



Universidad
Carlos III de Madrid
www.uc3m.es

DOCTORAL THESIS

MODELING AND CONTROL OF AN ANTHROPOMORPHIC ROBOTIC HAND

Author:

Choukri Bensalah

Director:

Mohamed Abderrahim

SYSTEMS ENGINEERING AND AUTOMATION
DEPARTMENT

Leganés, 2014

DOCTORAL THESIS

MODELING AND CONTROL OF AN
ANTHROPOMORPHIC ROBOTIC HAND

Author: Choukri Bensalah

Director: Mohamed Abderrahim

Examination Committee

Firma (Signature)

Chair: Prof. Anis Sahbani

Secretary: Prof. Fares Jawad Moh'D Abu-Dakka

Member: Prof. Claudio Rossi

Degree: Doctor of Philosophy in Electrical Engineering, Electronics and Automation

Grade:

Leganés,

de

de

Abstract

This thesis presents methods and tools for enabling the successful use of robotic hands. For highly dexterous and/or anthropomorphic robotic hands, these methods have to share some common goals, such as overcoming the potential complexity of the mechanical design and the ability of performing accurate tasks with low and efficient computational cost.

A prerequisite for dexterity is to increase the workspace of the robotic hand. For this purpose, the robotic hand must be considered as a single multibody system. Solving the inverse kinematics problem of the whole robotic hand is an arduous task due to the high number of degrees of freedom involved and the possible mechanical limitations, singularities and other possible constraints. The redundancy has proven to be of a great usefulness for dealing with potential constraints. To be able to exploit the redundancy for dealing with constraints, the adopted method for solving the inverse kinematics must be robust and extendable. Obviously, addressing such complex problem, the method will certainly be computationally heavy. Thus, one of the aims of this thesis is to resolve the inverse kinematics problem of the whole robotic hand under constraints, taking into account the computational cost. To this end, this thesis extends and reduces the most recent *Selectively Damped Least Squares* method which is based on the computation of all singular values, to deal with constraints with a minimum computational cost. New estimation algorithm of singular values and their corresponding singular vectors is proposed to reduce the computational cost. The reduced extended selectively damped least squares method is simulated and experimentally evaluated using an anthropomorphic robotic hand as a test bed. On the other hand, dexterity depends not only on the accuracy of the position control, but also on the exerted forces. The tendon driven modern

robotic hands, like the one used in this work, are strongly nonlinear dynamic systems, where motions and forces are transmitted remotely to the finger joints. The problem of modeling and control of position and force simultaneously at low level control is then considered. A new hybrid control structure based on the succession of two sliding mode controllers is proposed. The force is controlled by its own controller which does not need a contact model. The performance of the proposed controller is evaluated by performing the force control directly using the force sensor information of the fingertip, and indirectly using the torque control of the actuator. Finally, we expect that the applications of the methods presented in this thesis can be extended to cover different issues and research fields and in particular they can be used in a variety of algorithm that require the estimation of singular values.

To my Parents

&

To my Wife

Acknowledgements

This work was partially supported by the European project HANDLE, FP7-231640, and by the Spanish ministry MICINN through FPI scholarship within the project DPI-2005-04302. I am very thankful for the financial supports. I also thank the Robotics Lab, the laboratory at the Systems Engineering and Automation Department of the University Carlos III of Madrid for the opportunity granted to me to realize this PhD.

First of all, I would like to express my heartiest gratitude to my supervisor; Mohamed Abderrahim, for his valuable guidance and support in the course of the present work. I would also like to thank the members of my committee; Prof. Anis Sahbani, Prof. Fares Jawad Moh'D Abu-Dakka, Prof. Claudio Rossi for their willingness to serve on my committee and their suggestions. I also want to thank the staff of the Informatics Department of the University of Hamburg, particularly; Norman Hendrich, for the help in the experimental parts of the thesis.

I cannot omit to mention all my friends in our office and all staff of the Robotics Lab. Among the people I met during these years, I would like to thank Javier Gonzalez-Quijano. I thank also my friends outside of the university; Benyahia, Eddahbi, El Yaagoubi, Abdulla.

My greatest thanks go to my family. Thank you my mother, my father and my wife Amal, for always believing in me and pushing me to my limits. Thinking in your presence was always giving me the purposefulness to achieve the desired objective. Thanks to my sisters and brothers.

Last but not least, great thanks to Almighty God for give me strength during my PhD.

Contents

1	Introduction	1
1.1	Problem statement	5
1.2	Contribution	6
1.3	Organization of the thesis	7
2	Literature Review	11
3	Kinematic Modeling of Anthropomorphic Robotic Hand	23
3.1	Kinematics Modeling	27
3.2	Inverse Kinematics of Redundant Manipulators	28
3.3	Damped Least Squares	32
3.3.1	Estimation of the Smallest Singular Value for DLS method . . .	40
3.3.2	New Estimation Algorithm of all Small Singular Values for the DLS method	43
3.4	Selectively Damped Least Squares Method (SDLS)	45
3.4.1	A New Estimation Algorithm for Singular Value Decomposition (SVD)	49
3.4.2	An Extended Selectively Damped Least Squares (ESDLS) to Deal with Joint Limits Avoidance	53
3.5	Implementation Issues	54
3.5.1	Extended Selectively Damped Least Squares for Index Finger . .	56
3.5.2	Reduced ESDLS using SVD Estimation for Index Finger	59
3.5.3	Reduced ESDLS Using SVD Estimation for the whole Robotic Hand	63
3.6	Experimental results	65

CONTENTS

3.7	Conclusion	68
4	Modeling and Hybrid Control	71
4.1	Actuator system modeling	74
4.1.1	Electric model of the DC-motor	74
4.1.2	Mechanical equations of the motor and the gearbox	76
4.1.3	Friction modeling	78
4.1.4	Backlash	80
4.2	Hybrid Control	84
4.3	Sliding Mode Control	89
4.3.1	Problem statement	91
4.3.2	SMC design toward the hybrid controller	94
4.3.3	Simulated Speed/Current Control	95
4.3.4	Simulated Position/Speed/Current Control	100
4.4	Experimental Results	103
4.4.1	Sliding Mode Control using Pulse-Width-Modulation	103
4.4.2	Setup Hardware	104
4.4.3	Position/Speed/Current Control	105
4.4.4	Hybrid Position/Speed/Force Control	109
4.4.5	Hybrid Position/Speed/Torque Control	113
4.5	Conclusion	116
5	Conclusions and Future Work	119
5.1	Conclusions	119
5.2	Future Works	122
5.3	Publications	123
A	Denavit-Hartenberg's parameter for the Shadow Hand	125
A.1	Index finger	125
A.2	Middle finger	125
A.3	Ring finger	126
A.4	Little finger	126
A.5	Thumb finger	126
B	Robots hands	127

Bibliography

139

CONTENTS

1

Introduction

Day by day, robots are exhibiting more and more abilities to be widely integrated in human everyday life. For industrial applications, they are frequently used in intensive repetitive operations that require high level of precision and speed of execution. Nowadays, new types of robot are being designed for responding to various human needs, such as entertainment in the form of toys and animatronics, or services like the autonomous robot mowers, house floor cleaning and even security. The contribution of robots is constantly evolving and has been expanded to cover many disciplines, such as medicine, education, military, space, research, rescue, and more. Today, there are others having a humanoid character that can sing and dance, such as the humanoid robot Asimo, and robots showing an impressive walking in rough uneven terrain, such as the Boston Dynamics Big Dog quadruped robot (see Fig. 1.1) (1).

As the use of robots is in expansion, they are expected to achieve human hand skills. In order to make this leap, the robots must be able to manipulate objects with different shapes. However, in concrete terms, the robots' ability to perform such manipulations under realistic conditions is still far inferior compared with human skills. There are several reasons making this similarity almost impossible due, mainly, to the complex morphology of the human hand. It is well known that the human hand, although it is highly complex, is a prime example of a highly dexterous end-effectors exhibiting unrivaled manipulation abilities. In an attempt to match these abilities, many designs of robotic end-effectors have been proposed over the two past decades, starting from a simple parallel jaw gripper until the recent anthropomorphic robot hands. A thorough discussion of this advance is reported in the state of the art chapter. In addition to the

1. INTRODUCTION



Figure 1.1: The Boston Dynamics Big Dog quadruped robot showing walking abilities

challenge of design, the following reasons can also affect considerably the skill quality:

- The ability to perceive the environment.
- The sensorial information required throughout the manipulation of objects must be extracted, often from each contact points.
- Addressing dexterous manipulations requires efficient control strategies of different physical parts including both kinematics and dynamics aspects.
- There is a lack of efficient simulator frameworks that are able to reproduce manipulation tasks closely matching the reality.

Arguably, there are other hindrances that can be the result of diverse limitations which are relevant to both hardware and control. As for the ability to perceive the environment, the robot must use its sensors to operate in potentially unforeseen conditions. The vision, the common sensor system used to this aim, allows not only the recognition of the environment but also to perform grasps and manipulations of objects in a robust manner, subject to realistic levels of sensing error and noise. Performing a stable grasp means that any posterior manipulation task would be inevitably crowned with success.

Grasping, one of the main robotic hand tasks, allows the first physical interaction with the environment, a seemingly simple task, but the reality is absolutely the contrary. In addition to the aforementioned factors that decide the quality of grasping a can, for example (see Fig. 1.2), there are other important factors which can be related to the following potential questions: Where is the can located? From sets of *shape primitives*,



Figure 1.2: Fetch Hand grasping a can

which model matches the can? How to reach to the contact points? Is it full or empty?. The answer to the first question can be resolved by using perception systems. Usually, the matching processes of the perceived object are based on learning strategies. The *grasp planning*, the algorithm that leads to stable grasping task depends upon the state of the grasped object which can be found in many modes; fixed or manipulable. Furthermore, the environment may be more complex in the event of adding, moving, removing or modifying objects during the grasping process. The complexity can also be related to a matter of the robot capabilities (see Fig. 1.3). According to the type of the object knowledge, grasp planning methods can be performed either relying on a model or on an approximate shape. Despite good expected results when using the model knowledge, the correspondence process between the model and the real object results in heavy planning methods which affect appreciably the real-time usability. Whereas, for typical applications, the grasped object should be approximately structured but with

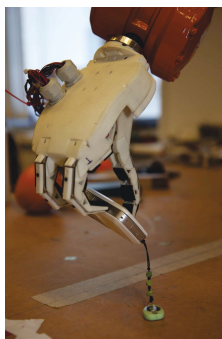


Figure 1.3: The KTHand manipulating a mobile phone

1. INTRODUCTION

an exact pose knowledge. Finding a good grasp by means of possible combinations of dynamics and kinematics design of the robot, different sensory information, object model or approximate shape, object pose, and robot control leads to a huge space of parameters which are difficult to control at once. This problem can be alternatively overcome concentrating on finding suitable contact points without considering neither kinematics nor dynamics of the robot (2, 3). The approach consists in extracting the outer contour of the grasped object and then applying a planar grasp to this contour (4). It is worth mentioning that, not considering the kinematics in these approaches refers to its ability and not to the number of finger of the robotic hand, which should be at least equal or more than the number of planned contact points. On the other hand, these approaches could fail when taking the constraints of robot kinematics into account. In the same context, another approach proposes to find the solutions to grasp planning through the use of prehensile posture for each primitive shape such as, cylinders, spheres, cones, boxes, etc (5).

A learning process by demonstration is another approach for grasp planning, in which the human shows the robot how to perform some tasks of grasping, and then the robot repeats the task. This approach, called programming by demonstration (6, 7, 8), is still far away from reproducing human skills due, basically to the dissimilarity of the morphology and to the unavailability of some information. Even if the demonstration is recorded using vision information and/or laser scanning, there is a lack of pertinent information on the contact points location and exerted forces which can be recorded separately, but leading to more complexity.

Furthermore, for a reasonable execution time, the grasp planning can include constraints that rely on both systems; robot and grasped object, such as the distributed contact points and their corresponding friction cone constraints, grasping style, value of contact forces exerted by fingers and inverse kinematics of robotic hand (9, 10, 11). Tokuo et al. proposed a grasp planning method for an anthropomorphic robotic hand attached at the tip of a humanoid robot's arm (12). The original contribution of their method consists in using the full robot kinematics; hand, arm and body, when the hand or the arm/hand alone are unable to lead to feasible grasping posture.

Modeling of contact points and contact forces, called the *grasp control*, are assumed a-priori defined before any launch of grasp planning method. In the literature on grasping, the modeling of contact points is called the *form-closure*, whereas for contact forces,

it refers to the *force-closure* (13). A wide spectrum of works are focused on improving the quality of grasping and maintaining the balance of grasped objects by involving many quality factors of both form and force closure (14, 15). In this framework, the advanced studies on both concepts allowed performance with a high level of dexterity, such as regrasping sequence (16) and sliding movement of fingers on the object (17, 18). Beyond the grasp planning, David William and Oussama Khatib (19) devised a new approach exploiting the characteristics of internal forces and moments during multi-grasp manipulation by means of robot manipulators. Recently, this approach is used *in-hand manipulation* tasks (20, 21).

The success rate of all these methods depends ultimately on the ability to perform the computed positions and forces in accurate and robust manner at high as well as low level control. Addressing this problem can open a wide range of robotic hand applications and move us one step closer to the possibility of reproducing human grasping and manipulation skills.

1.1 Problem statement

For grasping tasks, the fundamental aspects; perception, motion and force should be well controlled. At certain times, these concepts have to interact simultaneously either in explicit or implicit form according to the employed sensor information and control strategies. Usually, the perception based on a vision system can be employed for several objectives, including the reconstruction of unstructured environments, recognition of the grasped object, and tracking. In the actual work this topic is not treated, i.e. it is assumed that the grasped object is recognized and well located, and suitable contact points have been already computed.

Given a set of contacts on the surface of a known object according to a given quality criterion (22), it is rarely guaranteed that the contacts are physically reachable by a real robotic hand. Among several complex factors leading to this shortcoming is the fact that the motion of each finger is addressed by solving the inverse kinematics in an independent manner. Obviously, this process is less complex than when the inverse kinematics of the whole hand is addressed but it reduces considerably the workspace of the hand (23, 24, 25). It is worth noticing that, when each finger is addressed alone, it can be seen as a serial-chain manipulator with one end-effector. For serial-chain

1. INTRODUCTION

manipulators, the inverse kinematics problem has been widely studied where diverse approaches have been developed. However, few research efforts have been devoted to the inverse kinematics problem for the whole hand, which can be considered as a manipulator with tree multi-end effectors structure. Kinematically, there are other factors that should be taken into account related to realistic constraints caused by either the mechanical system itself or the environment. On the other hand, the efficiency of the approach used for solving the inverse kinematics is intrinsically related to the efficiency of the control strategies at actuator level. A relatively large spectrum of robotic hand designs use the tendon actuation system. This transmission modality, which is the same adopted in the addressed robotic hand in this thesis, is quite adequate for anthropomorphic robotic hands since it is physiologically inspired from the human muscle works. In such actuation modality, the tendon is the only component responsible of active force generation exerted on the manipulated object. However, controlling the position and the force of such actuator systems and overcoming all potential nonlinearities are still being an important issue to be resolved.

Now, we can formulate our problem statement as follows. *In robotic hands research, the increase in the workspace that can be achieved by considering the hand as a single multi-body system seems important for approaching human-level abilities. For this objective, resolving the inverse kinematics problem is quite complex that it has proven difficult to take into account the different constraints that can hinder the motion of the whole robotic hand. For a dextrous manipulation, a compromise between the position generated by the inverse kinematics approach and the exerted force has to be accomplished in a robust manner even for complex actuation systems.*

1.2 Contribution

In this work, the developed approaches have a dual usefulness which, even if the purpose was focused on robotic hand applications, they can be also applied on a wide range of mechanical systems. For this purpose, large research efforts have been undertaken while exploiting a good background knowledge of the theoretical control and by using sophisticated equipment. Despite being classical issues, each one of the developed approaches exhibits novelty in its design and efficiency in comparison with existing approaches. The main contributions of this thesis are:

- The original method for solving the inverse kinematics problem, *Selectively Damped Least Squares*, has been extended to deal with additional realistic constraints by extending the Jacobian matrix.
- On the basis of the estimation method of the smallest singular value, a new estimation algorithm to estimate all singular values and their corresponding input and output singular vectors, has been developed.
- It is shown how the estimation algorithm can be used to reduce the computational cost of the extended inverse kinematics method.
- A new approach has been developed to compute the overall solution of the inverse kinematics problem when only a part of all the singular values are estimated.
- On the basis of the robust discontinuous controller "*Sliding Mode Control*", the position/speed and the current controllers are placed in a cascade structure. In retrospect, such structure was not a recommendable control since the control law is sampled twice.
- An uncommon formulation of the hybrid position/speed/force controller in which the force is controlled separately, but in a continuous way without switching operation to alternate between the force and the position/speed control.

The validation of the inverse kinematics method has been simulated and experimented on a real Shadow hand (26, 27) carrying out a circular motion of a cylindrical knob. For the last point, more details are reported in the chapter of the hybrid control showing potential superiorities with respect to other hybrid controllers applied to the same experimental platform. Two versions of the hybrid control according to the type of the sensor information; either at the fingertip or at each joint, are proposed.

1.3 Organization of the thesis

This thesis is comprised of five chapters. The first chapter is the introduction, which is the current chapter. It is followed by a chapter covering the state of the art of existing robotic hands. The two following chapters cover much of the performed work and results. The last chapter is dedicated to discussing and drawing conclusions about the

1. INTRODUCTION

relevant results, along with the suggestions for future work. For completeness, more technical details are included as appendices. In the following part, each chapter will be more detailed.

Throughout **Introduction**, the different factors involved in the interaction of robotic hands with manipulated objects are discussed. Grasping tasks can be broadly split in two states; before and after the interaction. In the first state, the robotic hand takes a reachable grasp posture thanks to grasp planning. Once the interaction starts, the second state consists in controlling the force and the position simultaneously. Addressing these problems was the motivation behind this work, which consists in developing a potential inverse kinematics, in a way that is relevant for performing the grasp planning and to design a robust hybrid position/force controller.

Chapter 2 gives a literature review on robotic hands and the applicable technologies as well as a survey of some common robots.

Chapter 3 deals with the inverse kinematics problem for anthropomorphic robotic hand applications. Despite the problem is addressed for robotic hands, the designed method can be used for any type of robotic systems. The chapter starts with a chronogram of realized works in relation to the inverse kinematics issue, especially of those adopted in this work which are based on numerical approaches. The *Selectively Damped Least Squares* is one of the newest methods of this type. The method was extended in this work to deal with additional constraints. The singular values decomposition process is the core of this method. Thus, in order to reduce the computational cost, an algorithm for estimating singular values and their corresponding singular vectors is devised. In further detail, the estimation of only those singular values that generate singularity can substantially reduce the computing cost. It is shown how the overall solution can be computed using the Cholesky decomposition method. In the end, the proposed approach is successfully applied in grasping and for the rotating motion of a cylindrical object using the anthropomorphic robotic hand of the Shadow Robot Company.

Chapter 4 proposes solutions at low level control for one joint of the same robotic hand where the position and the force should be controlled simultaneously. An

experimental platform consisting of one finger and its drive mechanisms, was used to implement the designed controllers. In the beginning, a dynamical model of the whole system of the one joint, characterized by its strong nonlinearity, is identified. Using the Matlab framework, the position/speed in cascade structure with the current control of one joint are successfully simulated using two different versions of the discontinuous *Sliding Mode Control*. As for the hybrid control, a new control structure is suggested in which the force is controlled separately in a cascade structure with the position/velocity controller. With a force sensor providing the exerted force information at fingertip, and taking advantage of the obtained parameters of the simulated position/velocity controller, the hybrid force/position/velocity control is successfully implemented. Another alternative of hybrid control is also presented using, instead of the force, the torque control at joints level. Finally, the obtained results are compared with other works realized for the same objective.

Finally, **Chapter 5** concludes the thesis summarizing the main contributions and suggests some of the aspects to advance the work further and extend the obtained results.

1. INTRODUCTION

2

Literature Review

It is clear that the civilization inherited so far was the result of homo sapiens creativity. While, in the first era of its creation, this same homo sapiens was incapable of ensuring even basic requirements to survive. This raises an interesting question. What is the main feature that made the human himself enhances its abilities to be able to conquer the space?. The answer to this question can be formulated by resorting to a comparative study between the human and another specie, on condition that both spices share maximum common traits. Without a doubt, the specie showing quite homologous traits of the human physiology are the primates (28). Even if all classes of primates are distinguished by the mind; being the main distinguished feature of humans among all animals, the primates would not be able to innovate like humans. The hands of both species make the main distinction despite the fact that the number of bones and muscles are the same (see Fig. 2.1). Taxonomic trees of different classes of primates in a skeleton study is reported in (28). On the basis of this study, among all organs of both species, the hands make the exception in terms of the flexibility thanks to the number of bones which approximately represent a quarter of the total bones of the body (29). Human hands are distinguished by the treat of the thumb which is longer, more flexible and with more muscle. Primates' hands, on the other side, have relatively shorter and less flexible thumbs compared to all fingers. Indeed, primates' subfamilies use their knuckles for walking and the three middle fingers, which are longer and curved, to climb trees and to swing between branches (see Fig. 2.2). Because of the fact that the thumb is not involved in these tasks, it has not been evolved. This limitation makes primates' subfamilies incapable to fulfill dexterous manipulations, contrary to human

2. LITERATURE REVIEW



Figure 2.1: Skeleton hands

hands which exhibit high degrees of dexterity (30). Furthermore, the primates hands are not quite flexible at the wrist leading to much more reduction of the dexterity. It is worth noting that, physiologically, human hands are also more sensitive than primates subfamilies due to their thin skin. This feature allows to manipulate objects with different sizes, shapes and even quite fragile. But what is the relation between the mind, the hand and the creativity?, and why the hand has this privilege by comparison with all the organs?. This enigmatic has been addressed for the first time by two Greek philosophers; Anaxagoras (500-428 BC) and Aristotle (384-322 BC) (31). The authors attempted to treat the relation between the hands and the mind. After several debates, it was confirmed that the intelligence of the human has been developed thanks to the dexterity of the hands since by means of hands that virtual concepts become a reality. The dexterity of the human hands was and is still of concern of researchers to understand and to explain this complex concept. Biologically, the dexterity can be defined by the human hands aptitude to realize complex tasks by controlling small muscles of each finger under the nervous system. After having projected the frame of the manipulated object and the whole hand in the eyes' frame, the mind generates smooth movements by means of the learned motor skill. Artificially, mimicking such aptitude

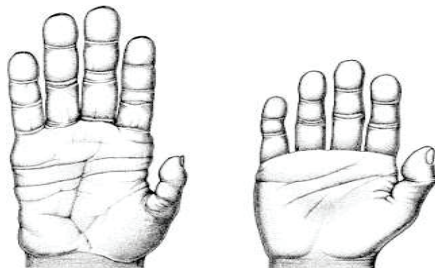


Figure 2.2: Primate and human hand

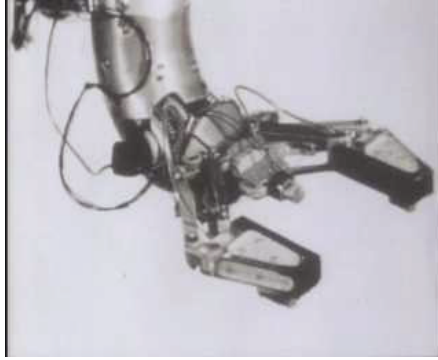
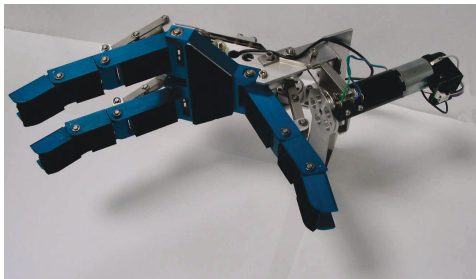


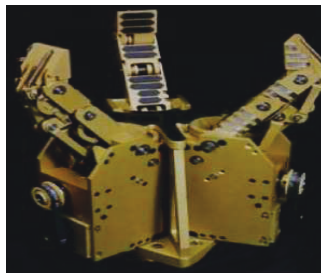
Figure 2.3: HM-1, the first robot hand

by artificial hands became a dream of researchers since the realization of the movie *Metropolis* (1927) which imagines the birth of the first female humanoid robot. In this context, after the first robot hand was developed by Heinrich Ernst in 1962, this field has known remarkable development over the past forty years. This first robot hand is far from being considered as humanoid robot hand since it consists of two opposite jaws gripper (see Fig. 2.3). Certainly, the dexterity requires at least three fingers distributed in such a way that one should be in opposite configuration. In this sense, many robotic hands based on three fingers are designed in different structures, such as; parallel (Fig. 2.4.a), circular (Fig. 2.4.b), changeable structure (Fig. 2.4.c) (32) or human hand structure (Fig. 2.4.d) (33). In order to strengthen the dexterity, these robotic hands, which are not the only ones, are endowed with either force sensor (Fig. 2.4.c) or tactile sensor (Fig. 2.4.a,b,d). Usually, the force sensing is exploited to fulfill grasping and manipulations performed at fingertips level. This type of sensors are required for quite complex applications, such as rolling and sliding in-hand manipulations. On the other side, if the robotic hand is expected to perform manipulations inside the hand, it would be adequate to cover all phalanxes and palm by a mesh of tactile sensors. It has been shown that the tactile sensor helps to estimate; the stability of the grasp (34), the pose of the manipulated object (35) and even identify the shape of the grasped object (36). It is quite evident that the dexterity of four fingered hand is better than three fingered. The DLR hand (37, 38) is the most advanced hands having four fingers. More examples of robotic hands with four fingers are reported in Appendix B. Note that, from four fingers, the robotic hands resemble much more human hands where one of the fingers

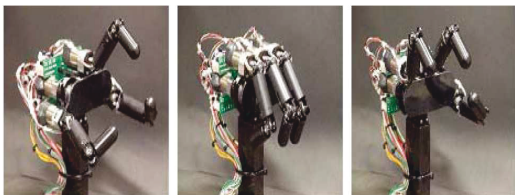
2. LITERATURE REVIEW



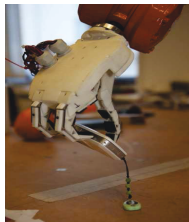
(a) Deft hand



(b) Robotic hand of Laval university



(c) HDS hand of Tokyo university



(d) KTHand

Figure 2.4: Different structures of three fingered robot hands

is placed in opposite to play the role of the thumb finger. From this number of fingers, the landmark of the anthropomorphism concept begins to emerge. Briefly, unlike the dexterity concept which involves the ability of the robotic hand to perform tasks with a certain level of complexity, the anthropomorphism deals much more with similarities and differences with respect to the human hand in terms of shape, size, weight, flexibility, skin suppleness, etc. Both concepts; dexterity and anthropomorphism have been exhaustively analyzed by Bicchi (31) and Biagiotti et al. (30), respectively. The authors have proposed some measurement factors to define the degrees of each concept. Fulfilling high degrees of each concept for a same robotic hand is still a challenging task. The evident requirement for ensuring a high degree of dexterity and anthropomorphism is to exploit robotic hands having five fingers. Usually, the added fifth finger plays the role of the little finger, and like the human hand, its length is often smaller than the three aligned fingers. Although robotic hands with four fingers provide high dexterity, the performance is still far from human-like hand tasks, notably for tele-operation tasks in slave-master protocol. This is due to the master part which, in several applications, is a data-glove reproducing the movement of human hands. Another application field considered as one of the most motivating factors towards the design of robotic hand

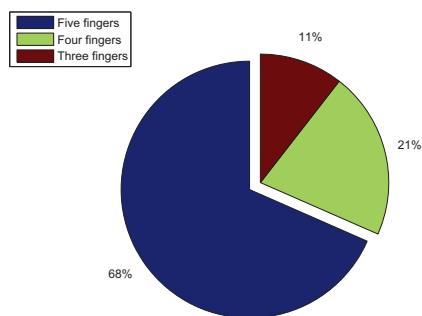


Figure 2.5: Number of finger in selected robotic hands

with five fingers, is for clinical use, and as prosthetic hands. The number of five-fingered robotic hands existing in the world represents the absolute majority among all types of robotic hands which are designed for research or meant for market. This conclusion is the result of surveying 57 of common anthropomorphic robotic hands throughout the last twenty years, excluding those considered as gripper tools. The survey is reported in Table. 2.1 in which the number of fingers, the DoF, the type of actuator and the transmission type are reported. As sketched in Fig. 2.5, almost two thirds of the reported hands are five fingered robotic hands. Whereas, the one third is shared between robotic hands with four and three fingers, in which four fingered robotic hands outweigh three fingered robotic hands in number. Note that, the most recent hands are those consisting of five fingers. Among the factors that lead to the growing interest of five-fingered robotic hands is the trend in the design of prosthetic robotic hands which approximately represent one third of this kind of hands. Whereas, it is very rare to find prosthetic hands with four or three fingers. The exception of this point of view is the RTR III hand, which is a three fingered prosthetic robotic hand (39). In addition to the categorization of the robotic hands in terms of the number of fingers, there are other decisive factors that determine their performance, such as; the actuators, the sensors, the transmission system between the actuators and the joints, and control strategies.

2. LITERATURE REVIEW

Table 2.1: Overview of anthropomorphic robotic hands

Name	Number of fingers	DoF	Actuator type	Transmission type
UB hand II	3	13	Electric	Tendons
Barrett hand	3	8	Electric + Mechanical	Linkages
Okada hand	3	11	Electric + Mechanical	Tendons
Stanford/JPL hand	3	9	Electric + Mechanical	Tendons
RTR hand III	3	6	Electric + Mechanical	Tendons
Salisbury	3	9	Electric	Tendons
DLR hand II	4	13	Electric	Motors
DIST hand	4	16	Electric + Mechanical	Tendons
Meka H2 Compliant hand	4	5	Electric + Mechanical	Tendons
MAC hand	4	12	Electric	Motors
LMS hand	4	16	Electric	Tendons
Utah/Mit hand	4	16	Pneumatic	Tendons
SKKU II hand	4	10	Electric	Motors
Goldfinger hand	4	12	Electric + Mechanical	Linkages
DLR-HIT-Hand I hand	4	13	Electric	Motors
LMS hand	4	16	Electric + Mechanical	Tendons
Sandia hand	4	13	Electric	Motors
NAIST-Hand I	4	12	Electric + Mechanical	Linkages
Shadow hand	5	20	Electric-Pneumatic	Tendons
Actuated Shaffied hand	5	12	Electric	Tendons
Elu-2 hand	5	9	Electric	Servo-actuated
Gifu hand III	5	16	Electric	Servo-actuated
Robonaut hand	5	14	Electric	Mechanical
EH1 Milano hand	5	16	Electric	Tendons
Dexhand	5	11	Electric	Tendons
Raphael hand	5	6	Pneumatic	Corrugated tubing
Ultralight hand	5	13	Pneumatic	Pneumatic
TUAT/Karlsruhe hand	5	13	Electric	Tendons
RCH-1 hand	5	16	Electric	Tendons
LARA hand	5	22	Electric + Mechanical	SMA wires
Rutger hand	5	20	Electric + Mechanical	SMA wires
UB hand III	5	16	Electric	Tendons
Harada hand	5	5	Electric	Tendons
Yokoi hand	5	13	Electric + Mechanical	Tendons
Dart hand	5	19	Electric	Tendons
Bionic hand	5	16	Pneumatic + Electric	Pneumatic
SBC hand	5	16	Electric + mechanical	SMA wires
ACT hand	5	20	Mechanical	Tendons
Anthrobot hand	5	20	Electric	Tendons
Asimo hand	5	20	Electric + Mechanical	Motors
Cyberhand	5	16	Electric + Mechanical	Tendons
Ng hand	5	6	Pneumatic	Pneumatic
IH1 Azzurra hand	5	Not found	Electric	DC Motors + Servomotors
DLRs Anthropomorphic Hand	5	19	Electric	Tendons
OCU hand	5	15	Electric	Servomotors
MiyazakiLab hand	5	Not found	Pneumatic	Tendons
Handroid hand	5	15	Pneumatic	Tendons
RP2009 hand	5	15	Electric + Mechanical	Motors
Bebionic v2 Hand	5	15	Electric + Mechanical	Motors + wires
TELESAR 2 hand hand	5	15	Electric + Mechanical	Tendons
SENSOPAC hand	5	6	Electric	Linkages
ZAR3 hand	5	12	Fluid + Mechanical	Tendons
Lucs Haptic habd III	5	12	Electric + Mechanical	Tendons
Keio hand	5	20	Electric + Mechanical	Mechanical with wires
MAENO hand	5	20	Electric + Mechanical	SMA wires
RP2009 hand	5	15	Electro-magnetic	Tendons
Belgrad/USC hand	5	18	Electric	Linkages

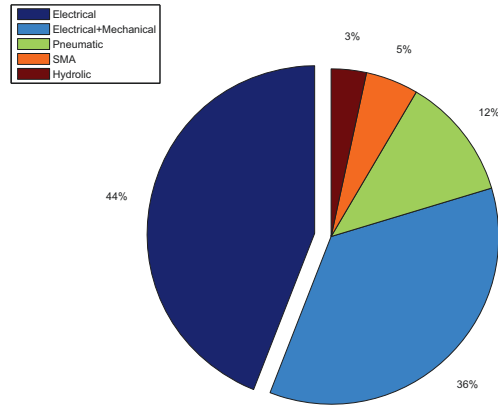


Figure 2.6: Types of actuation in robotic hands

As for actuators, they can be located either internally at the finger joints or very close, or externally in the palm, the forearm or even considered as a separated device. In the first category, the actuation of the joint does not require a transmission system. This actuation modality is the “ideal” actuation since the generated power is directly transmitted to the joints with a minimal loss. There are many technologies used for such actuation modality such as; pneumatic using the corrugated tubing, mechanical spring based and ultrasonic motors, but the most common are the electrical motors. However, the main drawback is related to the size and the weight of the hand. In the second category, the actuators are indirectly linked to the joints by means of transmission system. Unlike internal actuators, this actuation modality often suffers from the loss of transmitted power and nonlinear phenomena such as friction and backlash. Furthermore, in some cases, the control of this type of actuator becomes a tricky issue due to the time delay between the actuator and the joints. Whereas, the weight and the size are considerably reduced in comparison with the first actuation modality. External actuators can include also electrical, pneumatic, hydraulic, mechanical, SMA (Shape Memory Alloy), or combination of them. For both actuation categories, the widely used actuation system is the electrical rotary motor, and the DC motor in particular. This actuation system can be used alone or combined with other dynamical systems as a gearbox. In some external actuator examples, the joint is remotely actuated by

2. LITERATURE REVIEW

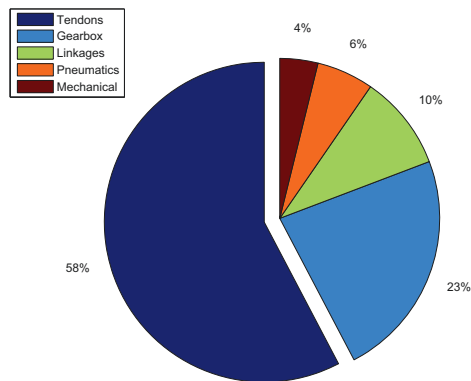


Figure 2.7: Methods of power transmission in robotic hands

electrical motor in a direction, and by means of spring device, the joint can move in the opposite direction. From the data collected in Table. 2.1, the percentage of each type of actuator is graphically displayed in Fig. 2.6. This figure shows that the electrical actuators represent the absolute majority of actuators, followed by mechanical, pneumatic, SMA wires and finally hydraulic actuators. Another study can be established by categorizing the methods of power transmission. Since the most used actuators are rotary motors in external actuation, then the tendons show a suitable means of power transmission in terms of simplicity of linkage, flexibility and, of course the price. This reasoning can be clearly seen in Fig. 2.7 in which more than half of robotic hands use tendons for power transmission.

Under-actuated robotic hands, are robots that have a lower number of actuators than de degrees of freedom. In this case, the fingers have coupled joints (called also passive joints). Their advantage is that they can adapt their structure to the shape of the object, even if it is not well known. Usually, the coupled joints are driven only in one direction and, thanks to passive spring device, the joint can move in the opposite direction.

The elastic aspect caused by the spring device in the design of the unactuated joints can considerably distribute the actuation torque making the hand to be constrained. Typically, the transmitted torque from the actuator to the joints by means of tendons

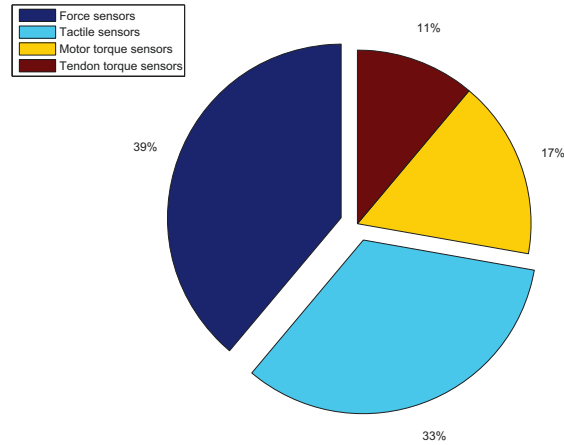


Figure 2.8: Different ways for the interaction measure

need $2n$ tendons functioning in antagonistic structure where n is the DoF of the finger. If the joint has a passive return system, then $n + 1$ tendons are needed. Usually, the tendons pass through sheaths and pulleys causing nonlinear effects that can complicate more the control of the finger. On the other hand, when the joint is rigidly coupled with the motor, the position measurements of the joint are directly obtained on the motor shaft. However, when the coupled joint is not rigid, additional position sensors are placed at joints level. Hall effect sensor is the device the most used in robotic hands, followed by the potentiometer, optical and flex sensors.

Touch sensor, or generally the force information, is certainly important as well as the position sensor, to perform interaction tasks of the robotic hand with the environment. Artificially, the interaction measurement can be done in four ways; force, tactile, motor and tendon torque sensors. According to some robotic hands reported in Table. 2.1, the force and the tactile sensors are the most frequently used, whereas the motor and the tendon torque sensors are used in fewer occasions (see Fig. 2.8). Force and tactile sensors can be found together to achieve more dexterity and can successfully recompense the gap in underactuated fingers in terms of torque sensor for coupled joints. When the contact force control is requested, it is evident that the force information and the tactile as well should be converted to actuator effort to be easily controlled. Hence, if the robotic hand is endowed with torque sensor at the actuator level, then

2. LITERATURE REVIEW

the previous process is notably avoided. This type of sensor is used in the internal actuation configuration. In the external actuation in which the joints are driven by tendons, the interaction effort can be sensed by the tendon torque by either elastomers or pushing on strain gauges sensor. The tendon torque sensor is not only used to predict the exerted force but also to prevent any potential damage of the tendon.

In general, on the basis of this diversity of sensors, the robot systems have to ensure a smooth and a stable interaction with the generic work environment. For robotic hands, in particular, when a grasping task is carried out, the control issue consists in controlling the transition phase in which the dynamic of the robotic hand switches from the free to the constrained motion. The crucial point in such tasks is in the detection of contact states. At switching phase, a large force can be arisen leading to a potential instability or to the damage of the grasped object. In the literature, the control of the behavior of a dynamic system in interaction tasks has been widely studied for many years and several control schemes have been proposed in continuous as well as discontinuous time (40). In the continuous case, the most commonly used control schemes is the impedance control which establishes the relationship between the position and the exerted force (41),(42). The position and the force have been also controlled separately (43),(44). Instead of the force control, other works propose the torque control at joint level such as used in Ultralight hand (45). In the appendix B, several control schemes adopted in different robotic hands are reported such as stiffness control (46), outer position and inner torque control (26), force/tactile control (47), pressure control (48) and current control of SMA (49).

On the other hand, during the impact transition, the speed control has to be taken into account to damp the motion and to dissipate the impact energy. In (40), the position and the force of a hybrid discrete-continuous system are controlled by the impedance control scheme. The speed is also used to detect the interaction with manipulated object. In the work covered in this thesis, position, speed and force are controlled simultaneously without any switching operation using continuous and discontinuous controllers.

