mathbf{n}\sin^2\phi) \\ &= \lambda\mathbf{n}.\end{aligned}$$

□

Fact A.7. Consider the unit quaternions $\mathbf{r}_1 = \cos\phi_1 + \mathbf{n}\sin\phi_1$ and $\mathbf{r}_2 = \cos\phi_2 + \mathbf{n}\sin\phi_2$; that is, quaternions that represent rotations $\phi_1/2$ and $\phi_2/2$ around the same axis \mathbf{n} . Hence,

$$\mathbf{r}_1\mathbf{r}_2 = \cos(\phi_1 + \phi_2) + \mathbf{n}\sin(\phi_1 + \phi_2).\tag{A.3}$$

Proof. Expanding the left-hand side gives

$$\mathbf{r}_1\mathbf{r}_2 = \cos\phi_1\cos\phi_2 + \mathbf{n}(\cos\phi_1\sin\phi_2 + \cos\phi_2\sin\phi_1) + \mathbf{n}\mathbf{n}\sin\phi_1\sin\phi_2.$$

Considering fact A.3 and the trigonometric identities related to $\cos(\phi_1 + \phi_2)$ and $\sin(\phi_1 + \phi_2)$, (A.3) is proven. □

Property 3 (Quaternion inverse). *The quaternion inverse is given by (Kuipers, 1999)*

$$\mathbf{h}^{-1} = \frac{\mathbf{h}^*}{\|\mathbf{h}\|^2}\tag{A.4}$$

Property 4 (Unit norm quaternions). *For unit norm quaternions, (A.4) is reduced to*

$$\mathbf{h}^{-1} = \mathbf{h}^*.\tag{A.5}$$

Hence

$$\mathbf{h}\mathbf{h}^* = \mathbf{h}^*\mathbf{h} = 1\tag{A.6}$$

Property 5 (Quaternion propagation).

The quaternion propagation equation is directly related with the derivative of the unit norm quaternion and is given by

$$\dot{\mathbf{r}}_1^0 = \frac{1}{2} \boldsymbol{\omega}_1^0 \mathbf{r}_1^0 \quad (\text{A.7})$$

Proof. Take the first derivative of (2.7) to obtain

$$\begin{aligned} \dot{\mathbf{p}}^0 &= \dot{\mathbf{r}}_1^0 \mathbf{p}^1 \mathbf{r}_1^{0*} + \mathbf{r}_1^0 \mathbf{p}^1 \dot{\mathbf{r}}_1^{0*} + \mathbf{r}_1^0 \dot{\mathbf{p}}^1 \mathbf{r}_1^{0*} \\ &= \dot{\mathbf{r}}_1^0 \mathbf{r}_1^{0*} \mathbf{p}^0 + \mathbf{p}^0 \mathbf{r}_1^0 \dot{\mathbf{r}}_1^{0*}. \end{aligned} \quad (\text{A.8})$$

Let $\boldsymbol{\alpha}_1^0 = \dot{\mathbf{r}}_1^0 \mathbf{r}_1^{0*}$, then (A.8) becomes

$$\dot{\mathbf{p}}^0 = \boldsymbol{\alpha}_1^0 \mathbf{p}^0 - \mathbf{p}^0 \boldsymbol{\alpha}_1^0. \quad (\text{A.9})$$

Considering fact A.2, $\text{Re}(\boldsymbol{\alpha}_1^0) = 0$; using definition A.1,

$$\dot{\mathbf{p}}^0 = 2\overset{\times}{\mathbf{H}}(\boldsymbol{\alpha}_1^0) \text{vec } \mathbf{p}^0.$$

Since the linear velocity is the cross product between the angular velocity and the position, it follows that

$$\dot{\mathbf{p}}^0 = \overset{\times}{\mathbf{H}}(\boldsymbol{\omega}_1^0) \text{vec } \mathbf{p}^0,$$

then

$$\frac{\overset{\times}{\mathbf{H}}(\boldsymbol{\omega}_1^0)}{2} = \overset{\times}{\mathbf{H}}(\boldsymbol{\alpha}_1^0).$$

Taking (A.2) into consideration leads to

$$\begin{aligned} \frac{\boldsymbol{\omega}_1^0}{2} &= \boldsymbol{\alpha}_1^0 \\ &= \dot{\mathbf{r}}_1^0 \mathbf{r}_1^{0*}. \end{aligned}$$

Finally, the quaternion propagation equation is

$$\dot{\mathbf{r}}_1^0 = \frac{1}{2} \boldsymbol{\omega}_1^0 \mathbf{r}_1^0$$

□

A.3 PROPERTIES OF HAMILTON OPERATORS

Fact A.8. *The following facts can be verified by direct calculation:*

1. $\overset{+}{\mathbf{H}}(\mathbf{h}^*) = \overset{+}{\mathbf{H}}(\mathbf{h})^T = \overset{+}{\mathbf{H}}(\mathbf{h})^{-1}$
2. $\overset{-}{\mathbf{H}}(\mathbf{h}^*) = \overset{-}{\mathbf{H}}(\mathbf{h})^T = \overset{-}{\mathbf{H}}(\mathbf{h})^{-1}$

Remark A.4. Facts A.8.1–2 have no analogous counterparts for dual quaternions. For instance, $\overset{+}{\mathbf{H}}(\underline{\mathbf{h}}^*) \neq \overset{+}{\mathbf{H}}(\underline{\mathbf{h}})^T$.

A.4 DUAL QUATERNION PROPERTIES

Property 6 (Analogy to standard complex numbers).

Since the real and imaginary parts of $\underline{\mathbf{h}} \in \mathcal{H}$ are a dual scalar and a dual vector,

$$\begin{aligned}\operatorname{Re}(\underline{\mathbf{h}}) &= \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) + \varepsilon \operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})), \\ \operatorname{Im}(\underline{\mathbf{h}}) &= \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}})) + \varepsilon \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}})),\end{aligned}$$

then

$$\underline{\mathbf{h}} = \operatorname{Re}(\underline{\mathbf{h}}) + \mathbf{i}_m \cdot \operatorname{Im}(\underline{\mathbf{h}}).$$

Proof. Recall that for $\underline{\mathbf{h}} \in \mathcal{H}$ and $\mathbf{h} \in \mathbb{H}$,

$$\begin{aligned}\underline{\mathbf{h}} &= \mathcal{P}(\underline{\mathbf{h}}) + \varepsilon \mathcal{D}(\underline{\mathbf{h}}), \\ \mathbf{h} &= \operatorname{Re}(\mathbf{h}) + \mathbf{i}_m \cdot \operatorname{Im}(\mathbf{h}).\end{aligned}$$

Hence,

$$\begin{aligned}\underline{\mathbf{h}} &= \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) + \mathbf{i}_m \cdot \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}})) + \varepsilon (\operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) + \mathbf{i}_m \cdot \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}}))) \\ &= \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) + \varepsilon \operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) + \mathbf{i}_m \cdot \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}})) + \varepsilon \mathbf{i}_m \cdot \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}})) \\ &= \operatorname{Re}(\underline{\mathbf{h}}) + \mathbf{i}_m \cdot \operatorname{Im}(\underline{\mathbf{h}}).\end{aligned}$$

□

Property 7 (Dual quaternion norm). *The norm of the dual quaternion $\underline{\mathbf{h}}$ is given by*

$$\begin{aligned}\|\underline{\mathbf{h}}\| &= \underline{\mathbf{h}}\mathbf{h}^* = \mathbf{h}^*\underline{\mathbf{h}} \\ &= \|\mathcal{P}(\underline{\mathbf{h}})\|^2 + 2\varepsilon \operatorname{vec} \mathcal{P}(\underline{\mathbf{h}}) \cdot \operatorname{vec} \mathcal{D}(\underline{\mathbf{h}}).\end{aligned}$$

Proof. Expand the first row to obtain

$$\begin{aligned}\underline{\mathbf{h}}^*\underline{\mathbf{h}} &= (\mathcal{P}(\underline{\mathbf{h}})^* + \varepsilon \mathcal{D}(\underline{\mathbf{h}})^*) (\mathcal{P}(\underline{\mathbf{h}}) + \varepsilon \mathcal{D}(\underline{\mathbf{h}})) \\ &= \mathcal{P}(\underline{\mathbf{h}})^* \mathcal{P}(\underline{\mathbf{h}}) + \varepsilon (\mathcal{P}(\underline{\mathbf{h}})^* \mathcal{D}(\underline{\mathbf{h}}) + \mathcal{D}(\underline{\mathbf{h}})^* \mathcal{P}(\underline{\mathbf{h}})).\end{aligned}$$

By definition 2.4, the primary part is obtained:

$$\mathcal{P}(\underline{\mathbf{h}}^*\underline{\mathbf{h}}) = \mathcal{P}(\underline{\mathbf{h}})^* \mathcal{P}(\underline{\mathbf{h}}) = \|\mathcal{P}(\underline{\mathbf{h}})\|^2.$$

The dual part is proved by direct calculation:

$$\begin{aligned}\mathcal{P}(\underline{\mathbf{h}})^* \mathcal{D}(\underline{\mathbf{h}}) &= (\operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) - \mathbf{i}_m \cdot \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}}))) (\operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) + \mathbf{i}_m \cdot \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}}))) \\ &= \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) \operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) + \mathbf{i}_m \cdot (\operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}})) - \operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}}))) \\ &\quad - (\mathbf{i}_m \cdot \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}}))) (\mathbf{i}_m \cdot \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}})))\end{aligned}$$

and

$$\begin{aligned}\mathcal{D}(\underline{\mathbf{h}})^* \mathcal{P}(\underline{\mathbf{h}}) &= \operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) + \mathbf{i}_m \cdot (\operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}})) - \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}}))) \\ &\quad - (\mathbf{i}_m \cdot \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}}))) (\mathbf{i}_m \cdot \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}}))).\end{aligned}$$

As $\mathcal{D}(\underline{\mathbf{h}}^* \underline{\mathbf{h}}) = \mathcal{P}(\underline{\mathbf{h}})^* \mathcal{D}(\underline{\mathbf{h}}) + \mathcal{D}(\underline{\mathbf{h}})^* \mathcal{P}(\underline{\mathbf{h}})$, then

$$\begin{aligned} \mathcal{D}(\underline{\mathbf{h}}^* \underline{\mathbf{h}}) &= 2 \operatorname{Re}(\mathcal{D}(\underline{\mathbf{h}})) \operatorname{Re}(\mathcal{P}(\underline{\mathbf{h}})) + 2 \operatorname{Im}(\mathcal{P}(\underline{\mathbf{h}})) \cdot \operatorname{Im}(\mathcal{D}(\underline{\mathbf{h}})) \\ &= 2 \operatorname{vec} \mathcal{P}(\underline{\mathbf{h}}) \cdot \operatorname{vec} \mathcal{D}(\underline{\mathbf{h}}). \end{aligned}$$

□

Property 8 (Dual quaternion inverse).

The dual quaternion inverse is given by

$$\underline{\mathbf{h}}^{-1} = \frac{\underline{\mathbf{h}}^*}{\|\underline{\mathbf{h}}\|}. \quad (\text{A.10})$$

Proof. The inverse must satisfy

$$\underline{\mathbf{h}}^{-1} \underline{\mathbf{h}} = 1 = \underline{\mathbf{h}} \underline{\mathbf{h}}^{-1} \quad (\text{A.11})$$

Using the first equality of (A.11)

$$\begin{aligned} \underline{\mathbf{h}}^{-1} \underline{\mathbf{h}} \underline{\mathbf{h}}^* &= \underline{\mathbf{h}}^* \\ \underline{\mathbf{h}}^{-1} \|\underline{\mathbf{h}}\| &= \underline{\mathbf{h}}^* \\ \underline{\mathbf{h}}^{-1} &= \underline{\mathbf{h}}^* \|\underline{\mathbf{h}}\|^{-1} \end{aligned} \quad (\text{A.12})$$

Analogously, the second equality of (A.11) results in

$$\underline{\mathbf{h}}^{-1} = \|\underline{\mathbf{h}}\|^{-1} \underline{\mathbf{h}}^*. \quad (\text{A.13})$$

As the dual quaternion norm is a dual scalar, and the multiplication of dual quaternions with dual scalars is commutative, hence the dual quaternion inverse expressed by (A.12) and (A.13) can be given simply by

$$\underline{\mathbf{h}}^{-1} = \frac{\underline{\mathbf{h}}^*}{\|\underline{\mathbf{h}}\|}. \quad (\text{A.14})$$

Interestingly, (A.10) is the dual version of the quaternion inverse given by (A.4). □

A.4.1 Unit dual quaternions and rigid motions

Section 2.2.3 has shown that dual quaternions can be used to represent rigid motions. Indeed, a unit dual quaternion represents a Plücker line with respect to two different frames and, as a consequence, a unit dual quaternion represents the point transformation between these two frames as well.

A Plücker line (Bottema & Roth, 1979), can be represented by the dual quaternion

$$\underline{\mathbf{l}} = \mathbf{l} + \varepsilon \mathbf{m},$$

where $\operatorname{Re}(\underline{\mathbf{l}}) = 0$, and the quaternion \mathbf{l} represents the direction of the line that passes through a point $\mathbf{p} \in \mathbb{H}$, and \mathbf{m} is the line moment; that is,

$$\mathbf{m} = \mathbf{p} \times \mathbf{l}.$$

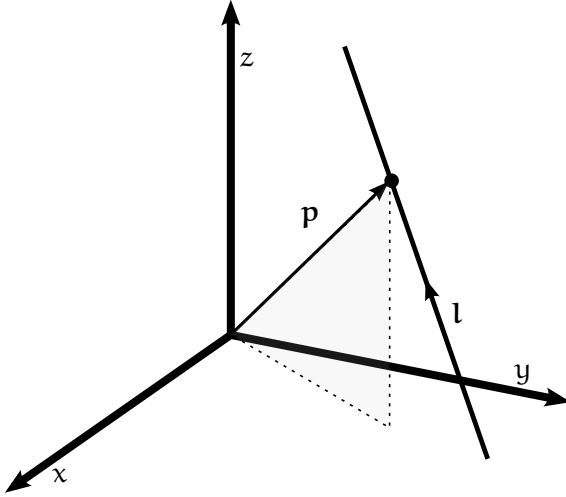


Figure 56: Plücker line.

Whereas \mathbf{l} represents the orientation of the line, the quaternion \mathbf{m} determines its spatial location. Figure 56 illustrates the representation of a Plücker line.

In the next paragraphs, first a Plücker line is represented with respect to two different frames; then, the result is used to show the existence of a unit dual quaternion that corresponds to a transformation between these two frames.

Consider two frames \mathcal{F}_a and \mathcal{F}_b , a reference frame \mathcal{F} , and the Plücker line $\underline{\mathbf{l}} = \mathbf{l} + \varepsilon \mathbf{m}$ shown in figure 57. The Plücker line with respect to \mathcal{F}_b is $\underline{\mathbf{l}}^b = \mathbf{l}^b + \varepsilon \mathbf{m}^b$ and, using the frame transformation (2.7) and fact A.4, one obtains

$$\mathbf{l}^b = \mathbf{r}_b^* \mathbf{l} \mathbf{r}_b$$

and

$$\mathbf{m}^b = \frac{1}{2} (\mathbf{p}^b \mathbf{l}^b - \mathbf{l}^b \mathbf{p}^b).$$

From figure 57, one can see that $\mathbf{p}^b = \mathbf{r}_b^{a*} \mathbf{p}^a \mathbf{r}_b^a - \mathbf{p}_{a,b}^b$, and thus

$$\mathbf{m}^b = \frac{1}{2} [(\mathbf{r}_b^{a*} \mathbf{p}^a \mathbf{r}_b^a - \mathbf{p}_{a,b}^b) \mathbf{l}^b - \mathbf{l}^b (\mathbf{r}_b^{a*} \mathbf{p}^a \mathbf{r}_b^a - \mathbf{p}_{a,b}^b)]$$

Using the frame transformation $\mathbf{l}^b = \mathbf{r}_b^{a*} \mathbf{l}^a \mathbf{r}_b^a$,

$$\mathbf{m}^b = \mathbf{r}_b^{a*} \mathbf{m}^a \mathbf{r}_b^a + \frac{1}{2} (\mathbf{r}_b^{a*} \mathbf{l}^a \mathbf{r}_b^a \mathbf{p}_{a,b}^b - \mathbf{p}_{a,b}^b \mathbf{r}_b^{a*} \mathbf{l}^a \mathbf{r}_b^a) \quad (\text{A.15})$$

The goal is to find the dual quaternion $\underline{\mathbf{h}}_b^a$ that satisfies $\underline{\mathbf{l}}^b = (\underline{\mathbf{h}}_b^a)^* \underline{\mathbf{l}}^a \underline{\mathbf{h}}_b^a$, where $\underline{\mathbf{l}}^a$ and $\underline{\mathbf{l}}^b$ are the Plücker line with respect to frames \mathcal{F}_a and \mathcal{F}_b , respectively; thus,