Coming back to Fig. 2.5, what can be ascertained through this survey is that the robotic hands have witnessed the trend to resemble human hands as closely as possible. Kinematically, it is illustrated that the morphology of most robotic hands, which consists of five fingers, are highly articulated. For kinematics control, there are two

fundamental aspects, one is the forward kinematics and the other one is the inverse kinematics. The forward kinematics is used to locate the loci of the fingertips and also to define the workspace of the hand. This aspect has been widely used to analyze and to reproduce realistic and natural hand movements by capturing human data. In order to understand the human hand behavior and to measure the range of the motion of each hand's joint (50), different techniques have been used, e.g goniometers (51), ultrasonic (52), data glove (53) and motion capture based on markers (54, 55, 56). The reproduction of human hand behaviors is often based on learning approaches (57). It is well known that for learning a specific human hand task, it requires to perform this task for many times and could not be used for other type of task. Instead, if the inverse kinematics approach is used, it is possible to generate any robotic hand task defined in the workspace, except for those configurations that coincide with the singularities. In fact, using the inverse kinematics for controlling the robotic hands was the first approach in this field of research. A good summarize of the evolution and the state of the art of the kinematics of hand in grasping tasks is reported in (58). However, the vast majority of realized works addressed each finger independently which reduces considerably the workspace and needs to other tools to coordinate all fingers between themselves. Alternatively, the work developed in this thesis considers and treats all fingers of the robotic hand as a single multibody system, in order to provide consistent solutions. The following chapter presents the methods and the contributions towards generating the kinematics of anthropomorphic robotic hands.

2. LITERATURE REVIEW

3

Kinematic Modeling of Anthropomorphic Robotic Hand

The diversity of robotic applications in non-conventional areas stimulate more researchers to understand and to control their behaviors with aid of mathematical tools. The most common tools are forward kinematics and inverse kinematics that describe the relationship between the different geometric concepts. These concepts are firstly assigned to serial chain manipulators to determine the relationship between the base coordinate frame, commonly located at the base of the manipulator, and the end-effector frame fixed to the Tool Center Point (TCP¹). Usually, the robotic systems may operate in cluttered and dynamic environments leading to a potential reduction in the workspace. Therefore, if the robot does not have enough DoF, it could lead into a limited manipulability. In addition to that, other factors may have serious impact on the manipulability and on the reachable workspace engendered from the mechanical design itself such as the joint bounds (or joint limits). One way of dealing with these limitations is to increase the DoF by proposing new designs to make the robot more flexible, and consequently more redundancy. The kinematic redundancy occurs when the robotic system possess more DoF than those required to perform a given task. This feature can be more beneficial to deal with additional tasks provided by different kinds of potential hindrances during the performing tasks. These tasks can be either internal such as the joint limits avoidance, or external which take place during the interaction with the environment such as the obstacles avoidance. For robot manipulators, the

¹The TCP is used to refer the focal point of the tool, typically located at the tip of the tool

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

interaction with the environment is defined in the neighborhood of the end-effector. Then, finding a suitable configuration of the manipulator, from which the end-effector moves from an initial position to a desired contact point, taking into account the different potential constraints is the *Inverse Kinematics (IK)* issue. Quite simply, the aim of *IK* approach consists of determining the joint values that correspond to a known end-effector position and orientation (pose). However, from some articular configurations, it is difficult to move the end-effector through a given direction. This critical configuration represents a *singularity*. An important research effort has been dedicated to solve the *IK* problem by using diverse mathematical tools, which can be classified in three types. Those approaches which are based on stochastic laws as learning (59). These approaches are more beneficial when the kinematic model of the manipulator is not accurately defined. They can be also considered as powerful tools to estimate the Jacobian matrix in visual servoing applications preventing the accumulation of errors caused by inaccurate calibration effects or sensor offsets (60). In the same context, a redundant manipulator is visually controlled with a stereo vision in an *eye-to-hand configuration* (61). One of the most important advantages of the learning approach is that it does not suffer from the singularities since it uses the forward kinematics model. However, there are some critical configurations that the learning approach has to resolve, such as the case of redundancy. Among the infinity solutions, the method has to ensure that the computed particular solution leads to a realistic solution. This problem has been solved by adopting to the real relationships between different joints during the movement of human hand (62). Another drawback of this approach is related to the computational cost which are very expensive. The second family of *IK* solvers are the heuristic iterative approaches. The most common methods that use this approach is the *Cyclic Coordinate Descent (CCD)* algorithm. The method is not complicate to be implemented and it needs a relatively small number of iterations. As the previous approach, the CCD method does not suffer from singularities since it treats, at each step, one joint angle. This provides a simplicity for dealing with local constraints at joint variables (such as joint limits avoidance), but it is more difficult to address global constraints (such as obstacle avoidance). After the first introduction of this method by Wang and Chen in 90's (63), the CCD algorithm has been extensively used in different application fields, such as computer games industry (64) and it has recently been used for protein structure prediction (65). In this latter reference, an

indirect way has been adopted to implement the constraints by adding them on the movement maps. This approach has been also addressed in (66) proposing CCD with constraints for biomechanics application. However, among the drawbacks of the CCD algorithm is the possibility of erratic results in local minimum cases (65, 66). Further, the method is not advisable for manipulator with multiple end-effectors (67). In order to generate realistic postures, an extension of the CCD algorithm has been proposed by using a *Uniform Posture Map (UPM)* (68, 69). The aim of this latter method is to prevent the generation of unrealistic output neurons. The combination of the CCD with the UPM makes the generation of a new algorithm called *Inductive Inverse Kinematics (IIK)*, which executes in two phases. First, a pre-calculated forward kinematics table for each output neuron is generated using the UPM algorithm. Afterwards, the data table is used to find the posture vector providing the smallest distance between the known position of the end-effector and the desired one. If the actual position is near to the target position, then the CCD is used in the second phase of the algorithm to reach the target position. On the basis of the same functioning principle of the heuristic approach, another IK method, termed *Triangulation* has been recently proposed (70). This method is based on the use of the cosine rule to calculate each joint angle. Unlike the CCD method that moves the joints starting at the outer joints, the Triangulation computes each joint separately starting at the root of the kinematic chain which makes the movement similar to human skill when the animation of human bodies are simulated. The Triangulation algorithm is a very fast algorithm since it requires one iteration to reach the target position. However, this method can exhibit the following drawbacks: the chain with more than one end-effector cannot be solved, and the target cannot be reached if the constraints are incorporated, since each joint angle is calculated independently without considering the constraints of neither the previous nor the next joints. The new recent method, called *FABRIK (Forward And Backward Reaching Inverse Kinematics)*, can be considered as a powerful heuristic method which can cope with all limitations exhibited by above mentioned methods. It is a method that has a low computational cost and produces realistic postures. The constraints can easily be incorporated and can treat also the case of kinematic chain with multiple end-effectors. However, one of the limitations of this method occurs in the case of prismatic joint and when the kinematic chain is completely straight. Furthermore, when the target to reach is located at the same line as the kinematic chain which,

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

this leads to an infinite loops. The third family of the IK methods are the numerical approaches whose solutions are approximately computed using the *Jacobian* matrix (71, 72, 73, 74, 75, 76, 77). It is well know that the Jacobian matrix gives the first-order approximation of the end-effector motion. The method is able also to generate smooth motions for the joint variables, and consequently for the end-effector. These numerical solutions are not complicate to be implemented and to add constraints to the main task. Here, the constraints can be included in two manners; by extending the Jacobian matrix (78) or by minimizing a cost function that includes the main task and the constraints (76). It requires to compute at each step the inverse of the Jacobian matrix, which is an expensive operation. However, the inversion of the matrix is not always advisable due to the number of conditions that must be accomplished beforehand. In order to avoid these conditions, the Jacobian inversion has been approximated by the *Jacobian Transpose* (79, 80). Therefore, the method does not suffer from the singularity and it executes in less computational cost than the inversion method. However, the method leads to oscillations near the target position and cannot reach it with precision. On the basis of the Jacobian inversion, there are several ways to compute the solution, such as, the pseudoinverse matrix, *Damped Least Squares (DLS)*, *DLS* with the Singular Value Decomposition (*SVD*), and the recent *Selectively Damped Least Squares (SDLS)* (71) method which is used for the developing of this work to deal with the IK of all fingers of a robotic hand.

In this chapter, the different steps toward the inverse kinematics solving of an anthropomorphic robotic hand are discussed. First, the forward and the inverse kinematics solving schemes of redundant manipulators are introduced using the Jacobian concept. Then after, the redundancy is exploited to extend the main task by incorporating additional constraints. To cope with the singularity, the *DLS* method is reviewed. The relevant component of this method; the damping factor, is chosen according to many criteria. The application of this factor causes some imperfections. The remedy for this problem is done by using the *SDLS* method which is analyzed with detail throughout the rest of this chapter. This method is extended to deal with joint limits and improved in terms of computational cost. To this end, a new estimation algorithm of the singular values and their corresponding singular vectors is developed. The algorithm will be simulated using one finger of the Shadow robotic hand. Afterward, the obtained resulted are exploited to cover all fingers of the hand. Finally, the efficiency of the

proposed method will be confirmed by an experimental implementation using the real Shadow hand. The chapter will end with conclusions.

3.1 Kinematics Modeling

Kinematics, which is a word derived from the Greek word *kinein* and it means to move, is the area of mechanical systems that deals with the description of an object in motion without taking into account its inertia and forces acting on it. Usually, a robotic system consists of multi bodies linked to each other by joints that can be rotational (also called revolute), or translational (known as prismatic). In both cases, rotational and translational joint, each one forms one DoF. Mathematically, it is possible to define between each succession of coordinate frames; frame i and frame $i + 1$ for example, a transformation matrix M_i (see Fig. 3.1). This matrix is concatenated by two components; translation vector $T(x_i, y_i, z_i)$ and rotation matrix $R(\theta_i)$ defining the homogenous transformation matrix between the joint i and its parent joint $i + 1$. Computing the position and the orientation (designated as *pose*) of the P , the tip (or the end-effector) of the last link of a serial chain manipulator, from the *joint configuration* $(\theta_1, \theta_2, \dots, \theta_n)$, with respect to the base, is a problem of *forward kinematics*, where n is the dimension of the *joint space*. The overall transformation from the base to the end-effector is given by the product of the intermediate matrices M_i ,

$$M = M_n M_{n-1} \cdots M_i \cdots M_2 M_1 \tag{3.1}$$

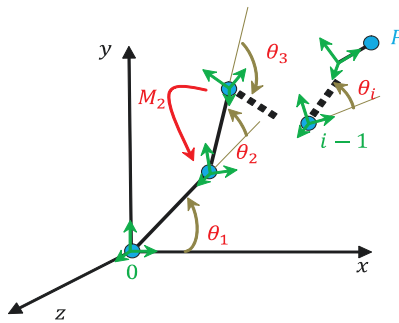


Figure 3.1: Forward kinematic chain

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

Using the D-H parameters, α_i , θ_i , a_i and d_i , the homogenous transformation matrix is given by (81):

$$M_i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \cos \theta_i & a_i \cos \alpha_i \\ \sin \theta_i & \cos \alpha_i \sin \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \alpha_i \\ 0 & \sin \theta_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

Another basic representation of the end-effector pose can be specified in terms of a minimal number of coordinates using the geometric relations of a known kinematic structure. In this way, the end-effector pose can be described by means of the $(m \times 1)$ vector, with $m \leq n$:

$$\begin{aligned} x : \mathbb{R}^n &\longrightarrow \mathbb{R}^m \\ \theta &\mapsto f(\theta) \end{aligned} \quad (3.3)$$

The vector x is defined in the space in which the manipulator tasks are specified, typically called *operational space* or *cartesian space* and m is its dimension. The forward kinematics is used to locate the pose of the end-effector by adjusting the rotation angles or the translations of each joints until fulfilling the desired location. However, this procedure can take an important time and can be only applied off-line. In contrast, the inverse kinematics concept can effectively deal with these problems providing the ability of on-line applications.

3.2 Inverse Kinematics of Redundant Manipulators

As previously said, the goal of the inverse kinematics is to place the end-effector at a specified desired target x , and find the corresponding joint configuration θ_i . In light of Eq. 3.3, the inverse kinematics is to solve the following problem:

$$\begin{aligned} \theta : \mathbb{R}^m &\longrightarrow \mathbb{R}^n \\ x &\mapsto f^{-1}(x) \end{aligned} \quad (3.4)$$

A task expressed Eq. 3.4 is often non-linear function which does not admit a trivial inverse. Particular attention has been dedicated to the study of the inverse kinematics problem. In this context, the relevant contribution of *D.E. Whitney* (72) can be considered as a primary source from which the inverse kinematics problem knew a jump towards the diversity and the perfection. The principal idea of this work is to control the rate motion rather than the position by computing the differential kinematics and instead of looking for direct and exact solution, a numerical approximative approach

3.2 Inverse Kinematics of Redundant Manipulators

has been adopted. This approach is also known as the Newton-Raphson algorithm applied to solve the inverse kinematics problem (73). Here, the task to perform is considered as the differential of vector functions of the joint configuration to obtain a Jacobian matrix:

$$J = \frac{\partial f(\theta)}{\partial \theta} \quad (3.5)$$

and instead of solving the nonlinear system Eq. 3.4, a numerical solution is computed by solving the linear matrix system:

$$\dot{x} = J\dot{\theta} \quad (3.6)$$

where $\dot{x} = \frac{\partial x}{\partial t}$, $\dot{\theta} = \frac{\partial \theta}{\partial t}$ and t is the time variable. The matrix J maps variations in the joint configuration θ to variations in the end-effector position x (82) where its dimension is $(m \times n)$. Since the linear system presented by the matrix J is only valid for small displacements, this matrix must be updated at each iteration. In addition to that, when the incremental time is short, the linear relationship Eq. 3.6 can be rewritten in the discontinuous form (83):

$$\delta x = J\delta\theta \quad (3.7)$$

In spite of the linearity of the relationship Eq. 3.7, the use of the ordinary inversion of J might not lead to solutions which depends essentially on its rank. However, for any matrix size including the case of not full rank matrix, the adopted method to solve the inverse kinematics problem consists in using the pseudoinverse method:

$$\delta\theta = J^\dagger \delta x \quad (3.8)$$

On the basis of this last linear representation and in the case of a redundant manipulator with a *degree of redundancy* $r = n - m (r > 0)$, the pseudoinverse method gives the best solution using the least squares approach:

$$\min_{\delta\theta} \|J\delta\theta - \delta x\| \quad (3.9)$$

and if J is the full row rank, then its pseudoinverse is given by:

$$J^\dagger = J^T (JJ^T)^{-1} \quad (3.10)$$

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

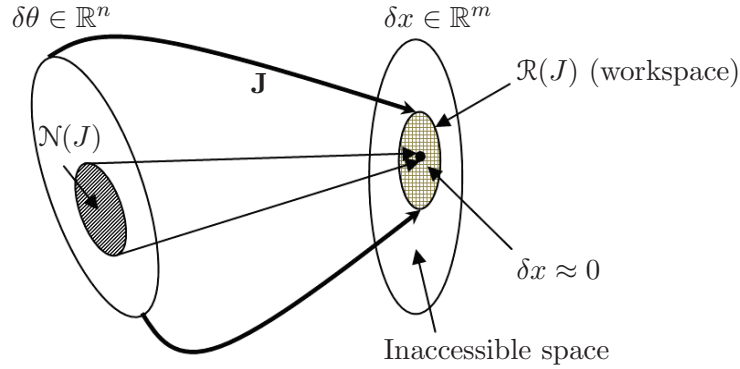


Figure 3.2: Geometric representation of the range and null space of J

The pseudoinverse of J is also called the *Moore-Penrose* (84). In this case, two fundamental subspaces are associated to define the overall solution which are the range and its null space (Fig. 3.2) (74). Ligeois has been the first to introduce the null space in the inverse kinematics problem to deal with the joint limits avoidance (75). The null space, denoted by $\mathcal{N}(J)$ is the subspace of \mathbb{R}^n from which the designed joint velocities belonging to this subspace, denoted by $\delta\theta_{\mathcal{N}}$, can be specified without affecting the velocities of the task space which can be defined by:

$$\mathcal{N}(J) = \{\delta\theta \in \mathbb{R}^n | J \cdot \delta\theta = 0\} \quad (3.11)$$

For redundant manipulators, the dimension of the null space $\mathcal{N}(J)$ is equal to $(n - m')$, where m' is the rank of matrix J . Then, any joint vector $\delta\theta$ satisfying Eq. 3.7 can be written as (85):

$$\delta\theta = \delta\theta_p + \delta\theta_{\mathcal{N}} \quad (3.12)$$

where $\delta\theta_p \in \mathbb{R}^n$ is a particular solution satisfying Eq. 3.7. The term $\delta\theta_{\mathcal{N}}$, called the internal motion, can be obtained by projection of an arbitrary n -dimensional vector φ to the null space \mathcal{N}_J . It is known that matrix $(I - J^\dagger J)$ performs a projection into the null space of J :

$$J(I - J^\dagger J)\varphi = 0 \quad (3.13)$$

Then, for all vectors, including the previously defined by the vector φ , the internal motion $\delta\theta_{\mathcal{N}}$ can be defined by:

$$\delta\theta_{\mathcal{N}} = (I - J^\dagger J)\varphi \quad (3.14)$$

This means that, the overall solution can be set as:

$$\delta\theta = J^\dagger \delta x + (I - J^\dagger J)\varphi \quad (3.15)$$

The vector φ is chosen to maximize or minimize secondary goals, denoted by $H(\cdot)$. In fact, the first term of the general Eq. 3.15 selects a joint velocity vector for which the end-effector must track the target position, while the second term exploits the redundancy of the manipulator by varying this joint velocity vector in such a way that $H(\cdot)$ is optimized without affecting the main task. Usually, the secondary goal φ (over this work, it is called the additional task) is selected as following (75):

$$\varphi = \nabla H(\cdot) = \frac{\partial H(\cdot)}{\partial \theta} = \left[\frac{\partial H(\cdot)}{\partial \theta_1} \dots \frac{\partial H(\cdot)}{\partial \theta_i} \dots \frac{\partial H(\cdot)}{\partial \theta_n} \right]^T \quad (3.16)$$

The objective function $H(\cdot)$ might depend on the kinematics parameters as well as the dynamics ones of the manipulator. With this choice of the vector φ , the solution given by Eq. 3.15 acts as a gradient optimization method which converges to a local minimum of the cost function (74). Many authors have used the null space method to satisfy different objectives, such as: torque and acceleration minimization (76), to deal with obstacle avoidance (77, 86, 87), joint limits avoidance (88, 89, 90), and to avoid the singularities (72, 78). Except for the singularity which can affect directly the tracking of the end-effector, the rest of the additional tasks have no influence on the tracking performance. The matrix is said to be singular when there is a linear dependence between the rows of the Jacobian matrix. For manipulators, the Jacobian matrix becomes singular means that after even small change in joint angle, this yields a large change at a certain end-effector direction. If the problem is regarded in the opposite way, the manipulator would not be able to perform any change in this direction. The pseudoinverse can still be applied to obtain a useful solution when J is near a singularity configuration using the null space method to avoid this critical configuration by maximizing the manipulability measure (91, 92). However, the pseudoinverse approach does not lead to a complete solution of the singularity, since when the manipulator passes through a singular configuration there are discontinuities in the elements of the computed pseudoinverse (82).

Benefiting from the redundancy in the DoF, the additional tasks have been treated by

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

another alternative called *extended Jacobian matrix* methods (93, 94). In this case, the Jacobian matrix has been augmented adding rows to obtain a square matrix (74):

$$J_{ext} = \begin{pmatrix} J \\ J_{add} \end{pmatrix} \quad (3.17)$$

where J_{ext} is the extended Jacobian matrix, J ($m \times n$) and J_{add} ($r \times n$) are the Jacobian matrices of the main and the additional task, respectively. Unlike the pseudoinverse method, the obtained solution is smoothly defined. The fact of extending the Jacobian matrix, the new kinematic system given by Eq. 3.17 is no longer redundant, hence the redundancy resolution is achieved and gives an unique solution. However, as result of concatenating rows to the main Jacobian matrix, possible artificial singularities can be formed where the extended Jacobian matrix could become deficient rank due to the potential linear dependance of the rows of J and J_{add} . Furthermore, the singularity configuration of J_{ext} can be arisen on account of adding either of the matrices J or J_{add} is singular. In such configurations, the solutions of Eq. 3.17 based on the inversion of the extended Jacobian matrix are unstable. In order to avoid the search of the exact solution near the singularities, an approximative solution has been adopted. The idea is to replace the exact solution of the linear Eq. 3.7 by a solution which takes into account at the same time the accuracy and the norm of the obtained solution. Such method is referred to as the *Damped Least Squares* which is inspired from the Levenberg-Marquardt optimization method (71, 74, 88, 95). The DLS method has much superior performance than the above mentioned methods; the pseudoinverse method and the solution given by Eq. 3.15.

3.3 Damped Least Squares

The damped least squares method is a numerical stable method which alternates between the achieving of the end-effector trajectory minimizing $\|J\delta\theta - \delta x\|$ and selecting $\delta\theta$ allowing the control of a joint rates by minimizing $\|\delta\theta\|$. Thus, one method of removing the discontinuity and also limiting the high solution norm, is to consider both criteria using the method of Lagrange multipliers. The least squares criterion for solving Eq. 3.7 is defined as follows:

$$\|J\delta\theta - \delta x\|^2 - \lambda^2\|\delta\theta\|^2 \quad (3.18)$$

where λ is a non-zero damping or singularity robustness factor. This factor gives the relative importance of the tracking accuracy of the inverse kinematics problem and the norm of angle changes. This is equivalent to replacing the original problem Eq. 3.7 by the augmented system of equations represented by:

$$\begin{bmatrix} J \\ \lambda I \end{bmatrix} \delta\theta = \begin{bmatrix} \delta x \\ 0 \end{bmatrix} \quad (3.19)$$

which is the result of the minimization of the quantity:

$$\left\| \begin{pmatrix} J \\ \lambda I \end{pmatrix} \delta\theta - \begin{pmatrix} \delta x \\ 0 \end{pmatrix} \right\| \quad (3.20)$$

The solution can be obtained by solving the consistent normal equations:

$$\begin{pmatrix} J \\ \lambda I \end{pmatrix}^T \begin{pmatrix} J \\ \lambda I \end{pmatrix} \delta\theta = \begin{pmatrix} J \\ \lambda I \end{pmatrix}^T \begin{pmatrix} \delta x \\ 0 \end{pmatrix} \quad (3.21)$$

which is equivalent to the following set of equations:

$$(J^T J + \lambda^2 I) \delta\theta = J^T \delta x \quad (3.22)$$

It is clear that $J^T J + \lambda^2 I$ is non-singular for $\lambda \in \mathbb{R}^*$. Therefore, the least squares solution is equal to:

$$\delta\theta = (J^T J + \lambda^2 I)^{-1} J^T \delta x \quad (3.23)$$

In practical application, rather than solving the Eq. 3.23 and in the redundancy case ($m < n$), an equivalent equation system is exploited given by:

$$\delta\theta = J^T (J J^T + \lambda^2 I)^{-1} \delta x \quad (3.24)$$

due to the clear equality: $(J^T J + \lambda^2 I)^{-1} J^T = J^T (J J^T + \lambda^2 I)^{-1}$. The advantage of Eq. 3.24 with respect to Eq. 3.23 is the number of iterations needed to solve the problem, where in the Eq. 3.24 the matrix being inverted is only $(m \times m)$, while in Eq. 3.23 the matrix to be inverted is $(n \times n)$. The damping factor must be carefully chosen in such a way that the Eq. 3.24 is numerically stable. When the manipulator is near singularities, this factor has to be quite large. In this manner, the joint changes $\delta\theta$ behave well and smoothly overcome the singularity configurations. However, if λ is chosen very large, this leads to a slow convergence rate. From Eq. 3.24, it is clear to show that, for $\lambda = 0$, the damped least squares formulation is the same as the

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

pseudoinverse approach. The selection of the damping factor value has been widely discussed based on several criteria, such as the maximum cartesian change, maximum joint change, configuration of the articulated system, condition number, etc (96, 97, 98, 99, 100, 101, 102, 103).

Near the singularities, the major problem is the numerical instabilities of the solution, which raises the matter of whether there is a tool to detect and to overcome this problem. Probably, the most powerful tools for analyzing the pseudoinverse and the damped least squares methods is the SVD (71, 103). The SVD theorem consists of writing any matrix as product of three matrices:

$$J = UDV^T \quad (3.25)$$

For a J ($m \times n$) matrix, D is a ($m \times n$) diagonal matrix with non-negative diagonal elements $d_{i,i}$ known as *singular values* and denoted by $\sigma_i = d_{i,i}$. When the Jacobian matrix is at a full rank, the singular values are arranged as: $\sigma_1 \geq \sigma_2 \dots \geq \sigma_m \geq 0$. If one or more than one singular value is zero, then J is singular matrix, whose rank equal to the largest value κ such that $\sigma_\kappa \neq 0$. Also, the singularity of J can be detected by computing the ratio of the largest singular value to the smallest one, called *condition number*¹: $\mathcal{K} = \frac{\sigma_1}{\sigma_\kappa}$, which is a measure of how ill-conditioned the matrix J is (82). Hence, the matrix J is ill-conditioned if its corresponding condition number is quite large. Moreover, the ill-conditioning is the responsible for the large joint changes generated by the pseudoinverse near singularities. This can be interpreted differently using the sensitivity of the solution of Eq. 3.8 (84):

$$\|\delta\theta\| \simeq \|J\| \|J^\dagger\| \frac{\|\delta x\|}{\|x\|} \|\theta\| \quad (3.26)$$

where $\|\bullet\|$ is the l_2 - norm or *Euclidean norm*. In this case, the condition number is given by:

$$\mathcal{K} = \|J\| \|J^\dagger\| = \frac{\sigma_1}{\sigma_\kappa} \quad (3.27)$$

The other matrices $U(m \times m)$ and $V(n \times n)$ of the Eq. 3.25 are orthonormal basis for the range matrix, establishing the output and the input singular vectors, respectively. The null space of J is formed by the vectors $(v_{\kappa+1}, \dots, v_n)$. Likewise, the non-zero

¹A problem with a low condition number is said to be well-conditioned, while a problem with a high condition number is said to be ill-conditioned. In this last case, a small unavoidable error taking place in numerical computing can bring undesirable solutions.

singular values $\sigma_{i(i=1,\dots,\kappa)}$ have their corresponding columns of U (u_1, \dots, u_κ) which present directions along which the end-effector can perform the obtained solution. The singular value decomposition of J can be written as:

$$J = \sum_{i=1}^m \sigma_i u_i v_i^T = \sum_{i=1}^{\kappa} \sigma_i u_i v_i^T \quad (3.28)$$

The pseudoinverse of J is equal to

$$J^\dagger = \sum_{i=1}^{\kappa} \sigma_i^{-1} v_i u_i^T \quad (3.29)$$

From the definition Eq. 3.28, it is clear to show that:

$$JJ^T + \lambda^2 = (UDV^T)(VDU^T) + \lambda^2(UU^T) = U(DD^T + \lambda^2)U^T \quad (3.30)$$

The matrix $DD^T + \lambda^2$ is an $(m \times m)$ diagonal matrix with $d_{ii} = \sigma_i^2 + \lambda^2$, which is, obviously non-singular and its inverse is the diagonal matrix $d_{ii}^{-1} = \frac{1}{\sigma_i^2 + \lambda^2}$.

Then, from Eq. 3.30:

$$J^T (JJ^T + \lambda^2 I)^{-1} = VD^T U (U^T (DD^T + \lambda^2 I)^{-1} U^T) = V \Sigma U^T \quad (3.31)$$

where Σ is a $(m \times m)$ diagonal matrix with $d_{ii} = \frac{\sigma_i}{\sigma_i^2 + \lambda^2}$. Thus, the SVD of damped least squares matrix can be given by:

$$J^T (JJ^T + \lambda^2 I)^{-1} = \sum_{i=1}^{\kappa} \frac{\sigma_i}{\sigma_i^2 + \lambda^2} v_i u_i^T \quad (3.32)$$

When comparing the above SVD with that in Eq. 3.29, a close relationship can be noticed. In both cases, the inverse of the Jacobian matrix is expressed by a general formula: $J = \sum_{i=1}^{\kappa} \sigma_i^* v_i u_i^T$. For the pseudoinverse case, $\sigma_i^* = \frac{1}{\sigma_i}$. As σ_i approaches to zero, the pseudoinverse method becomes unstable. Whereas, this situation never arises for the damped least squares method since $\sigma_i^* = \frac{\sigma_i}{\sigma_i^2 + \lambda^2}$ is always well defined for $\lambda \neq 0$. If the singular values σ_i are larger than λ , the damping factor in Eq. 3.32 has a little effect due to:

$$\frac{\sigma_i}{\sigma_i^2 + \lambda^2} \approx \frac{1}{\sigma_i} \quad (3.33)$$

which is identical to the pseudoinverse solution. The same behavior can be obtained by setting $\lambda = 0$. However, when the singular values are of the same order of the magnitude of λ or smaller, then σ_i and $\sigma_i/(\sigma_i^2 + \lambda^2)$ take different values and diverge.

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

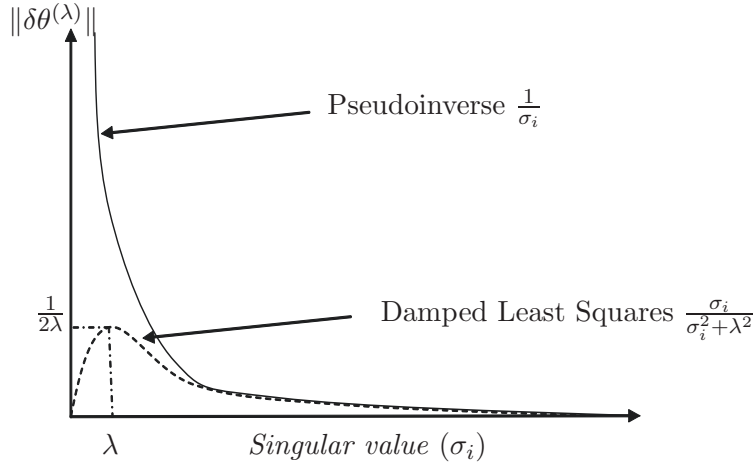


Figure 3.3: DLS and pseudoinverse solution in function of singular values

Furthermore, when $\sigma_i \rightarrow 0$, the pseudoinverse tends to infinity, whereas, for $\lambda > 0$, the damped least squares method tends smoothly to zero $\sigma_i/(\sigma_i^2 + \lambda^2) \rightarrow 0$. Fig. 3.3 shows the comparison between the obtained solution by two methods (82). Here, the damped least squares solution is considered as a function of singular values with distinguished value at $\sigma_i = \lambda$. At this value, the norm of the joint rates takes its maximum value. In all cases, the norm of the obtained solution using the damped least squares is always bounded by:

$$\|\delta\theta^{(\lambda)}\| \leq \frac{1}{2\lambda} \|\delta x\| \quad (3.34)$$

The above equation demonstrates that for a large value of λ , the norm of the joint rates is monotonically decreasing and tends to zero as $\lambda \rightarrow \infty$.

Once the damped least squares method is applied, the following task consists of determining the optimal value of the damping factor λ . This value should be carefully selected according to the physical properties of the manipulator and the kind of the task to be reached, especially the maximum allowable joint angle change and the end-effector tracking error (103). Before citing different methods that allow to select the value of λ , it is important to mention the method for which the singularity can be avoided without any use of the damping factor. The method consists of removing all the components of the solution corresponding to small singular values, called *truncated SVD* (104). The solution of this method, denoted by $\delta\theta^{(\tau)}$, is given by:

$$\delta\theta^{(\tau)} = \sum_{i=1}^{\tau} \frac{1}{\sigma_i} v_i u_i^T \delta x \quad (3.35)$$

where τ equal to the number of singular values greater than a certain threshold. Each singular value less than this threshold is considered as critical singular value which is truncated. It is clear that τ is less than or equal to the Jacobian rank κ . The effect of this method on the precision of Eq. 3.29 can be shown by computing the resultant residual function. Before this, it is important to recall that, the cartesian increment $\vec{\delta x}$ can be expressed as a linear combination of the columns of U , given by: $\vec{\delta x} = \sum_i^m \delta x_i u_i$ where $\delta x_i = \langle \vec{\delta x}, u_i \rangle$ and \langle, \rangle is the dot product. Then, the residual function is given by:

$$\|\delta x - J\delta\theta^{(\tau)}\|^2 = \sum_{i=m-\tau}^m \delta x_i^2 \quad (3.36)$$

The right hand side part of the above equation represents a portion of δx which is outside of the range space of J . Note that the magnitude of the resultant residual increases monotonically as τ becomes smaller than m . For the case where $\sigma_\tau \gg \sigma_{\tau+1}$ Maciejewski (103) proposed a damped factor with a midway value between σ_τ and $\sigma_{\tau+1}$.

A very practical approach for a dynamic setting of the damped factor is to choose it as a function of the manipulability defined as (91, 92, 96, 105):

$$\mu = \sqrt{\det(JJ^T)} \quad (3.37)$$

This choice is characterized by its computing simplicity. When the singularity configuration arises, the determinant of J is zero which involves a zero value of the manipulability. However, this parameter cannot give the prediction of the singularity and in which direction it will arise. This is due to definition of the manipulability which is computed by multiplying all singular values, whereas the relevant parameter allowing to prevent the singularity are the small singular values. Based on the damping factor concept, the effect of the damped least squares on the solution becomes clearly visible using the proprieties of the condition number of the matrix $(J^T J + \lambda^2 I)$ (82, 84, 104). To prevent the ill-conditioning problem, this quantity can be bounded applying the condition number definition on the damped system given by Eq. 3.32. Then, the condition number of the damped least squares matrix is defined by:

$$\mathcal{K}_{dls} = \frac{\sigma_1^2 + \lambda^2}{\sigma_n^2 + \lambda^2} \quad (3.38)$$

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

From the above equation and according to the knowledge of the maximum condition number, denoted by $\mathcal{K}_{max}(J)$, from which the matrix becomes ill-conditioned, the damping factor can be estimated as:

$$\lambda(\mathcal{K}_{max}(J)) = \sqrt{\frac{\sigma_1^2 - \mathcal{K}_{max}(J)\sigma_n^2}{\mathcal{K}_{max}(J) - 1}} \quad (3.39)$$

In practice, the boundaries of the joint changes are usually a necessary condition that should not be exceeded, so that a reasonable joint changes should be applied to satisfy the conditioning on the numerical resolution constraints. On the basis of this condition and from the relation Eq. 3.34, the damping factor can be specified without even considering the minimum singular value:

$$\delta\theta_{max}^{(\lambda)} = \frac{\|\delta x\|}{2\lambda} \Rightarrow \lambda = \frac{\|\delta x\|}{\delta\theta_{max}^{(\lambda)}} \quad (3.40)$$

This value of λ can be significative only for those cases where the minimum singular values is greater than the damping factor (see Fig. 3.3). In this context, the best way to choose the damping factor using the smallest singular value and satisfying the maximum joint changes, is to use the maximum norm of the damped solution $\|\delta\theta_{max}^{(\lambda)}\|$ given in general case as:

$$\|\delta\theta^{(\lambda)}\|^2 = \sum_{i=1}^{\kappa} \delta x_i^2 \left[\frac{\sigma_i}{\sigma_i^2 + \lambda^2} \right]^2 \quad (3.41)$$

where κ is the rank of the matrix J . Then, on the basis of the knowledge of $\|\delta\theta_{max}^{(\lambda)}\|$ which corresponds to the smallest singular value (see Fig. 3.3), the damping factor can be estimated as:

$$\lambda_{\delta\theta_{max}} = \sqrt{\|\delta x\| \frac{\sigma_{min}}{\|\delta\theta_{max}^{(\lambda)}\|} - \sigma_{min}^2} \quad (3.42)$$

Here, the minimum singular value is considered greater than the damping factor. Evidently, if $\sigma_{min} > \sqrt{\frac{\|\delta x_{\kappa}\|}{\|\delta\theta_{max}^{(\lambda)}\|}}$, then, the damped factor must be equal to zero.

The principal drawback of the damped solution is the resultant residual, which appears due to the addition of the damped factor to the solution. The residual, as function of the damping factor, can be given by computing the difference between the position change of the end-effector and the damped joint changes (103):

$$\|\delta x - J\delta\theta^{(\lambda)}\|^2 = \sum_{i=1}^{\kappa} \delta x_i^2 \left[\frac{\lambda^2}{\sigma_i^2 + \lambda^2} \right]^2 + \sum_{i=\kappa+1}^m \delta x_i^2 \quad (3.43)$$

The second term of the residual expression represents the range of J . It is clear that this part is not a function of the λ . However, the first term makes clear how this parameter can affect the resultant residual where its magnitude increases monotonically when λ takes high values. This magnitude reaches the minimum value at $\lambda = 0$. When the rank is full, the relative error, denoted by ΔR , between the end-effector tracking error and its velocity can be defined as:

$$\begin{aligned} \|\delta x - J\delta\theta^{(\lambda)}\|^2 &= \sum_{i=1}^m \delta x_i^2 \frac{\lambda^2}{\sigma_i^2 + \lambda^2} \\ \|\delta x - J\delta\theta^{(\lambda)}\|^2 &\leq \delta x^2 \frac{\lambda^2}{\sigma_m^2 + \lambda^2} \\ \left\| \frac{\delta x - J\delta\theta^{(\lambda)}}{\delta x} \right\|^2 = \Delta R &\leq \frac{\lambda^2}{\sigma_m^2 + \lambda^2} \end{aligned} \quad (3.44)$$

For a constraint on the maximum permitted relative error, denoted by ΔR_{max} , the specified value of the damping factor can be obtained from:

$$\lambda(\Delta R_{max}) = \sqrt{\frac{\sigma_m^2 \Delta R_{max}}{1 - \Delta R_{max}}} \quad (3.45)$$

However, this value of λ is inestimable for those cases where $\Delta R \geq 1$. If J is not full rank, then the components of the range space of J must be included. This corresponds to the second term in the right hand side of Eq. 3.43. Another simple choice of this factor using only the smallest singular as variable which is proposed in (74, 102) as:

$$\lambda^2 = \begin{cases} 0 & \text{if } \hat{\sigma}_m \geq \varepsilon \\ (1 - (\frac{\hat{\sigma}_m}{\varepsilon})^2)^2 & \text{otherwise} \end{cases} \quad (3.46)$$

So far, almost all proposed approaches for defining the damping factor values are based on the smallest singular value computation. However, it is known that the SVD computing suffers from high computational cost. Then, to overcome this problem, the recursive algorithm given in (103) allows the estimation of only the smallest singular value $\hat{\sigma}_m$ and its corresponding output singular vector \hat{u}_m . The following part explains in detail the different steps of the estimation algorithm and how to extend this algorithm to estimate also the input singular value \hat{v}_m .

3.3.1 Estimation of the Smallest Singular Value for DLS method

As shown in the previous subsection, the damped least squares solution of Eq. 3.22 can be obtained by solving either Eq. 3.23 or Eq. 3.24. Apart from the existing advantage between these latter expressions in terms of computing cost, each one allows to estimate only one type of singular vector. The estimation of the output singular values are fulfilled using the expression Eq. 3.24, whereas, the expression Eq. 3.23 is used to estimate the input singular value. In both expressions, the inverse part is always symmetric and therefore can be factorized using *Cholesky decomposition* method. Using this decomposition approach, the computation cost of the inversion operation is reduced significantly. To estimate the smallest singular value and the corresponding output singular vector, Maciejewski and Klein (103) proposed the following partitioned matrix equation:

$$(JJ^T + \lambda^2 I)[z \ ; \ \hat{u}'_m] = [\delta x \ ; \ \hat{u}_m] \quad (3.47)$$

where z is the intermediate solution obtained by solving the equations system:

$$(JJ^T + \lambda^2 I)z = \delta x \quad (3.48)$$

using the Cholesky decomposition method, and then substituting it in the following equation to obtain the overall damped solution:

$$\delta\theta^{(\lambda)} = J^T z \quad (3.49)$$

In this manner, the obtained solution is mathematically equivalent to the solution given by Eq. 3.24. The added vector \hat{u}_m is supposed to be the unit vector and it is designed to estimate the minimum singular value. Using the output singular values of the matrix J , the vector \hat{u}_m can be spanned as:

$$\hat{u}_m = \sum_{i=1}^m a_i u_i \quad (3.50)$$

The solution of left added vector of Eq. 3.47 can be given by solving:

$$\hat{u}'_m = \sum_{i=1}^m \frac{a_i}{\sigma_i^2 + \lambda^2} u_i \quad (3.51)$$

If J has a very small singular value, then the vector \hat{u}'_m will take a high value in the direction of the output singular value u_m , which is equivalent to $a_m \gg a_{i(i=1, \dots, m-1)}$ and then $a_m \simeq 1$, which leads to the following result:

$$\hat{u}'_m \simeq \frac{1}{\hat{\sigma}_m^2 + \lambda^2} u_m \quad (3.52)$$

Then, for a known value of λ , the square of the estimate $\hat{\sigma}_m$ of the smallest singular value can be found as:

$$\hat{\sigma}_m^2 = \frac{1}{\|\hat{u}'_m\|} - \lambda^2 \quad (3.53)$$

The different required steps to estimate both the smallest singular value and its corresponding output singular vector are summarized in Algorithm.1. Once the smallest singular value has been estimated, it is possible to compute the damping factor λ using one of the previous mentioned criteria.