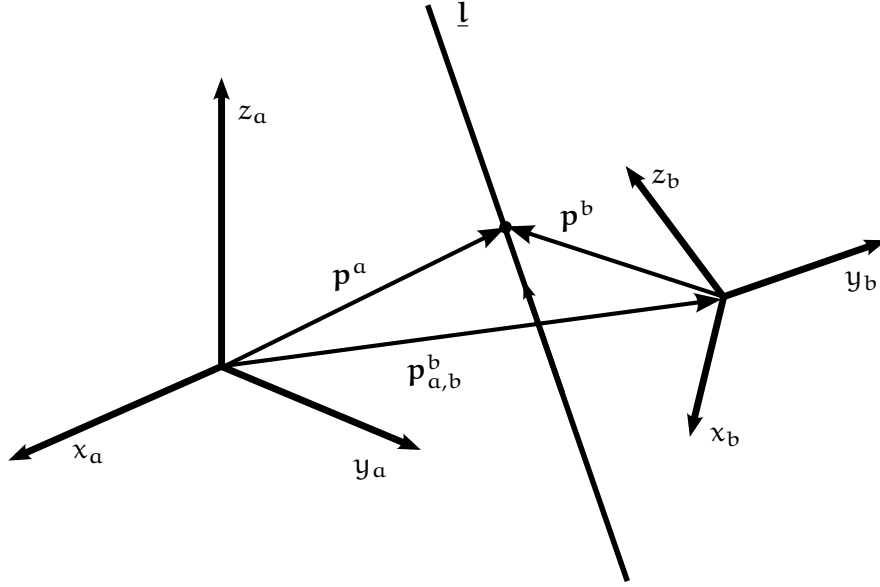
$$\underline{\mathbf{l}}^b = (\underline{\mathbf{h}}_b^a)^* \underline{\mathbf{l}}^a \underline{\mathbf{h}}_b^a \quad (\text{A.16})$$

$$\mathbf{l}^b + \varepsilon \mathbf{m}^b = (\mathcal{P}(\underline{\mathbf{h}}_b^a)^* + \varepsilon \mathcal{D}(\underline{\mathbf{h}}_b^a)^*) (\mathbf{l}^a + \varepsilon \mathbf{m}^a) (\mathcal{P}(\underline{\mathbf{h}}_b^a) + \varepsilon \mathcal{D}(\underline{\mathbf{h}}_b^a)) \quad (\text{A.17})$$

$$= \mathcal{P}(\underline{\mathbf{h}}_b^a)^* \mathbf{l}^a \mathcal{P}(\underline{\mathbf{h}}_b^a) + \varepsilon \mathbf{m}^b, \quad (\text{A.18})$$

where

$$\mathbf{m}^b = \mathcal{P}(\underline{\mathbf{h}}_b^a)^* \mathbf{l}^a \mathcal{D}(\underline{\mathbf{h}}_b^a) + \mathcal{P}(\underline{\mathbf{h}}_b^a)^* \mathbf{m}^a \mathcal{P}(\underline{\mathbf{h}}_b^a) + \mathcal{D}(\underline{\mathbf{h}}_b^a)^* \mathbf{l}^a \mathcal{P}(\underline{\mathbf{h}}_b^a). \quad (\text{A.19})$$

Figure 57: Plücker line with respect to frames \mathcal{F}_a and \mathcal{F}_b .

From (A.18), $\underline{l}^b = \mathcal{P}(\underline{\mathbf{h}}_b^a)^* \underline{l}^a \mathcal{P}(\underline{\mathbf{h}}_b^a)$, and hence $\mathcal{P}(\underline{\mathbf{h}}_b^a) = \mathbf{r}_b^a$. Compare (A.15) and (A.19) to obtain

$$\begin{aligned} \frac{1}{2} [\mathcal{P}(\underline{\mathbf{h}}_b^a)^* \underline{l}^a \mathcal{P}(\underline{\mathbf{h}}_b^a) \mathbf{p}_{a,b}^b - \mathbf{p}_{a,b}^b \mathcal{P}(\underline{\mathbf{h}}_b^a)^* \underline{l}^a \mathcal{P}(\underline{\mathbf{h}}_b^a)] \\ = \mathcal{P}(\underline{\mathbf{h}}_b^a)^* \underline{l}^a \mathcal{D}(\underline{\mathbf{h}}_b^a) + \mathcal{D}(\underline{\mathbf{h}}_b^a)^* \underline{l}^a \mathcal{P}(\underline{\mathbf{h}}_b^a), \end{aligned}$$

which implies

$$\mathcal{D}(\underline{\mathbf{h}}_b^a) = \frac{1}{2} \mathcal{P}(\underline{\mathbf{h}}_b^a) \mathbf{p}_{a,b}^b. \quad (\text{A.20})$$

Because $\mathcal{P}(\underline{\mathbf{h}}_b^a) = \mathbf{r}_b^a$ and thanks to the frame transformation (2.7),

$$\mathbf{p}_{a,b}^b = (\mathbf{r}_b^a)^* \mathbf{p}_{a,b}^a \mathbf{r}_b^a,$$

and thus

$$\underline{\mathbf{h}}_b^a = \mathbf{r}_b^a + \varepsilon \frac{1}{2} \mathbf{p}_{a,b}^a \mathbf{r}_b^a. \quad (\text{A.21})$$

A.5 PROPERTIES OF THE GENERALIZED JACOBIAN

Fact A.9. *Let the generalized Jacobian be partitioned as*

$$\mathbf{G} = 2 \begin{bmatrix} \Psi_L & \Psi_R \\ \Omega_L & 0_4 \end{bmatrix},$$

then its inverse is

$$\mathbf{G}^{-1} = \frac{1}{2} \begin{bmatrix} 0_4 & \Omega_L^T \\ \Omega_L^T & -\Omega_L^T \Psi_L \Omega_L^T \end{bmatrix}.$$

Proof. Since the matrix inverse must satisfy

$$\mathbf{G}\mathbf{G}^{-1} = \mathbf{1}, \quad (\text{A.22})$$

let

$$\mathbf{G}^{-1} = \begin{bmatrix} \mathbf{G}_{1,1} & \mathbf{G}_{1,2} \\ \mathbf{G}_{2,1} & \mathbf{G}_{2,2} \end{bmatrix},$$

and, observing that $\Psi_{\mathbf{R}} = \Omega_{\mathbf{L}}$, solve (A.22) to obtain $\mathbf{G}_{1,1} = \mathbf{0}_4$, $\mathbf{G}_{1,2} = \Omega_{\mathbf{L}}^{-1}$, $\mathbf{G}_{2,1} = \Omega_{\mathbf{L}}^{-1}$, $\mathbf{G}_{2,2} = -\Omega_{\mathbf{L}}^{-1}\Psi_{\mathbf{L}}\Omega_{\mathbf{L}}^{-1}$. Because $\Omega_{\mathbf{L}}$ is a Hamilton operator, the relation $\Omega_{\mathbf{L}}^{-1} = \Omega_{\mathbf{L}}^{\mathbf{T}}$ holds (see Fact A.8). \square

A.6 PROPERTIES OF THE DECOMPOSITIONAL MULTIPLICATION

Fact A.10 (Priority over the standard multiplication).

$$\begin{aligned} \underline{\mathbf{h}}_1 \underline{\mathbf{h}}_2 \otimes \underline{\mathbf{h}}_3 &= \underline{\mathbf{h}}_1 (\underline{\mathbf{h}}_2 \otimes \underline{\mathbf{h}}_3) \\ &\neq (\underline{\mathbf{h}}_1 \underline{\mathbf{h}}_2) \otimes \underline{\mathbf{h}}_3 \end{aligned}$$

Fact A.11 (Associativity).

$$(\underline{\mathbf{h}}_1 \otimes \underline{\mathbf{h}}_2) \otimes \underline{\mathbf{h}}_3 = \underline{\mathbf{h}}_1 \otimes (\underline{\mathbf{h}}_2 \otimes \underline{\mathbf{h}}_3)$$

Fact A.12 (Identity).

$$\underline{\mathbf{h}} \otimes \mathbf{1} = \mathbf{1} \otimes \underline{\mathbf{h}} = \underline{\mathbf{h}}$$

Fact A.13 (Commutativity).

$$\begin{aligned} \underline{\mathbf{t}}(\underline{\mathbf{h}}_1) \otimes \underline{\mathbf{h}}_2 &= \underline{\mathbf{h}}_2 \otimes \underline{\mathbf{t}}(\underline{\mathbf{h}}_1) \\ &= \underline{\mathbf{t}}(\underline{\mathbf{h}}_1) \underline{\mathbf{h}}_2 \end{aligned}$$

Fact A.10 is given directly by definition 2.17 and facts A.11–A.13 can be verified by direct calculation.

DENAVIT-HARTENBERG PARAMETERS OF THE ROBOTS

Table 8: Standard D-H parameters of the KUKA LWR 4 (Giordano, 2007).

i	θ_i [rad]	d_i [m]	a_i [m]	α_i [rad]
1	0	0.31	0	$\pi/2$
2	0	0	0	$-\pi/2$
3	0	0.4	0	$-\pi/2$
4	0	0	0	$\pi/2$
5	0	0.39	0	$\pi/2$
6	0	0	0	$-\pi/2$
7	0	0	0	0

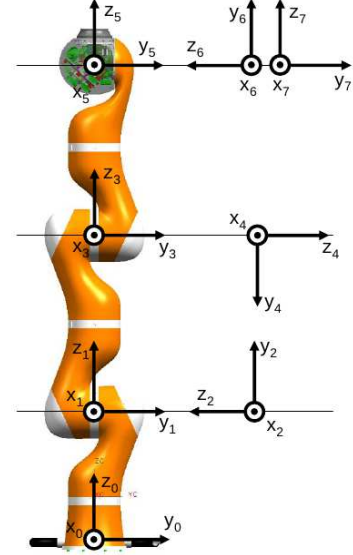


Table 9: Modified D-H parameters of the Hoap-3's arms.

	i	α_i [rad]	a_i [m]	θ_i [rad]	d_i [m]
Left	1	$-\pi/2$	0	θ_1	0.111
	2	$\pi/2$	0	$\theta_2 + \pi/2$	0
	3	$\pi/2$	0	$\theta_3 + \pi/2$	0.111
	4	$\pi/2$	0	θ_4	0
	End-effector	$-\pi/2$	0	-	0.171
Right	1	$-\pi/2$	0	θ_1	-0.111
	2	$\pi/2$	0	$\theta_2 + \pi/2$	0
	3	$\pi/2$	0	$\theta_3 + \pi/2$	0.111
	4	$\pi/2$	0	θ_4	0
	End-effector	$-\pi/2$	0	-	0.171

These D-H parameters are different from the ones of the Hoap-3's documentation (but they are equivalent).