Algorithm 1 Smallest singular value estimation and the corresponding output singular vector

Input: u_m, λ_0 .

- 1: $\hat{u}_m \leftarrow u_m$
 - 2: $\lambda \leftarrow \lambda_0$
 - 3: $\hat{u}'_m \leftarrow (JJ^T + \lambda^2 I)\hat{u}_m = \hat{u}_m$ (using Cholesky decomposition).
 - 4: $\hat{\sigma}_m \leftarrow \sqrt{\frac{1}{\|\hat{u}'_m\|} - \lambda^2}$
 - 5: $\hat{u}_m \leftarrow \frac{\hat{u}'_m}{\|\hat{u}'_m\|}$
 - 6: $\lambda \leftarrow \lambda(\hat{\sigma}_m)$
 - 7: **return** $\hat{\sigma}_m, \hat{u}_m, \lambda$
-

According to (103) and (96), the initialization of \hat{u}_m given by u_m must be computed using the SVD of J . Whereas, this may involve relatively extra computational cost and, consequently, results as a contradiction with the objective of the estimation requirement. An intuitive convenient initialization of \hat{u}_m has been proposed in (106) choosing $\hat{u}_m = e_m$ which represents the unit vector in direction m .

Based on the same philosophy, the smallest singular value and its corresponding input singular vector can be estimated substituting the system Eq. 3.24 by Eq. 3.23. The estimation algorithm using the input singular vector is described in the Algorithm.2. Like the previous algorithm, the initialization of the estimated input singular vector is chosen as $\hat{v}_n = e_n$ where e_n is the unit vector in direction n . On the other hand, both

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

algorithms are unable to estimate neither null-space nor *left null-space*¹ since both spaces are not associated to the singular value.

Algorithm 2 Smallest singular value estimation and the corresponding input singular vector

Input: v_m, λ_0 .

- 1: $\hat{v}_m \leftarrow v_m$
 - 2: $\lambda \leftarrow \lambda_0$
 - 3: $\hat{v}'_m \leftarrow (J^T J + \lambda^2 I) \hat{v}'_m = \hat{v}_m$ (using Cholesky's decomposition).
 - 4: $\hat{\sigma}_m \leftarrow \sqrt{\frac{1}{\|\hat{v}'_m\|} - \lambda^2}$
 - 5: $\hat{v}_m \leftarrow \frac{\hat{v}'_m}{\|\hat{v}'_m\|}$
 - 6: $\lambda \leftarrow \lambda(\hat{\sigma}_m)$
 - 7: **return** $\hat{\sigma}_m, \hat{v}_m, \lambda$
-

Chiaverini and Siciliano (96) proposed a new estimation scheme allowing the estimation not only the smallest singular value but also the second smallest singular value using the input singular vector. The procedure consists of taking away the estimated part obtained during the estimation of the smallest singular value from the Jacobian matrix and repeating the estimation algorithm for the second smallest singular value assuming that the estimation of $\hat{\sigma}_m$ and its corresponding input singular vector are available. On the basis of this assumption, the new matrix allowing the estimation of the second smallest singular value and its corresponding input singular vector are given by:

$$H = J^T J + \lambda^2 I - (\hat{\sigma}_n^2 + \lambda^2) \hat{v}_n \hat{v}_n^T \quad (3.54)$$

However, on the contrary to what is mentioned in their work, which expects that this matrix leads to the estimation of the second smallest singular value, the rerunning of the estimation algorithm of the matrix H leads to zero singular value. To overcome this problem, we propose in the following subsection a new correct scheme to estimate not only the second smallest singular value, but also all small singular values and their corresponding singular vectors.

¹are the last $(m-r)$ columns of U where r is the rank of J and U forms all output singular vectors.

3.3.2 New Estimation Algorithm of all Small Singular Values for the DLS method

It may happen in robotic applications that the system suffers from many singularity directions and should be properly damped by different damping factors. This means that the Jacobian matrix has more than one singular value close to zero. In order to compute the damping factors without using the SVD method, we propose a new algorithm allowing the estimation of all those singular values which generate singularities. Before starting the description of the algorithm, the Jacobian matrix is assumed a square matrix ($m = n$), which is the condition to ensure that the estimation of $\hat{\sigma}_m$, \hat{u}_m and \hat{v}_m are available. The trick would be to replace the estimated smallest singular value and its corresponding singular vector far from the second smallest singular value and not be eliminated as suggested in the method explained above. In this manner, the second smallest singular becomes the smallest one since the singular values are arranged in order from the smallest to the largest value. Therefore, the reapplication of the estimation algorithm allows to obtain the second smallest singular value and its corresponding singular vector.

In detail, when J is a square matrix, then its SVD leads to the following result:

$$\sigma_1 > \sigma_2 > \dots > \sigma_{m-1} > \sigma_m$$

Now, let σ_{max} an a priori known positive value satisfying the following condition:

$$\sigma_{max} \geq \|J\| \tag{3.55}$$

where $\|\cdot\|$ is the Euclidean norm. Then, the inequality

$$\sigma_{max} \geq \sigma_1 > \sigma_2 > \dots > \sigma_{m-1} > \sigma_m$$

is also satisfied. On the basis of the smallest singular value estimation $\hat{\sigma}_m \simeq \sigma_m$, so the following inequalities are always true:

$$\sigma_{max} + \hat{\sigma}_m > \sigma_{max} \geq \sigma_1 > \sigma_2 > \dots > \sigma_{m-1}$$

Therefore, the required procedure to be done consists of positioning the smallest singular value in place of the largest one using the corresponding estimated basis vectors. By means of one of the above definitions of the damping factor given by Eq. 3.42, Eq.

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

3.45 and Eq. 3.46, let μ_1 the appropriate damping factor associated to the estimated smallest singular value $\hat{\sigma}_m$. Whereas, λ is still considered fixed at small amount allowing the validity of the estimation algorithm. It is worth noting that the particular feature of μ_1 against λ is that the last factor is uniformly applied to all singular values, whereas, μ_1 has an effect only on the last singular value. In this case, the solution of the new updated matrix is given by:

$$\delta\theta^{(\mu_1, \lambda)} = J^T (JJ^T + (\mu_1^2 - \lambda^2)\hat{u}_m\hat{u}'_m + \lambda^2 I)^{-1} \delta x \quad (3.56)$$

By using the SVD method, the solution norm of the above equation can be defined as:

$$\|\delta\theta^{(\mu_1, \lambda)}\|^2 = \sum_{i=1}^{m-1} \delta x_i^2 \left[\frac{\sigma_i}{\sigma_i^2 + \lambda^2} \right]^2 + \delta x_m^2 \left[\frac{\sigma_m}{\sigma_m^2 + \mu_1^2} \right]^2 \quad (3.57)$$

Clearly, with an appropriate choice of λ_1 , the estimated smallest singular value $\hat{\sigma}_m$ can be more damped than the others without any disturbance to the rest of components. Note that the above equation has been introduced with some modifications in (103). In this latter reference, a so-called *numerical filtering* has been applied only to the smallest singular value, while, in the following, this approach will be extended to filter more than one. Starting from the Eq. 3.56, the estimation of the second smallest singular value $\hat{\sigma}_{m-1}$ can be achieved by replacing Eq. 3.47 with

$$(JJ^T + (\sigma_{max}^2 + \mu_1^2 - \lambda^2)\hat{u}_m\hat{u}'_m + \lambda^2 I)[z : \hat{u}'_{m-1}] = [\delta x : \hat{u}_{m-1}] \quad (3.58)$$

If we denote by H_1 the left new matrix, then the Algorithm.1 is repeated with a slight modification in Line.3, where H_1 replaces $(JJ^T + \lambda^2 I)$:

$$H_1 \hat{u}'_{m-1} = \hat{u}_{m-1} \quad (3.59)$$

Then, the estimation of the second smallest singular value of J is found from:

$$\hat{\sigma}_{m-1} = \sqrt{\frac{1}{\|\hat{u}'_{m-1}\|} - \lambda^2} \quad (3.60)$$

The overall solution after the estimation of two smallest singular values is given by:

$$\delta\theta^{(\mu_1, \mu_2, \lambda)} = J^T (JJ^T + (\mu_1^2 - \lambda^2)\hat{u}_m\hat{u}'_m + (\mu_2^2 - \lambda^2)\hat{u}_{m-1}\hat{u}'_{m-1} + \lambda^2 I) \delta x \quad (3.61)$$

3.4 Selectively Damped Least Squares Method (SDLS)

where μ_2 is the appropriate damping factor of $\hat{\sigma}_{m-1}$. The solution norm of the above equation can be written as:

$$\|\delta\theta^{(\mu_1, \mu_2, \lambda)}\|^2 = \sum_{i=1}^{m-2} \delta x_i^2 \left[\frac{\sigma_i}{\sigma_i^2 + \lambda^2} \right]^2 + \delta x_m^2 \left[\frac{\sigma_m}{\sigma_m^2 + \mu_1^2} \right]^2 + \delta x_{m-1}^2 \left[\frac{\sigma_{m-1}}{\sigma_{m-1}^2 + \mu_2^2} \right]^2 \quad (3.62)$$

Through this subsection, the damping of two estimated small singular values and their corresponding output singular vectors are detailed. This proposed approach is still valid to deal with more than two small singular values. In the case of k small singular values, the overall solution norm is generalized to take the following expression:

$$\|\delta\theta^{(\mu_1, \dots, \mu_k, \lambda)}\|^2 = \sum_{i=1}^{m-k} \delta x_i^2 \left[\frac{\sigma_i}{\sigma_i^2 + \lambda^2} \right]^2 + \sum_{j=0}^{k-1} \delta x_{m-j}^2 \left[\frac{\sigma_{m-j}}{\sigma_{m-j}^2 + \mu_{j+1}^2} \right]^2 \quad (3.63)$$

Another alternative to achieve the last equation is to use the estimation of the singular values and their corresponding input singular vectors. In this case, the Eq. 3.56 is replaced by:

$$\delta\theta^{(\mu_1, \lambda)} = (J^T J + (\mu_1^2 - \lambda^2)\hat{v}_m \hat{v}_m' + \lambda^2 I)^{-1} J^T \delta x \quad (3.64)$$

and the rest of the procedure is maintained valid to estimate all damping factors for each small singular value.

On the other hand, as shown in the overall solution given by either Eq. 3.63 or Eq. 3.64, the damping factor λ is still affecting those singular values with a good behaviors, while they should not be disturbed. Taking inspiration from the appropriate damping of the smallest singular value more than the others without affecting the rest, a new approach proposed by Buss and Kim (71) has been devised to treat each singular value individually. This approach, called the *Selectively Damped Least Squares* method is the base of this work and will be widely analyzed in order to improve it, in terms of the computational cost and the ability to deal with motion constraints.

3.4 Selectively Damped Least Squares Method (SDLS)

SDLS is considered as one of the most efficient numerical approaches against the singularity configurations. Its contribution is similar to the DLS method, but it adopts selectively a different damping factor value for each singular value. In this way, the problem of the uniform damping factor applied to all singular values is well overcome.

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

As mentioned above, this problem has been partially solved by the proposed numerical filtering whose small singular values are more damped than the rest, whereas the SDLS filters all singular values. From the selective damping constant point of view, the novelty of this approach is that this constant depends not only on the actual articular configuration of the manipulator, or explicitly on the singular values as the DLS method, but also on the current positions of both end-effector and target. Another advantage of the SDLS method consists in treating each joint angle individually by controlling its movement via the displacement of the end-effector engendered by this joint. Then, if the joint generates a displacement of the end-effector greater than the distance from which this joint has been calculated, then the movement of this joint must be more damped to prevent any excess in the target position. Since this damping process applies directly on the joint angle, then the damping constant should be defined according to the maximum permissible change in joint angles, denoted by γ_{max} . On the other hand, the increment of the end-effector should be clamped to avoid the large increments. It is a necessary condition for generating accurate motions since the Jacobian matrix gives only the first-order approximation of the movement.

$$\delta x = \begin{cases} \delta x & \text{if } \|\delta x\| \leq \delta x_{max} \\ \delta x_{max} \frac{\delta x}{\|\delta x\|} & \text{otherwise} \end{cases} \quad (3.65)$$

where δx_{max} is the maximum permissible increment in cartesian space displacement. At first, the SDLS method has been proposed to solve the inverse kinematics of a tree multi-body simulated system (71). In a similar application, we intend through this thesis to solve the inverse kinematics of the whole robotic hand, which can be seen as a tree multibody system with multi end-effectors. In order to simplify the problem and to detail the different steps making up this algorithm, a simple serial multibody system with one end-effector is addressed. In addition, through this part, the pose of the the end-effector is limited only to the position $\delta x \in \mathbb{R}^3$, while $\delta \theta \in \mathbb{R}^n$.

Let δx_i the component of δx in the direction of singular value u_i , such that:

$$N_i = \delta x_i = \langle \delta x, u_i \rangle \quad (3.66)$$

Recall that, the overall solution given by the SVD theory in ordinary case:

$$\delta \theta = \sum_{i=1}^r \sigma_i^{-1} v_i u_i^T \delta x = \sum_{i=1}^r \sigma_i^{-1} v_i \delta x_i \quad (3.67)$$

3.4 Selectively Damped Least Squares Method (SDLS)

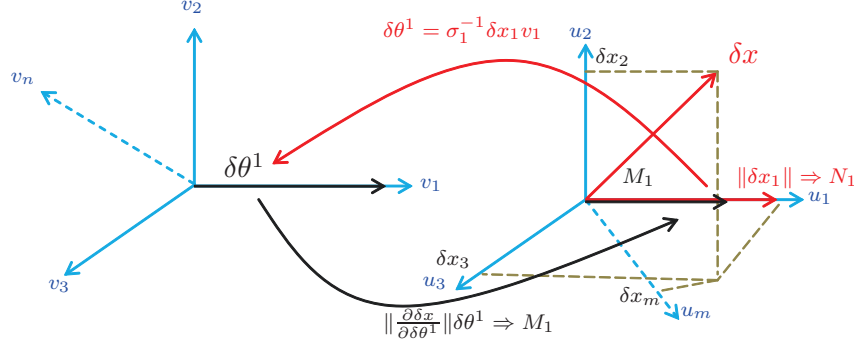


Figure 3.4: SDLS principal

$$\delta\theta = \sum_{i=1}^r \delta\theta^i \quad (3.68)$$

where r is the rank of J and $\delta\theta^i$ is the component of $\delta\theta$ in the direction v_i . Now, starting from $\delta\theta^i$, the goal is to find the maximum variation generated by this angle change on the component δx . Mathematically, this reasoning is equivalent to computing the variation of δx with respect to the small variation of $\delta\theta^i$:

$$\frac{\partial \delta x}{\partial \delta \theta^i} = \frac{\partial \delta x}{\partial \delta \theta_1} \delta \theta_1 + \dots + \frac{\partial \delta x}{\partial \delta \theta_n} \delta \theta_n \quad (3.69)$$

where $\delta\theta^i = (\delta\theta_1, \dots, \delta\theta_n)^T$. Then, the maximum possible variation of the end-effector in the direction u_i , denoted by M_i , is given by:

$$M_i = \left\| \frac{\partial \delta x}{\partial \delta \theta_1} \right\| \sigma_i^{-1} \delta x_i \|v_{1,i}\| + \dots + \left\| \frac{\partial \delta x}{\partial \delta \theta_n} \right\| \sigma_i^{-1} \delta x_i \|v_{n,i}\| \quad (3.70)$$

where $v_{i,j}$ is the component of the input singular vector at i th row and j th column. The amount $\left\| \frac{\partial \delta x}{\partial \delta \theta_i} \right\|$ represents the relative magnitude of the end-effector position engendered by the small change of the i th joint angle which represents the i th column norm of the Jacobian matrix. If we set $J = (J_1 J_2 \dots J_n)$, then

$$M_i = \sigma_i^{-1} \delta x_i \sum_{j=1}^n \|J_j\| \|v_{j,i}\| \quad (3.71)$$

A representative figure which outlines all steps of the SDLS algorithm has been proposed in Fig. 3.4. When $M_i > N_i$, this implies that the joint actuating in this direction causes the end-effector to move in the opposite direction of $\vec{\delta x}$ or it moves in the same direction but with too great displacement. In this case, the mechanical system is at singularity configuration and requires a large amount of damping to reduce the joint change in

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

this direction. From the *damping condition* $\frac{N_i}{M_i}$, a new permissible maximum angle is defined as:

$$\gamma_i = \frac{N_i}{M_i} \gamma_{max} \quad (3.72)$$

This last value should be applied to each vector $\delta\theta^i$ in such a way that each joint angle constituting this vector must be less than the new maximum angle change. The following function allows this bounding:

$$\vartheta_i = \begin{cases} \sigma_i \delta x_i v_i & \text{if } \|\sigma_i \delta x_i v_i\|_\infty \leq \gamma_i \\ \frac{\sigma_i \delta x_i v_i}{\|\sigma_i \delta x_i v_i\|_\infty} \gamma_i & \text{otherwise} \end{cases} \quad (3.73)$$

Finally, by substituting the variable $\delta\theta_i$ by ϑ_i in Eq. 3.68, the overall solution must also be clamped by applying again the maximum permissible change angle γ_{max} :

$$\delta\theta = \begin{cases} \sum_i^r \vartheta_i & \text{if } \|\sum_i^r \vartheta_i\|_\infty \leq \gamma_{max} \\ \frac{\sum_i^r \vartheta_i}{\|\sum_i^r \vartheta_i\|_\infty} \gamma_{max} & \text{otherwise} \end{cases} \quad (3.74)$$

One of the important advantages of the SDLS method with respect to the DLS method is in terms of the damping factor choice. It has been seen in the SDLS method that the damping contribution given by Eq. 3.72, Eq. 3.73 and Eq. 3.74 depends only on γ_{max} which is simple to implement, robust and more realist representing a concrete physical factor. However, for the DLS method and as previously mentioned, there are several ways to choose this factor based on maximizing or minimizing a given criterion. Nevertheless, these choices are still optimal ones and should be redefined after each change in the conditions the tracking task, such as; the maximum permitted relative error ΔR and the maximum condition number \mathcal{K}_{max} .

On the other hand, the SDLS method is expected to suffer from the computational cost due to the SVD calculation. In addition, on a trial basis, it has been noted that the damping condition of the SDLS method ($M_i > N_i$) is achieved only for small singular values, which means that there is an over computational cost caused by the fact of treating all singular values. Then, it is obvious to use only the singular values that meet the condition. From this suggestion, the question that may arise is: how to obtain a part of the singular values without resorting to the complete SVD computation? Solving this problem means that the execution of the SDLS method leads to less computational cost while maintaining its robustness against the singularity configurations. On the basis of the same philosophy presented in the subsection 3.3.2, the objective of the following

subsection consists of estimating this part of the singular values and their corresponding singular vectors, but adopting some modifications.

3.4.1 A New Estimation Algorithm for Singular Value Decomposition (SVD)

The singular value decomposition is one of the most fundamental matrix calculations in numerical linear algebra, which is applied to many fields, including digital image processing for example, where the SVD is used for *Image Compression* (107). When the image is SVD transformed, it is not compressed, but it is possible to use only a few singular values to represent the image with little difference from the original image. In this way, the image uses less storage space. But the SVD approach has the disadvantage that it is not fast from the computational point of view. In the robotic field, this tool has been widely used in particular, in the inverse kinematics control. However, for the real-time applications, this tool is still the last one to be exploited owing to its high computational cost. Despite this, it is considered as the most effective tools to deal with the singularity. In the following, a new estimation algorithm is proposed, which estimates not only the small singular values, but also all singular values. Here, unlike to the above estimation algorithm given in subsection 3.3.2, the Jacobian is not damped by any factor. In other words, the SVD of the original matrix J is estimated. Hence, the Eq. 3.47 used to estimate the damped singular value should be replaced by

$$(JJ^T)[z : \hat{u}'_m] = [\delta x : \hat{u}_m] \quad (3.75)$$

setting $\lambda = 0$. Now, let β a positive constant defined as $\beta \gg \|J\|$ used to replace each singular value at the first column after it has been estimated. It is known that the accuracy of the estimated singular value procedure depends enormously on the difference in value between the singular value to be estimated and the following one, and must be large enough, in such a way that the following relationship is maintained:

$$\hat{u}'_m \simeq \frac{u_m}{\hat{\sigma}_m^2} \quad (3.76)$$

Like the previous damped case, the singular value is now estimated as:

$$\hat{\sigma}_m = \sqrt{\frac{1}{\hat{u}'_m}} \quad (3.77)$$

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

However, the matrix in general cases does not necessarily possess small enough singular value. It could happen that this value has the same order as the others. Thus, the estimation algorithm could lead to erroneous results. Intuitively, it has been noted that the obtained results can be improved by repeating this algorithm several times. In fact, repeating the estimation process is also required in the damping case, where its impact on the results will be highlighted in subsection 4.3.3. This new estimation algorithm is reported in Algorithm.3 where the initial vector $u_0 = e_m$ and e_m is the unit vector in direction m and the parameter k is the number of times that the estimation should be repeated. This parameter depends essentially on the size of the matrix. Its value is experimentally defined. This algorithm can be adapted for input singular

Algorithm 3 Estimation of all singular values and their corresponding output singular vectors

Input: J, u_0, β

- 1: $H^0 \leftarrow JJ^T$
- 2: **for** $i = 0$ to $m - 1$ **do**
- 3: $\hat{u}_{m-i} \leftarrow u_0$
- 4: **for** $j = 1$ to k **do**
- 5: $\hat{u}'_{m-i} \leftarrow H^i \hat{u}_{m-i} = \hat{u}_{m-i}$ (Cholesky Decomposition)
- 6: $\hat{\sigma}_{m-i}^2 \leftarrow \frac{1}{\|\hat{u}'_{m-i}\|}$
- 7: $\hat{u}_{m-i} \leftarrow \frac{\hat{u}'_{m-i}}{\|\hat{u}'_{m-i}\|}$
- 8: **end for**
- 9: $H^{i+1} \leftarrow H^i + (\beta + \hat{\sigma}_{m-i})\hat{u}_{m-i}\hat{u}_{m-i}^T$
- 10: **end for**
- 11: **return** $\hat{\sigma}_i, \hat{u}_i, \hat{v}_i$.

values, changing the Line.2 by $H^0 \leftarrow J^T J$. Naturally, the expected results from the SVD estimation is to provide for each singular value its corresponding output and input singular vector. In such order, the estimation algorithm should be executed twice; using the output and the input singular vectors to estimate $\hat{\sigma}_i, \hat{u}_i$ and \hat{v}_i . However, this can cause two undesirable phenomena. The first one is the eventual high computational cost due to the execution of the algorithm twice which contradicts the desired objective consisting of real-time implementation. After several trials, it has been noted that for some cases and comparing with the obtained results from the ordinary SVD method, the output and the input singular vector take opposite signs which can considerably

3.4 Selectively Damped Least Squares Method (SDLS)

affect the accuracy of the results. This is the second undesirable phenomena that may occur. Then, to overcome this, the estimation algorithm should be done by choosing one type of singular vector to estimate the singular value, and after that, the other type of singular vector can be computed from the following relationship:

$$J^T \hat{u}_i = \hat{\sigma}_i \hat{v}_i \quad (3.78)$$

In order to reduce the computational cost, this last equation is destined to compute \hat{v}_i assuming that the estimation algorithm was based on the output singular vector \hat{u}_i . While, in the opposite case, when the estimation algorithm is performed by using \hat{v}_i , then the output singular vector can be computed from the following relationship:

$$J \hat{v}_i = \hat{\sigma}_i \hat{u}_i \quad (3.79)$$

In this way, and tacking into account the partial singular values involvement in the SDLS method, the overall computation time can reduced substantially. However, the global problem is not solved yet, where the overall solution given by Eq. 3.73 and Eq. 3.74 can not be computed if the rest of singular values are not computed yet. Then, the proposed solution consists of treating the problem as if this was the damped problem. Setting $\lambda = 0$, Eq. 3.24 can be written as

$$\delta\theta = J^T (J^T J)^{-1} \delta x \quad (3.80)$$

Extending the same analogy as the damping case, the solution of this last equation is performed in two parts, attempting to avoid an explicit inversion by using the symmetric appropriate of $J^T J$ by means of the Cholesky decomposition procedure. Every time a decision on the damping condition has been taken, a new damping constant according to the SDLS method is added to the first part of Eq. 3.75. By analogy with Eq. 3.48, and if we denote ℓ damping constants, the the first part to be solved is defined as:

$$\left(J J^T + \sum_{i=1}^{\ell} \alpha_i \hat{u}_i \hat{u}_i^T \right) z = \delta x \quad (3.81)$$

where each factor α_i represents a sequence of coefficients defined during the SDLS functioning. Then, if the damping condition is verified and if the second condition of the clamping function given by Eq. 3.73 is also verified, then the total joint angle should be damped by:

$$\delta\theta = \frac{\gamma_i}{\|\delta\theta_i\|_{\infty}} \delta\theta_i \quad (3.82)$$

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

From the equations Eq. 3.81 and Eq. 3.82, the new damping constant of the SDLS method is given by:

$$\alpha_i = \frac{1}{\frac{\gamma_i}{\|\delta\theta_i\|_\infty}} = \frac{M_i \|\delta\theta_i\|_\infty}{N_i \gamma_{max}} \quad (3.83)$$

Accordingly, once these constants are computed, the overall solution is computed by solving the same equation given in Eq. 3.49.

Finally, the process that sums up all steps of the reduced SDLS method based on the estimation of the SVD components is given in Algorithm.4. From this algorithm, it is

Algorithm 4 Reduced SDLS approach using the new SVD Estimation algorithm

Input: $J, u_0, \beta, \gamma_{max}, \delta x, \theta_0$

```

1:  $\theta \leftarrow \theta_0$ 
2:  $H^0 \leftarrow JJ^T$ 
3:  $G^0 \leftarrow JJ^T$ 
4: for  $i = 0$  to  $m - 1$  do
5:    $\hat{u}_{m-i} \leftarrow u_0$ 
6:   for  $j = 1$  to  $k$  do
7:      $\hat{u}'_{m-i} \leftarrow H^i \hat{u}_{m-i} = \hat{u}_{m-i}$ 
8:      $\hat{\sigma}_{m-i}^2 \leftarrow \frac{1}{\|\hat{u}'_{m-i}\|}$ 
9:      $\hat{u}_{m-i} \leftarrow \frac{\hat{u}'_{m-i}}{\|\hat{u}'_{m-i}\|}$ 
10:  end for
11:   $\hat{v}_{m-i} \leftarrow \frac{J^T \hat{u}_{m-i}}{\hat{\sigma}_{m-i}}$ 
12:   $N_{m-i} \leftarrow$  Using the equation 3.66
13:   $M_{m-i} \leftarrow$  Using the equation 3.71
14:  if  $M_{m-i} > N_{m-i}$  then
15:     $\delta\theta_{m-i} \leftarrow \hat{\sigma}_{m-i}^{-1} \hat{v}_{m-i} \hat{u}_{m-i} \delta x$ 
16:     $\alpha_{m-i} \leftarrow \frac{M_{m-i} \|\delta\theta_{m-i}\|_\infty}{N_{m-i} \gamma_{max}}$ 
17:     $G^i \leftarrow G^i + \alpha_{m-i} \hat{u}_{m-i} \hat{u}_{m-i}^T$ 
18:     $H^{i+1} \leftarrow H^i + (\beta + \hat{\sigma}_{m-1}) \hat{u}_{m-i} \hat{u}_{m-i}^T$ 
19:    go to 4
20:  end if
21:   $z \leftarrow G^i z = \delta x$  (Cholesky Decomposition Procedure)
22:   $\delta\theta \leftarrow J^T z$ 
23:   $\theta \leftarrow \theta + \delta\theta$ 
24: end for
25: return  $\theta$ 

```

clear that the proposed approach does not call upon the inversion Jacobian matrix. The eventual advantages brought by this approach against the original one are explained with more detail in the simulation section. Before proceeding to this topic, the SDLS is extended to deal with an additional task. This task can not be realized if the Jacobian matrix does not exhibit a redundancy in its DoF, while, it should be reminded that the aim of this work is to solve the inverse kinematics problem for cases with redundancy.

3.4.2 An Extended Selectively Damped Least Squares (ESDLS) to Deal with Joint Limits Avoidance

The additional task concept for the redundant manipulators can be interpreted in two ways; explicitly by multiplying this task by a null-space and add it to the main one leading to the overall solution as given by Eq. 3.15, or implicitly, where the Jacobian of both tasks are concatenated to form the extended Jacobian matrix as presented by Eq. 3.17. In this work, this later approach has been adopted to deal with joint limits avoidance (JLA), since the SDLS method is based on matrix computation. The JLA constraint consists of enforcing the joint variables to be executed far from their upper and lower bounds without affecting the performance of the main task.

Let $\psi(\theta)$ represents this constraint, then the corresponding Jacobian matrix of this constraint can be defined as:

$$J_{add} = \frac{\partial}{\partial \theta} \left(N_e^T \frac{\partial \psi(\theta)}{\partial \theta} \right) \quad (3.84)$$

where N_e ($n \times r$) is the null-space of the main Jacobian matrix. There are several ways to compute the null-space of a matrix. The most common choice of N_e is constructed from the concatenation of two matrices as follows (78): $N_e = [I_r : J_1^T (J_2^T)^{-1}]$, where J_1 and J_2 are ($m \times r$) and ($m \times m$) partitions of the main Jacobian: $J = [J_1 : J_2]$ and I_r is ($r \times r$) identity matrix with $r = n - m$ is the redundancy degrees. However, this definition requires J_2 to be a full rank to make it invertible which is not always satisfied. Baillieul (108) proposed a recursive approach to compute the null-space of a matrix, but, once again the matrix should be full rank since this approach is based on dividing by the determinant. To avoid all types of restrictions, the adopted null-space in this work is computed from SVD of J formed by the r 's last columns of input singular

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

vectors V . The most common objective function allowing the JLA is given by:

$$\psi(\theta) = \sum_{i=1}^n \left\| \frac{\theta_i - \bar{\theta}_i}{\theta_{u_i} - \theta_{l_i}} \right\|^p \quad (3.85)$$

where θ_{u_i} and θ_{l_i} denote the upper and the lower bounds, respectively, and $\bar{\theta}_i$ is the middle value of the joint range θ_i . Minimizing this function means that all joint angles attempt to reach their corresponding middle values. In order to improve the JLA task, several order norms are introduced. In addition, to make the objective function meaningful, the objective function has been weighted by K_i , allowing more activation of those joints evolving near the bounds. For a smooth incorporation of the JLA task with the main task, the weighting factor K_i was chosen as a function depending on $\theta_i, \theta_{u_i}, \theta_{l_i}$ and τ_i which is a region margin defined between the bounds and the center, such that when the joint variable is located outside of this region, the weight is zero, and on entering into the region, the weight factor increases gradually towards its maximum value. The structure of K_i is defined by:

$$K_i = \begin{cases} 0 & \text{if } \theta_i \leq \theta_{u_i} - \tau_i \text{ and } \theta_i \geq \theta_{l_i} + \tau_i \\ \frac{K_{max}}{2} (1 + \cos \pi \frac{\theta_{u_i} - \theta_i}{\tau_i}) & \text{if } \theta_{u_i} - \tau_i \leq \theta_i \leq \theta_{u_i} \\ \frac{K_{max}}{2} (1 + \cos \pi \frac{\theta_i - \theta_{l_i}}{\tau_i}) & \text{if } \theta_{l_i} \geq \theta_i \geq \theta_{l_i} + \tau_i \\ K_{max} & \text{if } \theta_i > \theta_{u_i} \text{ or } \theta_i < \theta_{l_i} \end{cases} \quad (3.86)$$

This parametrization has been proposed in (109) as a weight matrix for the weighted damped least squares solving, where K_{max} is the user-defined constant representing the maximum weight. In this way, the SDLS application for the new extended Jacobian matrix leads to more than one objective: tracking towards the desired positions, avoiding the potential singularity configuration and allowing joint motion far from the bounds in smooth manner.

After having discussed the SDLS method from different aspects; robustness against singularities, less computational cost and finally the ability to deal with additional tasks, the next subsection aims to confirm this obtained theoretical results through several tests and using different simulation platforms.

3.5 Implementation Issues

As indicated in the title, the global aim consists of solving the inverse kinematics problem for an anthropomorphic robotic hand. Kinematically, each anthropomorphic

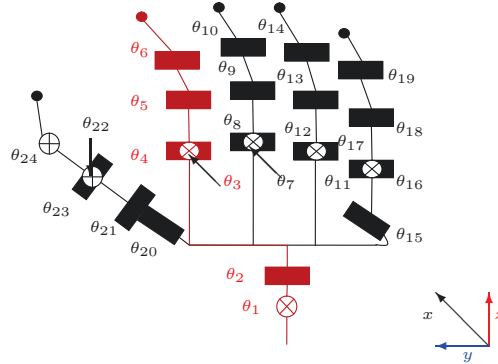


Figure 3.5: Kinematic model of Shadow hand and distributed joints naming

robotic hands has its own design, but the majority of the designs consists of several fingers connected to a palm (27, 44, 110). As a basic example of robotic hand is the *MA-I* hand (111) which consists of the thumb finger designed to be opposite to three aligned fingers. Whereas, there are other hands exhibiting more dexterity, such as the case of the Shadow hand where the thumb finger can be either aligned or in opposite position with respect to the rest of the fingers thanks to its complex structure. The Shadow hand has 24 joints and 20 DoF (see Fig. 3.5). The first joint of each of the four fingers ($\theta_6, \theta_{10}, \theta_{14}, \theta_{19}$) is coupled with the second ($\theta_5, \theta_9, \theta_{13}, \theta_{18}$). The coupling factor between these joints is not linear, otherwise it has a dynamical dependence, particularly when the distal or the middle phalange of the finger interacts with the environment. In addition, the joints of this robotic hand are remotely driven by DC-motors and a transmission system based on tendons. More details concerning the dynamical part of this hand and the control are reported in the next chapter. The robotic hand has the structure of a tree, called also, *kinematic tree*, where all joints fingers share both joints of the wrist. Recall that the Jacobian matrix can be obtained through the partial derivative of the successive products of homogenous transformation matrices expressed with the Denavit-Hartenberg's parameter of each finger (see Appendix A). On the other hand, since the expected work is to study the robotic hand as a single multi-body system, the resulting Jacobian matrix should be computed by the concatenation of all Jacobian matrices of each finger chain defined from the common frame located at the wrist joint to the fingertip frame. First, the computational aspects of different approaches are

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

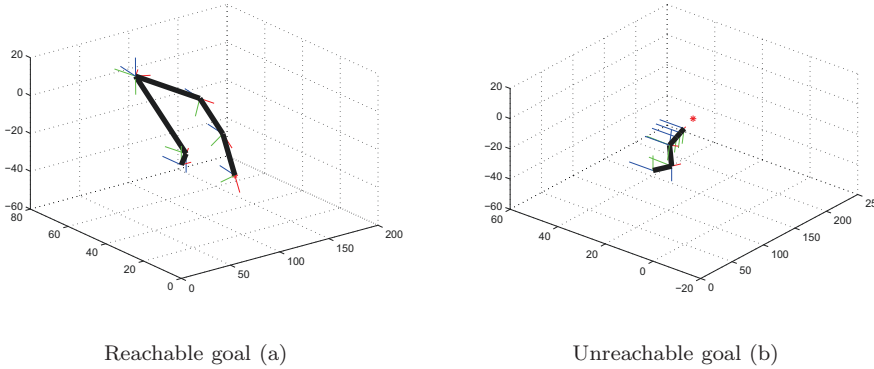


Figure 3.6: ESDLS results in reachable and unreachable tracking point of the index finger robot

discussed through the implementation on the index finger case. The obtained results are afterward extended to cover all fingers of the Shadow hand. In the first case, the index finger is considered as a serial chain of links making up a manipulator. Let x , the position of the fingertip represented by one point $x \in \mathbb{R}^3$. Assuming that all joint variables are independent, the joint space dimension is given by the number of these joints, then $\theta \in \mathbb{R}^6$. Clearly, the mechanical system is kinematically redundant, where the Jacobian matrix is a no-square matrix $J(3 \times 6)$ with the degree of redundancy $r = n - m = 3$. Then, according to the null-space definition, N_e is (6×3) matrix. Before starting with the reduced SDLS method, the extended SDLS is developed and discussed in the next subsection.

3.5.1 Extended Selectively Damped Least Squares for Index Finger

On the basis of the redundancy propriety of the finger, the null-space of the main Jacobian matrix is given by:

$$N_e = [v_4, v_5, v_6] \quad (3.87)$$

where v_4, v_5 and v_6 are the three last columns of the input singular vector V of J . In order to select a suitable norm-order, several trials are performed in different configurations, beginning with a reachable and then after an unreachable point (see Fig. 3.6). For all simulations, the buffer parameter $\tau = 6^\circ$ and the constant $K_{max} = 10$. On the basis of Eq. 3.85, it is important to define the value of order-norm. For this

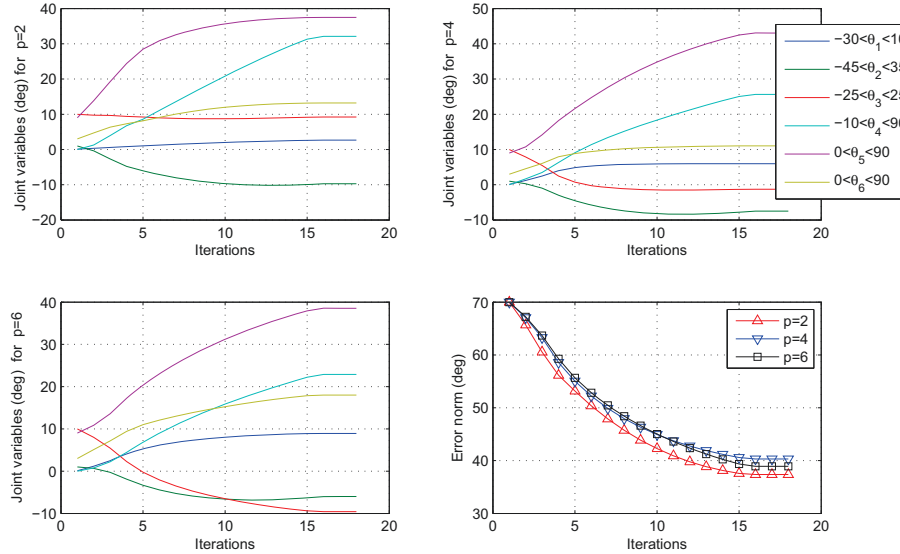


Figure 3.7: ESLDS response without singularities for $p=2$, $p=4$ and $p=6$

purpose, three values; for $n = 2$, $n = 4$ and $n = 6$ are analyzed. This latter order is the most pertinent choice for practical considerations according to (74). Fig. 3.7 shows the joint variables for each order-norm, while, the bottom-right figure shows the error norm between the joint and its corresponding center. Note that, as indicated in this figure, the tracking time is the same for all p where the convergence time is achieved at the 16th iteration. Furthermore, none of the joint variables violate their limits and all perform far from them. On the other hand, and from the performance point of view, the motion of the joint variables for $p = 2$ is closer to their centers than in the case where $p = 4$ and $p = 6$, since its associated error norm curve is placed at the bottom of both orders $p = 4$ and $p = 6$. Therefore, the function cost for 2 order-norm leads to the better performance.