EXPERIMENTAL SETUP FOR THE HUMAN-ROBOT COLLABORATION

C.1 “TELEOPERATION WITH COLLABORATION” EXPERIMENT

As seen in figure 43 on page 98, the robot hand is related to \mathcal{F}_T and the pose of the human hand is related to \mathcal{F}_M , the coordinate system of the motion tracker. However, (5.1) on page 96 requires that the poses of both human hand and robot hand be expressed with respect to a common frame, and thus the relation between \mathcal{F}_M and \mathcal{F}_T must be known.

Instead of obtaining the exact relation between \mathcal{F}_M and \mathcal{F}_T , which can be tedious and difficult in practice, the goal is to use a methodology that gives a reasonable invariance with respect to the physical placement of the robot and the motion tracker. In order to develop the method, first consider a teleoperation task, where the robot imitates the subject’s movement—afterward, this idea will be extended to other cooperative tasks.

In the teleoperation task, the robot has to perform—with respect to its own torso—the same movement that the subject does with respect to his/her torso. In this case, the task is defined as $\underline{x}_{\text{task}} = 1$, which implies $\underline{x}_H = \underline{x}_R$. In addition, the initial poses of both the robot hand and human hand¹, $\underline{x}_{R_0}^T$ and $\underline{x}_{H_0}^M$, are the same pose with respect to their own torsos.

In order to address this problem of invariance, first a virtual torso \mathcal{F}_{T_v} is defined at the same origin of \mathcal{F}_M , but aligned with \mathcal{F}_T , as illustrated in figure 58, and the frame $\mathcal{F}_{H_{v0}}$ is defined such that $\underline{x}_{R_0}^T = \underline{x}_{H_{v0}}^{T_v}$. Defining these two virtual frames is important in order to achieve invariance with respect to translational displacement between the robot, the motion tracker, and the person. Indeed, for practical reasons, it is not easy to superpose \mathcal{F}_{H_0} to $\mathcal{F}_{H_{v0}}$, meaning that $\underline{x}_{H_v}^{T_v} \neq \underline{x}_H^{T_v}$, and hence $\underline{x}_H^{T_v}$ cannot simply be used as the reference to be passed to the robot.

In order to ensure that the movement performed by the subject with respect to its initial configuration \mathcal{F}_{H_0} will be replicated by the robot, $\mathcal{F}_{H_{v0}}$ is used as the reference. The decompositional multiplication turns out to be quite useful in this case, since the goal is to represent $\underline{x}_H^{H_0}$ considering the transformation $\underline{x}_{H_{v0}}^{H_0}$; that is, the frame $\mathcal{F}_{H_{v0}}$ must be modified by the transformation $\underline{x}_H^{H_0}$, but using \mathcal{F}_{H_0} as the reference frame. Thus,

$$\underline{x}_{H_v}^{H_0} = \underline{x}_{H_{v0}}^{H_0} \otimes \underline{x}_H^{H_0}.$$

Based on this information, $\underline{x}_{H_v}^{T_v}$ can be computed as

$$\underline{x}_{H_v}^{T_v} = \underline{x}_{H_0}^{T_v} \underline{x}_{H_v}^{H_0}, \quad (\text{C.1})$$

1. Incidentally, the motion tracker provides the rotation matrix \mathbf{R}_H^M —from where the quaternion \mathbf{r}_H^M is extracted—and the translation \mathbf{p}_H^M . The dual quaternion representing the pose of the marker with respect to the motion tracker is $\underline{x}_H^M = \mathbf{r}_H^M + \varepsilon(1/2)\mathbf{p}_H^M \mathbf{r}_H^M$.

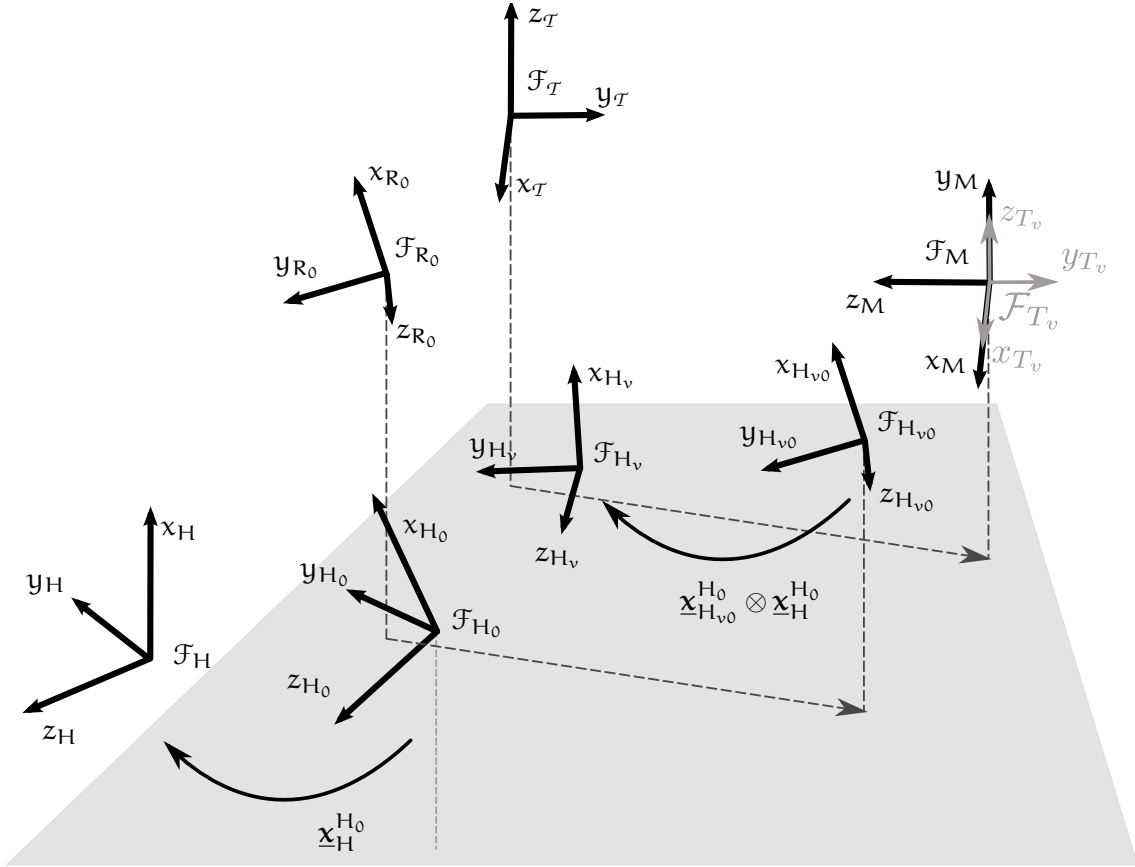


Figure 58: Representation of the frames in the “teleoperation with collaboration” task.

where

$$\underline{x}_{H_0}^{T_v} = (\mathbf{r}_{T_v}^M)^* \underline{x}_{H_0}^M. \quad (\text{C.2})$$

Finally, it is important to underline that, in order to calculate $\underline{x}_{H_0}^{T_v}$, the relationship between \mathcal{F}_T and \mathcal{F}_M is not completely specified. In fact, as shown by (C.2), only the rotation $\mathbf{r}_{T_v}^M$ must be specified *a priori*, because the system is invariant with respect to the translational displacement between the robot’s torso and the motion tracker.

C.2 “WATER POURING” AND “MIRRORED MOVEMENTS FOLLOWED BY SIMULTANEOUS HANDLING” EXPERIMENTS

The previous section developed a reasoning about the need of obtaining an invariance with respect to the physical placement between the robot and the motion tracker. This reasoning was necessary only because the robot hand and the human hand are expressed in different coordinate systems, and it is not practical to find a precise relation between them. Furthermore, the teleoperation task was used as an example to illustrate how to obtain this invariance.

With respect to the other tasks, the only difference is that the initialization phase assumes that the person is facing the robot, as illustrated in figure 43; hence, $\underline{x}_{H_{v0}}^{T_v} = \underline{x}_{R_{v0}}^{T_v} \mathbf{r}_{x,\pi}$. The algorithm C.1 summarizes both the initialization phase and the execution

Algorithm C.1 Cooperation algorithm

-
- 1: Define $\mathbf{x}_{T_v}^M$ according to the configuration between the robot and the motion tracker.
 - 2: $\mathbf{x}_{H_0}^{T_v} \leftarrow (\mathbf{r}_{T_v}^M)^* \mathbf{x}_{H_0}^M$
 - 3: $\mathbf{r}_{\text{inv}} = \begin{cases} 1 & \text{if teleoperation} \\ \mathbf{r}_{\chi, \pi} & \text{otherwise} \end{cases}$
 - 4: $\mathbf{x}_{H_{v0}}^{H_0} \leftarrow (\mathbf{x}_{H_0}^{T_v})^* \mathbf{x}_{R_{v0}}^{T_v} \mathbf{r}_{\text{inv}}$
 - 5: **while not** *end_of_cooperation* **do**
 - 6: $\mathbf{x}_H^{H_0} \leftarrow (\mathbf{x}_{H_0}^M)^* \mathbf{x}_H^M$
 - 7: $\mathbf{x}_{H_v}^{T_v} \leftarrow \mathbf{x}_{H_0}^{T_v} (\mathbf{x}_{H_{v0}}^{H_0} \otimes \mathbf{x}_H^{H_0})$
 - 8: $\boldsymbol{\theta}_{R,k} = \boldsymbol{\theta}_{R,k-1} + \mathbf{J}_R^+ \mathbf{K} \text{vec} (\mathbf{x}_{H_v}^{T_v} \mathbf{x}_{\text{task}}^* - \mathbf{x}_{R_m}^T)$
 - 9: **end while**
-

of the cooperation tasks. Note that the initial position of the subject—facing the robot or mimicking it—is given by the parameter \mathbf{r}_{inv} . This parameter is not used to describe the task. Instead, it is used only in the initialization, since the robot has no information (in the current configuration of the experiment) about the initial position of the human’s hand.

Also, the motion tracker was placed on the robot’s left, such that (see fig. 50 on page 106 and fig. 58)

$$\mathbf{r}_{T_v}^M = \mathbf{r}_T^M = \mathbf{r}_{\chi, (-\pi/2)}. \quad (\text{C.3})$$

Recall that (C.3) does not include the information about the translation because the system setup is invariant with respect to the translational displacement between the robot torso and the motion tracker.

C.3 CALIBRATION PROCEDURE FOR THE “BALL IN THE HOOP” EXPERIMENT

In order to simplify the experiment, the robot is fixed with respect to the motion tracker, and the subjects’ elbows are located roughly at the same place and with the same configuration during the whole experiment, as shown in figure 51a on page 109. Because \mathbf{x}_H^M and \mathbf{x}_R^T are given with respect to the motion tracker and the robot’s torso, respectively, a calibration procedure is executed in order to find the relation between \mathcal{F}_M and \mathcal{F}_T ; that is, \mathbf{x}_T^M .

First, the marker is put on the robot hand such that their coordinate systems coincide (e.g., $\mathcal{F}_R \leftarrow \mathcal{F}_{\text{marker}}$), and the measure $\mathbf{x}_{R_0}^M = \mathbf{x}_{\text{marker}}^M$ is stored. Then the marker is put on the object and the subject places his/her arm in the initial configuration while holding the object. In this case, $\mathcal{F}_H \leftarrow \mathcal{F}_{\text{marker}}$, and the measure $\mathbf{x}_{H_0}^M$ is stored.

The relation between the \mathcal{F}_T and \mathcal{F}_M is then given by

$$\mathbf{x}_T^M = \mathbf{x}_{R_0}^M (\mathbf{x}_{R_0}^T)^*. \quad (\text{C.4})$$

Algorithm C.2 Calibration

- 1: Initialize the system with the configuration depicted in fig. 51a on page 109.
- 2: Align the marker with the robot hand, then with the human elbow, human wrist, and finally with the object, obtaining the measures $\underline{x}_{R_0}^M$, $\underline{x}_{\text{elbow}}^M$, $\underline{x}_{\text{wrist}}^M$ and $\underline{x}_{H_0}^M$.
- 3: Place the ball at the desired relative configuration with respect to the hoop and get the measure $\underline{x}_{H_{\text{task}}}^M$.
- 4: Calculate and store:

$$\begin{aligned}\underline{x}_T^M &\leftarrow \underline{x}_{R_0}^M \left(\underline{x}_{R_0}^T \right)^* \\ \underline{x}_{\text{task}} &\leftarrow \left(\underline{x}_{R_0}^M \right)^* \underline{x}_{H_{\text{task}}}^M \\ \underline{x}_{\text{wrist}}^H &\leftarrow \left(\underline{x}_{H_0}^M \right)^* \underline{x}_{\text{wrist}}^M\end{aligned}$$

The procedure is repeated in the same manner with the subject's elbow and wrist to obtain $\underline{x}_{\text{elbow}}^M$ and $\underline{x}_{\text{wrist}}^M$, respectively. The relation between the grasped object and the wrist is also considered constant and given by

$$\underline{x}_{\text{wrist}}^H = \left(\underline{x}_{H_0}^M \right)^* \underline{x}_{\text{wrist}}^M \quad (\text{C.5})$$

Finally, in order to define the task, the subject's hand is positioned with respect to the robot hand and the measure $\underline{x}_{H_{\text{task}}}^M$ is stored. Because the desired task is also constant and defined in terms of the relative dual position between the robot hand and the grasped object, then

$$\underline{x}_{\text{task}} = \left(\underline{x}_{R_0}^M \right)^* \underline{x}_{H_{\text{task}}}^M \quad (\text{C.6})$$

The calibration procedure, summarized in algorithm C.2, is performed just once and with just one subject. Also, it is important to underline that throughout the experiment (but not during the calibration process) the marker was fixed on the ball's surface (see fig. 50 on page 106); thus, $\mathcal{F}_H \leftarrow \mathcal{F}_{\text{marker}}$ and $\underline{x}_H^M = \underline{x}_{\text{marker}}^M$.

BIBLIOGRAPHY

- Adams, J., Bajcsy, R., Kosecka, J., Kumar, V., Mandelbaum, R., Mintz, M., Paul, R., Wang, C., Yamamoto, Y., & Yun, X. (1995). Cooperative material handling by human and robotic agents: module development and system synthesis. *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, 11(2), 200–205.
- Adorno, B. V., Bó, A. P. L., & Fraitse, P. (2011a). Interactive Manipulation Between a Human and a Humanoid : When Robots Control Human Arm Motion. In *2011 Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Adorno, B. V., Bó, A. P. L., Fraitse, P., & Poignet, P. (2011b). Towards a Cooperative Framework for Interactive Manipulation Involving a Human and a Humanoid. In *2011 IEEE International Conference on Robotics and Automation*.
- Adorno, B. V. & Borges, G. A. (2009). iARW: An incremental path planner algorithm based on adaptive random walks. In *The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 988–993). St. Louis, USA.
- Adorno, B. V. & Borges, G. A. (2011). Planejamento de Rotas. In D. Wolf, F. Osório, E. Prestes, & R. A. F. Romero (Eds.), *Robótica Móvel* chapter 5. Editora Elsevier/LTC.
- Adorno, B. V. & Fraitse, P. (2011). “Extended cooperative dual task-space and its application to interactive manipulation in mobile manipulators (submitted for publication)”. *Advanced Robotics*.
- Adorno, B. V., Fraitse, P., & Druon, S. (2010). Dual position control strategies using the cooperative dual task-space framework. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 3955–3960). Taipei: IEEE.
- Akyar, B. (2008). Dual quaternions in spatial kinematics in an algebraic sense. *Turk. J. Math.*, 32(4), 373–391.
- Andaluz, V., Roberti, F., & Carelli, R. (2010). Robust control with redundancy resolution and dynamic compensation for mobile manipulators. *2010 IEEE International Conference on Industrial Technology*, 1109(5400), 1469–1474.
- Asfour, T., Regenstein, K., Azad, P., Schroder, J., Bierbaum, A., Vahrenkamp, N., & Dillmann, R. (2006). ARMAR-III: An Integrated Humanoid Platform for Sensory-Motor Control. *2006 6th IEEE-RAS International Conference on Humanoid Robots*, (pp. 169–175).
- Bernardes, M. C., Adorno, B. V., Poignet, P., Zemiti, N., & Borges, G. A. (2011a). Adaptive path planning for steerable needles using duty-cycling. In *2011 Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Bernardes, M. C., Adorno, B. V., Zemiti, N., Poignet, P., & Borges, G. A. (2011b). Path planning for steerable needles using duty-cycling. In *CARS 2011 Computer Assisted Radiology and Surgery* Berlin.