In the same way, in order to simulate a singularity behavior, the robotic finger performs a pointing task toward unreachable point. Fig. 3.8 shows that, except for $p = 2$, the rest of orders exhibit oscillations at the singularity. The curves of each norm-error demonstrate that, once again, the order 2-norm leads to better results. Consequently, $p = 2$ is an appropriate choice for all types of real-time applications due essentially

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

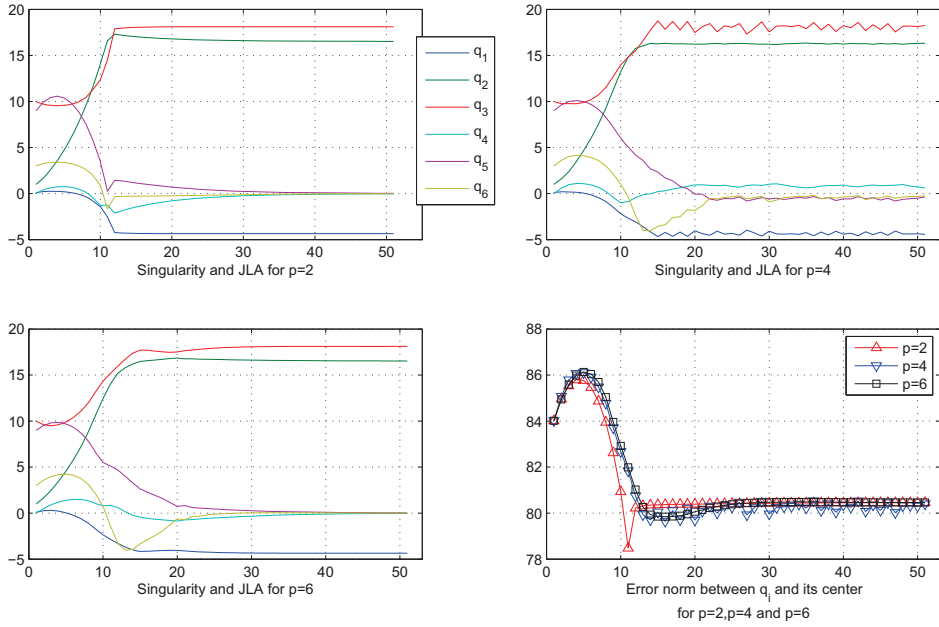


Figure 3.8: ESLDS response with singularities for $p=2$, $p=4$ and $p=6$

to the computational cost and the best provided performance toward the joint limits avoidance. For these reasons, this order will be adopted through the rest of this work. The above simulations are carried out with a maximum cartesian increment $\delta x = 3.5 \text{ mm}$ and a maximum joint angle $\gamma_{max} = 45^\circ$. These both parameters have considerable influence on the tracking time. Theoretically, for a given reachable goal, the tracking time becomes shorter whenever these values are increased. Indeed, δx_{max} can be optimally selected by exploiting the condition number definition:

$$\frac{\|\delta\theta\|}{\|\theta\|} \leq \kappa(J) \frac{\|\delta x\|}{\|x\|} \quad (3.88)$$

where $\kappa = \frac{\sigma_1}{\sigma_r}$ and $r = \min(m, n)$. Then, in the extreme case corresponding to the maximum permitted joint angle γ_{max} , it is clear to verify that:

$$\delta x_{max} = \frac{\sigma_1}{\sigma_r} \frac{\|\gamma_{max}\|}{\|\theta\|} \|x\| \quad (3.89)$$

This approach is valid when J is well-conditioned. Further, the maximum increment in the real implementation case depends essentially on γ_{max} which is related to the

k	1	2	3	4
$\ J_{ext}\ _F - \ \hat{J}_{ext}\ _F$	77.4127	0.5916	4.3552e-004	6.3014e-004

Table 3.1: Impact of the parameter k on the accuracy of the estimation algorithm

physical aspects of the robotic system, such as the permissible velocity of the actuator and evidently the position of the joint angle with respect to its limits.

3.5.2 Reduced ESDLs using SVD Estimation for Index Finger

Once the extended Jacobian matrix is a square matrix, the estimation algorithm can be successfully applied. Recall that one of the key parameters of the estimation algorithm is the number of times that the algorithm must be repeated. In order to show the impact of this factor on the performance of the estimation algorithm, a comparative study between a given Jacobian matrix and the same matrix but computed from the estimation of its singular values and singular vectors is done. The given matrix is the Jacobian matrix at a singularity configuration, in which the index finger attempts to reach an unreachable goal (see Fig. 3.6(b)). In such configuration, the matrix has at least one smallest singular value. By using Matlab tool, singular values of J_{ext} computed using the ordinary SVD method are $\sigma = (217.7112, 197.6391, 9.3577, 4.0407, 3.9182, 0.1278)^T$. The estimation of the smallest singular value $\hat{\sigma}_6 = 0.1278$, according to its reliability principle given by Eq. 3.52, expands to provide an accurate value. However, the estimation of $\hat{\sigma}_5$ is not reliable since, the difference between σ_5 and σ_4 is respectively large. Then, the proposed solution of this problem consists of executing this algorithm several times. Table. 3.1 presents the error of Frobenius norm ¹ between the given Jacobian matrix J_{ext} and its estimated matrix \hat{J}_{ext} computed as

$$\hat{J}_{ext} = \sum_{i=1}^r \hat{\sigma}_i \hat{u}_i \hat{v}_i^T \tag{3.90}$$

Notice that, from $k = 3$ the error norm becomes quite small and it is of the order of 10^{-4} . Thence, for a good SVD estimation of a matrix (6×6), the estimation algorithm must be repeated at least three times $k = 3$, since it corresponds to the shortest computational

¹From the three definitions of the Frobenius norm, the adopted one in this problem is given by $\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n \|a_{i,j}\|^2}$

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

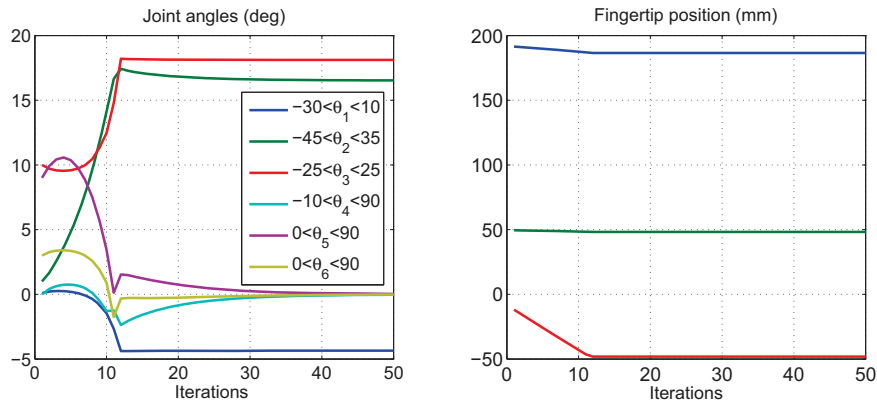
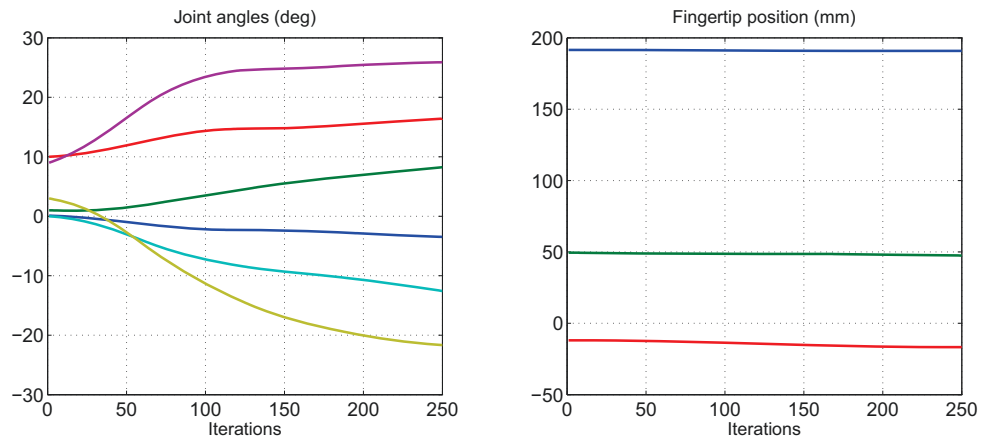


Figure 3.9: The ESDLS method results using the ordinary SVD approach

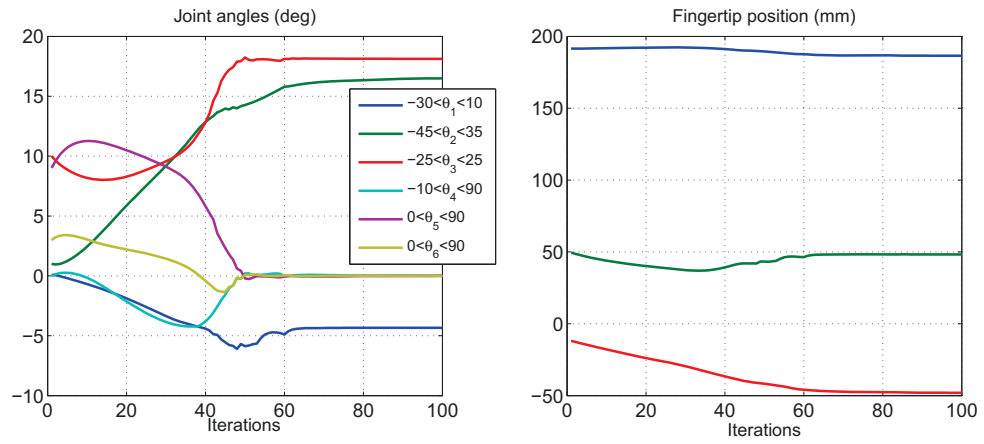
cost leading to a high accuracy. This choice can be justified by comparing the simulation results of the ESDLS method using the estimated Jacobian matrix instead of the original one. Going back to the above simulation corresponding to the singularity case, the figures shown in Fig. 3.10 illustrate the impact of k on both accuracy and convergence time of the ESDLS method. Fig. 3.9 represents the joint variations and fingertip position obtained by the application of the ESDLS method based on the ordinary SVD method computed in Matlab. Maintaining the same simulation parameters, Fig. 3.10 represents the simulation results of the estimated SVD for different values of k .

For $k = 1$ (see Fig. 3.10(a)), the convergence toward the desired position requires too much time to be achieved. Whereas, less iterations are required when $k = 2$ (see Fig. 3.10(b)). In this case, the finger exhibits some oscillation when coming close to the singularity. The best performance has been noted for $k = 3$ (Fig. 3.10(c)) where the fingertip reaches the desired point at the same time as the case of the ordinary SVD. Furthermore, the resulting joint variables show more smoothness than the ordinary case which show a high change at the iteration 11.

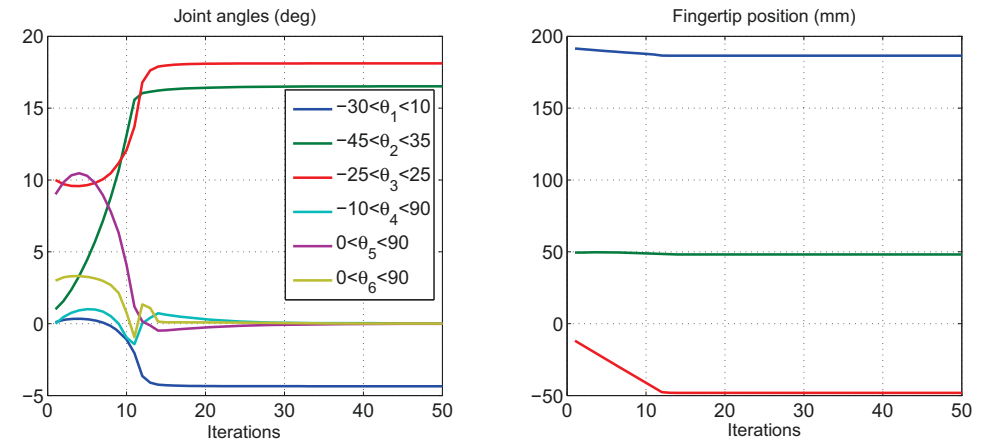
However, in terms of computational cost, the estimation of all singular values and their corresponding singular vectors is computationally expensive compared to the ordinary SVD. Whereas, the damping condition according to the SDLS method is accomplished only for those small singular values, which leads to a partial estimation and not the estimation of all singular values. In order to illustrate this reduction, an extreme case of the number of singular values that accomplish the damping condition according to the



$k=1$ (a)



$k=2$ (b)



$k=3$ (c)

Figure 3.10: The ESDLs method results using the estimated SVD for different values of k

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

Number of damped singular values	0	1	2	3
Reduced ESDLS over the ordinary SVD use (%)	86.58	67.78	48.99	22.26

Table 3.2: Percentage gains in terms of the number of operations between the reduced and the original SDLS method

SDLS method is adopted. Here, it is assumed that the index finger can take up to three singular values that should be damped. On the basis of this assumption, and taking into account the number of times k , a comparative study of the computational cost between the original and the reduced SDLS methods is done. The overall computational cost is provided by Matlab tool using the *flops* command. The number of flops is the total number of *floating-point operations* without appeal to time unit.

The comparative study involves only those changed parts since both approaches share several procedures of the SDLS method, such as the clamping function of the cartesian increment. For the original SDLS, the singular value decomposition procedure requires 4283 flops. Both magnitudes N_i and M_i for all singular values require 199 and 296 flops, respectively. The total number of flops used to compute the joint angles after the clamping operation of the cartesian increment is 176 flops (Eq. 3.73 and Eq. 3.74). Therefore, the total number of operations is 4954. With regard to the reduced approach based on the SVD estimation, the computation of $J_{ext}J_{ext}^T$, which is employed many times, costs 432 flops. The estimation of one singular value and its corresponding output singular vector as defined in Algorithm.4 requires 192 flops. As previously discussed, this procedure should be executed three times to achieve the best performance. The estimation of the input singular vector (Line.11 of Algorithm.4) and the new matrix defined in Line.17 require 79 and 72 flops, respectively. Concerning Line.18, the term $\hat{u}_i\hat{u}_i^T$ has been already computed in the above line, which allows to reduce the cost by 72 flops. Then, the total number of flops of this approach is computed by multiplying the sum of all flops defined up to here, by the number of the singular values which should be damped, and afterward add it to the cost of the two last lines; Line.21 and Line.22 which are 161 and 72 flops respectively. In the extreme case, the total number of flops is 3851 which represents a gain of 22,26% with respect to the total flops of the original method. Different percentage gains according to the number of damped singular values are reported in Table. 3.2. Note that, the percentage gain between the

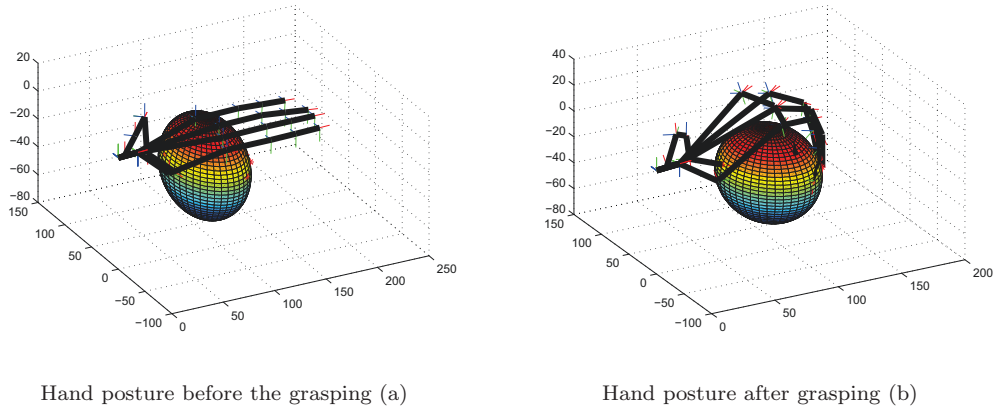


Figure 3.11: Simulated Grasping Task

reduced and the original SDLS is calculated from the following expression:

$$G(\%) = \frac{np_{originalSDLS} - np_{reducedSDLS}}{np_{originalSDLS}} 100 \quad (3.91)$$

where $np_{originalSDLS}$ and $np_{reducedSDLS}$ are the number of operations of the original and the reduced SDLS method, respectively. Table. 3.2 shows that when the SDLS method does not damp any singular value, this means that the finger evolves far from the singularity, the reduced SDLS method provides a reduction of up to 86.58% of the total number of operations of the original method. Obviously, this percentage decreases when this number increases.

In the following subsection, the obtained results for the index finger are extended to cover all the joints of the robotic hand to fulfill a grasping task. The previous methods are modified to achieve the solution of all fingers together as a single complex multibody system.

3.5.3 Reduced ESDLs Using SVD Estimation for the whole Robotic Hand

Assuming that, the task-space of all fingertips is limited to the position, then the whole robotic hand can be modeled by $x \in \mathbb{R}^{15}$ and $\theta \in \mathbb{R}^{24}$. In this case, the robotic hand shows a redundancy in the DoF where the Jacobian matrix is $J(15 \times 24)$ and the degree of redundancy $r = 9$, which is also the dimension of the null space $N_e^T (9 \times 24)$. By

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

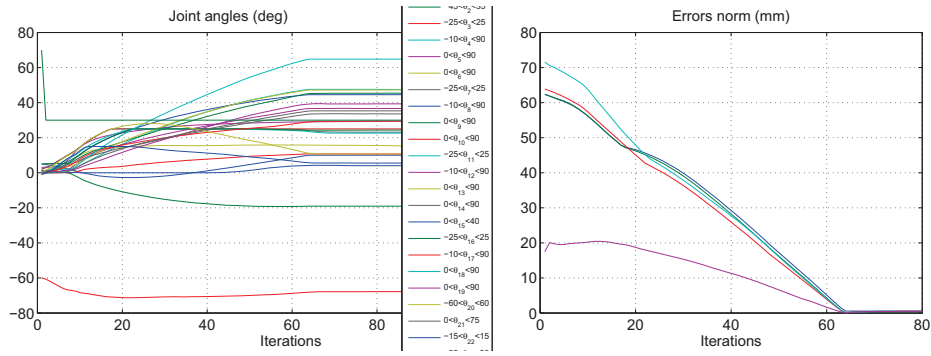


Figure 3.12: The ESDLS method results using the ordinary SVD approach

analogy with the previous case:

$$N_e = [v_{16}, \dots, v_{24}] \quad (3.92)$$

Similar to the work applied on the index finger, the joint limits avoidance is incorporated as an additional task following the methodology of Extended Jacobian matrix $J_{ext}(24 \times 24)$. Once again, the estimation algorithm of singular values and singular vectors are applied to reduce the computational cost of the ESDLS method. The procedure of finding the suitable k is achieved through the simulation of all fingers simultaneous movement performing a grasping task of a spherical object whose contact points are supposed known (see Fig. 3.11). Before the analysis is started, the application of the ESDLS method based on the ordinary SVD is presented in Fig. 3.12. These results use the same simulation parameters as in the index finger case.

In order to justify the choice of the parameter k , the errors norm variables of each fingertip are presented in Fig. 3.13. This latter figure shows that, for $k = 3$, the estimation algorithm costs more iterations to converge compared with the ordinary case Fig. 3.12(b). For $k = 5$ (see Fig. 3.13(b)), the cost has been reduced considerably to be close to the ordinary case. However, from $k = 7$, the obtained results are quite similar to the case of the ordinary SVD method (see Fig. 3.13(c) and Fig. 3.13(d)). It is worth mentioning that the parameter k has a proportionality with the dimension of the matrix, whenever the matrix has a high dimension, the estimation algorithm should be iterated more times. As previously mentioned, the estimation of all singular values and their singular vectors costs more operations than the ordinary SVD, but if only a part of singular values is estimated, the algorithm leads to less computational

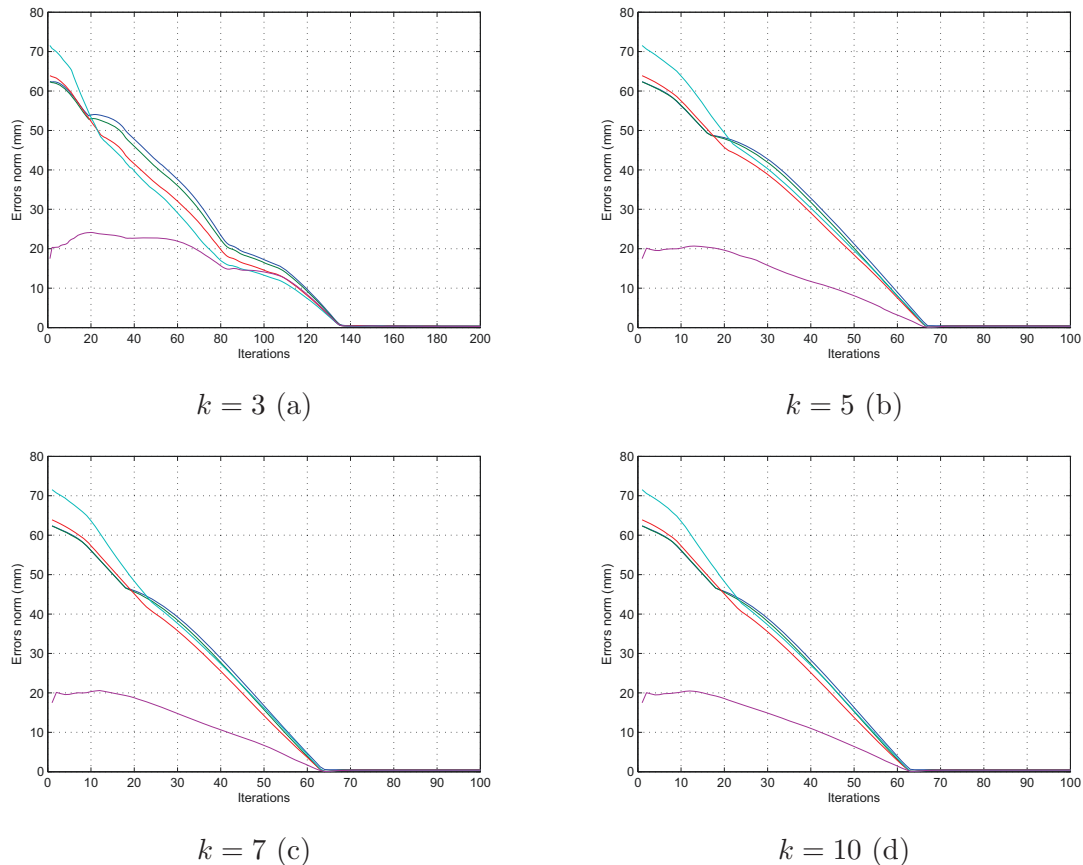


Figure 3.13: The ESDLS method results using the estimation of SVD for different values of k

cost. Table. 3.3 shows the different percentage gains when a part of the singular values are estimated. The extreme number of singular values that might lead to singularity in accordance with the damping condition of the SDLS method is assumed equal to 11.

3.6 Experimental results

In order to validate the simulation results of the proposed algorithm, several experiments have been conducted on a real system. Some experiments were conducted on an electrically motorized Shodow hand at our labs and others were carried out during a research stay at the University of Hamburg on a pneumatically actuated version of the same robotic hand. The experimental platform consists in the Shadow robotic hand which is mounted on the end-effector of a PA10 MITSUBISHI robot manipulator

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

Number of damped singular values	0	1	3	5	7	9	10	11
Reduced ESDLS over the ordinary SVD use (%)	84.13	77.54	64.34	51.15	37.95	24.76	18.16	11.57

Table 3.3: Percentage gains in terms of the number of operations between the reduced and the original SDLS method

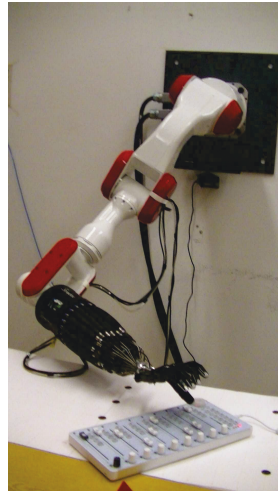


Figure 3.14: Experiment set-up used to validate the reduced ESDLS algorithm

(see Fig. 3.14). In this experiment, the objective is solving the inverse kinematics of the robotic hand. In this case, the robotic arm is assumed to be fixed in such a way that the manipulated objects belong to the workspace of the hand. The experimental evaluation suggests that the robotic hand performs two sequences: grasping a cylindrical knob by means of fingertips and performing a circular motion of the object, caused by the synchronized motion of the fingertips involved. The experimental setup is performed primarily using simulations of the hand and the knob object developed in Matlab environment. The generation of the fingertips trajectories as well as the joint angle motions by applying the reduced ESDLS algorithm are also implemented in Matlab. By means of the ROS (Robot Operating System) framework, all joint trajectories are transmitted to the real hand. After executing these trajectories, the actual joint angles are fed back into Matlab to update the fingertips trajectories generation (see Fig. 3.15). Experimentally, due to the object size which is relatively small, only the index and the thumb fingers are involved to perform the the mentioned tasks. In order to illustrate the efficiency of the reduced ESDLS algorithm, the knob object is initially

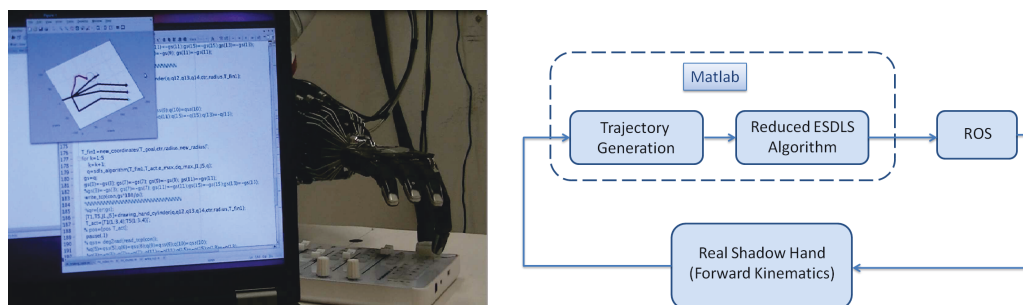


Figure 3.15: Experimental procedure

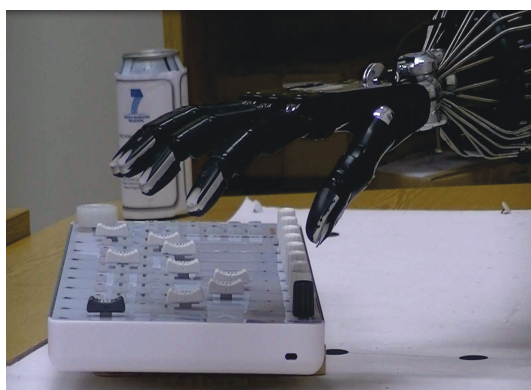


Figure 3.16: Initial configuration of the hand

placed far from the palm (see Fig. 3.16). Starting from this configuration, each finger takes an aligned from making it in singularity configuration and some joint angles are at their limits. But, just after, the fingers take more secure configurations. Thanks to the reduced ESDLS algorithm, the two joints of the wrist are involved in the performing of the task to compensate the critical configurations. Once the knob object is grasped, the second sequence consists in generating a circular motion to turn it. The sequences of this motion can be split into the circular motion (second row of Fig. 3.17) and the reposition of the fingertips in an adequate grasp to resume the circular motion (first row of Fig. 3.17). Nonetheless, these sequences offer several challenges, some of which are:

- Control of grasping force: since the robotic hand is kinematically controlled without the use of the force sensor, it is difficult to maintain contacts between fingertips and manipulated object. In this case, the problem is overcome by assuming

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

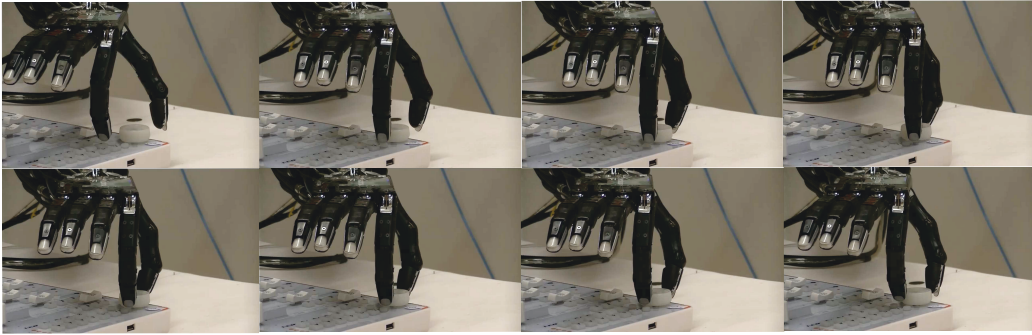


Figure 3.17: Sequences of the execution of the circular motion generated using the reduced ESDLS method

that the contact points are slightly located inside the cylindrical object and not at its surface.

- Adapting the simulated environment to the real case: to successfully replicate the simulation results, the real environment must be known with acceptable precision.

Once again, to evaluate the effectiveness of the algorithm, the hand fingertips are instructed to perform these sequences, but for small cylindrical object radius. For such object, the generated motions of different sequences are quite precise and require efficient algorithms, which are one of the particularities of the reduced ESDLS method.

3.7 Conclusion

This chapter presents the basic concepts needed for the analysis of the numerical inverse kinematics for redundant manipulators in general, and particularly for anthropomorphic robotic hands. Starting from the most popular method, the Damped Least Squares, it has been seen that the method exhibits abilities to overcome the singularity configurations successfully. However, due to the uniform way in which the damping factor is applied to all singular values, the method represents imperfections in terms of accuracy. This is explained by the influence of the damping factor on the singular values which have a good behaviors. Rather, only the smallest singular values should be damped since they are the origin of the singularities. The literature of damping factor proposes an alternative solution to mitigate this effect by damping those singular values that generate singularity more than the others. Nevertheless, the problem persists and still produces undesirable results since the selective damping factor is once again added to

the one applied uniformly to all singular values. To overcome this problem, the Selectively Damped Least Squares method is devised. The method damps each singular value by a suitable damping factor without affect the rest. The method outperforms the DLS method in the accuracy and the manner by which it selects the damping factor is more practical. It has been seen how the damping factor can be selected by taking into account some physical parameters such as, the permissible maximum angle change and the cartesian target that has to be reached. However, its main drawback is the computational cost since the basic idea behind this method consists of computing the SVD of the Jacobian matrix. During many trials, it has been noted that only a part of all singular values accomplish the damping condition of this method. This means that, the damping procedure is applied only for this part of singular values. On the basis of this criteria, the question that may come to mind, why all singular values are computed if only a part of them must be damped?. One of the most contributions of this work is the development of an estimation algorithm, which calculates some or all the singular values and their corresponding singular vectors. The idea of the algorithm is to estimate the smallest singular value and its corresponding input and output singular vectors. Before executing the estimation algorithm for the second smallest singular value, the already estimated singular value is placed far from the greater singular value using some properties of SVD components. In this way, the second smallest singular value becomes the smallest one. Therefore, by estimating only those singular values that should be damped, the computational cost of the SDLS method is reduced substantially when compared to the original method. Since only a part of all singular values is computed by the estimation algorithm, then, the overall solution is computed by adapting the Cholesky decomposition to the solution given by the estimated singular values. It is worth noting that, the proposed algorithm works correctly only when the Jacobian matrix is square. Taking advantage of the redundancy characteristic, the Jacobian matrix is extended to deal with the additional tasks. The SDLS is extended to avoid the joint limits. To this end, the objective function of the JLA task is weighted by factors to ensure smooth motions when joint angles are close to their bounds. Finally, the proposed estimation algorithm was applied to the ESDLS method to reduce the overall computational cost. In order to evaluate the effectiveness of the reduced ESDLS method, the inverse kinematics of the redundant index finger of the anthropomorphic robotic Shadow hand is simulated. Afterwards, the results are extended to cover the simulation of all fingers of to fulfill a grasping task of a spherical object. In this case, all fingers are considered as a single complex multi-body system. The simulation results yield a satisfactory success in terms of accuracy and provide an important reduction

3. KINEMATIC MODELING OF ANTHROPOMORPHIC ROBOTIC HAND

in the computational cost. The theoretical results are experimentally validated by performing the grasping and manipulation of a cylindrical object using the real Shadow hand. The experimental results demonstrate that the developed algorithm is a reliable algorithm which provides real time application, accurate and smooth solutions.

With respect to the contact control, we have seen that to keep the manipulated object in the force closure grasp configuration, the contact points are assumed to be inside the object rather than on its surface. This is not reliable for fragile objects and risk to damage the manipulated object. Therefore, in order to control the exerted forces, the robotic system must be dynamically controlled. This is the aim of the next chapter.

4

Modeling and Hybrid Control

In robotic grasping and manipulation, control of the interaction between robot and the generic work environment is still an open issue and an active area to research. This is due to the diversity of shape of manipulated objects, the type of the robotic tasks to achieve and mainly the actuator system. In the human case, this latter element refers to the muscles placed in the lower arm that control fingers to interact with the environment. As mentioned in the state of the art Chapter 2, emulating this actuation modality allows the development of significantly smaller and lighter yet powerful robotic hands. Furthermore, such mechanical systems are designed to control not only desired hand configurations but also to control interactive forces. The control of hand robotic systems in interaction tasks and robotic control in general can take several forms and strategies relying on the actuation modality, the type of sensor information and its location in the robotic system. In the robotic hand area, the morphology of most hands is quite similar, being composed by serial phalanges connected to a palm. Each phalange is connected to other phalanges by rotational joints. This is one of the reasons why most robotic hands are actuated by means of electric motors to generate a rotary motion of joints, whereas, as previously mentioned, muscles, in biology, perform their work in linear motions. On the basis of this biologic inspiration and using the motor devices, the idea for combining both mechanisms requires converting rotations to lineal motions. This can be feasible by endowing each joint with a pulley mechanism on which a tendon is spooled. There are other mechanisms that exist, such as using bar linkages or screws. However, these solutions can lead to more complexity. In addition to the use of electric motors (112),(113), (26),(114), pneumatic and hydraulic actuators provide an efficient way of creating linear motions (48),(27). Typically, the efficiency of pneumatic and hydraulic systems are lower than the efficiency of electric motors.

4. MODELING AND HYBRID CONTROL

The implementation of both pneumatic and hydraulic systems is not trivial, and the control of position and force is relatively more complicated (requiring more time to be achieved) than working with electric motors. Furthermore, pneumatic and hydraulic systems require additional support systems such as compressors. When a robotic hand is expected to be used in prosthetics robotics, pneumatic and hydraulic actuator are reliable, since they offer natural compliance.

In the case of remote actuation, position and force of the fingertip are transmitted by driving each pulley located at the finger joint using tendons. This transmission modality is not confined only to robotic hands, but also has been employed in many robotic applications since the early 80's, such as in robot manipulators (115) and Stewart platforms (116). Among the motivations that made this transmission modality attract researchers are its simplicity of implementation, the potential reduction in the weight and the cost of the hand and essentially, the possibility to transmit high forces by using powerful actuator systems (26) without changing the design of the hand. However, this transmission modality makes the system strongly nonlinear due mainly to the friction along the tendon routing from the actuator to the joint and the backlash caused by the looseness of the tendon. On the other side, in the anatomic organization of the human, there are at least two muscles working in antagonistic configuration to move the human limbs (112). Inspired from this, at least two mechanical actuators should be working simultaneously to drive one joint, ensuring, thus, a bidirectional movement. This structure enables an efficient control of the stiffness of the device, and has been seen in mechanical hand finger systems (117, 118), and also in several robot applications (119, 120, 121), particularly in those applications in which the robot interacts with an unknown environment or with other robots. The simultaneous control of the position and the stiffness in such systems is considered a complex task to achieve. However, another structure exists that plays the same role as the antagonistic one using only one actuator is used. Such actuation modality is called Series Elastic Actuator (SEA) (122). This structure provides many desirable useful features which include the compliance that allows to overcome the difficult force control in stiff robots. This is a very important feature to make robots safe in interaction with humans. On the other side, the most common actuator types used in robotic hands are electric motors, whose main disadvantage is their inability to generate large forces at slow speed. Therefore, a gearbox is required to overcome this problem. Nevertheless, introducing a gearbox to the electric motor can bring some drawbacks, in particular the reflected inertia of the motor, the backlash and potential damage of the gearbox teeth resulting from unexpected collisions of the linked stalk to the gearbox with the environment. Using

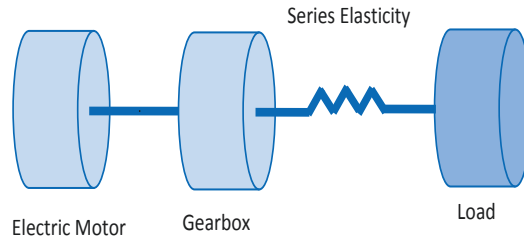


Figure 4.1: Series Elastic Actuator principle

a SEA system between the gearbox and the load allows substantially reducing these drawbacks. The working principle of this structure is schematically presented in Fig. 4.1. This structure is adopted by the Shadow robotic hand where the elasticity feature is arisen from the transmission system, whereas the motors have no elasticity. More details regarding the used robotic hand can be found in the dynamic modeling section (Section 4.1). Typically, in such structure, position control is less complex than the antagonistic one. Knowing the stiffness of the tendon, for example, would also make the force control less complex by establishing the relationship between the tendon elasticity and the force. However, the difficulty of this approach lies in the lack of the elasticity measurement, which leads to the resort to use another measurement devices providing the force or its equivalent. Starting from this topic, a great effort has been devoted to develop controllers allowing to regulate the position (or position and speed) and the force simultaneously. Whitney (123) gave different methods for achieving hybrid control. Among the methods defined by the author are stiffness control, impedance control, damping control, hybrid force-position control, and implicit and explicit force control. A good discussion of these types of control is reported at the hybrid control section (Section 4.2). Note that, even if these approaches have been developed for manipulator robots, they are valid for the robotic hand area. Interestingly, each approach is intrinsically based on a mathematical model describing the whole dynamics of the system. In order to develop a robust controller for the position and the force of the used robotic hand, the latter issue of modeling will be the first topic to be addressed in the next chapter. Afterwards, the proposed dynamic model will be simulated in an open loop. Next, a robust discontinuous control based on the “Sliding Mode Control” approach is developed. The different steps to justify the stability of this controller will be discussed. Subsection 4.3.3 gives some simulated results about the proposed controller, where the position and the speed are simultaneously controlled. The very same

controller used for the simulation is used in a real experimental set-up to evaluate the simulation results. Finally, a hybrid position/speed/force control will be designed and validated in two different ways.

4.1 Actuator system modeling

The actuator device of each joint of the Shadow hand can be split up in three parts: the motor and gearbox, a tendon-sheath transmission system, and a joint (or pulley). The overall dynamical system is the combination of the dynamical model of each part:

- The DC-motor, which creates a rotational motion with its accompanying gearbox. Its overall dynamics is the result of the combination of its electrical and mechanical properties. The electrical part has faster dynamics than the mechanical one, and can be modeled by establishing the relationship between the power supply and the mechanical variables such as the position, the speed and the torque. Whereas, the mechanical model can be obtained by using the fundamental laws of dynamics.
- The tendon-sheath transmission system is characterized by two nonlinear phenomena: friction and backlash. This part of system is experimentally identified.
- The joint (or pulley) is endowed by a position encoder that measures the magnetic field orientation of a magnet mounted on it.

4.1.1 Electric model of the DC-motor

DC-motors have a well-known model which binds the electrical and mechanical behavior. This model is composed by a resistance R_a , an inductor L_a , and an input supply V_b . A schematic representation of an armature controlled DC-motor is shown in Fig. 4.2.