- Bluethmann, W., Ambrose, R., Diftler, M., Askew, S., Huber, E., Goza, M., Rehnmark, F., Lovchik, C., & Magruder, D. (2003). Robonaut: A Robot Designed to Work with Humans in Space. *Autonomous Robots*, 14(2-3), 179–197.
- Bó, A. P. L. (2010). *Compensation active de tremblements pathologiques des membres supérieurs via la stimulation électrique fonctionnelle*. Phd thesis, Université Montpellier 2.
- Bohren, J., Rusu, R., Jones, E., Marder-Eppstein, E., Pantofaru, C., Wise, M., Mösenlechner, L., Meeussen, W., Holzer, S., & Garage, W. (2011). Towards Autonomous Robotic Butlers: Lessons Learned with the PR2. In *IEEE International Conference on Robotics and Automation* (pp. 5568–5575). Shanghai: IEEE.
- Borst, C., Wimbock, T., Schmidt, F., Fuchs, M., Brunner, B., Zacharias, F., Giordano, P., Konietzschke, R., Sepp, W., Fuchs, S., & Others (2009). Rollin' Justin-Mobile platform with variable base. In *IEEE International Conference on Robotics and Automation, ICRA'09* (pp. 1597–1598): IEEE.
- Bottema, O. & Roth, B. (1979). *Theoretical kinematics*, volume 24. North-Holland Publishing Company.
- Bradski, G. & Kaehler, A. (2008). *Learning OpenCV - Computer Vision with the OpenCV Library*. O'Reilly, first ed. edition.
- Breazeal, C., Takanishi, A., & Kobayashi, T. (2008). Social Robots that Interact with People. In B. Siciliano & O. Khatib (Eds.), *Handbook of Robotics* (pp. 1349–1369): Springer.
- Bronshstein, I., Semendyayev, K., Hirsch, K., & Mühlig, H. (2007). *Handbook of mathematics*. Springer New York, NY, 5th edition.
- Caccavale, F., Chiacchio, P., & Chiaverini, S. (2000). Task-space regulation of cooperative manipulators. *Automatica*, 36(6), 879–887.
- Caccavale, F., Chiacchio, P., Marino, A., & Villani, L. (2008). Six-DOF Impedance Control of Dual-Arm Cooperative Manipulators. *IEEE/ASME Transactions on Mechatronics*, 13(5), 576–586.
- Caccavale, F., Natale, C., Siciliano, B., & Villani, L. (1999). Six-DOF impedance control based on angle/axis representations. *IEEE Transactions on Robotics and Automation*, 15(2), 289–300.
- Caccavale, F. & Uchiyama, M. (2008). *Handbook of Robotics*, chapter 29. Cooper, (pp. 701–718). Springer.
- Campa, R. & de la Torre, H. (2009). Pose control of robot manipulators using different orientation representations: A comparative review. In *2009 American Control Conference*, number 1 (pp. 2855–2860): IEEE.
- Carriker, W., Khosla, P., & Krogh, B. (1989). An approach for coordinating mobility and manipulation. In *1989 IEEE International Conference on Systems Engineering* (pp. 59–63): IEEE.

- Cheng, P., Kumar, V., Arkin, R., Steinberg, M., & Hedrick, K. (2009). Cooperative control of multiple heterogeneous unmanned vehicles for coverage and surveillance [From the guest editors]. *IEEE Robotics & Automation Magazine*, 16(2), 12–12.
- Chiacchio, P., Chiaverini, S., & Siciliano, B. (1996). Direct and Inverse Kinematics for Coordinated Motion Tasks of a Two-Manipulator System. *J. Dyn. Sys., Meas., Control*, 118, 691–697.
- Chiaverini, S. (1997). Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *Robotics and Automation, IEEE Transactions on*, 13(3), 398–410.
- Choset, H., Lynch, K. M., Hutchinson, S., G. Kantor, W. B., Kavraki, L. E., & Thrun, S. (2005). *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press.
- Chou, J. C. K. (1992). Quaternion kinematic and dynamic differential equations. *IEEE Transactions on Robotics and Automation*, 8(1), 53–64.
- Clifford (1873). Preliminary Sketch of Biquaternions. s1-4(1), Proc. London Math. Soc.
- Connolly, T. H. & Pfeiffer, F. (1994). Cooperating manipulator control using dual quaternion coordinates. In *Proceedings of the 33rd IEEE Conference on Decision and Control, 1994*, volume 3 (pp. 2417–2418). Lake Buena Vista, FL, USA.
- Corke, P. (1996). A robotics toolbox for MATLAB. *IEEE Robotics & Automation Magazine*, 3(1), 24–32.
- Diftler, M., Mehling, J., Abdallah, M., Radford, N., Bridgwater, L., Sanders, A., Askew, S., Linn, D., Yamokoski, J., Permenter, F., & Others (2011). Robonaut 2 – the first humanoid robot in space. In *IEEE International Conference on Robotics and Automation*, volume 1 (pp. 2178–2183).
- Dombre, E. & Khalil, W. (2007). *Robot Manipulators: Modeling, Performance Analysis and Control*. Wiley-ISTE.
- Dooley, J. & McCarthy, J. (1991). Spatial rigid body dynamics using dual quaternion components. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, number April (pp. 90–95). Sacramento, California: IEEE Comput. Soc. Press.
- Dooley, J. & McCarthy, J. (1993). On the geometric analysis of optimum trajectories for cooperating robots using dual quaternion coordinates. In *Proceedings IEEE International Conference on Robotics and Automation* (pp. 1031–1036): IEEE Comput. Soc. Press.
- Edsinger, A. & Kemp, C. (2008). Two Arms Are Better Than One: A Behavior Based Control System for Assistive Bimanual Manipulation. *Recent Progress in Robotics: Viable Robotic Service to Human*, (pp. 345–355).
- Evrard, P., Mansard, N., Stasse, O., Kheddar, A., Schauss, T., Weber, C., Peer, A., & Buss, M. (2009). Intercontinental, multimodal, wide-range tele-cooperation using a humanoid robot. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 5635–5640): IEEE.

- Fukao, T., Nakagawa, H., & Adachi, N. (2000). Adaptive tracking control of a nonholonomic mobile robot. *IEEE Transactions on Robotics and Automation*, 16(5), 609–615.
- Gharbi, M., Cortés, J., & Siméon, T. (2008). A sampling-based path planner for dual-arm manipulation. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (pp. 383–388).: IEEE.
- Giordano, P. (2007). *Visual Estimation and Control of Robot Manipulating Systems*. PhD thesis.
- Gu, Y.-L. & Luh, J. (1987). Dual-number transformation and its applications to robotics. *IEEE Journal of Robotics and Automation*, 3(6), 615–623.
- Hamilton, W. R. (1844). On Quaternions, Or On a New System Of Imaginaries In Algebra. Edited by David R. Wilkins. 18 vols. *Philosophical Magazine*.
- Han, D.-P., Wei, Q., & Li, Z.-X. (2008). Kinematic control of free rigid bodies using dual quaternions. *International Journal of Automation and Computing*, 5(3), 319–324.
- Hockstein, N. G., Gourin, C. G., Faust, R. a., & Terris, D. J. (2007). A history of robots: from science fiction to surgical robotics. *Journal of Robotic Surgery*, 1(2), 113–118.
- Huber, M., Rickert, M., Knoll, A., Brandt, T., & Glasauer, S. (2008). Human-robot interaction in handing-over tasks. In *The 17th IEEE International Symposium on Robot and Human Interactive Communication* (pp. 107–112).
- Iwata, H. & Sugano, S. (2009). Design of human symbiotic robot TWENDY-ONE. In *2009 IEEE International Conference on Robotics and Automation* (pp. 580–586).: IEEE.
- Kane, T., Likins, P., & Levinson, D. (1983). *Spacecraft dynamics*, volume 1. McGraw-Hill Book Company.
- Kanoun, O. (2009). *Contribution à la planification de mouvement pour robots humanoïdes*. PhD thesis, Université Toulouse III - Paul Sabatier.
- Kemp, C., Edsinger, A., & Torres-Jara, E. (2007). Challenges for robot manipulation in human environments [Grand Challenges of Robotics]. *IEEE Robotics & Automation Magazine*, 14(1), 20–29.
- Kemp, C. C., Fitzpatrick, P., Hirukawa, H., Yokoi, K., Harada, K., & Matsumoto, Y. (2008). *Handbook of Robotics*, chapter 56. Humano, (pp. 1307–1333). Springer.
- Khatib, O. (1987). A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal of Robotics and Automation*, 3(1), 43–53.
- Khatib, O. (1988). Object manipulation in a multi-effector robot system. In *Proceedings of the 4th international symposium on Robotics Research* (pp. 137–144). Cambridge, MA, USA: MIT Press.
- Khatib, O. (1999). Mobile manipulation: The robotic assistant. *Robotics and Autonomous Systems*, 26(2-3), 175–183.

- Khatib, O., Yokoi, K., Brock, O., Chang, K., & Casal, A. (1999). Robots in human environments: Basic autonomous capabilities. *The International Journal of Robotics Research*, 18(7), 684.
- Khatib, O., Yokoi, K., Chang, K., Ruspini, D., Holmberg, R., & Casal, A. (1996). Decentralized Cooperation between multiple manipulators. In *5th IEEE International Workshop on Robot and Human Communication* (pp. 183–188). Tsukuba, Japan: Dept. of Comput. Sci., Stanford Univ., CA.
- Kim, J.-H. & Kumar, V. R. (1990). Kinematics of robot manipulators via line transformations. *Journal of Robotic Systems*, 7(4), 649–674.
- Kim, M.-J., Kim, M.-S., & Shin, S. Y. (1996). A Compact Differential Formula for the First Derivative of a Unit Quaternion Curve. *The Journal of Visualization and Computer Animation*, 7(1), 43–57.
- Krebs, H. I., Hogan, N., Aisen, M. L., & Volpe, B. T. (1998). Robot-aided neurorehabilitation. *IEEE Transactions on Rehabilitation Engineering*, 6(1), 75–87.
- Kröger, T., Finkemeyer, B., Thomas, U., & Wahl, F. (2004). Compliant motion programming: The task frame formalism revisited. In *Mechatronics & Robotics, Aachen, Germany*.
- Kuffner, J. J. (2004). Effective Sampling and Distance Metrics for 3D Rigid Body Path Planning. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation* (pp. 3993–3998). New Orleans, LA.
- Kuipers, J. B. (1999). *Quaternions and Rotation Sequences - A Primer With Applications To Orbits, Aerospace, And Virtual Reality*. Princeton University Press.
- Liegeois, A. (1977). Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms. *IEEE Transactions on Systems, Man and Cybernetics*, 7(12), 868–871.
- Lozano-Perez, T., Mason, M. T., & Taylor, R. H. (1984). Automatic Synthesis of Fine-Motion Strategies for Robots. *The International Journal of Robotics Research*, 3(1), 3–24.
- Luh, J., Walker, M., & Paul, R. (1980). Resolved-acceleration control of mechanical manipulators. *IEEE Transactions on Automatic Control*, 25(3), 468–474.
- Lum, S. P., Burgar, C. G., Shor, P. C., Majmundar, M., & der Loos, M. V. (2002). Robot-assisted movement training compared with conventional therapy techniques for the rehabilitation of upper-limb motor function after stroke. *Arch Phys Med Rehabil*, 83(7), 952–959.
- Lynch, C. & Popovic, M. (2008). Functional Electrical Stimulation. *IEEE Control Systems Magazine*, 28(2), 40–50.
- Mansard, N. (2006). *Enchaînement de Tâches Robotiques*. PhD thesis, Université de Rennes 1.
- Mansard, N. & Chaumette, F. (2007). Task sequencing for high-level sensor-based control. *Robotics, IEEE Transactions on*, 23(1), 60–72.