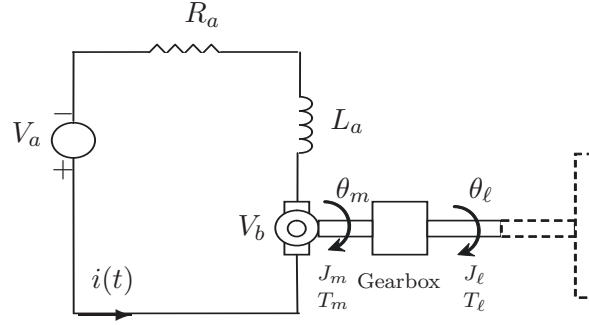


Figure 4.2: Armature DC-Motor

The resistance R_a represents the Joule loss due to the current flow into the copper conductor. The inductor behavior derives from the shape of the motor wires, which are wound in the middle of the rotor. Lastly, the generator V_b supplies a voltage proportional to the motor speed. The circuit is controlled through the voltage supply V_a . In fact, there are other factors that can reduce the motor efficiency, such as the resistance due to dispersions. However, in some recent DC-motors, such as the used MAXON DC-motor, that effect is almost neglected. The differential equation binding the armature current with both the back electromotive force (e.m.f) V_b and the armature voltage V_a is obtained by applying Kirchhoff's voltage law:

$$L_a \frac{di_a}{dt} + R_a i_a + V_b = V_a \quad (4.1)$$

Note that for an armature controlled DC-motor, the back e.m.f. induced by the armature rotation is directly proportional to the armature angular speed $\omega_m(t) = \frac{d\theta_m(t)}{dt}$, where $\theta_m(t)$ is the angular position of the motor shaft. Thus, V_b can be given as:

$$V_b = K_b \frac{d\theta_m(t)}{dt} \quad (4.2)$$

where K_b is the motor constant or the speed constant. Therefore, Eq. 4.1 can be rewritten as:

$$V_a = L_a \frac{d i_a}{dt} + R_a i_a + K_b \frac{d \theta_m(t)}{dt} \quad (4.3)$$

Another important characteristic of the DC-motor binding is that of the armature current i_a with torque motor T_M :

$$T_M = K_t i_a \quad (4.4)$$

4. MODELING AND HYBRID CONTROL

where K_t is the torque constant. This constant establishes the relationship between the electrical and the mechanical parts of the motor.

4.1.2 Mechanical equations of the motor and the gearbox

It is worth to note that the modeling of the mechanical part, as well as the electrical part, should be coherent with the mechanism used in the Shadow robotic hand. As mentioned above, all fingers are actuated by DC-motors with accompanying gearboxes. To simplify the treatment, it is assumed that all parts coming after the gearbox, primarily the payload and the friction, are included in the inertia load J_ℓ (see Fig. 4.2). A simple model for the motor with the gearbox is shown in Fig. 4.3. In this latter figure, several variables have been introduced, such as: J_m is the rotor moment of inertia around the motor axis, J_ℓ is the load inertia, ω_ℓ is the angular speed of the load, T_{ℓ_m} is the load torque in the motor axis, T_ℓ is the load torque and T_{m_ℓ} is the motor torque in the load axis.

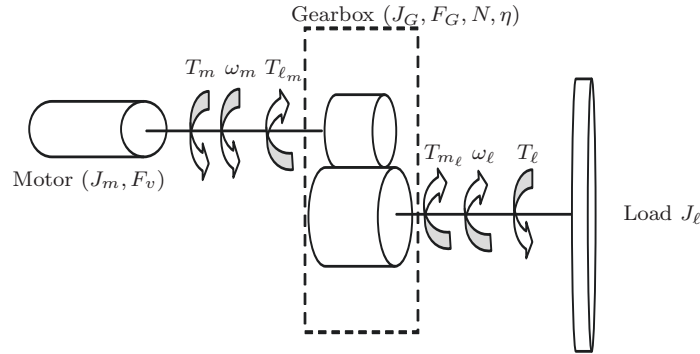


Figure 4.3: Motor-gearbox and load system

The gearbox has a reduction ratio N defined by the ratio between the motor ω_m and the load ω_ℓ velocities ($N \triangleq \frac{\omega_m}{\omega_\ell}$). Another important parameter of the gearbox is the conversion efficiency η , which relates the mechanical power of the motor axis P_m to the load axis one P_ℓ :

$$\begin{aligned} P_m \eta &= P_\ell \\ \omega_m T_{\ell_m} \eta &= \omega_\ell T_{m_\ell} \end{aligned} \quad (4.5)$$

From Newton's laws, the conservation of linear momentum at the motor output shaft

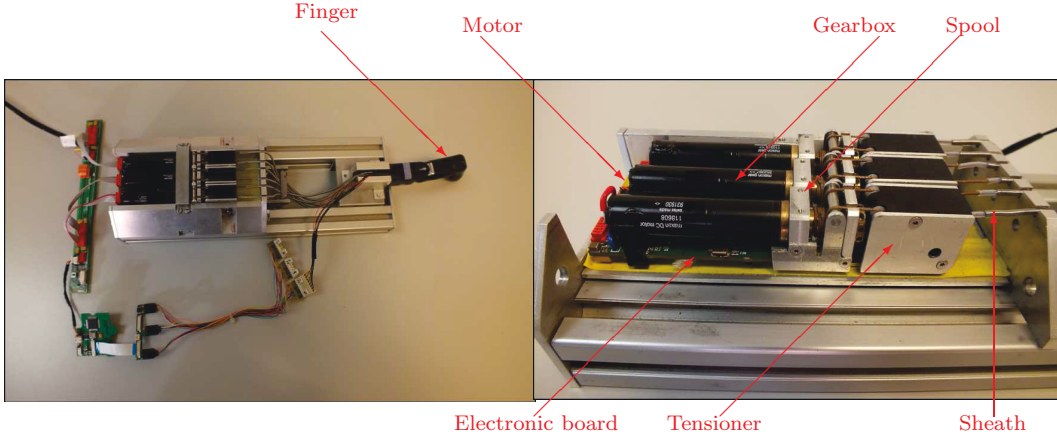


Figure 4.4: One fingered hand prototype of the Shadow hand

yields:

$$T_m - T_{\ell_m} = J_m \frac{d \omega_m}{dt} + F_v \omega_m \quad (4.6)$$

where F_v is the viscous friction coefficient of the motor. Likewise, the dynamic model of the gearbox system can be given as:

$$T_{m_\ell} - T_\ell = J_G \frac{d \omega_\ell}{dt} + F_G \omega_\ell \quad (4.7)$$

where F_G and J_G denote the internal damping and the internal inertia of the gearbox, respectively. By using Eq. 4.6 and Eq. 4.7, the equality given by Eq. 4.5 can be rewritten as:

$$\left(J_m + \frac{J_G}{\eta N^2} \right) \frac{d \omega_m}{dt} + \left(F_v + \frac{F_g}{\eta N^2} \right) \omega_m = T_m - \frac{T_\ell}{\eta N} \quad (4.8)$$

This latter equation describes the dynamic model of the system with respect to the motor speed ω_m . Whereas, the same system can be modeled by another dynamic in which the speed variable is the load speed ω_ℓ :

$$(J_m N^2 \eta + J_G) \frac{d \omega_\ell}{dt} + (F_v N^2 \eta + F_g) \omega_\ell = N \eta T_m - T_\ell \quad (4.9)$$

Through this work, the dynamics given by Eq. 4.9 will be adopted due to the fact that the joint finger is remotely connected to the spool which is directly attached to the gearbox axis, and not to the motor axis (see Fig. 4.4). In Fig. 4.5, the actuator model of each joint is simplified. Through this figure and according to Eq. 4.9, all possible nonlinear phenomena that could be arisen in the actuator system are included in the load torque T_ℓ . Among these phenomena, the most significant effect is the friction effect

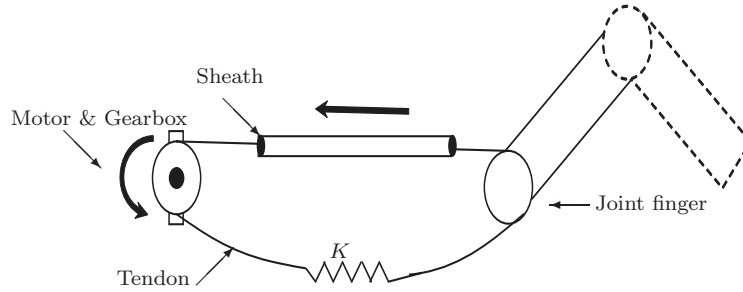


Figure 4.5: Model of actuator

due mostly to the routing of the tendon in the sheath and the contact of the tendon with different parts of the system as the spool housing, the tensioner and the pulley (27). Furthermore, when the tendon is loosened in one direction and tightened in the opposite one, the actuator will be exposed to the backlash phenomenon (called also dead-zone). The actuator suffers from this phenomenon once it changes the direction of rotation. Both the friction and the backlash can be seen as two phenomena with opposite effects. When friction is taking place, the motor requires great power to put the joint in motion, whereas the backlash leads to high motion even if a very small power is applied to the motor. Finally, the load torque T_ℓ can be split into the friction ($F_{c(1,2)}(\theta, \text{sign}(\omega))$), load torque ($T_{\ell_{object}}$), and backlash ($H(K, \theta, \omega)$) effects. The equivalent load torque can be formulated as follows:

$$T_\ell = (1 - \alpha)(F_{c(1,2)}(\theta, \text{sign}(\omega)) + T_{\ell_{object}}) + \alpha H(K, \theta, \omega) \quad (4.10)$$

where the constant $\alpha = \{0, 1\}$ is a selection parameter which alternates between the different phenomena of the torque load T_ℓ . When $\alpha = 1$, the motor changes the training motion, and only the backlash effect is taking place. Once the backlash is completely executed, the value $\alpha = 0$, and only the first sum term of T_ℓ is considered. Thus, the overall dynamic system can be structured into three states. This classification depends upon the actual speed direction and also the expected one.

4.1.3 Friction modeling

Tribology, the science of friction, has been widely studied in the last century, especially in the field of engineering, to understand, to model, and to control this phenomenon. In the friction modeling area, most works analyze friction according to speed (124, 125). For industrial applications, this analysis is widely studied, where speed control of machines is very important and should be well analyzed (126). However, in the used

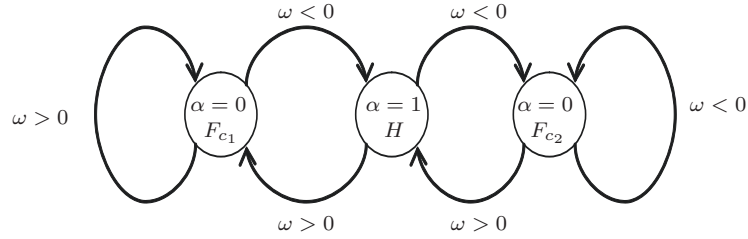


Figure 4.6: Different working states according to stated state and the direction

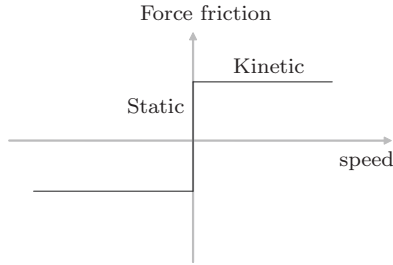


Figure 4.7: Force friction

robotic hand, the actuator system is expected to drive the joint angle only about ten degrees with relatively slow speed. Moreover, the friction effect in the actual system depends on the motor-gearbox system as well as on the joint finger position. For these reasons, instead of speed, the joint position is used to model the friction effect. Assume that, for all the subsequent times, the overall friction force is equivalent to the static phase, which includes the friction of both the motor-gearbox and the tendons-sheath system (see Fig. 4.7).

The friction model is identified by using a random set of angular positions. On the basis of the proportionality existing between torque and current of the DC-motor, the static force is computed by adopting a simple circuit consisting of a voltage supply and a variable resistance connected to the prototype finger. The empirical test consists of changing the resistance value until the motor starts to move. On the basis of the knowledge of the resistance value and the tension supply, the circulating current can be calculated and converted to the torque (T_{m0}) by using the torque constant K_t (Eq. 4.4). During the empirical collection of data, it has been shown that, by applying T_{m0} , the joint finger continues moving constantly, which reinforces the hypothesis that the static friction force can be extended to be the dynamic friction force. In Fig. 4.8, the static force friction curves are presented using one joint of the finger where its range angle is

4. MODELING AND HYBRID CONTROL

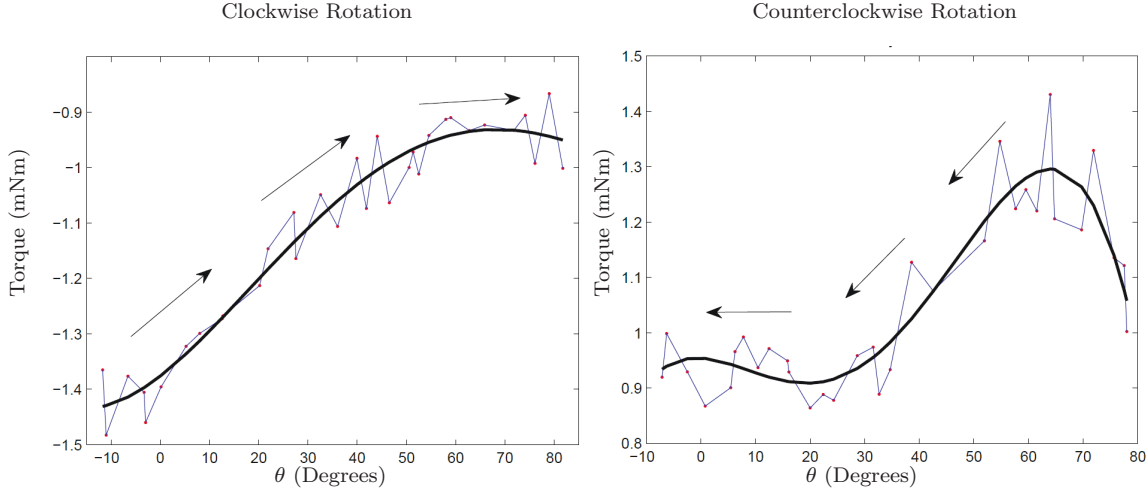


Figure 4.8: Static friction torque versus joint angular position

$\dot{\theta}$	a_1	a_2	a_3	a_4	a_5
> 0	1.4554 10e-8	-3.297202 10e-6	1.51229149 10e-5	6.930741273 10e-4	-1.376147477
< 0	-1.52716 10e-8	1.6828153 10e-6	-3.58629942 10e-5	-6.03088395 10e-4	0.9542918409

Table 4.1: Numerical values of coefficients of the polynomial equation

defined as $\theta \in [-10^\circ, 90^\circ]$. The blue line represents the real static force friction which is analyzed in both directions of the joint finger; clockwise direction (Fig. 4.8 in the left) and counterclockwise direction (Fig. 4.8 in the right). For each case, a mathematical model has been proposed which is approximated by a fourth order polynomial equation given by:

$$F_c(\theta, \dot{\theta}) = a_1\theta^4 + a_2\theta^3 + a_3\theta^2 + a_4\theta + a_5 \quad (4.11)$$

The unknown coefficients $a_i = \{a_1, a_2, a_3, a_4, a_5\}$ are identified using the Levenberg Marquart optimization method of Matlab, where the objective function to be minimized is given by:

$$\min_{a_i} \|F(\theta) - F_c(\theta, \dot{\theta})\| \quad (4.12)$$

where $F(\theta)$ is the actual friction given by the blue line. The approximated smooth curves are given in Fig. 4.8 by the dark solid line. The obtained values of these coefficients are reported in Table. 4.1.

4.1.4 Backlash

In addition to friction, the second nonlinear phenomenon that can affect considerably the performance of mechanical systems is backlash. Among the systems that suffer

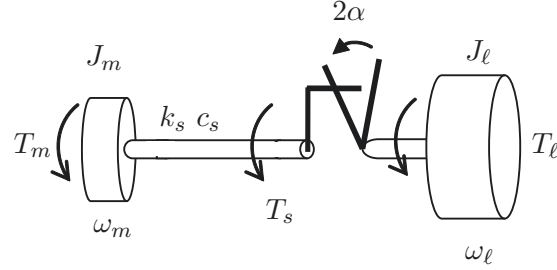


Figure 4.9: Backlash of the system motor-gearbox-shaft-load

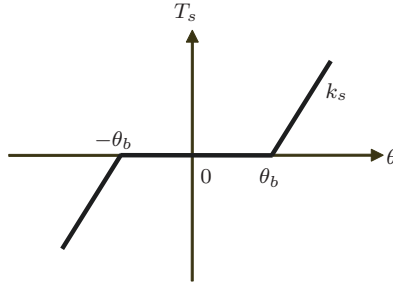


Figure 4.10: Dead zone model

from this phenomenon are those where the actuator device is not directly connected to the driven part. In particular, for systems driven by motors, the backlash occurs in the gearbox and the transmission system which is, generally, the shaft located between the gearbox and the load (see Fig. 4.9). On the gearbox, the backlash could be the result of two occurrences; when the motor has to change the direction of rotation or when the load is disturbed. In both instances, the motor loses the contact with the load. In this case, only the actuator system is moving and not the load. In regard to the transmission system, the backlash occurs when the shift twist is taken place. To counteract the problem, the most of the proposed works are based on the compensation of this effect. To be able to compensate this effect, the backlash has to be mathematically modeled. For this target, Nordin and Gutman had outlined this phenomenon by giving a survey, in which they revealed 96 references (127). According to the place where the backlash occurs, the authors have exposed different models. Once the backlash is compensated, the resulted systems become linear. Here, many linear controllers have been adopted such as, P, PI, PID, adaptive controller and observer based controller using the Luenberger observer. Note that, the control of the backlash consists of the angle lag ($\theta_b \leq |\alpha|$) related to the shaft torque by:

$$T_s = k_s \theta_s + c_s \dot{\theta}_s \quad (4.13)$$

4. MODELING AND HYBRID CONTROL

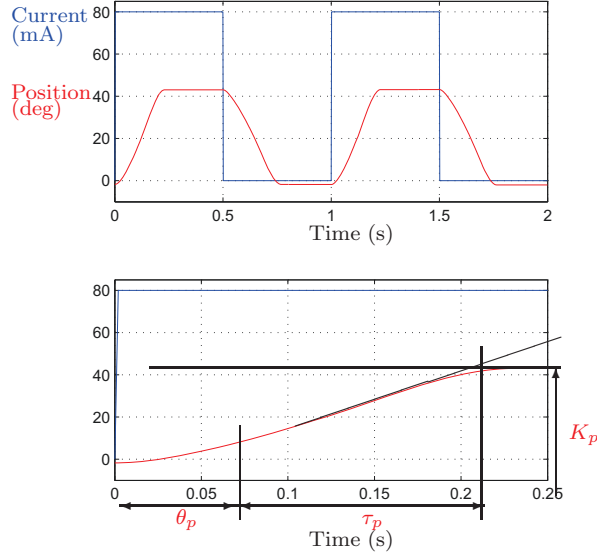


Figure 4.11: Backlash response

where $\theta_s = \theta_d - \theta_b$ represents the shaft twist, θ_d is the difference angle between the motor and the load, k_s is the elasticity of the shaft, and c_s is the viscous damping. There is another simplified model in which the inner shaft damping is neglected and the dead zone model is taken place (see Fig. 4.10). In this model, the shaft torque is proportional defined to the shaft twist:

$$T_s = k_s \theta_s = k_s D_\alpha(\theta_d) \quad (4.14)$$

where the dead zone function $D_\alpha(\theta_d)$ is defined by:

$$D_\alpha = \begin{cases} \theta_d - \alpha & \text{if } \theta_d > \alpha \\ 0 & \text{if } |\theta_d| \leq \alpha \\ \theta_d + \alpha & \text{if } \theta_d < -\alpha \end{cases} \quad (4.15)$$

Regarding the mechanical system given in Fig. 4.5, the experimental backlash model $H(K, \theta, \omega)$ is obtained by applying a maximum torque that makes the actuator “absorbs” the slack tendon without getting the joint finger moving. The response model from the real experiment is shown in Fig. 4.11. It can be shown that the obtained curve is the response of the first order system which is identical to the dynamical model given by Eq. 4.13. Furthermore, the response shows a dead time interpreted by the existence of an inflexion point.

From the graphical time response, the mathematical model describing a first order

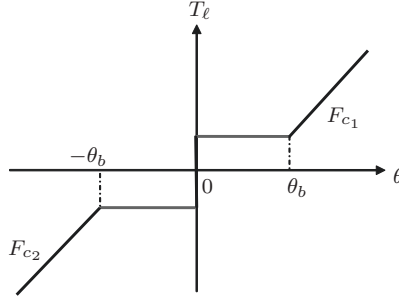


Figure 4.12: Dead zone model with friction model

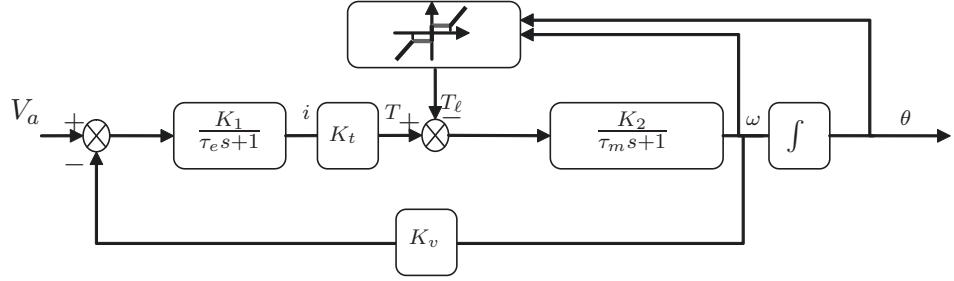


Figure 4.13: A block diagram for the overall dynamical system

system with time-delay can be given by:

$$H(s) = \frac{K_p e^{-\theta_p s}}{\tau_p s + 1} \quad (4.16)$$

However, this response represents the gearbox angle position obtained by applying a fixed torque, whereas the actual objective consists of defining the torque load brought by backlash. Therefore, the model in Eq. 4.16 can be oversimplified to a dead-zone model. In Fig. 4.12, the backlash model is associated to the friction model F_{c1} and F_{c2} in order to form a single model. The overall dynamic model of the system; actuator, tendon-sheath transmission system can be presented by the block diagram depicted in Fig. 4.13, where τ_e and τ_m represent the time constants of the electrical and mechanical parts, respectively. This structure is valid only under the assumption that the value of the inductance component of the DC-motor is small enough. In addition to these nonlinear effects, the load torque T_ℓ can be also affected during the interaction of the finger with the environment by the weight of the manipulated object or by applying an external force. In order to validate the different modeled parts of the whole system, a simulation application has been performed in an open loop. All parameters and rating of the motor driver are reported in Table. 4.2. The responses the whole system

4. MODELING AND HYBRID CONTROL

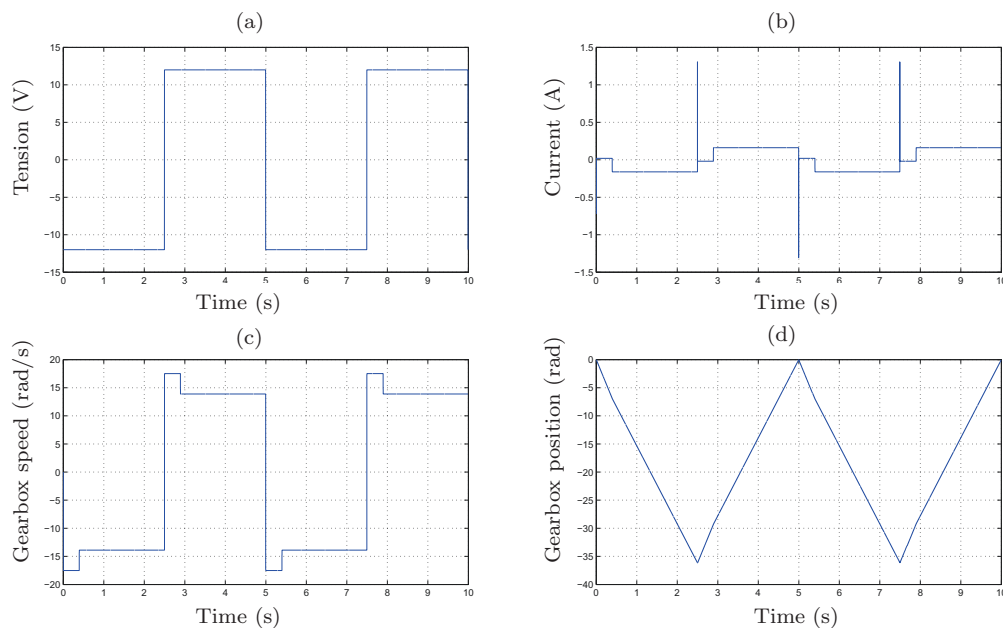


Figure 4.14: Simulation responses of the system in the open loop

without external interaction in open loop simulation are given in Fig. 4.14.b,c,d. Here, the input signal (tension supply) is a square signal (see Fig. 4.14.a). At each change in the input, the speed response starts with high value (backlash phenomenon) and afterwards it decreases and becomes stable. This is due to the fact that the motor turns in unload system. In fact, despite of the speed values seem constant, in reality is not the case, because these values depend essentially on the finger posture. For example, the friction effect generated by the last finger joint when the finger is stretched is not the same when one of the preceded joint is bent. Taken into account all these potentially nonlinear behaviors, the control of the position and the force should be robust vis-à-vis the friction and should have the ability to react as fast as possible given that the system including a time-delay due to the backlash.

4.2 Hybrid Control

The combination of position and force control into a single topic of control scheme is termed a *hybrid position/force control* and currently is reduced to a *hybrid control*. The problem has been firstly addressed for robot manipulators by Raibert and Craig in 1979 (128). On the basis of the same control scheme of the Raibert-Craig method,

Parameter	Value	Unit	Definition
R_a	14.1	Ω	Terminal Resistance (or Armature Resistance)
L_a	0.485	mH	Terminal Inductance (or Armature Inductance)
K_v	$10.4 * 10^{-3}$	V.s/rad	Speed Constant
K_t	$10.4 * 10^{-3}$	Nm/A	Torque Constant
τ_m	6.91	ms	Time Constant
J_M	0.541	gcm^2	Rotor Inertia
J_G	0.015	gcm^2	Gear-box Inertia
N	67		Gear ratio
η	75	%	Gearbox Efficiency
F_g	($\simeq 0$)	mN	Internal Damping Factor of Gear-box

Table 4.2: Parameters value of both motor and gear-box

Zhan and Paul (129) proposed a modified hybrid control in which the control defined in the cartesian space has been transformed to the joint space. In both cases, the control of position and force are achieved separately and afterward both controllers are combined to lead to a single torque control. Using the same control scheme, William and Mujtaba have proposed a correction into the position formulation of the hybrid position/force control scheme due to the error that may take place by using the inverse of Jacobian matrix (130). The hybrid control is also used in an interaction application between many manipulator robots to manipulate an object (131, 132). In such application, the hybrid control depends not only on the dynamic models of the robots, but also on the manipulated object. Recently, these results are extended to cover robotic hand area in which multifingered robotic hand has the same role as parallel manipulator robots (133, 134, 135, 136, 137, 138). The strategies of the hybrid control can be divided in two types, the direct and the indirect control (137). In the indirect case, the force control is implicitly controlled such as: the stiffness control (139), the impedance control (135), and the compliance control (140). In these strategies, the environment should be dynamically modeled and injected in the control loop. However, in the direct hybrid control, the force is explicitly controlled, independently of the position control. Once again, the direct hybrid control can be structured in two ways, the cascade structure (137) (see Fig. 4.15.a) where the force precedes the position control and the parallel structure (128, 130) (see Fig. 4.15.b). The parallel and the cascade structures have many common advantages. However, the parallel structure of control is valid only when the torque data is accessible, whereas, when the force sensor at fingertip is available,

4. MODELING AND HYBRID CONTROL

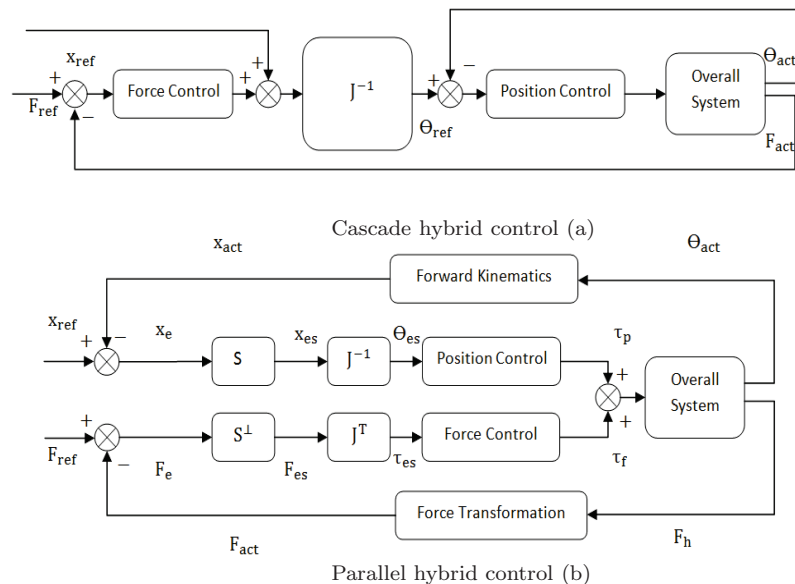


Figure 4.15: Direct hybrid control in its two structures

the cascade structure can be truly useful using only the position data of the joint. Due to the numerous advantages of the cascade structure over all presented approaches, it will be adopted to achieve the hybrid control. Among these advantages:

- The structure does not require any transformation matrix.
- The ability to deal with unmodeled environments.
- Very useful for multi-finger cooperation.
- A good performance even if an imperfect model of the finger is used.

One of the work aims of the Handle Project is to develop an efficient low level control. In order to do that, many partners of this project were involved in this aim, where the university of Carlos III of Madrid (UC3M) was chosen to be the responsible for this task. In addition to the UC3M, Université Pierre et Marie Curie, Paris (UPMC), Commissariat d'Énergie Atomique (CEA), Kings College of London (KCL) and the Shadow Robot Company are also involved in this work. For this purpose, the CEA's partner has designed an electronic board to control three motors driving three joints of one fingered robotic hand prototype (see Fig. 4.16). The designed controller deals with

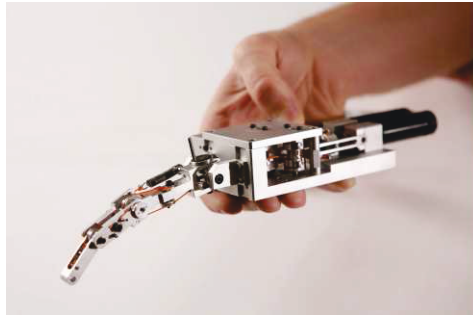


Figure 4.16: Prototype of CEA's fingered robotic hand

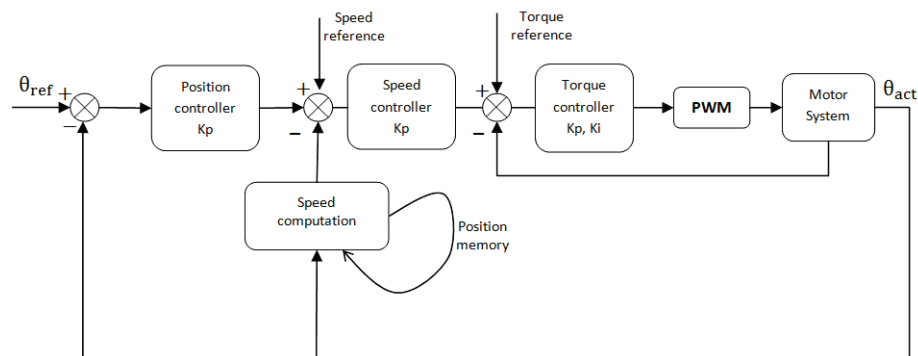


Figure 4.17: CEA's block diagram of the control loop cascade

the hybrid control of position/speed/torque. To this end, a proportional controller has been proposed to control the position and the speed, whereas, for the torque control, a PI controller has been adopted. As depicted in Fig. 4.17, three controllers are arranged in cascade structure forming inner and outer control loops. However, despite the control scheme comprises good results (141), it was out of use in Shadow hand control, due to the gap in the design between both systems. It is noteworthy to mention that the CEA's prototype has a negligible friction and backlash effects.

On the other hand, the Shadow hand company has already designed a hybrid position/speed/force (or torque) controller implemented in the electronic board of each motor. An illustrative block diagram that explains this hybrid control structure is given by Fig. 4.18. The controllers are based on a simple PID. Also, it can be shown that the force control is running in the inner control loop of the outer position/speed con-

4. MODELING AND HYBRID CONTROL

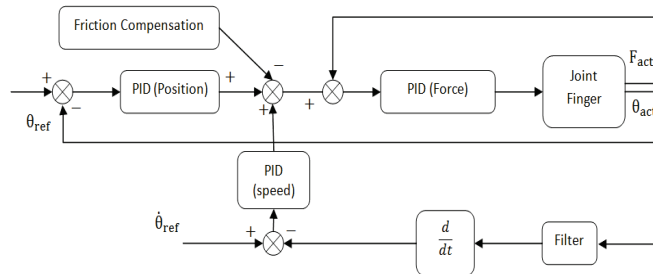


Figure 4.18: Shadow diagram block control

control loop. Since the system actuator-tendon-sheath is strongly subjected to the friction effect, the controller is endowed with friction compensation to remove the nonlinearity, which justifies the use of the linear PID controllers. However, the compensated friction has been identified when the hand is working in free motion without interactions. In the opposite case, the friction should be re-identified again, since the weight of the manipulated object, as well as the external forces can significantly perturb the identified friction model. Despite a high performance of the position/speed controller, so far, the PID controller of the torque is not yet used. Instead, an upper bounding of the current of the motor has been used to limit the generated torque. Under this condition, the torque can not take values far beyond a maximum torque equivalent to the upper-bounded current. This is due at least to the following two reasons. First, this controller was really suffering from the inaccuracy of the torque sensor, in which the measure tends to drift due, basically, to the temperature fluctuations and high exerted forces. Second, the imperfection of the measured data through the way to which the tendons exert on the strain gauges. To be more clear, the two tendons exert on the same shared beam connected to the strain gauges, which involves an overlapped measurement. This problem has been recently solved by making one tendon to be on the loose, in such a way that the other is exerting on the beam. Hence, the torque sensor reading ensures a measure more reliable. It should be noted that even if the torque control at the motor level exhibits a significant effectiveness, this value does not reflect the real torque applied at each joint fingers. This is due to the location of the strain gauges sensor which are placed just after the gearbox of the motor (a better reading of the torque measure is at the joint itself). The other reason is that the torque control structure does not take into account the friction effect. Nevertheless, this controller

could be employed to prevent any damage of tendons. Despite everything that has been said so far, it is possible to use this sensor as a mean to control high force values with respect to the friction effects.

In order to overcome these critical situations and all nonlinearities of the system, a new high-performance structure of control based on the nonlinear discontinuous controller, the *Sliding Mode Control* (SMC) has been developed. From a structural point of view, the novelty of this controller consists of achieving a hybrid position/speed/force (and torque) control under the current control of the motor. Moreover, the proposed structure of the hybrid control combines between SMC and PID controller to fulfill, on one hand, the position/speed/current control and on the other hand, the force (and torque) control, respectively. In the position/speed/current control, a succession of two versions of the SMC are used, in which the position/speed and the current control use different versions. Before to validate the efficiency and the robustness of the overall hybrid controller, the speed/current and afterward the position/speed/current control will be simulated and experimented. Finally, the hybrid position/speed/force (and torque) control under the current control will be experimentally evaluated using the prototype of one fingered Shadow hand.

4.3 Sliding Mode Control

The sliding mode control theory has been introduced for the first time by USSR's researchers in early 1950s, where Utkin and Emelyanov were among the founder of this control theory (142). This control approach, which is purely nonlinear, is dedicated to a wide spectrum of systems, especially for *Variable Structure Systems* (VSS) (143, 144), in which the system alternates between a set of continuous subsystems (called *structures*) using a proper switching process and hence the control laws are discontinuous functions (see Fig. 4.19). The outstanding feature of this controller is its robustness with regard to the parameters' uncertainties and the external disturbances. Excluding the VSS, the sliding mode control exhibits a high performance for controlling, nonlinear systems (144), multivariable systems (145), discrete-time systems (146) and large scale with high order systems. The sliding mode control is designed to enforce the state trajectory of the system, using a discontinuous control input, to swing around a manifold spanned by the variable state spaces which are expected to be controlled. Thus, the state trajectory is said to be in a *sliding mode*. This manifold is called *sliding surface* (or *sliding manifold*) and it is denoted by $S(x)$, where $x \in \mathbb{R}^n$ describes the properties of the desired dynamic plants, such as the stability and the tracking. In Fig. 4.20, an

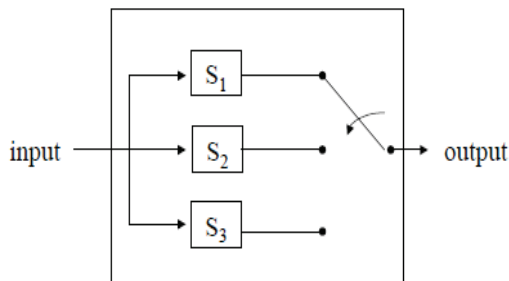


Figure 4.19: The representation of a controlled VSS

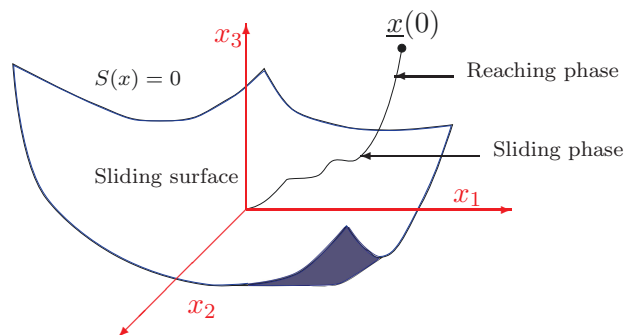


Figure 4.20: Sliding surface of three state spaces system

example of state trajectory of a system with three state spaces is sketched (147). As can be shown in this figure, the convergence of the state trajectory starting from the initial state $\underline{x}(0)$ away from the sliding surface, can be achieved passing by two phases; the reaching and the sliding phases. In the first phase, the variable state is driven by a feedback control that allows the convergence toward the sliding surface in finite time. Once the state trajectory reaches the sliding surface, the control switches at a high frequency to eliminate the deviation from the sliding surface. In real applications, the oscillation of the state trajectory caused by the switching of the control leads to an undesirable effect, called *shattering*, which could cause damages for dynamical systems. However, when dealing with electrical systems driven by switching circuit, such as the *Pulse-Width Modulation (PWM)*, this effect can be greatly reduced. One of the most advantages of the SMC is in its reduced-order plant dynamics, where instead of designing a controller which treats the full-order system, the SMC splits the problem in two low dimensional problems. The first one is used to build the sliding surface and the other is used to design the discontinuous control law. In addition to the reduced-order

advantage, and as previously mentioned, the SMC exhibits a high performance in the case of systems with changed parameters or in the face of external disturbances. This is due to the fact that the control law depends especially on the direction made between the actual position of the state and the sliding surface. It is known in the theory of sliding mode control that the robustness can be improved if the *matching condition* is accomplished. This condition implies that the control law should not be affected by the perturbations by choosing suitable control gains able to reject the upper bounds of these perturbations. Thus, the state trajectory tends to the sliding surface in finite time and remains on it regardless of all potential disturbances.

In the following subsection, the theoretical principle of the SMC is highlighted for nonlinear systems. Different steps leading to the stability of the equivalent control law and how this control can react in presence of potential disturbance in the system are presented as well.

4.3.1 Problem statement

Assume a nonlinear system presented by the state space equation of the form:

$$\dot{x} = f(x, t) + g(x, t)u \quad (4.17)$$

where $x(t) \in \mathbb{R}^n$, is the state vector and $u \in \mathbb{R}^m$ is the control vector. Furthermore, $f : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$ and $g : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^{n \times m}$ are assumed to be nonlinear functions that include the uncertainties of the dynamic system. Note that, the full-order system Eq. 4.17 can be re-written in the *regular form* forcing the control vector to appear only in the last equation and the rest of dynamic system takes a diagonal form. The diagonal nonlinear system of Eq. 4.17 consists of two subsystems with reduced-order:

$$\begin{cases} \dot{x}_1 &= f_1(x_1, x_2, t) \\ \dot{x}_2 &= f_2(x_1, x_2, t) + G_2(x_1, x_2, t)u \end{cases} \quad (4.18)$$

where $x_1(t) \in \mathbb{R}^{n-m}$, $x_2(t) \in \mathbb{R}^m$ and G_2 is a $m \times m$ nonsingular matrix. The nonsingularity of G_2 requires that the function vector $g(x, t)$ should be of full rank. The upper subsystem of Eq. 4.18 does not depend on the control. On the other part, the lower subsystem, which has the same dimension as the control law, is directly controlled by the input u . Hence, the upper subsystem of the state vector x_1 is “indirectly” controlled by x_2 . In the sliding mode control design, the state x_1 is used to define the sliding surface whereas the state x_2 is used to design the control law.

As regards the first phase of SMC designing, the state vector x_2 plays the role of the

4. MODELING AND HYBRID CONTROL

control input, in such a way the upper-subsystem should be asymptotically stable. On that ground, the new control law is defined as feedback control:

$$x_2 = \phi(x_1, t) \quad (4.19)$$

Then, the substitution of the control law Eq. 4.19 into the upper-subsystem of Eq. 4.18 leads to:

$$\dot{x}_1 = f_1(x_1, \phi(x_1, t), t) \quad (4.20)$$

which can be easily stabilized employing, for example, the *Lyapunov* approach. Now, let $x^* = (x_1^*, x_2^*)^T$ the desired state vector and $\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T = (x_1 - x_1^*, x_2 - x_2^*)^T$ the error signal between the desired and the actual state vector.

One of the most component constituting the SMC approach is the sliding mode surface defined by the following criterion:

$$S(t) = \{(x_1, x_2)^T | \sigma(x_1, x_2, t) = 0\} \quad (4.21)$$

where $\sigma(x, t)$ is the sliding surface defined by:

$$\sigma(x_1, x_2, t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} \tilde{x} \quad (4.22)$$

with $\lambda > 0$. It is clear that, on the sliding surface $S(t)$, tracking errors are governed by the following equation:

$$\left(\frac{d}{dt} + \lambda \right)^{n-1} \tilde{x} = 0 \quad (4.23)$$

This dynamic converges exponentially to zero with a rate depends on the value of λ . The asymptotic stability of the tracking errors can be reached in other way by pre-multiplying by a polynomial having all zeros in the left half plane:

$$\sigma(x_1, x_2, t) = p(s) \cdot \tilde{x} \quad (4.24)$$

where the variable “ s ” is the Laplace’s factor, then:

$$p(s) = s^{n-1} + a_{n-1}s^{n-2} + \dots + a_2s + a_1 \quad (4.25)$$

Once the sliding surface has been defined, the second phase of the SMC design consists of establishing the system input $u(t)$ so that the state vector x_2 of the lower-subsystem reaches $\phi(x_1, t)$ while maintaining the condition $\sigma(x_1, x_2, t) = 0$. In other words, the control law u has to drive any state trajectory, which is initially far away from the sliding surface, to converge towards this surface and to keep it there. Employing the

Lyapunov approach, the strategy of the convergence can be proved, by selecting as a Lyapunov candidate function, the square norm of the sliding surface:

$$V(x_1, x_2, t) = \frac{1}{2} \sigma^T \sigma \quad (4.26)$$

Differentiating of $V(x_1, x_2, t)$ respect to time gives:

$$\frac{dV(x_1, x_2, t)}{dt} = \dot{\sigma}^T \sigma \quad (4.27)$$

The development of the sliding surface given by Eq. 4.22 can be written as follows:

$$\sigma = \tilde{x}^{(n-1)} + \lambda \tilde{x}^{(n-2)} + \dots + \lambda^{n-1} \tilde{x} \quad (4.28)$$

Differentiating:

$$\dot{\sigma} = \tilde{x}^{(n)} + \lambda \tilde{x}^{(n-1)} + \dots + \lambda^{n-1} \dot{\tilde{x}} \quad (4.29)$$

where

$$\tilde{x}^{(n)} = \dot{\tilde{x}}_2 = \dot{x}_2^* - \dot{x}_2 \quad (4.30)$$

Substituting Eq. 4.30 in Eq. 4.29 using the lower-subsystem of Eq. 4.18 gives:

$$\begin{aligned} \dot{\sigma} &= f_2(x_1, x_2, t) + G_2(x_1, x_2, t)u - \dot{x}_2^* + \lambda \tilde{x}^{(n-1)} + \dots + \lambda^{n-1} \dot{\tilde{x}} \\ &= f_2(x_1, x_2, t) + G_2(x_1, x_2, t)u + h(x, \tilde{x}, t) \end{aligned} \quad (4.31)$$

It is known that, to achieve the asymptotic stability using the *Lyapunov* function, it is necessary to check that $\sigma^T \dot{\sigma} < 0$. This condition can be guaranteed during all subsequent time by controlling the sign of $\dot{\sigma}$ to have an opposite sign of σ by means of the control law u appearing in $\dot{\sigma}$. Finally, the control law should have the following form:

$$u = -u_0 \operatorname{sign}(\sigma) \quad (4.32)$$

where

$$u_0 > \| (G_2(x_1, x_2, t))^{-1} (f_2(x_1, x_2, t) + h(x, \tilde{x}, t)) \| \quad (4.33)$$

and

$$\operatorname{sign}(\sigma) = \begin{cases} 1 & \text{if } \sigma \geq 0 \\ -1 & \text{if } \sigma < 0 \end{cases} \quad (4.34)$$

The function $h(x, \tilde{x}, t)$ can also include all nonlinearities and perturbations. From the definition of the control law Eq. 4.32, it is clear to state that the control law depends only on the sliding surface that defines the tracking error of the states and the magnitude u_0 that can include the maximum variation of the different components of the system, even if these components are uncertainly defined.

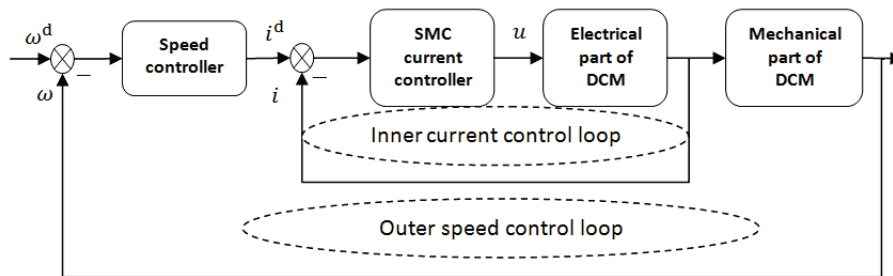


Figure 4.21: Block diagram of speed-current control

4.3.2 SMC design toward the hybrid controller

After having introduced the structure of the hybrid control which will be used in developing this work, the following objective is to implement the sliding mode control for the position and the speed. Of course, the controller will be implemented at the motor level where its success depends on the performance of the control of current signal of the motor. In this context, Utkin, in his book (148) has proposed a speed control of the DC-motor under the current control in accordance with the control scheme given by Fig. 4.21. The control of the current is structured as an inner control loop performing in an outer speed control loop. In this structure, the author reveals two topics dealing with the control loops. As for the inner current loop control, a SMC has been exploited. In order to reduce the shattering effect produced by the real value of the current, a reduced-order observer of this measure has been exploited to design the sliding surface. As for the outer speed control loop, the author has required that the input of the inner control loop should be continuously defined. For this purpose, the author used a simple PID to achieve the speed control.

On the basis of this latter structure of control, our position/speed control problem will be formalized in the same way. However, unlike the Utkin's work, the position/speed outer control loop is also based on SMC. Consequently, the control law of the inner control loop is purely a discontinuous signal. Whereas, through the simulation testing, the control of the current exhibits interruptions in which the infinity values of the current have been pointed out. This means that the inner current control loop can not be able to deal with the shattering effect resulted from the outer control loop. An alternative solution to overcome this problem is that the sampling period of the simulated model must be chosen very small value. Even if the simulation led to good performances,

the whole controller becomes very slow and can not be used to perform real-time applications. The problem has been counteracted by resorting to other strategy of control, where another version of the SMC has been used in the inner control loop to control the current. Thanks to this version, the current suffer less than the usual from the shattering effects. The condition matching of this version is given by (149):

$$\sigma \dot{\sigma} = -\xi \|\sigma\| - \gamma \sigma^2 \quad (4.35)$$

Then, the reaching rate to the sliding surface is defined as follows:

$$\dot{\sigma} = -\xi \operatorname{sgn}(\sigma) - \gamma \sigma \quad (4.36)$$

where ξ and γ are positive constants. This law forces the switching $\sigma(x)$ to vanish at time $T = \frac{1}{\gamma} \ln \frac{\xi|\sigma| + \gamma}{\gamma}$.

Therefore, on the basis of the above considerations, the following objectives consist of designing the hybrid control for the prototype finger, starting by simulating the speed control, and after the position/speed control using the first version of the SMC. Both control schemes are implicitly driven by the current, using the second version of the SMC. Afterwards, the position/speed control under the current control is experimentally evaluated. As for the hybrid control, two controllers are experimented; the position/speed/force control using an external force sensor placed at the fingertip and the position/speed/torque using once again the current signal.

4.3.3 Simulated Speed/Current Control

As it was shown in the modeling section, the overall dynamical model of the actuator-finger system can be split into two dynamics; the dynamic of the electrical part of the DC-motor and the dynamic of the mechanical part including all nonlinearities included in the load component T_ℓ . The overall dynamical model can be simplified by:

$$\begin{cases} \frac{di}{dt} &= a_1 u - a_2 i - a_3 \omega \\ \frac{d\omega}{dt} &= b_1 i - b_2 T_\ell \end{cases} \quad (4.37)$$

where $a_1 = \frac{1}{L_a}$, $a_2 = \frac{R_a}{L_a}$, $a_3 = \frac{K_v}{L_a}$, $b_1 = \frac{K_t}{J}$ and $b_2 = \frac{1}{J}$, in which the viscous friction of the motor is neglected with respect to the external frictions. As mentioned above, the overall control system of the speed control at the gearbox level is split into inner current control loop and outer speed control loop, where each loop has its proper sliding surface and reaching law (see Fig. 4.22). As for the outer speed control loop, it is the lower subsystem of Eq. 4.37 that governs the speed by means of the current signal

4. MODELING AND HYBRID CONTROL

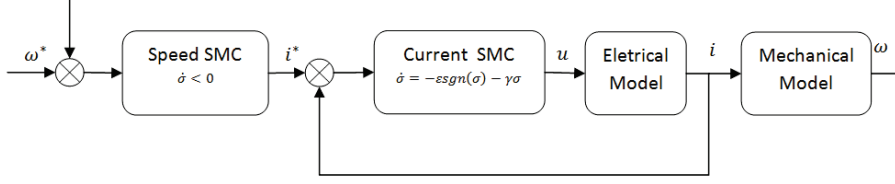


Figure 4.22: Bloc diagram of speed/current control using two versions of the SMC

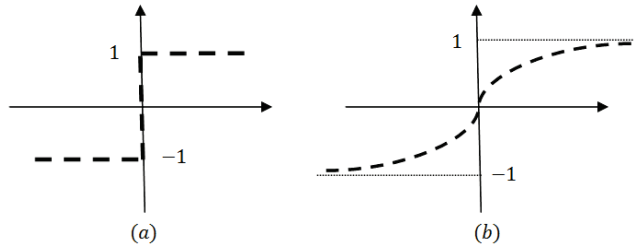


Figure 4.23: (a) *sign* function and (b) *tanh* function

considered, here, as the input. By comparing with the regular form, this part is already written as its lower subsystem form. As per the expression given by Eq. 4.22, where $n = 1$, then, the sliding surface for the tracking problem is given by:

$$\sigma = \omega^* - \omega \quad (4.38)$$

where ω^* is the desired reference speed. Then, by considering the first version of the SMC, the discontinuous control that leads to this surface is given by:

$$i = i_0 \text{sign}(\sigma) \quad (4.39)$$

where i_0 should fulfill the condition:

$$i_0 > \frac{1}{b_1} \|\dot{\omega}^* + b_2 T_\ell\| \quad (4.40)$$

This magnitude can be selected to take a high value when the state trajectory is far from σ while decreasing when it becomes too close. This can be done by adding a positive term depending on the value of σ :

$$i_0 = \frac{1}{b_1} (\|\dot{\omega}^* + b_2 T_\ell\| + c\|\sigma\|) \quad (4.41)$$

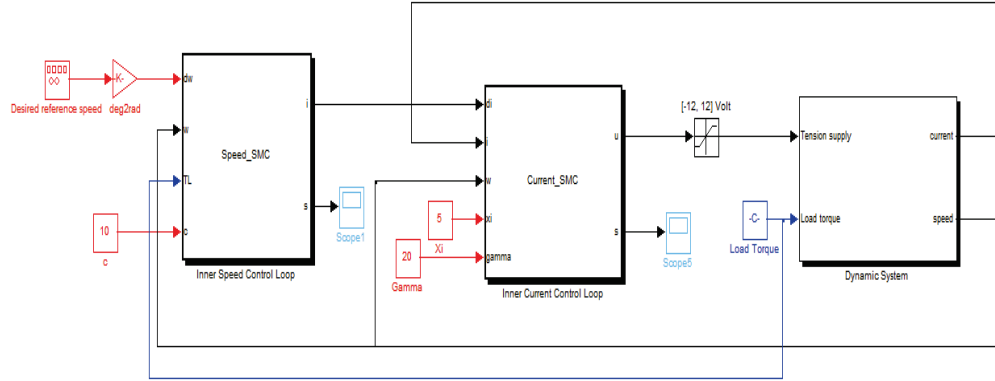


Figure 4.24: Simulink block diagram of Speed/Current control

where $c > 0$. Furthermore, in order to reduce the shattering effect, the function *sign* (see Fig. 4.23.a) has been changed by the function *hyperbolic tangent*, denoted by *tanh* (see Fig. 4.23.b).

Similar to the speed control, the sliding surface of the inner current control loop is given by:

$$\sigma = i^* - i \quad (4.42)$$

Now, i^* which is the desired reference current, it is also the output of the speed control. As mentioned above, the control law should fulfill the reaching rate given by Eq. 4.36. On the basis of this reaching law and the upper subsystem of Eq. 4.37, the input voltage of the motor that allows the control of the current is given by:

$$\frac{di^*}{dt} - (a_1 u - a_2 i - a_3 \omega) = -\xi \tanh(\sigma) - \gamma \sigma \quad (4.43)$$

4. MODELING AND HYBRID CONTROL

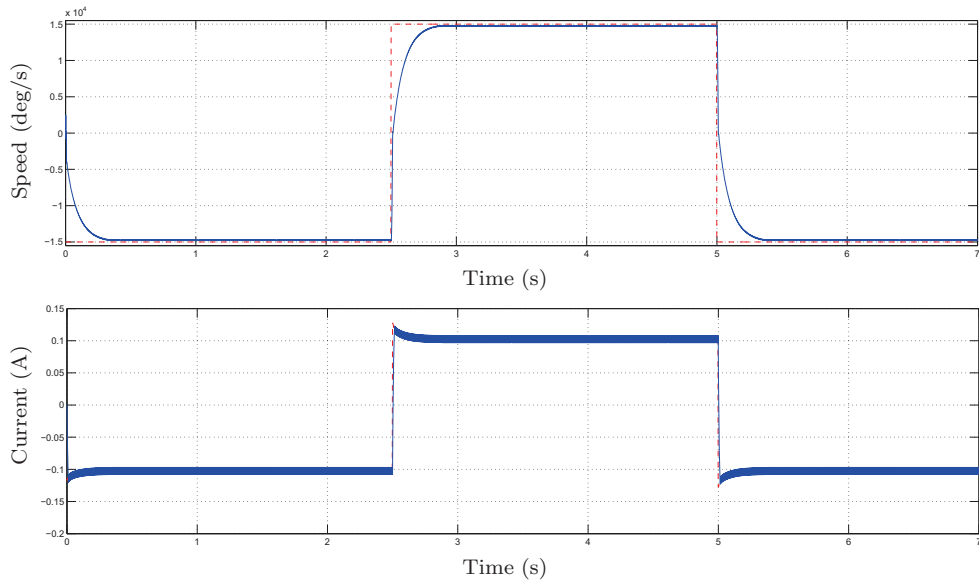


Figure 4.25: Speed/Current response for $\xi = 5$ and $\gamma = 100$. Upper figure: (--) reference speed, (—) actual speed. Lower figure: (--) reference current, (—) actual current.

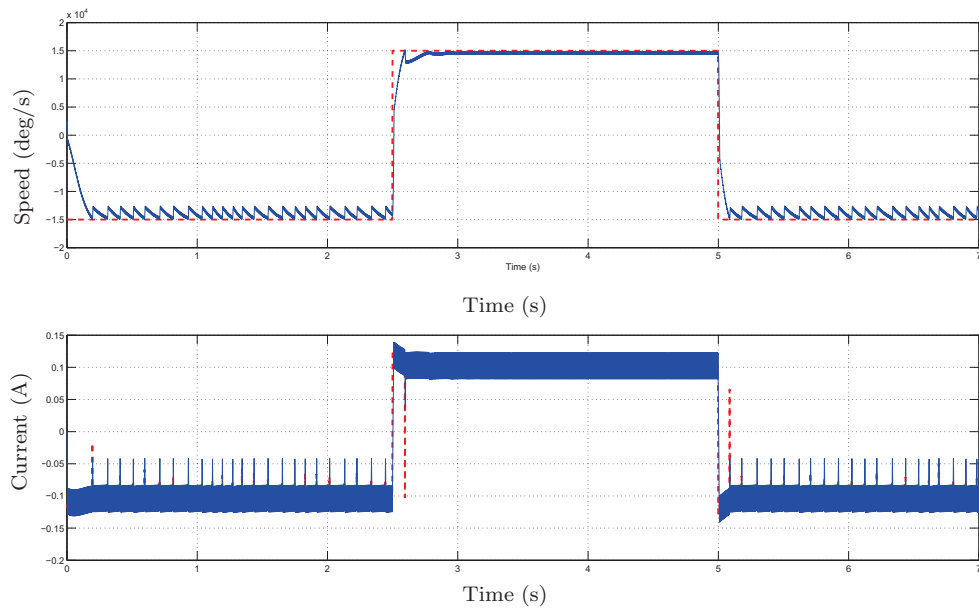


Figure 4.26: Speed/Current response for $\xi = 20$ and $\gamma = 20$. Upper figure: (--) reference speed, (—) actual speed. Lower figure: (--) reference current, (—) actual current.

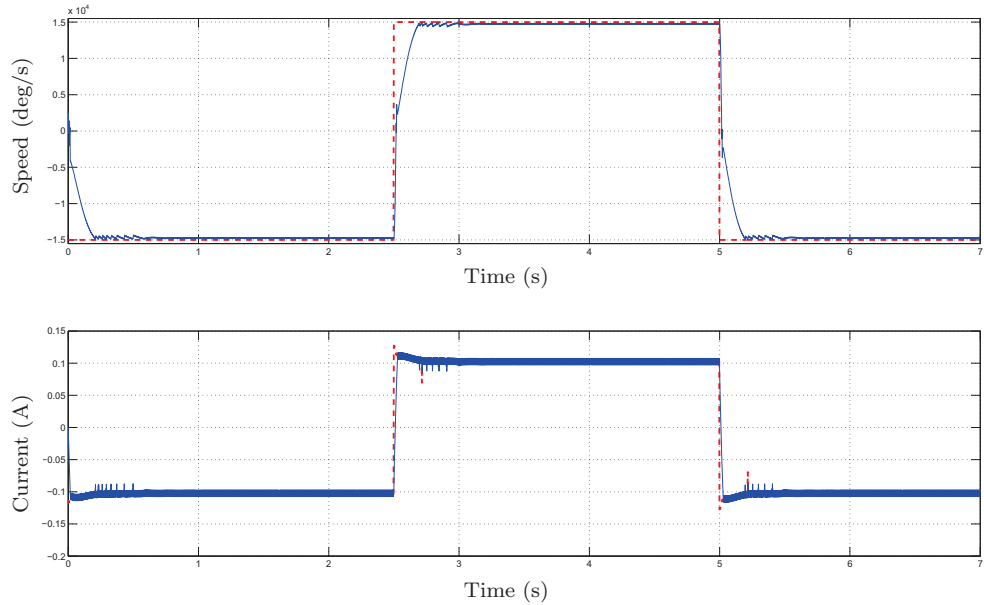


Figure 4.27: Speed/Current response for $\xi = 5$ and $\gamma = 20$. Upper figure: (---) reference speed, (—) actual speed. Lower figure: (---) reference current, (—) actual current.

which leads to

$$u = \frac{1}{a_1} (-\xi \text{sign}(\sigma) - \gamma \sigma) + a_2 i + a_3 \omega \quad (4.44)$$

The effectiveness of this control law depends essentially on the constants ξ and γ . Using the Simulink tool of Matlab, the proposed control algorithm has been simulated (see Fig. 4.24). In this case, the input voltage has been bounded to $\pm 10V$ to simulate the same permissible voltage supply of the experimental platform presented in the end of this section. The dependance of the controller performance on the parameters ξ and γ is shown in Fig. 4.25, Fig. 4.26 and Fig. 4.27, where the desired reference speed is alternated between two high opposite values. A high value of γ leads to quite smooth speed and less shattering effect, whereas the reaching rate has been relatively increased (see Fig. 4.25). Moreover, the speed and the current response exhibit a steady state error. On the other hand, the increasing of ξ leads to a considerable reduction in the reaching rate but strongest shattering effect has been arisen (see Fig. 4.26). After many trials, the better results are obtained for $\xi = 5$ and $\gamma = 20$, where the speed and the current responses reach their references in less time with less shattering effects and steady state error (see Fig. 4.27).

4. MODELING AND HYBRID CONTROL

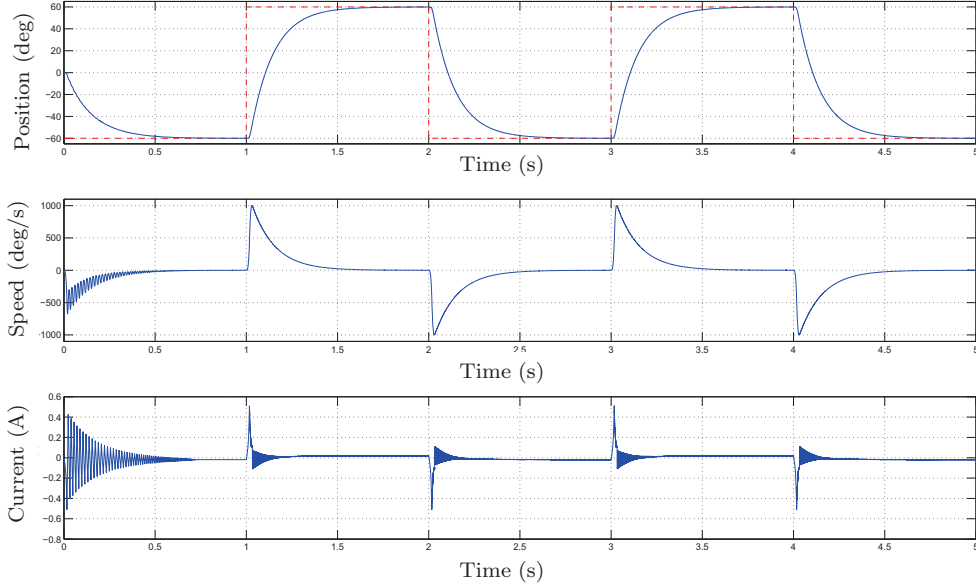


Figure 4.28: Position/Speed/Current response for $\lambda = 8$. Upper figure: (- -) reference position, (-) actual position. Middle figure: actual speed. Lower figure: actual current.

4.3.4 Simulated Position/Speed/Current Control

In fact, the speed/current control addressed in the above subsection was a step towards the whole objective in which the position/speed control under the current control is requested, relying on the parameter values found so far. In reality, in such dynamical system the position control plays a role more important than the speed control. The difference in the priority can be structured in form of the weighting during the design of the sliding surface, which is formulated by the position and the speed. As per the regular form given by Eq. 4.18 and the sliding surface designing given by Eq. 4.22, in which the order of σ is $n = 2$, then the corresponding sliding surface takes the following form:

$$\sigma = \tilde{x}_2 + \lambda \tilde{x}_1 \quad (4.45)$$

where $\tilde{x}_1 = \theta^* - \theta$ and $\tilde{x}_2 = \omega^* - \omega$ with θ^* and θ are the desired and the actual position of the motor, respectively. In such considerations, the mechanical part of the system is extended to have the following system of equations:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_2(x_1, x_2) + g(x_1, x_2)u \end{cases} \implies \begin{cases} \frac{d\theta}{dt} = \omega \\ \frac{d\omega}{dt} = b_2 T_\ell + b_1 i \end{cases} \quad (4.46)$$

4.3 Sliding Mode Control

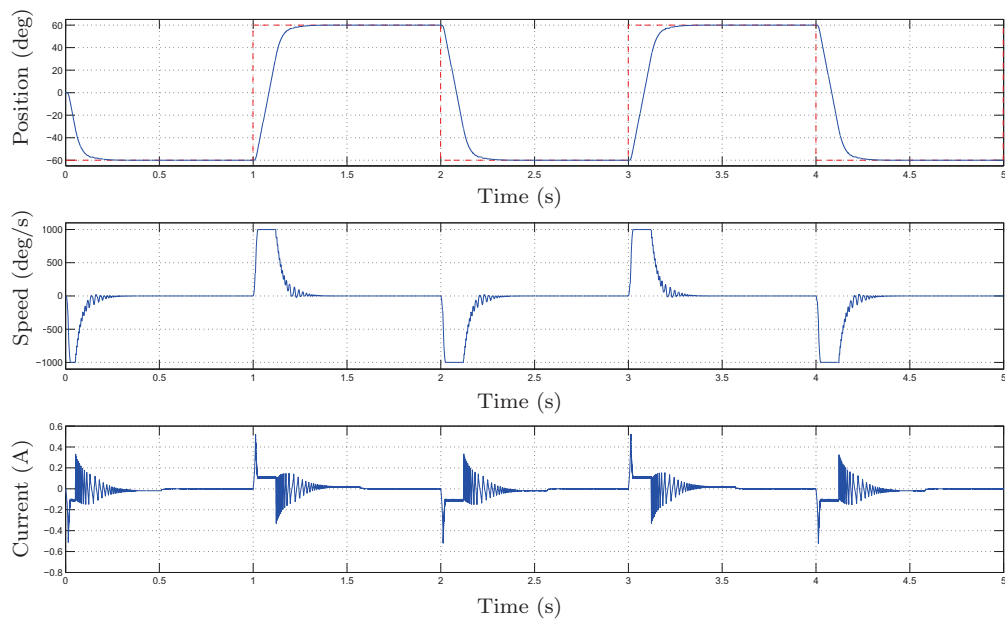


Figure 4.29: Position/Speed/Current response for $\lambda = 30$. Upper figure: (- -) reference position, (-) actual position. Middle figure: actual speed. Lower figure: actual current.

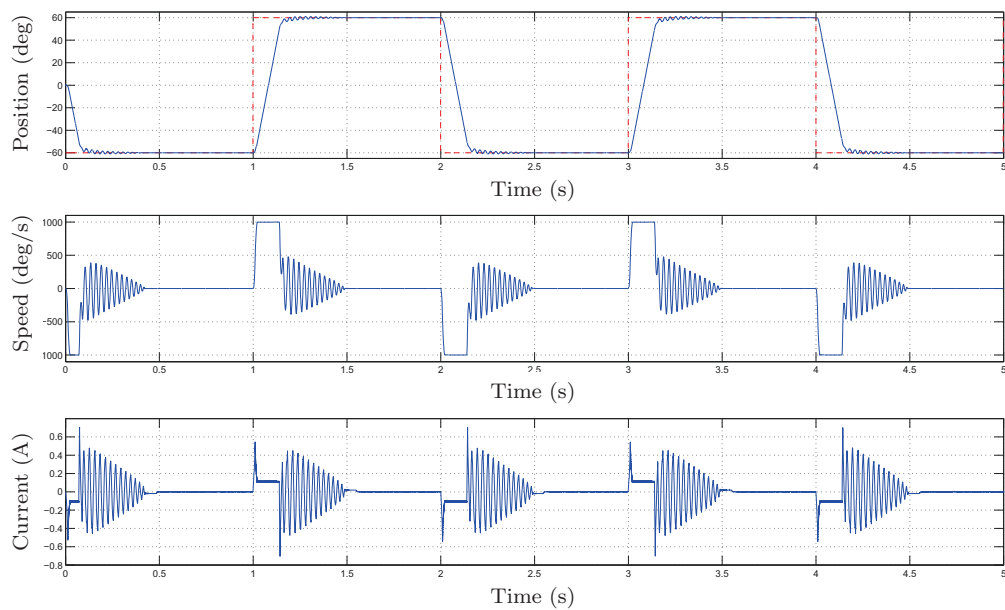


Figure 4.30: Position/Speed/Current response for $\lambda = 80$. Upper figure: (- -) reference position, (-) actual position. Middle figure: actual speed. Lower figure: actual current.

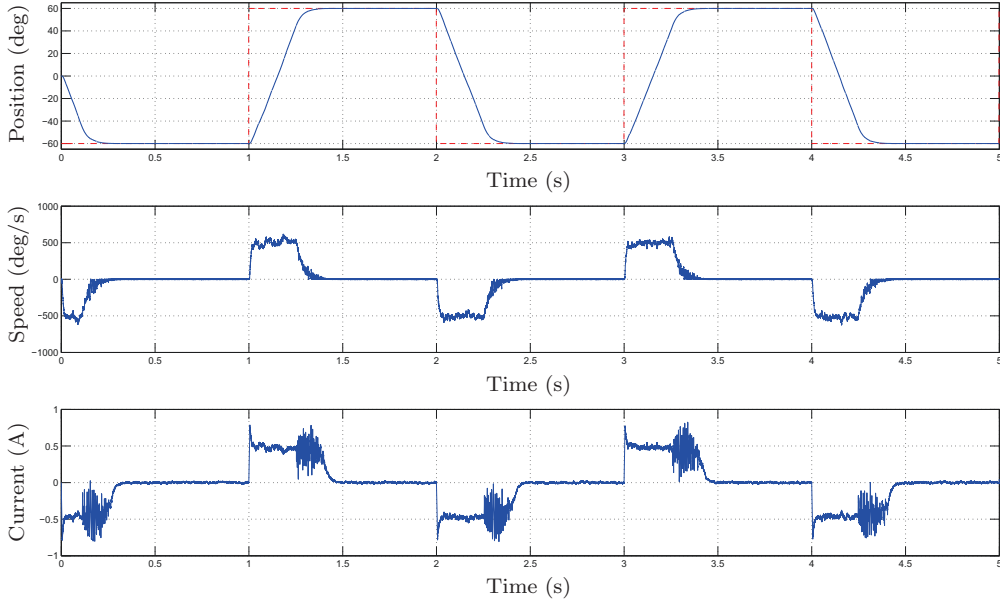


Figure 4.31: Robustness of the position/speed control using the SMC

As previously mentioned, the position/speed control scheme is the same for the speed control (see Fig. 4.22), in which the discontinuous control law driving towards the sliding surface is given by Eq. 4.39. Now, the magnitude i_0 should fulfill the following condition:

$$i_0 > \frac{1}{b_1} \|\dot{\omega}^* + \lambda\omega^* - \lambda\omega + b_2 T_\ell\| \quad (4.47)$$

Likewise, a weighted absolute value of σ has been added to the above magnitude making this last condition true for the subsequent time. Therefore, the discontinuous control law can be defined by:

$$i = \frac{1}{b_1} (\|\dot{\omega}^* + \lambda\omega^* - \lambda\omega + b_2 T_\ell\| + c\|\sigma\|) \cdot \tanh(\sigma) \quad (4.48)$$

As for the electrical part, the current control is the same as defined for the speed control relying on the same parameter values. Once the parameters of the current control are fixed, the accuracy of the results becomes strictly dependent on the parameter λ . The figures; Fig. 4.28, Fig. 4.29 and Fig. 4.30 show that for the small values of λ , the actual position reaches the reference in relatively long time, but smooth trajectories are generated. However, when λ is larger, the reaching time is small and leads to high frequency chattering. The suitable value $\lambda = 30$ gives a good trade-off between the reaching rate and the chattering effect.

In order to demonstrate the robustness of the developed control, the amplitude of the

disturbance load torque has been arbitrary changed to simulate the potential increasing in the load torque. The maximum value of this disturbance is selected to reach 500% of the maximum value of the resistive load dominated by the friction effect. The obtained results show that the overall performance are extremely high in terms of robustness, accuracy and time response in spite of large perturbations applied on the system (see Fig. 4.31).

As for the simulation of the hybrid control, it was difficult to simulate the force (or the torque) due to the lack of the simulated interaction data. However, this control can be successfully realized in the real applications since the interaction can be detected, using the force measurement devices or thanks to the torque measurement equivalent to the current of the motor.

4.4 Experimental Results

In order to validate the simulation results and also to include the force (and the torque) control with the position/speed control to function together, some experiments have been conducted. It is shown that, in the cascade structure of the hybrid control, the motor is driven by a discontinuous control law. This means that, it is important to interface the control law to the motor by a hardware device which has to be well adapted to the this type of control law. The following subsection discusses one of the efficient tools that allows not only the above adaptation, but also a potential interface between the low and the high power systems.

4.4.1 Sliding Mode Control using Pulse-Width-Modulation

Pulse Width Modulation (PWM) converter has been widely used to drive the electrical motors that are numerically controlled. When the SMC is exploited, the PMW converter is considered as a suitable tool to interface this controller to the motors (148),(149). This is due to the close similarities between the SMC and PWM converter in terms of the working principle. In fact, if assuming that the magnitude of the discontinuous control law of the application of SMC given by the Eq. 4.32 has a maximum value $u_0 = U_0$, then:

$$u = \begin{cases} +U_0 & \text{if } \sigma > 0 \\ 0 & \text{if } \sigma = 0 \\ -U_0 & \text{if } \sigma < 0 \end{cases} \quad (4.49)$$

In DC-motor application, the control u will be the average of the input voltage applied at the motor level and U_0 is the fixed output tension supply. On the other hand, the

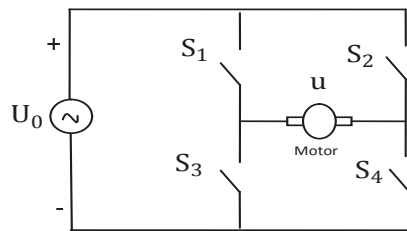


Figure 4.32: Motor and supply converter connection

above control can be depicted as the converter circuit given by the Fig. 4.32 which represents, in fact, the same concept of PWM in its bipolar switching formulation. Due to the shattering behaviors of the SMC, the electrical motor is exposed to the inertias of the mechanical part. This undesired effect can be treated by using a high switching frequency of the control law under, obviously, a suitable magnitude. Maintaining this requirement, the switching frequency depends also on the PWM frequency which should be significantly higher than the switching frequency of the control law (148). In the experiments, the PWM frequency is high enough $f_{PWM} = 40KHz$, whereas the switching frequency of the control is equal to $f_{smc} = 50Hz$. Note that, the PWM is also used by the Shadow company to control the motors. This was also one of the most reasons to use this tool for the SMC to be finally easy to implement in the firmware of the Shadow hand.

4.4.2 Setup Hardware

The experimental set-up of the hybrid control is schematically shown in the Fig. 4.33. There are three PCs that communicate with each other by means of the Internet network. The PC2 and PC3 are of use to provide the position data of each joint angle using a Shadow's equipment and the 6D force measure thanks to JR3 measurement device, respectively. Both PCs are connected to the main PC1 through TCP/IP and UDP communication protocol, respectively. The developed controllers are run in the main PC1 using the Simulink toolbox of Matlab version 2009.a. The control law is transmitted to the motor by means of the powerful measurement and control board Q4 integrated with Matlab/Simulink/RTW via Quanser's WinCon solution (150). The low level hardware interfaces between the electronic and electrical part is depicted in Fig. 4.33 by the electronic converter box. This part of system includes PWM converter and the Quansers WinCon electronic board. The DC-motor is the same as used in the

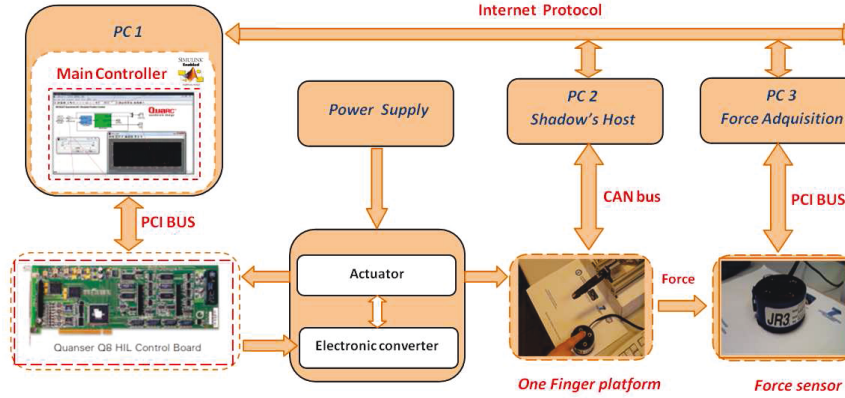


Figure 4.33: Experimental set-up used for hybrid control position/speed/force control validation

simulations. Moreover, the motor is endowed with an absolute encoder.

4.4.3 Position/Speed/Current Control

As mentioned earlier, the efficiency of the hybrid control depends intimately on the accuracy of the position control. However, the developed position control is performed at the motor level, whereas the “ideal” for this type of application is to control the position at the joint level. It is worth clearing that, the control at the motor level or at the joint level referring to the place of the measurement device that provides the position feedback information. This privilege is because of the backlash effect. In other words, performing the control at the joint level allows to omit this effect. Otherwise, if the controller is carried out at the motor level, which is the case in this work, the backlash should be precisely modeled. One of the main reasons why it is so difficult to perform the position control at the finger joint is due to the time-delay caused by the communication system of this set-up. Reading the joint angle from Matlab in the *PC2* and send it by means of TCP/IP protocol using Simulink tool of Matlab to the main program run in *PC1* requires about a hundred of milliseconds (some 400 ms) which is seriously a heavy time. It is well known that the delay can be reformulated by an exponential negative amount which can be approximated by a first order system with a zero in numerator:

$$G(s) = e^{-\tau s} \simeq \frac{1 - \tau/2s}{1 + \tau/2s} \quad (4.50)$$

where τ is the delay. Obviously, the new request control law should be able to control

4. MODELING AND HYBRID CONTROL

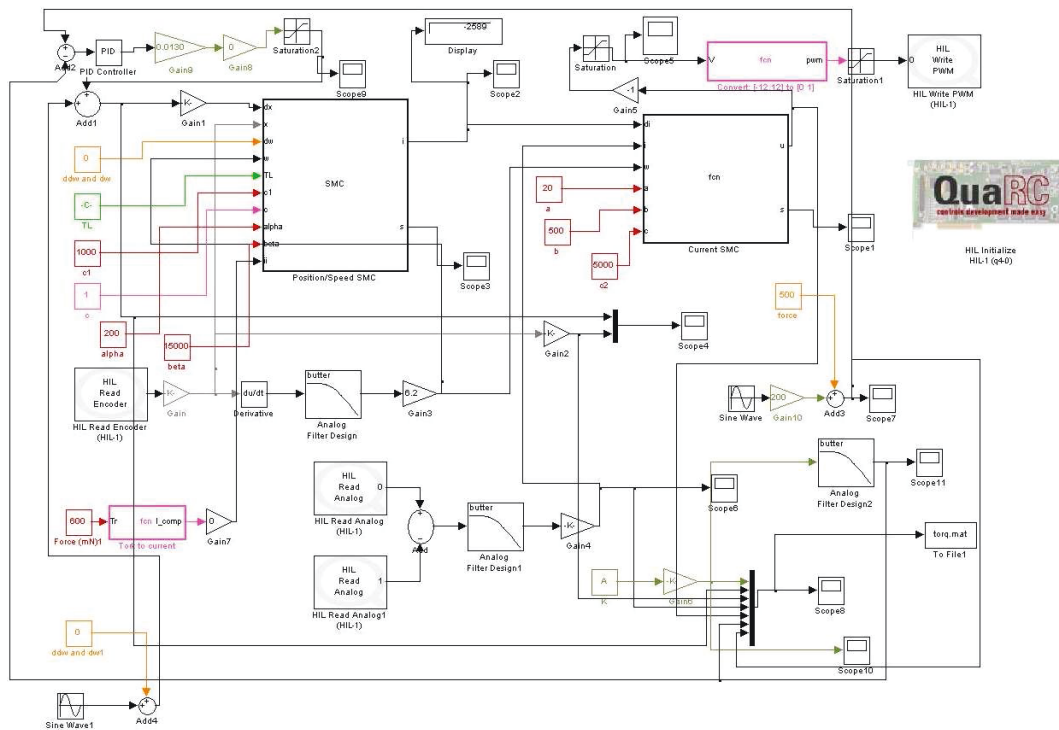


Figure 4.34: Simulink block diagram of position/speed/current control in realtime application

the new system modeled by the overall dynamic system given by Eq. 4.37 and the delay system Eq. 4.50. Designing a discontinuous control law based on SMC for the overall system leads to a derivative action of the control due to the zero effect in $G(s)$ which would result a high peaking phenomenon. Additionally, the experimental implementation of the SMC with delay leads to oscillating responses. Counteracting the time-delay system using the SMC has been of lively interest in the control field in both continuous and discontinuous systems (151, 152, 153, 154, 155). A similar topic has been treated in (151), with the aim of controlling a system by means of network channel. The proposed solution consists of applying a particular linear transformation that allows to reformulate a system with time-delay to another without time-delay. Chou and Cheng proposed an adaptive variable structure control to stabilize systems under the time-delay and perturbations by means of the Lyapunov theory (153). In aforementioned papers, the delay is assumed to be relatively small, which refers to the switching time provided in PWM converter.