- Mansard, N. & Chaumette, F. (2009). Directional Redundancy for Robot Control. *IEEE Transactions on Automatic Control*, 54(6), 1179–1192.
- Mansard, N., Khatib, O., & Kheddar, A. (2009). A Unified Approach to Integrate Unilateral Constraints in the Stack of Tasks. *IEEE Transactions on Robotics*, 25(3), 670–685.
- Marani, G., Antonelli, G., & Yu, J. (2010). Marine Robotic Systems [From the Guest Editors]. *IEEE Robotics & Automation Magazine*, 17(1), 18–18.
- Marchand, E., Spindler, F., & Chaumette, F. (2005). ViSP for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robotics & Automation Magazine*, 12(4), 40–52.
- McCarthy, J. (1990). Introduction to theoretical kinematics.
- Morin, P. & Samson, C. (2008). Motion Control of Wheeled Mobile Robots. In B. Siciliano & O. Khatib (Eds.), *Springer Handbook of Robotics* chapter 34, (pp. 799–826). Springer.
- Murray, R., Li, Z., & Sastry, S. (1994). *A mathematical introduction to robotic manipulation*. CRC.
- Nakamura, Y., Hanafusa, H., & Yoshikawa, T. (1987). Task-Priority Based Redundancy Control of Robot Manipulators. *The International Journal of Robotics Research*, 6(2), 3–15.
- Okada, K., Ogura, T., Haneda, A., Fujimoto, J., Gravot, F., & Inaba, M. (2005). Humanoid motion generation system on HRP2-JSK for daily life environment. In *IEEE International Conference Mechatronics and Automation, 2005*, number July (pp. 1772–1777): IEEE.
- Ott, C., Eiberger, O., Friedl, W., Bauml, B., Hillenbrand, U., Borst, C., Albu-Schaffer, A., Brunner, B., Hirschmuller, H., Kielhofer, S., Konietschke, R., Suppa, M., Wimbock, T., Zacharias, F., & Hirzinger, G. (2006). A Humanoid Two-Arm System for Dexterous Manipulation. In *2006 6th IEEE-RAS International Conference on Humanoid Robots* (pp. 276–283).
- Paul, R. (1981). *Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators*. The MIT Press.
- Perez, A. & McCarthy, J. M. (2004). Dual Quaternion Synthesis of Constrained Robotic Systems. *Journal of Mechanical Design*, 126(3), 425.
- Perrier, C. (1998). *Génération de mouvements pour un manipulateur mobile non-holonome*. PhD thesis, Université Montpellier 2.
- Perrier, C., Dauchez, P., & Pierrot, F. (1997). Towards the use of dual quaternions for motion generation of nonholonomic mobile manipulators. In *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems.*, volume 3 (pp. 1293–1298): IEEE.
- Perrier, C., Dauchez, P., & Pierrot, F. (1998). A global approach for motion generation of non-holonomic mobile manipulators. In *1998 Proceedings of IEEE International Conference on Robotics and Automation*, volume 4 (pp. 2971–2976): IEEE.

- Pham, H.-L., Perdereau, V., Adorno, B. V., & Fraitse, P. (2010). Position and orientation control of robot manipulators using dual quaternion feedback. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 658–663). Taipei: IEEE.
- Rodriguez-Angeles, A. & Nijmeijer, H. (2004). Mutual synchronization of robots via estimated state feedback: a cooperative approach. *IEEE Transactions on Control Systems Technology*, 12(4), 542–554.
- Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N., & Fujimura, K. (2002). The intelligent ASIMO: system overview and integration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3 (pp. 2478–2483 vol.3).
- Schmidt, F., Wimb, T., Birbach, O., Dietrich, A., Fuchs, M., Friedl, W., Frese, U., Borst, C., Grebenstein, M., Eiberger, O., & Hirzinger, G. (2011). Catching Flying Balls and Preparing Coffee: Humanoid Rollin' Justin Performs Dynamic and Sensitive Tasks. In *IEEE International Conference on Robotics and Automation* (pp. 3443–3444).
- Sentis, L. (2007). *Synthesis and control of whole-body behaviors in humanoid systems*. PhD thesis.
- Seraji, H. (1993). An on-line approach to coordinated mobility and manipulation. [1993] *Proceedings IEEE International Conference on Robotics and Automation*, (pp. 28–35).
- Siciliano, B. & Khatib, O. (2008). *Springer Handbook of Robotics*. Springer.
- Siciliano, B., Sciavicco, L., & Villani, L. (2009). *Robotics: modelling, planning and control*. Springer Verlag.
- Siciliano, B. & Slotine, J.-J. (1991). A general framework for managing multiple tasks in highly redundant robotic systems. In *Fifth International Conference on Advanced Robotics, 1991. 'Robots in Unstructured Environments'* (pp. 1211 –1216 vol.2).
- Sisbot, E. A., Marin-Urias, L. F., Broquère, X., Sidobre, D., & Alami, R. (2010). Synthesizing Robot Motions Adapted to Human Presence. *International Journal of Social Robotics*, 2(3), 329–343.
- Soyama, R., Ishii, S., & Fukase, A. (2004). Selectable Operating Interfaces of the Meal-Assistance Device “My Spoon”. In Z. Bien & D. Stefanov (Eds.), *Rehabilitation* (pp. 155–163).: Springer Berlin / Heidelberg.
- Spong, M., Hutchinson, S., & Vidyasagar, M. (2006). *Robot modeling and control*. John Wiley & Sons.
- Stasse, O., Evrard, P., Perrin, N., Mansard, N., & Kheddar, A. (2009). Fast foot prints re-planning and motion generation during walking in physical human-humanoid interaction. *2009 9th IEEE-RAS International Conference on Humanoid Robots*, (pp. 284–289).
- Sun, D. & Mills, J. (2002). Adaptive synchronized control for coordination of multirobot assembly tasks. *IEEE Transactions on Robotics and Automation*, 18(4), 498–510.
- Tenorth, M., Klank, U., Pangercic, D., & Beetz, M. (2011). Web-Enabled Robots. *IEEE Robotics & Automation Magazine*, 18(2), 58–68.

- The University of Chicago, Ed. (2010). *The Chicago Manual of Style*. The University of Chicago Press, 16th edition.
- Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., Lau, K., Oakley, C., Palatucci, M., Pratt, V., Stang, P., Strohband, S., Dupont, C., Jendrossek, L.-e., Koelen, C., Markey, C., Rummel, C., van Niekirk, J., Jensen, E., Alessandrini, P., Bradski, G., Davies, B., Ettinger, S., Kaehler, A., Nefian, A., & Mahoney, P. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9), 661–692.
- Tinos, R. & Terra, M. (2002). Control of cooperative manipulators with passive joints. In *Proceedings of the 2002 American Control Conference*, volume 2 (pp. 1129–1134): IEEE.
- Uchiyama, M. & Dauchez, P. (1988). A symmetric hybrid position/force control scheme for the coordination of two robots. In *Proceedings of 1988 IEEE International Conference on Robotics and Automation* (pp. 350–356): IEEE Comput. Soc. Press.
- Ueha, R., Pham, H. T. T., Hirai, H., & Miyazaki, F. (2009). A simple control design for human-robot coordination based on the knowledge of dynamical role division. In *IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 3051–3056).
- Wampler, C. (1986). Manipulator Inverse Kinematic Solutions Based on Vector Formulations and Damped Least-Squares Methods. *IEEE Transactions on Systems, Man, and Cybernetics*, 16(1), 93–101.
- Whitney, D. E. (1969). Resolved Motion Rate Control of Manipulators and Human Prostheses. *IEEE Transactions on Man-Machine Systems*, 10(2), 47–53.
- Williams, D. & Khatib, O. (1993). The virtual linkage: a model for internal forces in multi-grasp manipulation. In *Proc. IEEE International Conference on Robotics and Automation* (pp. 1025–1030).
- Yamamoto, Y. & Yun, X. (1994). Coordinating locomotion and manipulation of a mobile manipulator. *IEEE Transactions on Automatic Control*, 39(6), 1326–1332.
- Yamano, M., Kim, J.-s., Konno, A., & Uchiyama, M. (2004). Cooperative Control of a 3D Dual-Flexible-Arm Robot. *Journal of Intelligent and Robotic Systems*, 39(1), 1–15.
- Yang, A. T. (1963). *Application of quaternion algebra and dual numbers to the analysis of spatial mechanisms*. PhD thesis, Columbia University.