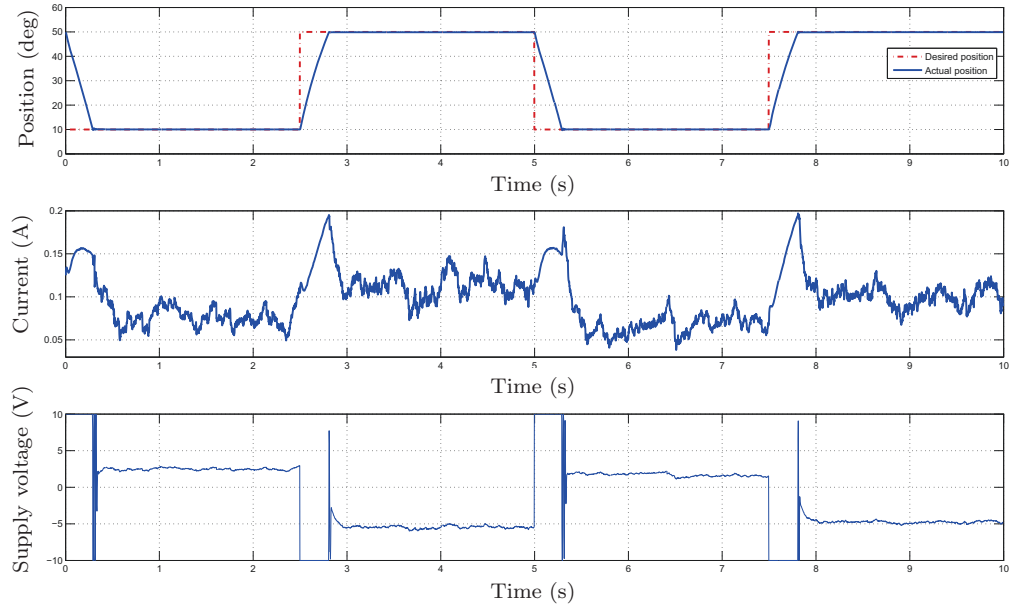


Figure 4.35: Response of position/speed/current control for realtime application with square reference position

The fact that the backlash has been well modeled before, the control at the motor and at the fingers joint levels lead to the similar results. The Simulink block diagram of the position/speed control under the current control is depicted in Fig. 4.34. The model runs on PC1 and generates the control law for the motor at each $20ms$, representing the sample time of the control law, which is, afterwards converted to PWM using the following relationship:

$$PWM = \frac{u + U_0}{2U_0} \quad \text{with } PWM \in [0, 1] \quad (4.51)$$

where $u \in [-U_0, +U_0]$ is the equivalent control law used by the inner current control loop given by Eq. 4.44 and U_0 is the maximum input voltage applied to the motor through the H-bridge concept (see Fig. 4.32). The input voltage U_0 is fixed to $10V$ due to the limitation of the electronic board $Q4$. This limitation can considerably affect the performance of the force control. The effects of this limitation is clearly shown through the Fig. 4.35 where the position, the current and the control law (supply voltage) are reported. Based on the perfect knowledge of the backlash and the position ratio between the motor and the joint angle, the results of the position trajectory tracking are equivalent to the joint finger, despite the feedback information is read from the encoder

4. MODELING AND HYBRID CONTROL

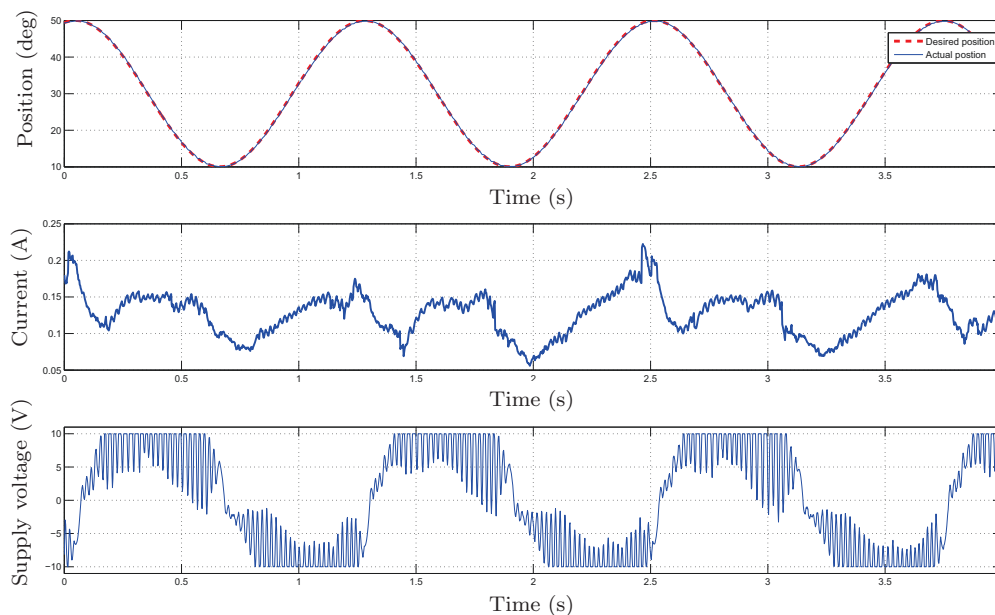


Figure 4.36: Response of position/speed/current control for realtime application with sine wave position reference

of the motor. It is also possible to see that the tracking error is small, but the system has a delay of almost $250ms$ to reach the desired position. The main reason of this delay is due to the limited input voltage. It is possible to see that during the reaching phase, the control law (voltage supply) stabilizes at the permissible value, which leads to a bounded speed. The purpose of applying a large step joint position variation is to demonstrate the performance of the controller. Whereas, in the robotic hand applications, the fingers are not expected to perform a large displacement. However, on the contrary, they often perform small displacements as generated, for example, by the inverse kinematics approach based on a numerical solution. In Fig. 4.36, the same width of the square wave is now generated in the form of sine wave chosen equal to $f = 0.8Hz$ which is a high frequency for such mechanical system. It can be clearly shown that the proposed controller reacts instantaneously with high accuracy. It has to be pointed out that for all obtained experimental results, the reference trajectory of the speed is chosen to equal to zero.

Now that the inner control loop based on SMC is defined, it is possible to point out that the efficiency of the force (and torque) control will depend certainly on its own controller, since the inner control loop leads to high performance.



Figure 4.37: Shadow hand equipped by 6D force sensors at the fingertips

4.4.4 Hybrid Position/Speed/Force Control

Despite the existence of the pressure sensor at the fingertips of the Shadow hand, its information is not yet used for the force control. One of the main reasons is due to the inaccuracy of this sensor. To overcome this problem, the UPMC's team proposed a hybrid control by using 6D-force sensors that take place of the distal phalanges of each finger (see Fig. 4.37). For such experimental setup, the team proposed two ways to control the force using the same control structure as used in the present work. Whereas, the difference is related to the inner position control loop. In the first experiment, the inner position control loop is the same controller developed by the Shadow company, i.e. the position is controlled by a PID controller. The corresponding experiment results are shown in Fig. 4.38 (156). The experiment consists of testing the force response when the reference takes high and small values. For high reference value (equal to 1 N), the response is relatively slow and very noisy. While, when the reference is small (equal to 0.6 N), the force response oscillates around the reference without reaching it. For both cases, the force is controlled by a simple integrator. The Fig. 4.39 reports the obtained force response by using an adaptive nonlinear controller in the inner position control loop. For two different values of the integrator gain of the force control, the expected experiment is to reach the same references defined in the above test. The controller, in this case, exhibits a significant decreasing in the time response but sometimes the response suffers from high oscillations. Most probably, the oscillations are related to the position controller which is insufficient to provide good performance to control the force. Thereby, the success of the hybrid control depends intrinsically on the efficiency of the position control.

The laboratory prototype of the one fingered hand is not endowed with the force sensor device. In order to realize the force control, the feedback signal is coming from an

4. MODELING AND HYBRID CONTROL

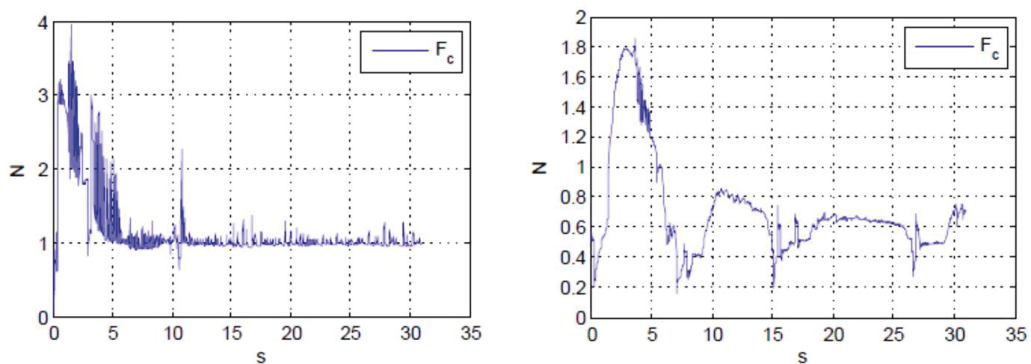


Figure 4.38: Experiment results of force control using the Shadow company position controller

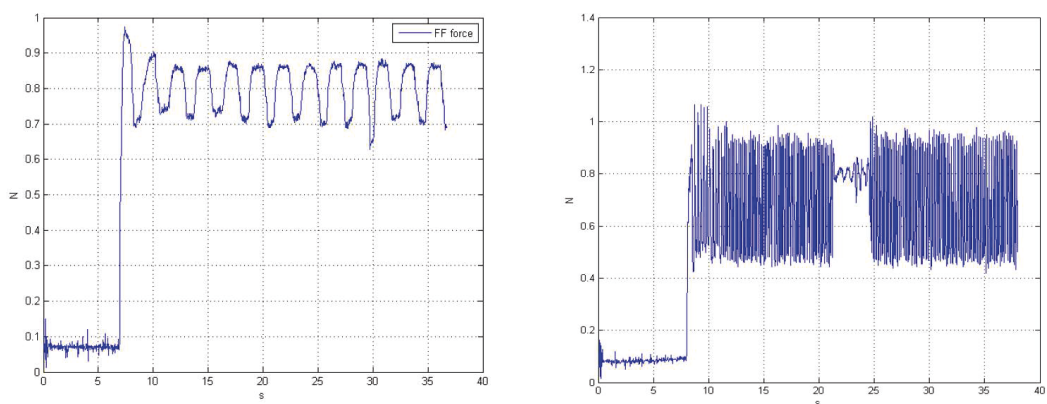


Figure 4.39: Experiment results of hybrid control using the UPMC's position controller

external force sensor (JR3) (see Fig. 4.33). Intuitively, when the desired position is reached and the finger becomes in contact with the environment, but the desired force is not reached yet, then the force control tries to add more displacement to the position reference until the desired force is reached. In the inner control loop, the position control attempts to reach this new reference by applying more torque at the motors, which leads to more force at the contact point. When the actual force is greater than the desired one, then the force controller requests to the position controller to loosen up by subtracting displacement from the position reference. Conceptually, a PID controller can be easily adapted to these requirements. Therefore, the proposed control structure of the hybrid position/speed/force control using the inner current control loop is depicted by the diagram given in Fig. 4.40. The force, the position

4. MODELING AND HYBRID CONTROL

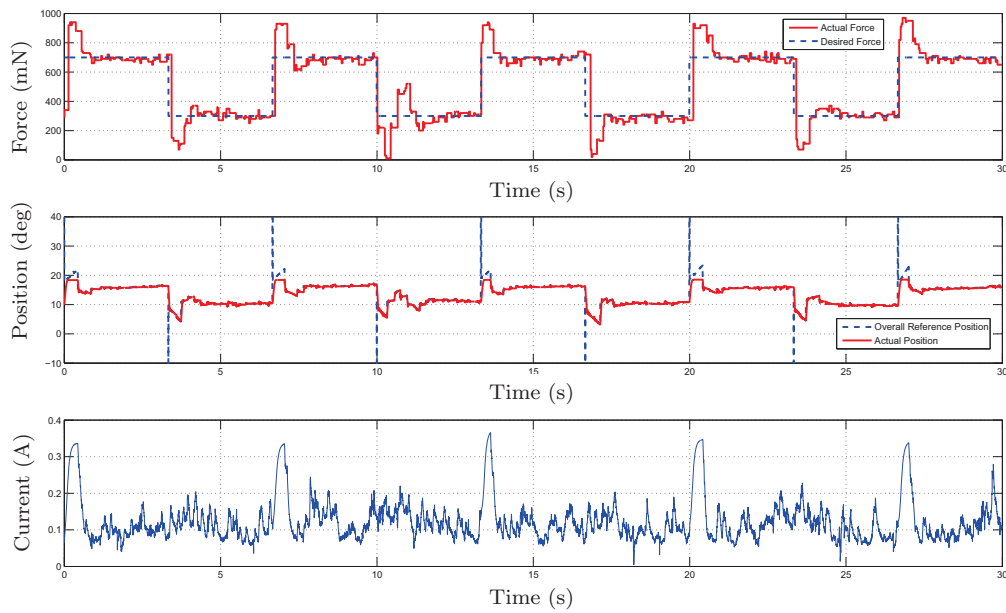


Figure 4.41: Hybrid position/speed/force control results for a square signal force reference. The PID force control parameters are $K_p = 1.5$, $K_i = 1.5$ and $K_d = 0.3$

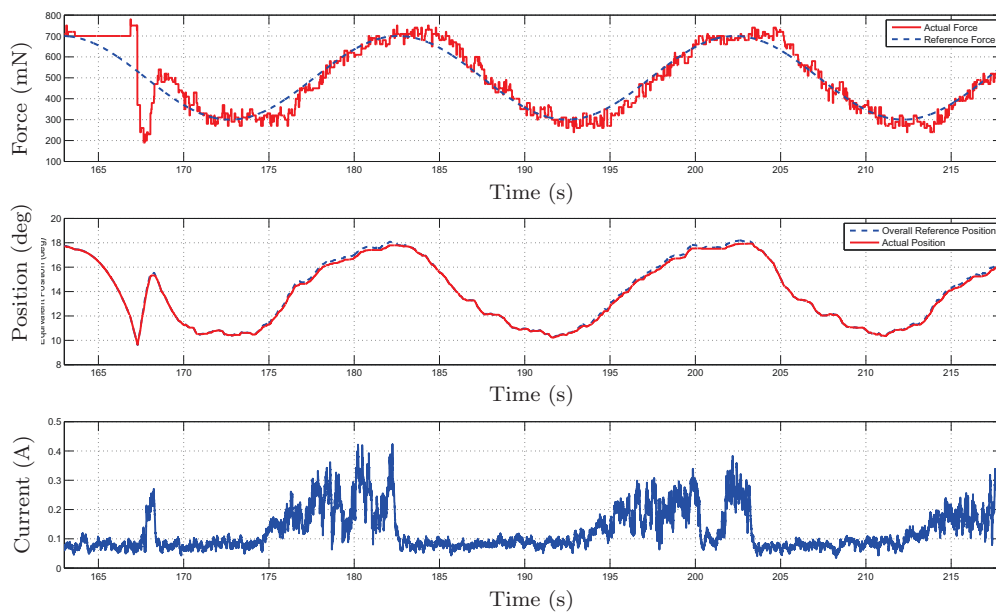


Figure 4.42: Hybrid position/speed/force results with PID force controller ($K_p = 1.5$, $K_i = 1.5$ and $K_d = 0.3$) for sine wave force reference

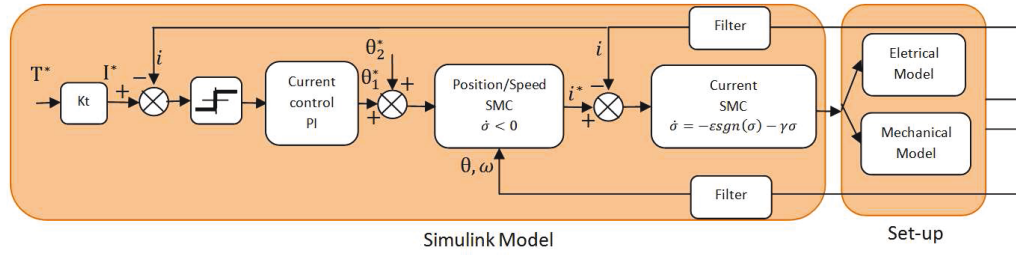


Figure 4.43: Hybrid Position/Speed/Torque Diagram

4.4.5 Hybrid Position/Speed/Torque Control

On the basis of the considered control structure, the hybrid control can be achieved by two methods. The exerted force can be directly controlled by using the force sensor or indirectly by controlling the motor torque. The latter alternative is useful when force sensor at the fingertip is not available. It is well known that the torque is proportionally related to the current of the motor. This relationship is an advantage that can be used to control the torque using the control current due to the unreliability of the torque sensor. The overall control structure of the hybrid position/speed/torque is shown in the Fig 4.43 where the torque is converted to the current thanks to the torque constant K_t . The complexity of this control structure resides in the fact that the current measure is implicitly employed in two loops, at the inner control loop to drive the position and also the speed, and at the outer control loop to drive the torque. Unlike the force control, the current has a fast dynamic, hence, it is not necessary to use a derivative action to be controlled. Therefore, a PI controller is more than enough. To illustrate the effectiveness of the proposed control, a square signal torque is injected as reference signal to be tracked assuming that the desired position is reached, i.e. the finger is in contact configuration. In Fig. 4.44, different responses of the current, the tension supply and the position are depicted. The figure shows that the current response tracks the reference in relatively small time, without overshoot and with small tracking errors. In position response, the reference generated by the torque controller can not be reached, which means that the position controller attempts to reach this reference beyond the permissible compliance. Once again, to prove the efficiency of this designed control structure, a sine wave reference of the torque is performed (see Fig. 4.45).

4. MODELING AND HYBRID CONTROL

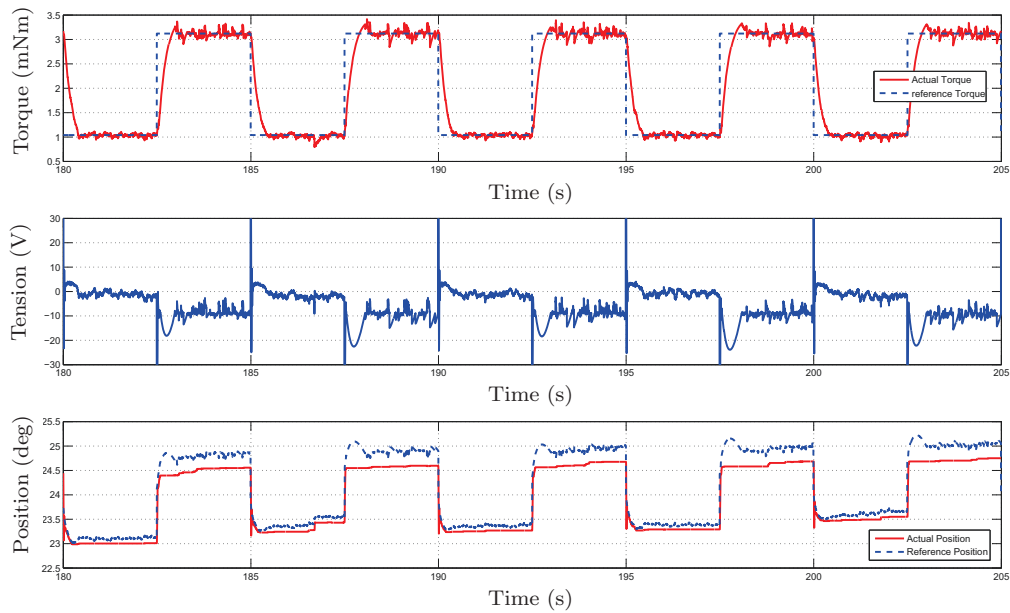


Figure 4.44: Hybrid position/speed/torque results with PI current controller ($K_p = 10$ and $K_i = 50$) for square wave torque reference

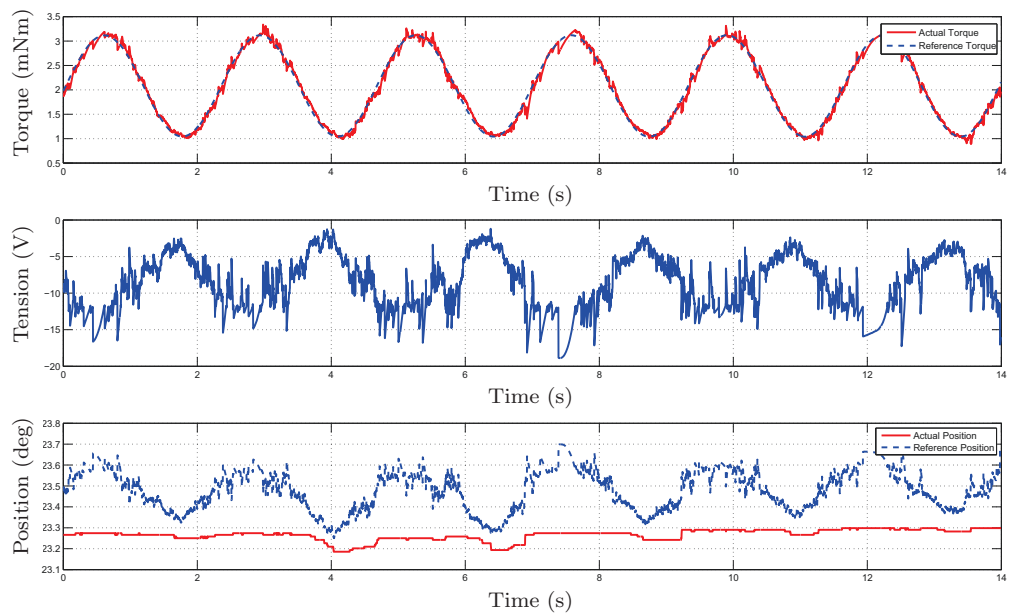


Figure 4.45: Hybrid position/speed/torque results with PI current controller ($K_p = 10$ and $K_i = 50$) for sine wave torque reference

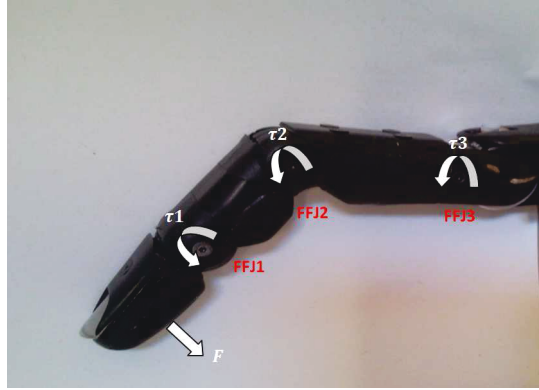


Figure 4.46: Index finger prototype of Shadow hand

From the mechanical point of view, the reference is chosen to have a high frequency. Nevertheless, the torque response demonstrates a tracking with high accuracy and no delay has been noted. For both tests, in order to avoid high peaks, a saturation in the current control action has been implemented. The usefulness of the saturation action can be particularly noted when the finger is performing in free motion. In this phase, the integration action of the PID controller provides a high control action, i.e. an important position is added to the reference, which could lead to a potential no-permitted speed. However, once an important increase in current is detected, i.e. the finger becomes in contact with the environment, the saturation action bandwidth should be extended for responses as fast as possible.

It is important to note that, both hybrid control structures are tested using one joint of the one fingered Shadow hand prototype, conventionally noted by FFJ3 (abbreviation of *First Finger 3rd Joint*) (see Fig. 4.46). Usually, in the robotic hand manipulation area, the position and force are often planned in the cartesian space and should be transformed to reference signals to be afterwards executed at the joint level. By using the concept of the Jacobian matrix, this transformation is feasible. It has been seen in Chapter 3, how the joint angles have been computed from the cartesian position increment of each fingertip by using the concatenated Jacobian matrix of the whole hand. Thus, it is always possible to split the whole Jacobian matrix to matrices, each one leads to the relationship between cartesian and joint spaces for each finger. Now, let the forward kinematics of the prototype finger:

$$\delta x_1 = J_{FFJ3} \delta \theta \quad (4.52)$$

where J_{FFJ3} is the Jacobian matrix of the prototype finger. Therefore, from a given

4. MODELING AND HYBRID CONTROL

force F , the desired torque at each joint finger is computed by the following relation:

$$\tau = J_{FFJ3}^T F \quad (4.53)$$

where $\tau = (\tau_1, \tau_2, \tau_3)^T$. Throughout all experimental tests of all proposed control structures, the prototype finger has been at a straight-line posture. However, by using the standard relation given by Eq. 4.53, all proposed control structures will be always valid for any posture of the fingers.

4.5 Conclusion

An innovative hybrid control for an actuation system based on a tendon-sheath mechanism of one fingered robotic hand prototype is proposed. The hybrid control involves position, speed and force (or torque) controllers and the adopted strategy to function together. Without any switching concept to alternate between the position/speed and the force control, the adopted strategy consists of performing the position/speed control in an inner control loop of the force control. In such structure, the efficiency of the force control depends not only on itself, but also on the position/speed control. Before the hybrid control was developed and because of the nonlinearity effects of the actuator system due ultimately to the tendon-sheath transmission modality, a dynamic model is proposed as well. Requirement on using an efficient inner control loop able to overcome the strong nonlinearity of the actuator prompts to use a robust nonlinear control. On the basis of the modeled system, the position/speed control under the current control are simulated using two different versions of the sliding mode control. The simulation results are quite satisfactory despite the fact that the control is sampled twice. Nevertheless, the simulations show that an additional improvement of the control performance could be introduced if the saturation in the control law was not considered. In the real case, this limitation is considered in accordance with the power supply of the real setup. Taking advantage of the simulation results, the hybrid control has been elaborated in two ways. Using an external force sensor, the experimental results show that the hybrid position/speed/force control leads to better results when comparing to those obtained by another work realized on the same actuator system. Note that, despite the accuracy of the force control, the response could be faster if all components of the actuator system are implemented in the same control computer. In the absence of the force measurement device, the hybrid control can be indirectly achieved by controlling the torque of the motor using the feedback of the current measure. It is shown that, the dynamics of such controller is faster than the force, the

reason for which the derivative action of the PID controller used for the force control, is not used for the torque case.

So far, the hybrid control whether based on the force or the torque feedback are validated for one joint of the one finger prototype. Whereas, since all joints finger are actuated in the similar manner, the proposed controller remains valid to cover all joints by adapting its parameters according to the friction and backlash factors that can be different. However, the validity excludes the coupled joints which are under-actuated systems. An interesting point for the future work will be the evaluation of the proposed controllers when all joints are involved to perform a cartesian task of the fingertip.

5

Conclusions and Future Work

5.1 Conclusions

In robotics, multi-fingered anthropomorphic hands have become more widespread in many disciplines by the virtue of their dexterity and multifunctional like grasping and object manipulation. For these kind of tasks, it is important to develop control strategies capable of overcoming all kinds of complexities and potential nonlinearities of the different parts of the hand. The present work has proposed new approaches for solving two important problems related to the inverse kinematics of the whole hand, and hybrid control to ensure better interaction. As for the first topic, the adopted approach is based on numerical solutions, where the inversion of the Jacobian matrix is required at each computation cycle. Despite the usefulness of this approach for a wide variety of robotic systems, the matrix risks to be ill-conditioned in some critical configurations. Different ways have been presented for overcoming such critical configuration, but the most useful is that which consists of damping small singular values that generate singularities.

Selectively Damped Least Squares, the method that relies on singular values computation has been adopted for solving the inverse kinematics of the whole hand. This provides an efficient damping factor for each singular value without affecting the rest. On the other hand, it leads to high computational cost since all singular values and their corresponding singular vectors have to be computed. The first purpose of this work was to reduce this cost by estimating only those singular values that generate the singularity. The proposed algorithm is inspired from an estimation algorithm which estimates only the smallest singular value. The various adaptations in designing the estimation algorithm are as follows:

5. CONCLUSIONS AND FUTURE WORK

- The new estimation algorithm estimates not only the smallest singular value, but also all singular values and their corresponding singular vectors.
- To be able to estimate any value of the singular values, the new estimation algorithm is repeated more than once. The number of times the algorithm is repeated depends on the dimension of the matrix.
- For a good estimation of the next singular value, it is necessary to place the estimated singular values far from those which are not estimated yet.

The efficiency of the proposed estimation algorithm requires that the matrix must have a square dimension. However, the redundancy of the whole robotic hand makes the Jacobian matrix a non-square matrix. Taking advantage of the redundancy to obtain a square matrix, the Jacobian matrix has been extended to deal with additional constraints. In this thesis, the joint limits avoidance is addressed as an additional task. On the basis of the estimation algorithm, the computation cost of the Extended SDLS method has been substantially reduced with respect to the original method. Both simulation results and experimental tests using the Shadow robotic hand confirm the efficiency of the reduced Extended SDLS method. To summarize, the relevant results obtained through the remedy of the inverse kinematics problem are as follows:

- Developing a new estimation algorithm of singular values and singular vectors which can be used in many areas outside the robotic discipline.
- In order to reduce the computation cost, the SDLS method has been changed in such a way that the overall solution is computed by combining the estimated singular values which should be damped and the Cholesky decomposition method.
- Extending the SDLS method to deal with additional constraints.
- The extended SDLS method allows simultaneous tracking tasks and the avoidance of singularities and joint limits as well.
- Compared to the original SDLS method, the proposed method leads to an important reduction in computation cost.
- A realtime implementation has been successfully experimented on the real anthropomorphic robotic hand of the Shadow Robot Company.

As for the second topic, some problems related to the modeling and the hybrid position/speed/force control of tendon actuated robotic hand have been addressed. This

actuation modality suffers mainly from friction of the tendon with sheath and backlash due to looseness of the tendon. Each phenomenon has been experimentally identified. The understanding of the dynamic model of the system helps in the design of hybrid control with better performance. For the hybrid control, a new structure of control is proposed in which the position/speed control and the force control form a cascade structure. Unlike the impedance control, the force is controlled by its own controller and does not need the contact point to be modeled. The position/speed control can be considered as the core of the force control since it performs in the inner control loop of the force control. For reasons of simplicity, the position/speed control has been addressed separably using the Sliding Mode Control. The effectiveness of the proposed hybrid control has been experimentally illustrated on a single-joint of one fingered robotic hand prototype. The most important results obtained from the research activities involved in this topic are:

- The development of new structure of control whose position/speed are controlled under the current control. Both controllers are also placed in cascade structure.
- Both controllers are based on the discontinuous nonlinear control; the Sliding Mode Control.
- The cascade structure makes the control law sampled twice which can lead to instability. To overcome this problem two different versions of the SMC have been used for each controller.
- The hybrid structure can control the finger during the free motion as well as when there is interaction, without switching process.
- Using the force sensor at the fingertip, the hybrid control has been successfully implemented, where force is controlled by means of a PID controller.
- In the case of the absence of force sensor, another alternative of the hybrid control has been proposed by controlling the torque. Here, the current has been used simultaneously as feedback for two control loops; the position/speed and the torque control.
- Since the current has fast dynamics, the torque control is achieved by a PI controller without derivative action.

5.2 Future Works

Future research activities can be addressed to further improve each topic individually, in such a way that, they can be integrated together to improve the grasp and the manipulation quality. In the following, some of the problems encountered during the development of this thesis, suggestions, and a possible linkage between the two topics are discussed.

Inverse kinematics: The problem has been solved by the proposed method; reduced ESDLS method. The basic principle that is used in the development of this method is the estimation algorithm of singular values. Recall that, the only reason this algorithm was developed was to reduce the computation cost. Note that the cost of the estimation algorithm is intrinsically related to the number of times that this algorithm would be repeated, which increases proportionally with respect to the dimension of the matrix. However, the value of this number and the convergence of the algorithm are evaluated empirically from simulation, and they remain to be mathematically proven.

Hybrid control: The obtained results can be further improved in terms of response time if all the components of the actuator system are executed in one machine, thus avoiding the time of communication between them. The one fingered Shadow robotic hand has a coupled joint. The hybrid control of the non-actuated joint is a complex problem when the force sensor is not available, in other words, when hybrid control is done under the torque control. Whereas, the problem can be solved when hybrid control is done using the force feedback at the fingertip.

Dexterity: Dextrous manipulation of an object requires a great trade-off between the posture of the hand, and its interaction with the object. So far, these requirements are defined by the inverse kinematics and the hybrid control, respectively, which are addressed in a separate manner, whereas, the dexterity requires that, both topics have to act interdependent. Fig. 5.1 suggests a schematic of how to structure the overall control that includes both topics is depicted. The reference force F_i of the hybrid control block controls whether the force or the torque is the equivalent effort that has to be applied at only one joint. As for the coupled joints, the hybrid control is a more complex task, since the coupling factor in the free motion is not the same as when the finger exerts a force on the object.

Hardware limitations: The hardware used for evaluating the inverse kinematics method, as well as the hybrid control, has some limitations principally caused

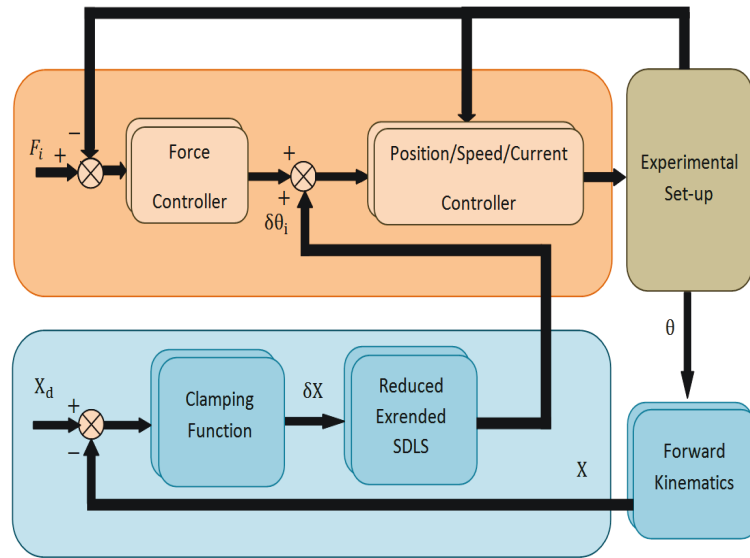


Figure 5.1: Inverse kinematics approach with low level hybrid control

by the communication rate between the different machines. It would be much better if all programs and algorithms are executed on one machine which communicates directly with the firmware of the Shadow hand, in which the low level hybrid control could be implemented.

5.3 Publications

The main contributions of the work presented in this thesis has either been published in various peer-reviewed articles, or is being prepared for submission for publication in an international journal. The following list summarizes all the publications accomplished during the period of working towards this thesis.

- Choukri Bensalah, Javier Gonzalez-Quijano, Jianwei Zhang and Mohamed Abderrahim. "A New Extended SDLS to Deal with the JLA in the Inverse Kinematics of an Anthropomorphic Robotic Hand". ROBOT2013: First Iberian Robotics Conference. Advances in Intelligent Systems and Computing. pp 661-673. 2013.
- C.Bensalah, J.G. Quijano, N. Hendrich M.Abderrahim: Anthropomorphic Robotics hand inverse kinematics using estimated SVD in an extended SDLS approach. IEEE, International Conference on Advanced Robotics (ICAR), Nov, 2013.

5. CONCLUSIONS AND FUTURE WORK

- C.Bensalah; J.G.Quijano; M.Abderrahim. Selectively Damped Least Squares with Joint Limits Avoidance as Inverse Kinematics approach for Shadow Hand. HANDLE Workshop Benicassim. Feb, 2012.
- C.Bensalah; M.Abderrahim; J.González-Gómez. A new finger inverse kinematics method for an anthropomorphic hand. 2011 IEEE International Conference on Robotics and Biomimetics (ROBIO), Dec, 2011.

During the period leading to the preparation of this thesis and through the participation in the HANDLE project, I collaborated in the work of my colleagues through discussions and the design of some algorithms. Although the work is not part of my thesis it produced some publications among which I cite following.

- J.G.Quijano; M.Abderrahim; C.Bensalah; S.Rodríguez. RoMPLA: An Efficient Robot Motion and Planning Learning Architecture. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Tokyo. Japan. Nov, 2013.
- J.G.Quijano; M.Abderrahim; C.Bensalah; A.Al-kaff. A human-based genetic algorithm applied to the problem of learning in-hand manipulation tasks. IROS 2012 workshop: Beyond Robot Grasping - Modern Approaches for Learning Dynamic Manipulation. Villamoura. Portugal. Oct, 2012.

Appendix A

Denavit-Hartenberg's parameter for the Shadow Hand

Denavit-Hartenberg's parameters of each finger are reported in tables. In the last column of these table, it is defined the margin of variation of each joint.

A.1 Index finger

Joint i	a_i (mm)	d_i (mm)	α_i [rad]	θ_i [rad]	θ_{l_i} [deg]	θ_{u_i} [deg]
1	20	0	$\pi/2$	θ_1	10	-30
2	80	33	$\pi/2$	θ_2	35	-45
3	0	0	$-\pi/2$	θ_3	25	-25
4	45	0	0	θ_4	90	-10
5	25	0	0	θ_5	90	0
6	24	0	0	θ_6	90	0

A.2 Middle finger

Joint i	a_i (mm)	d_i (mm)	α_i [rad]	θ_i [rad]	θ_{l_i} [deg]	θ_{u_i} [deg]
1	20	0	$\pi/2$	θ_1	10	-30
2	90	11	$\pi/2$	θ_2	35	-45
7	0	0	$-\pi/2$	θ_7	25	-25
8	45	0	0	θ_8	90	-10
9	25	0	0	θ_9	90	0
10	24	0	0	θ_{10}	90	0

A. DENAVIT-HARTENBERG'S PARAMETER FOR THE SHADOW HAND

A.3 Ring finger

Joint i	a_i (mm)	d_i (mm)	α_i [rad]	θ_i [rad]	θ_{l_i} [deg]	θ_{u_i} [deg]
1	20	0	$\pi/2$	θ_1	10	-30
2	80	-11	$\pi/2$	θ_2	35	-45
11	0	0	$-\pi/2$	θ_{11}	25	-25
12	45	0	0	θ_{12}	90	-10
13	25	0	0	θ_{13}	90	0
14	24	0	0	θ_{14}	90	0

A.4 Little finger

Joint i	a_i (mm)	d_i (mm)	α_i [rad]	θ_i [rad]	θ_{l_i} [deg]	θ_{u_i} [deg]
1	20	0	$-\pi/2$	θ_1	10	-30
2	0	-50.02	$-55\pi/180$	$\theta_2 - \pi/2$	35	-45
15	26.20	67.1027	$-\pi/2$	$\theta_{15} + \pi/2$	40	0
16	0	0	$-\pi/2$	$\theta_{16} + 55\pi/180$	25	-25
17	45	0	0	θ_{17}	90	-10
18	25	0	0	θ_{18}	90	0
19	24	0	0	θ_{19}	90	0

A.5 Thumb finger

Joint i	a_i (mm)	d_i (mm)	α_i [rad]	θ_i [rad]	θ_{l_i} [deg]	θ_{u_i} [deg]
1	20	0	$-\pi/2$	θ_1	10	-30
2	8.5	20.88	$-\pi/4$	$\theta_2 - \pi/2$	35	-45
20	0	41.35	$-\pi/2$	θ_{20}	60	-60
21	38	0	$\pi/2$	$\theta_{21} + \pi/2$	75	0
22	0	0	$-\pi/2$	θ_{22}	15	-15
23	32	0	$\pi/2$	θ_{23}	30	-30
24	25	0	0	θ_{24}	10	-90

Appendix B

Robots hands

In this appendix, a number of known robots hands is categorized in terms of number of fingers, number of DOF, sensorial system, actuation mechanism, algorithms of the force control as well as the position and other features.