PUBLICATIONS

Some part of this thesis, or the concepts developed herein, has already been published in the following papers:

JOURNALS:

1. **Adorno, Bruno Vilhena**, and Philippe Fraisse. 2011. "Extended cooperative dual task-space and its application to interactive manipulation in mobile manipulators (submitted for publication)". *Advanced Robotics*.

FULL PAPER IN INTERNATIONAL CONFERENCES:

1. **Adorno, Bruno Vilhena**, Philippe Fraisse, and Sebastien Druon. 2010. Dual position control strategies using the cooperative dual task-space framework. In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 3955-3960. Taipei: IEEE, October.
2. Pham, Hoang-Lan, Veronique Perdereau, **Bruno Vilhena Adorno**, and Philippe Fraisse. 2010. Position and orientation control of robot manipulators using dual quaternion feedback. In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 658-663. Taipei: IEEE, October.
3. **Adorno, Bruno Vilhena**, Antônio Padilha Lanari Bó, Philippe Fraisse, and Philippe Poignet. 2011. Towards a Cooperative Framework for Interactive Manipulation Involving a Human and a Humanoid. In 2011 IEEE International Conference on Robotics and Automation.
4. **Adorno, Bruno Vilhena**, Antônio Padilha Lanari Bó, and Philippe Fraisse. 2011. Interactive Manipulation Between a Human And a Humanoid: When Robots Control Human Arm Motion. In The 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. San Francisco.

Other publications during the period of the thesis, but related to different topics:

BOOK CHAPTERS:

1. **Adorno, Bruno Vilhena**, and Geovany Araújo Borges. "Planejamento de Rotas (to be published)." In *Robótica Móvel*, edited by Denis Wolf, Fernando Osório, Edson Prestes, and Roseli Aparecida Francelin Romero. Editora Elsevier/LTC, 2011.

FULL PAPER IN INTERNATIONAL CONFERENCES:

1. **Adorno, Bruno Vilhena**, and Geovany Araújo Borges. iARW: An incremental path planner algorithm based on adaptive random walks. In *The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 988-993. St. Louis, USA, 2009.
2. Bernardes, Mariana Costa, **Bruno Vilhena Adorno**, Nabil Zemiti, Philippe Poignet, and Geovany Araújo Borges. 2011. "Adaptive Path Planning for Steerable Needles Using Duty-Cycling". In *The 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. San Francisco.

EXTENDED ABSTRACT IN INTERNATIONAL CONFERENCES:

1. Bernardes, Mariana Costa, **Bruno Vilhena Adorno**, Nabil Zemiti, Philippe Poignet, and Geovany Araújo Borges. 2011. "Path planning for steerable needles using duty-cycling". In *CARS 2011 Computer Assisted Radiology and Surgery*. Berlin.

INDEX

- control
 - distance, 75
 - functional electrical stimulation, 98
 - generic law, 75
 - human arm, 98
 - human arm (Cartesian space), 99
 - human arm (FES), 98
 - primitives, 72, 77
 - prioritized, 70
- cooperative dual task-space, 50
 - complete description, 62, 64, 65
 - extended, 86
 - kinematic control, 67
 - object manipulation, 56
 - twists, 60
 - wrenches, 60
- cost
 - decompositional multiplication, 44
 - forward kinematic model, 41
 - Jacobian, 42
- decompositional multiplication, 29
 - definition, 31
 - homogeneous transformation matrices, 34
 - properties, 125
- Denavit-Hartenberg
 - modified, 36, 40
 - standard, 36, 39
- dual number, 23
 - addition, 23
 - dual part, 23
 - inverse, 23, 117
 - multiplication, 23
 - primary part, 23
 - properties, 117
 - subtraction, 23
 - Taylor expansion, 117
- dual quaternion
 - conjugate, xvi, 24
 - decompositional multiplication, 31
 - definition, 23
 - dual part, xvi, 23
 - exponential, 25
 - Hamilton operators, 24
 - imaginary part, 24
 - inverse, 122
 - inverse under decompositional multiplication, 34
 - logarithm, 25
 - multiplication, 24
 - norm, 25, 121
 - power, xvii, 25
 - primary part, xvi, 23
 - real part, 24
 - rigid motion, 28
 - set, xv
 - unit norm, 28
 - vec operator, xvii, 24
- dual unit, xvi, 23
- forward kinematic model, 36
- frame rotation, 26
- functional electrical stimulation, 98
- generalized
 - coordinates, 57
 - Jacobian, 57
 - speeds, 57
 - wrench, 58
- Hamilton operators, xvi
- human-robot collaboration, 95
- imaginary unit, xv
- imaginary vector unit, xvi, 21
- Jacobian
 - absolute dual quaternion, 54
 - conjugate, 53
 - cooperative, 52
 - dual quaternion, 38
 - generalized, 57, 124
 - geometrical, 59

- relative dual quaternion, 54
- translation, 54
- mirrored movements, 103
- nonholonomic mobile base, 88
- nullspace, 60
 - projection, 68
- Plücker line, 122
- point rotation, 27
- quaternion
 - conjugate, xvi, 21
 - cross product, xvii, 118
 - definition, 20
 - Hamilton operators, 22
 - imaginary part, 21
 - imaginary vector unit, 21
 - inverse, 119
 - norm, 21
 - propagation, 119
 - real part, 21
 - rigid motion, 27
 - rotation, 26
 - set, xv
 - translation, 26
 - unit norm, 26, 119
 - vec operator, 22
- rigid body motion, 25
 - decompositional, 29
 - dual quaternion, 28
 - quaternion, 27
- serially coupled kinematic chain, 84
- simultaneous handling, 103
- twist, 56
 - absolute, 60
 - relative, 60
- virtual sticks, 60
- wrench, 56
 - absolute, 60
 - relative, 60

ABSTRACT

This thesis is devoted to the study of robotic two-arm coordination/manipulation from a unified perspective, and conceptually different bimanual tasks are thus described within the same formalism. In order to provide a consistent and compact theory, the techniques presented herein use dual quaternions to represent every single aspect of robot kinematic modeling and control.

A novel representation for two-arm manipulation is proposed—the cooperative dual task-space—which exploits the dual quaternion algebra to unify the various approaches found in the literature. The method is further extended to take into account any serially coupled kinematic chain, and a case study is performed using a simulated mobile manipulator. An original application of the cooperative dual task-space is proposed to intuitively represent general human-robot collaboration (HRC) tasks, and several experiments were performed to validate the proposed techniques. Furthermore, the thesis proposes a novel class of HRC tasks wherein the robot controls all the coordination aspects; that is, in addition to controlling its own arm, the robot controls the human arm by means of functional electrical stimulation (FES).

Thanks to the holistic approach developed throughout the thesis, the resultant theory is compact, uses a small set of mathematical tools, and is capable of describing and controlling a broad range of robot manipulation tasks.

Keywords: *Two-arm manipulation, Dual quaternion, Kinematic control, Intuitive human-robot collaboration, Physical human-robot interaction, Whole-body motion, Mobile manipulator, Cooperative dual task-space.*

RÉSUMÉ

Cette thèse est consacrée à l'étude de la manipulation et de la coordination robotique à deux bras ayant pour objectif le développement d'une approche unifiée dont différentes tâches seront décrites dans le même formalisme. Afin de fournir un cadre théorique compact et rigoureux, les techniques présentées utilisent les quaternions duaux afin de représenter les différents aspects de la modélisation cinématique ainsi que de la commande.

Une nouvelle représentation de la manipulation à deux bras est proposée - l'espace dual des tâches de coopération - laquelle exploite l'algèbre des quaternions duaux afin d'unifier les précédentes approches présentées dans la littérature. La méthode est étendue pour prendre en compte l'ensemble des chaînes cinématiques couplées incluant la simulation d'un manipulateur mobile.

Une application originale de l'espace dual des tâches de coopération est développée afin de représenter de manière intuitive les tâches principales impliquées dans une collaboration homme-robot. Plusieurs expérimentations sont réalisées pour valider les techniques proposées. De plus, cette thèse propose une nouvelle classe de tâches d'interaction homme-robot dans laquelle le robot contrôle tout les aspects de la coordination. Ainsi, au-delà du contrôle de son propre bras, le robot contrôle le bras de l'humain par le biais de la stimulation électrique fonctionnelle (FES) dans le cadre d'applications d'interaction robot / personne handicapée.

Grâce à cette approche générique développée tout au long de cette thèse, les outils théoriques qui en résultent sont compacts et capables de décrire et de contrôler un large éventail de tâches de manipulations robotiques complexes.

Mots-clés: *Manipulation à deux bras, Quaternion dual, Commande Cinématique, Coopération homme-robot intuitive, Interaction physique homme-robot, Contrôle du corps complet, Manipulateur mobile, Espace dual des tâches de coopération.*