B. ROBOTS HANDS

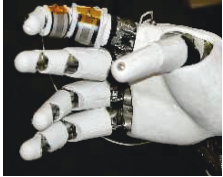

Name	Robonaut Hand (43)	Shadow C6 hand (26)
Picture	 A photograph of the Robonaut Hand, a white, five-fingered prosthetic hand with visible mechanical joints and actuators.	 A photograph of the Shadow C6 hand, a dark, five-fingered prosthetic hand with a more organic, skin-like appearance, holding a white object.
Center or company	NASA	Shadow robot company
Country	USA	UK
Year	1999	2011
Number of fingers	Five	Five
Number of joints	22 (20 hand+ 2 wrist)	24 (22 hand + 2 wrist)
DOF	14 (12 hand + 2 wrist)	20 (18 hand + 2 wrist)
Transmission types	lead-screw + flex-shaft	Tendons
Actuator types	Electrical revolute motors	DC motors (Pneumatic in the old version)
Location of actuators	Remote	Remote
Sensorial system	incremental encoders at each motor, tactile force sensor	Joint position sensors based on Hall effect, motor current, tendon force sensor and force sensor at fingertip, temperature sensor
Feedback control	force control, position control	impedance control; outer position control loop and inner force control loop

Table B.1: Robonaut and Shadow hands' features



Name	Actuated Sheffield Hand	EH1 Milano Hand (42)
Picture	 A photograph of the Actuated Sheffield Hand, a complex, multi-fingered prosthetic hand with a transparent, mechanical structure. It is mounted on a black base. A small caption below the image reads: "PROTOTYPE DEVELOPED FOR OREBRO UNIVERSITY BY ELMOTION LTD".	 A photograph of the EH1 Milano Hand, a white, sleek, and modern prosthetic hand. It has five fingers and is mounted on a white, cylindrical base.
Center or company	Sheffield Hallam University	Prensilia S.R.L
Country	UK	Italy
Year	2007	2009
Number of fingers	Five	Five
Number of joints	Not found	20
DOF	12	16
Transmission types	Tendons	Tendons
Actuator types	Brushed DC motors	Brushed DC motors
Location of actuators	Remote	Remote
Sensorial system	Not found	Force sensor, position sensor, motor current
Feedback control	Not found	Implemented control loops; position, current, force

Table B.2: Actuated Sheffield and EH1 Milano hands' features

B. ROBOTS HANDS

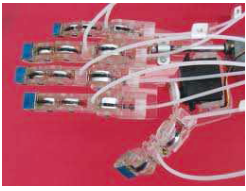
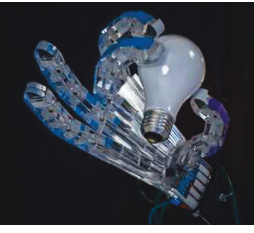
Name	Dexhand	Raphael hand (48)
Picture		
Center or company	SURABAYA UNIVERSITY	Robotics and Mechanisms Laboratory at Virginia Technology
Country	Indonesia	USA
Year	2006	2010
Number of fingers	Five	Five
Number of joints	16	Not indicated
DOF	11	6
Transmission types	Tendons	Corrugated Tubing
Actuator types	Brushed DC motors	Pneumatic
Location of actuators	Remote	Remote
Sensorial system	Joint position sensors (potentiometer)	Flex sensors for Position, Force Sensitive, Resistors for Force
Feedback control	Position control	Pressure control of the compressed air

Table B.3: Dexhand and Raphael hands' features

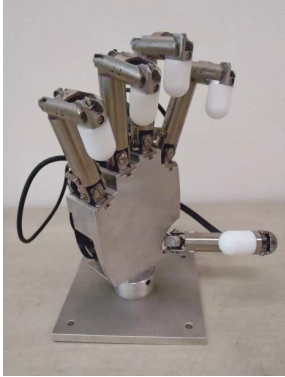

Name	Gifu hand (44)	Ultralight hand (45)
Picture		
Center or company	Gifu University	Inst. of applied computer science research center of Karlsruhe
Country	Japan	Germany
Year	2001	2001
Number of fingers	Five	Five
Number of joints	20	18
DOF	16	13 (10 hand + 3 wrist)
Transmission types	Gears	Fluidic actuator
Actuator types	Servomotors	Fluidic actuator
Location of actuators	At fingers' level	At fingers' level
Sensorial system	Motor position sensor, tactile sensors, force sensors at each fingertip	Flex sensor, touch sensor
Feedback control	Position control, force control	Position control, torque control

Table B.4: Gifu and Ultralight hands' features

B. ROBOTS HANDS


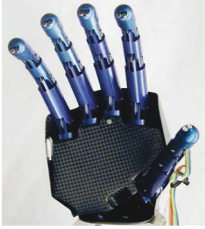
Name	TUAT/Karlsruhe Humanoid Hand (157)	RCH-1 hand (49)
Picture		
Center or company	Tokyo and Karlsruhe Universities	Scuola Superiore SantAnna Pontedera and Waseda University
Country	Japan	Italy and Japon
Year	2000	2003
Number of fingers	Five	Five
Number of joints	24	16
DOF	20	16
Transmission types	Tendons	Tendons
Actuator types	Electrical revolute motors	DC motors
Location of actuators	Remote	Remote
Sensorial system	Motor position sensor	Motor position sensor, 3D force sensor at fingertips, tactile sensors
Feedback control	Motor position control	Motor position control

Table B.5: Gifu and RCH-1 hands' features


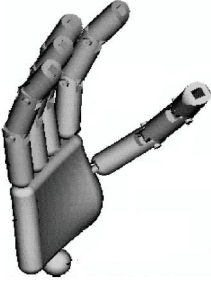
Name	LARA Hand	Rutger hand (49)
Picture		
Center or company	Technical University Darmstadt	Rutger University
Country	Germany	USA
Year	2007	1999-2002
Number of fingers	Five	Five
Number of joints	22 (20 hand + 2 wrist)	20
DOF	22	20
Transmission types	SMA actuator	SMA actuator
Actuator types	SMA actuator	SMA actuator
Location of actuators	Remote	Remote
Sensorial system	Inductive linear position encoder	Senor spots, joint position sensor based on Hall effect
Feedback control	Current control of SMA	Current control of SMA

Table B.6: Lara and Rutger hands' features

B. ROBOTS HANDS



Name	UB hand III (113)	Harada hand (158)
Picture	 A photograph of the UB hand III, a prosthetic hand with five green, segmented fingers. The hand is mounted on a white wrist-like base, which is connected to a complex assembly of metal rods and wires, likely actuators and sensors.	 A photograph of the Harada hand, a prosthetic hand with five black, segmented fingers. The hand is mounted on a silver-colored metal base, which is connected to a complex assembly of metal rods and wires, likely actuators and sensors.
Center or company	Bologna university	California Institute of Technology
Country	Italy	Japan
Year	2004	2001
Number of fingers	Five	Five
Number of joints	20	14
DOF	16	5
Transmission types	Tendons	Tendons
Actuator types	small DC motors with high reduction ratio	DC motors
Location of actuators	Remote	Remote
Sensorial system	Position sensors (potentiometer), Tendon force sensor	No sensor
Feedback control	Stiffness/position control	No control

Table B.7: UB hand III and Harada hands' features

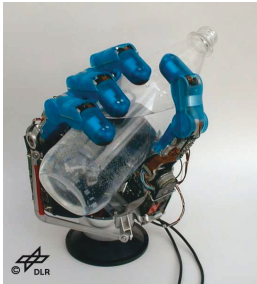
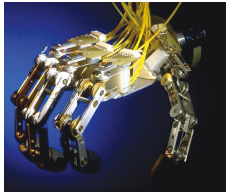
Name	DLR hand II (159)	DIST hand (160)
Picture		
Center or company	DLR-Germany Aerospace center	Genova university
Country	Germany	Italy
Year	2001	1998
Number of fingers	Four	Four
Number of joints	17	16
DOF	13	16
Transmission types	Brushless DC-motor	Tendons
Actuator types	Gearbox	DC motors
Location of actuators	At fingers' level	Remote
Sensorial system	Motor position sensor (potentiometers), strain-gauged based joint torque sensors, temperature sensor	Joint position sensors based on Hall effect, force sensor at fingertip and tactile sensor
Feedback control	Position control, joint level impedance control, force control	Force control all fingertips together, joint position control

Table B.8: DLR hand II and DIST hands' features

B. ROBOTS HANDS

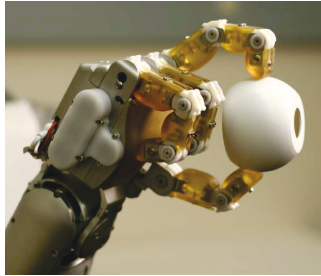
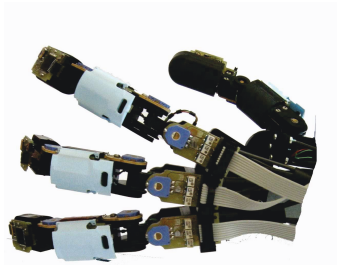
Name	Meka H2 Compliant Hand	MAC hand (47)
Picture	 A close-up photograph of the Meka H2 Compliant Hand, a yellow and white prosthetic hand, holding a white egg.	 A photograph of the MAC hand, a black and white prosthetic hand, shown in a flexed position.
Center or company	San Francisco	University of Genoa
Country	USA	Italy
Year	2008	2005
Number of fingers	Four	Four
Number of joints	12	12
DOF	5	12
Transmission types	Series Elastic Actuator	DC motor
Actuator types	DC motor	DC motors
Location of actuators	In the hand	In the hand
Sensorial system	Hall Effect sensor measuring the Series Elastic Actuator spring displacement giving the force feedback, encoder motor sensor, motor temperature and current sensors	Joint position sensors, force sensors, tactile sensors
Feedback control	Force closure control, joint position control, joint stiffness control	Force/tactile control, position control

Table B.9: H2 Compliant and MAC hands' features

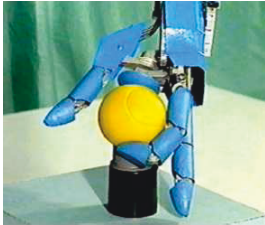

Name	UB Hand II (46) Hand	Barret hand (161)
Picture		
Center or company	Bologna university	MIT Barrett technology, Inc
Country	Italy	USA
Year	1992	2000
Number of fingers	Three	Three
Number of joints	13	9
DOF	13 (11 hand + 2 wrist)	8
Transmission types	Tendon	Gearbox
Actuator types	Electrical revolute motors	Electrical revolute motors
Location of actuators	Remote	At each joint
Sensorial system	Joint position measurement based on the Hall-effect sensor, encoder at each motor, Intrinsic tactile force sensor placed on each phalanx and on the palm	Motor position sensors (optical incremental encoders), joint torque sensor
Feedback control	Position/torque control	Joint position control, joint velocity control

Table B.10: UB hand II and Barrett hands' features

Bibliography

- [1] G. Nelson R. Playter M. Raibert, K. Blankespoor and the BigDog Team. Bigdog, the rough-terrain quadruped robot. *Proceedings of the 17th World Congress The International Federation of Automatic Control*, pp. 10822-10825, 2008. 1
- [2] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. *In Proc. IEEE Int. Conf. on Robotics and Automation*, vol 1, pp. 348-353, 2000. 4
- [3] D. Ding; Y.H. Liu and S. Wang. Computing 3-d optimal formclosure grasps. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 3573-3578, 2000. 4
- [4] A. Morales; E. Chinellato; A. H. Fagg and A.P. del Pobil. Using experience for assessing grasp reliability. *Int. Journal of Humanoid Robotics*, 1(4), pp. 671-691, 2004. 4
- [5] A.T. Miller; S. Knoop; H.I. Christensen and P.K. Allen. Automatic grasp planning using shape primitives. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1824-1829, 2003. 4
- [6] H. Friedrich; R. Dillmann and O. Rogalla. Interactive robot programming based on human demonstration and advice. *Sensor Based Intelligent Robots*, pp. 96-119, 1998. 4
- [7] M.N. Nicolescu and M.J. Mataric. Natural methods for robot task learning: instructive demonstrations, generalization and practice. *Int. Conf. on Autonomous Agents*, pp. 241-248, 2003. 4
- [8] S. Ekvall and D. Kragic. Grasp recognition for programming by demonstration. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 748-753, 2005. 4
- [9] T. Tsuji; K. Harada; K. Kaneko; F. Kanehiro and Y. Kawai. Motion for humanoid robots with a multi-fingered hand. *IEEE-RAS Int. Conf. on Humanoid Robots*, pp. 54-60, 2008. 4

BIBLIOGRAPHY

- [10] T. Tsuji; K. Harada and K. Kaneko. Easy and fast evaluation of grasp stability by using ellipsoidal approximation of friction cone. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009. 4
- [11] T. Tsuji; K. Harada; K.Kaneko; F.Kanehiro and M.Morisawa. Grasp planning for a multi-fingered hand with a humanoid robot. *Int. Jour. of Robotics and Mechatronics*, 2009. 4
- [12] T. Tsuji; K. Harada; K. Kaneko; F. Kanehiro and K. Maruyama. Grasp planning for a humanoid hand. *The Future of Humanoid Robots - Research and Applications*, Dr. Riadh Zaier (Ed.), ISBN: 978-953-307-951-6, 2012. 4
- [13] A. Bicchi. On the closure properties of robotics grasping. *Int. Jour. of Robotics Research*, 14(4), pp. 319-334, 1995. 5
- [14] C. Ferrari and J. Canny. Planning optimal grasps. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2290-2295, 1992. 5
- [15] R.D. Hester; M. Cetin; C. Kapoor and D. Tesar. A criteria-based approach to grasp synthesis. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1255-1260, 1999. 5
- [16] T. Omata and K. Nagata. Planning reorientation of an object with multifingered hand. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 3104-3110, 1994. 5
- [17] A. Sudsang and T. Phoka. Regrasp planning for a 4-fingered hand manipulating a polygon. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2671-2676, 2003. 5
- [18] N. Niparnan; T. Phoka; P. Pipattanasomporn and A. Sudsang. Regrasp planning of four-fingered hand for parallel grasp of a polygonal object. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 7961-2676, 2005. 5
- [19] D. Williams and O. Khatib. The virtual linkage: a model for internal forces in multi-grasp manipulation. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1025-1030, 1993. 5
- [20] T. Wimboeck; C. Ott and G. Hirzinger. Passivity-based object-level impedance control for a multifingered hand. *Int. Conf. on Intelligent Robots and Systems*, pp. 4621-4627, 2006. 5

- [21] T. Wimboeck; C. Ott and G. Hirzinger. Analysis and experimental evaluation of the intrinsically passive controller (ipc) for multifingered hands. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 278-284, 2008. 5
- [22] M. Roa and R. Suarez. Geometrical approach for grasp synthesis on discretized 3d objects. *IEEE-RSJ International Conference on Intelligent Robots and Systems*, pp. 3283-3288, 2007. 5
- [23] N. Rezzoug and P. Gorce. A biocybernetic method to learn hand grasping posture. *IKybernetes*, 32(4), pp. 478-490, 2003. 5
- [24] R. Platt; A. H. Fagg and R. Grupen. Nullspace composition of control laws for grasping. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1717-1723, 2002. 5
- [25] R. Platt; A. H. Fagg and R. Grupen. Manipulation gaits: Sequences of grasp control tasks. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 801-806, 2004. 5
- [26] <http://www.shadowrobot.com/hand>. 7, 20, 71, 72, 128
- [27] Shadow c6m hand user manual. 2012. 7, 55, 71, 78
- [28] Department of Anthropology The university of Texas at Austin. "anthropoids." eskeletons. n.p., n.d. <http://www.eskeletons.org/treecat.html>, 2013. 11
- [29] G.A. Thibodeau and K.T. Patton. The human body in health and disease. *The university of Michigan, Mosby*, 2002. 11
- [30] L. Biagiotti; F. Lotti; C. Melchiorri and G.Vassura. How far is the human hand? a review on anthropomorphism robotic end-effectors. *DIES Internal Report, University of Bologna, Tech. Rep*, 2004. 12, 14
- [31] A. Bicchi. Hands for dexterous manipulation and robust grasping; a difficult road toward simplicity. *IEEE Tran. on Robo. and Aut. Vol. 16, no 6*, pp. 652-662, 2000. 12, 14
- [32] Y. Namiki; M. Imai and M. Ishikawa. Development of a highspeed multifingered hand system and its application to catching. *IEEE/RSJ Int Conf on Intelligent Robots and Systems*, pp. 2666-2671, 2003. 13

BIBLIOGRAPHY

- [33] J.E.F. Tegin. Tactile grasping for domestic service robots, simulations, experiments and hand design. *PhD thesis, KTH School of Industrial Engineering and Management Department of Machine Design, Sweden*, 2009. 13
- [34] Y. Bekiroglu; J. Laaksonen; J. A. Jorgensen and V. Kyrki. Learning grasp stability based on haptic data. *Robotics: Science and Systems Workshop on Representations for Object Grasping and Manipulation in Single and Dual Arm Tasks*, 2010. 13
- [35] V. Christopoulos and P. Schrater. Handling shape and contact location uncertainty in grasping two-dimensional planar objects. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1557-1563, 2007. 13
- [36] S. Chitta; M. Piccoli and J. Sturm. Learning grasp stability based on haptic data. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2342-2348., 2010. 13
- [37] C. Borst; M. Fischer; S. Haidacher; H. Liu and G Hirzinger. Dlr hand ii: Experiments and experiences with an antropomorphic hand. *In Proc. IEEE Int. Conf. on Robotics and Automation, volume 1*, pp. 702-707, 2003. 13
- [38] H. Liu; P. Meusel; G. Hirzinger; M.J.Y. Liu and Z. Xie. The modular multisensory dlr-hit-hand: Hardware and software architecture. *IEEE Trans. on Mechatronics*, 14, 2008. 13
- [39] B. Massa; S. Roccella; M.C. Carrozza and P. Dario. Design and development of underactuated prosthetic hand. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 3374-3379, 2002. 15
- [40] M. Buss T. Schlegl and G. Schmidt. A hybrid systems approach toward modeling and dynamical simulation of dextrous manipulation. *IEEE/ASME Transaction on Mechatronics*, 8(3), pp. 352-361, 2003. 20
- [41] Z. Chen; N.Y. Lii; T. Wimboeck; S. Fan; M. Jin; C.H. Borst and H. Liu. Experimental study on impedance control for the five-finger dexterous robot hand dlr-hit ii. *The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5867-5874, 2010. 20
- [42] <http://www.prensilia.com>. 20, 129
- [43] C. Lovchik and M. Diftler. The robonaut hand: A dexterous robot hand for space. *In Proc. IEEE Int. Conf. on Automation and Robotics*, pp. 907-912, 1999. 20, 128

- [44] T. Mouri; H. Kawasaki; K. Yoshikawa; J. Takai and S. Ito. Anthropomorphic robot hand: Gifu hand iii. *ICCAS2002, Muju Resort, Jeonbuk, Korea*, pp. 1288-1293, 2002. [20](#), [55](#), [131](#)
- [45] S. Schulz; C. Pylatiuk and G. Bretthauer. A new ultralight anthropomorphic hand. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2437-2441, 2001. [20](#), [131](#)
- [46] A. Eusebi; C. Fantuzzi; C. Melchiorri; M. Sandri and A. Tonielli. The u.b. hand i1 control system: Design features and experimental results. *Int. Conf. on Industrial Electronics, Control and Instrumentation*, pp. 782-787, 1994. [20](#), [137](#)
- [47] C. Giorgio and M. Maggiali. An embedded tactile and force sensor for robotic manipulation and grasping. *IEEE-RAS International Conference on Humanoid Robots*, 2005. [20](#), [136](#)
- [48] <http://www.romela.org/main/RAPHAEL:Robotic-Air-Powered-Handwith-Elastic-Ligaments>. [20](#), [71](#), [130](#)
- [49] S. Roccella; M.C. Canozza; G. Cappiello; M. Zecca; H. Miwa; K. Itoh and M. Matsumoto. Design, novel fabrication and preliminary results of a anthropomorphic hand for humanoid robotics: Rch-1. *Proc. 01 2004 IEEWRSJ International Conference on Intelligent Robots and Systems*, pp. 266-271, 2004. [20](#), [132](#), [133](#)
- [50] G. Stillfried and P. Van Der Smagt. Movement model of a human hand based on magnetic resonance imaging (mri). *1st International Conference on Applied Bionics and Biomechanics*, 2010. [21](#)
- [51] M. Jenkins; H. Bamberger; L. Black and R. Nowinski. Thumb joint exion. what is normal? *Journal of Hand Surgery, Vol. 23*, 1998. [21](#)
- [52] P. Hahn; H. Krimmer; A. Hradetzky and U. Lanz. Quantitative analysis of the linkage between the interphalangeal joints of the index finger: An in vivo study. *Journal of Hand Surgery, Vol. 20*, 1995. [21](#)
- [53] J. Park and J. Cheong. Analysis of collective behavior and grasp motion in human hand. *In Proc. IEEE Int. Conf. on Control, Automation and systems*, pp. 2514-2518, 2010. [21](#)
- [54] I. Carpinella; P. Mazzoleni; M. Rabuketti; R. Thorsen and M. Ferrarin. Experimental protocol for the kinematic analysis of the hand: definition and repeatability. *Gait and Posture, Vol. 23*, 2006. [21](#)

BIBLIOGRAPHY

- [55] N. Miyata; M. Kouchi; T. Kurihara and M. Mochimaru. Modeling human hand link structure from optical motion capture data. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004. 21
- [56] C. Metcalf; S. Notley; P. Chappell; J. Burridge and V. Yule. Validation and application of a computational model for wrist and hand movements using surface markers. *IEEE Transactions on Biomedical Engineering*, Vol. 55, No. 3, 2008. 21
- [57] S. Ekvall and D. Kragic. Learning task models from multiple human demonstration. In *Proc. IEEE Int. Workshop on Robot and Human Communication*, 2006. 21
- [58] A. Bicchi and V. Kumar. Robotic grasping and contact: a review. In *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Vol. 1, pp. 348-353, 2000. 21
- [59] A. D'Souza; S. Vijayakumar and S. Schaal. Learning inverse kinematics. *Proc. of the 2001 IEEE/RSJ, International Conference on Intelligent Robots and Systems*, pp. 298-303, 2001. 24
- [60] P.P. Kumar and L. Behera. Visual servoing of redundant manipulator with jacobian matrix estimation using self-organizing map. *Robotics and Autonomous Systems*, Vol. 58, pp. 978-990, 2010. 24
- [61] C. Bensalah. Control visual de un manipulador redundante. *Oficial Master in robotic and Automation*, 2005. 24
- [62] P.P. Kumar and L. Behera. Supervised learning with a distal teacher. *Cognitive science* 16, pp. 307-345, 1992. 24
- [63] L.C.T. Wang and C.C. Chen. A combined optimization method for solving the inverse kinematics problem of mechanical manipulators. *IEEE transactions on Robotics and Automation*, No. 7, pp. 489-499, 1991. 24
- [64] J. Lander. Game developer 5(3), pp. 15-22. *Making kinematics more flexible*, 1998. 24
- [65] A.A. Canutescu and R.L. Dunbrack. Cyclic coordinate descent: a robotics algorithm for protein loop closure. *Protein Science* 12 (5), pp. 963-972, 2003. 24, 25

- [66] C. Welman. Inverse kinematics and geometric constraints for articulated figure manipulation. *Master Dissertation, Simon Fraser University, Department of Computer Science*, 1993. 25
- [67] D. Merrick and T. Dwyer. Skeletal animation for the exploration of graphs. In *Neville Churcher and Clare Churcher, editors, Australasian Symposium on Information Visualisation, (invis.au04), volume 35 of CRPIT, pages 61-70, Christchurch, New Zealand*, 2004. 25
- [68] J.O. Kim; B.R. Lee; C.H. Chung; J. Hwang and W. Lee. The inductive inverse kinematics algorithm to manipulate the posture of an articulated body. In *Proc. of the Int. Conf. on Computational Science, ICCS 2003, volume 2657 of Lecture Notes in Computer Science, pages 305-313. Springer*, 2003. 25
- [69] J.O. Kim; B.R. Lee and C.H. Chung. On the uniform posture map algorithm for an articulated body. *IEEE International Symposium on Virtual and Intelligent Measurement Systems Anchorage*, 2002. 25
- [70] R. Muller-Cajar and R. Mukundan. Triangulation: A new algorithm for inverse kinematics. *Proceedings of the Image and Vision Computing New Zealand*, pp. 181-186, 2007. 25
- [71] S.R. Buss and J.S. Kim. Selectively damped least squares for inverse kinematics. In *Journal of Graphics Tools*, pp. 37-49, 2005. 26, 32, 34, 45, 46
- [72] D.E. Whitney. Resolved motion rate control of manipulators and human prostheses. *IEEE Transactions on Man-Machine Systems*, pp. 47-53, 1969. 26, 28, 31
- [73] A. Goldenberg; B. Benhabib and R.G. Fenton. A complete generalized solution to the inverse kinematics of robots. *IEEE Journal of Robotics and Automation*, Vol. Ra-1, No. 1, 1985. 26, 29
- [74] R.V. Patel and F. Shadpey. *Control of Redundant Robot Manipulators, Theory and Experiments*. Springer-verlag berlin heidelberg edition, 2005. 26, 30, 31, 32, 39, 57
- [75] A. Ligeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Transactions on Systems, Man and Cybernetics*, pp. 868-871, 1977. 26, 30, 31

BIBLIOGRAPHY

- [76] H. Seraji. Task options for redundancy resolution using configuration control. *30th IEEE Conf. on Decision and Control*, pp. 2793-2798, 1991. 26, 31
- [77] A.A. Maciejewski and C.A. Klein. Obstacle avoidance for kinetically redundant manipulators in dynamically varying environments. *International Journal of Robotic Research*, Vol. 4, pp. 109-117, 1985. 26, 31
- [78] H. Seraji and R. Colbaugh. Singularity-robustness and task-prioritization in configuration control of redundant robots. *Proc. of the 29th Conf. on Decision and Control*, pp. 3089-3095, 1990. 26, 31, 53
- [79] W.A. Wolovich and H. Elliot. A computational technique for inverse kinematics. *In Proc. IEEE Int. Conf. on Decision and Control*, pp. 1359-1363, 1984. 26
- [80] L. Sciavicco and B. Siciliano. A dynamic solution to the inverse kinematic problem for redundant manipulators. *International Conference on Robotics and Automation - ICRA*, 1987. 26
- [81] J. Denavit and R. Hartenberg. A kinematic notation for lower-pair mechanisms based matrices. *Journal of Applied Mechanics*, Vol.22, pp. 215-221, 1955. 28
- [82] Anthony A. Maciejewski. Dealing with the ill-conditioned equations of motion for articulated figures. *IEEE Computer Graphics and Applications*, pp. 63-71, 1990. 29, 31, 34, 36, 37
- [83] Tzu-Chen Liang and Jing-Sin Liu. An improved trajectory planner for redundant manipulators in constrained workspace. *Journal of Robotic Systems*, 16(6), pp. 339-351, 1999. 29
- [84] Adi Ben-Israel and Tomas N.E. Greville. *Generalized Inverses: Theory and Applications*. Springer-verlag new york, inc (second edition) edition, 2003. 30, 34, 37
- [85] R.V. Dubey; J.A. Euler and S.M. Babock. An efficient gradient projection optimization scheme for a seven-degree-of-freedom redundant robot with spherical wrist. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 28-36, 1988. 30
- [86] R. Colbaugh; H. Seraji and K. Glass. Obstacle avoidance of redundant robots using configuration control. *Int. Journal of Robotics Research*, Vol.6, pp. 721-744, 1989. 31

- [87] S.I. Choi and B.K. Kim. Obstacle avoidance control for redundant manipulators using collidability measure. *Robotica*, Vol. 18, pp. 143-151, 2000. 31
- [88] L. Villani B. Siciliano, L. Sciavicco and G. Oriolo. *Robotics: Modelling, Planning and Control*. Springer-Verlag London Limited, 2009. 31, 32
- [89] H. Zghal; R.V. Dubey and J.A. Euler. Efficient gradient projection optimization for manipulators with multiple degrees of redundancy. *In Proc. IEEE Int. Conf. on Robotics and Automation*, Vol. 2, pp. 1006-1011, 1990. 31
- [90] T.F. Chan and R.V. Dubey. A weighted least-norm solution based scheme for avoiding joint limits for redundant joint manipulators. *IEEE Transactions on Robotics and Automation*, Vol. 11, pp. 286-292, 1995. 31
- [91] T. Yoshokawa. Manipulability of robotic mechanisms. *International Journal of Robotics Reaserch*, Vol. 4, pp.3-9, 1985. 31, 37
- [92] T. Yoshokawa. Dynamic manipulability of robot manipulators. *Journal of Robotic Systems*, Vol. 2, pp. 113-124, 1985. 31, 37
- [93] O. Egeland. Task-space tracking with redundant manipulators. *IEEE Journal of Robotics and Automation*, Vol. 3, pp. 471-475, 1987. 32
- [94] L. Sciavicco and B. Siciliano. A solution algorithm to the inverse kinematic problem for redundant manipulators. *IEEE Journal of Robotics and Automation*, Vol.4, pp. 403-410, 1988. 32
- [95] C.W. Wampler. Manipulator inverse kinematic solutions based on vector formulations and damped least squares methods. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 16, pp. 93-101, 1987. 32
- [96] S. Chiaverini; B. Siciliano and O. Egeland. Review of damped least-squares inverse kinematics with experimenets on an industrial robot manipulator. *IEEE Transactions on Control Systems Technology*, Vol. 2, pp. 123-134, 1994. 34, 37, 41, 42
- [97] A.S. Deo and I.D Walker. Robot subtask performance with singularity robustness using optimal damped least-squares. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 434-441, 1992. 34

BIBLIOGRAPHY

- [98] A.S. Deo and I.D Walker. Adaptative non-linear least squares for inverse kinematics. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 186-193, 1993. 34
- [99] A.A. Maciejewski and C.A. Klein. The singular value decomposition: Computation and applications to robotics. *International Journal of Robotic and Research*, Vol. 8, pp. 63-79, 1989. 34
- [100] R.V. Mayorga; A.K. Wong and N. Milano. A fast procedure for manipulator inverse kinematics evaluation and pseudoinverse robustness. *IEEE Transactions on system, man and cybernetics*, Vol. 22, pp. 790-798, 1992. 34
- [101] S.K. Chan and P.D. Lawrence. General inverse kinematics with the error damped pseudoinverse. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 834-839, 1988. 34
- [102] S. Chiaverini. Estimate of the two smallest singular values of the jacobian matrix: Applications to damped least-squares inverse kinematics. *Journal of Robotics Systems*, 10, pp. 991-1008, 1988. 34, 39
- [103] A.A. Maciejewski and C.A. Klein. Numerical filtering for the operation of robotic manipulators through kinematically singular configurations. *Journal of Robotics Systems*, 5(6), pp. 527-552, 1988. 34, 36, 37, 38, 39, 40, 41, 44
- [104] J.M; Varah. A practical examination of some numerical methods for linear discret ill-posed problems. *SIAM Review*, 21(1), pp. 100-111, 1979. 36, 37
- [105] Y. Nakamura and H. Hanafusa. Inverse kinematic solutiond with singularity robustness for manipulator control. *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 108, pp. 163-171, 1987. 37
- [106] O. Egeland; J.R. Sagli and I. Spangelo. A damped least-squares solution to redundancy resolution. *In Proc. IEEE Int. Conf. on Robotics and Automation*, 1991. 41
- [107] P. Sivakumar; K. Magesway and M. Rajaram. *Proc. of 2011 Int. Conf. on Signal Processing, Communication, Computing and Networking Technologies (ICSCCN 2011)*. 49
- [108] J. Baillieul. Kinematic programming alternatives for redundant manipulators. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 722-728, 1985. 53

- [109] M. Shah and R.V. Patel. Inverse jacobian based hybrid impedance control of redundant manipulators. *Conf. on Mecha. and Auto. Niagara Falls, Canada*, 2005. 54
- [110] H. Liu; K. Wu; P. Meusel; N. Seitz; G. Hirzinger; M.H. Jin; Y.W. Liu; S.W.Fan; T. Lan and Z.P.Chen. Multisensory five-finger dexterous hand: The dlr/hit hand ii. *IEEE/RSJ International Conference on Intelligent Robots and Systems Acropolis Convention Center*, 2008. 55
- [111] C. Rosales; J.M. Porta; R. Suarez and L. Ros. Finding all valid hand configurations for a given precision grasp. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1634-1640, 2008. 55
- [112] N. Hogan. Adaptative control of mechanical impedance by coactivation of antagonist muscles. *IEEE Transactions on Automatic Control*, Vol. Ac-29, No. 8, 1987. 71, 72
- [113] L. Biagiotti; F. Lotti; C.Melchiorri; G. Palli; P. Tiezzi and G. Vassura. Development of ub hand 3: Early results. *In Proc. IEEE Int. Conf. on Robotics and Automation*, 2005. 71, 134
- [114] C.S. Lovchik and M.A. Difler. The robonaut hand: A dexterous robotic hand for space. *In Proc. IEEE Int. Conf. on Robotic and Automation*, pp. 907-912, 1999. 71
- [115] Y.J. Ou and L.W. Tsai. Theory of isotropic transmission for tendon-driven manipulators. *In Robotics: Kinematics, Dynamics and Control*, ASME, pp. 53-61, 1994. 72
- [116] R. Verhoeven; M. Hiller and S. Tadokoro. Workspace, stiffness, singularities and classification of tendon-driven stewart-platforms. *In Proceedings of the 6th International Symposium on Advances in Robot Kinematics*, ARK 98, pp. 105-114, 1998. 72
- [117] G. Palli; C. Melchiorri; T. Wimbeck; M. Grebenstein and G. Hirzinger. Feedback linearization and simultaneous stiffness-position control of robots with antagonistic actuated joints. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 4367-4372, 2007. 72

BIBLIOGRAPHY

- [118] S. Wolf and G. Hirzinger. A new variable stiffness design: Matching requirements of the next robot generation. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1741-1746, 2008. 72
- [119] C.E. English and D. Russell. Mechanics and stiffness limitations of a variable stiffness actuator for use in prosthetic limbs. *Mechanism and Machine Theory* 34, pp. 7-25, 1999. 72
- [120] C. English and D. Russell. Implementation of variable joint stiffness through antagonistic actuation using rolamite springs. *Mechanism and Machine Theory* 34, pp. 27-40, 1999. 72
- [121] F. Petit; M. Chalon; W. Friedl; M. Grebenstein; A.A. Schaffer and G. Hirzinger. Bidirectional antagonistic variable stiffness actuation: Analysis, design and implementation. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 4189-4196, 2010. 72
- [122] G.A. Pratt and M.M. Williamson. Series elastic actuators. *Intelligent Robots and Systems 95. Human Robot Interaction and Cooperative Robots*, pp. 399-406, 1995. 72
- [123] D. Whitney. Historical perspective and state of the art in robot force control. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 262-268, 1985. 73
- [124] P.E. Dupont. Stick-slip and control in low-speed motion. *IEEE transactions on Automatic Control*, Vol. 30 No. 10, pp. 1094-1097, 1993. 78
- [125] T. Tjahjowidodo; F. Al-Bender and H.V. Brussel. Friction identification and compensation in a dc motor. *Proc. of the 16th IFAC World Congress*, pp. 92-98, 2005. 78
- [126] P.E. Dupont. Avoiding stick-slip through pd control. *IEEE transactions on Automatic Control*, Vol. 39 No. 5, pp. 1094-1097, 1994. 78
- [127] M. Nordin and P. Gutman. Controlling mechanical systems with backlash. *Automatica* 38, pp. 1633-1649, 2002. 81
- [128] J.J. Craig and M.H. Raibert. A systematic method of hybrid position/force control of a manipulator. *Computer Software and Applications Conference, IEEE Computer Society*, pp. 446-451, 1979. 84, 85

- [129] H. Zhang and R.P. Paul. Hybrid control of robot manipulators. *In Proc. IEEE Int. Conf. on Robotics and Automation, IEEE Computer Society, pp. 602-607, 1985.* 85
- [130] W.D. Fisher and M.S. Mujtaba. A kinematically stable hybrid position/force control scheme. *International Journal of Robotics Research, 1991.* 85
- [131] M. Uchiyama and P. Dauchez. A symmetric hybrid position/force control scheme for coordination of two robots. *In Proc. IEEE Int. Conf. on robotics and Automation, pp. 350-356, 1988.* 85
- [132] M. Yamano; J.S. Kim and M. Uchiyama. Hybrid position/force control of two cooperative flexible manipulators working in 3d space. *In Proc. IEEE Int. Conf. on robotics and Automation, pp. 350-356, 1998.* 85
- [133] T. Yoshikawa and K. Nagai. Manipulating and grasping forces in manipulation by multifingered robot hands. *IEEE Transaction On Robotics And Automation, Vol. 7. No. 1, pp. 67-77, 1991.* 85
- [134] Li. Zexiang; P. Hsu and S. Sastry. Dynamic coordination of a multiple robotic system with point contact. *American Control Conference, 1988.* 85
- [135] P. Minyong; K. Mouri; H. Kitagawa; T. Miyoshi and K. Terashima. Hybrid impedance and force control for massage system by using humanoid multi-fingered robot hand. *IEEE International Conference on System, Man and Cybernetics, pp. 3021-3026, 2007.* 85
- [136] Li. Zexiang; P. Hsu and S. Sastry. Computing of the sub-optimal grasping forces for manipulation object by multifingered robot hand. *In Proc. IEEE Int. Conf. on Robotics and Automation, 1996.* 85
- [137] V. Perdereau and M. Drouin. Hybrid position/force control for multi-fingered robot hand. *Proc. of the IEEE-SMC IMACS Multiconference on Computational Engineering in Systems Applications, 1998.* 85
- [138] L. Zexiang; P. Hsu and S. Sastry. Online computation of grasping force in multi-fingered hands. *Intelligent Robots and Systems, (IROS 2005), pp. 1223-1228, 2005.* 85
- [139] M. Buss and K.P. Kleinmann. Multi-fingered grasping experiments using real-time grasping force optimization. *In Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1801-1806, 1996.* 85

BIBLIOGRAPHY

- [140] B.H. Kim; B.J. Yi; S.R. Oh and I.H. Suh. Task-based compliance planning for multi-fingered robotic manipulations. *Advanced Robotics, Vol. 18, No. 1, pp. 23-44*, 2004. [85](#)
- [141] Partner responsible: UC3M. Low level control algorithms. deliverable 22. *HANDLE Developmental pathway towards autonomy and dexterity in robot in-hand manipulation*, 2012. [87](#)
- [142] V. Utkin. Variable structure systems with sliding modes. *IEEE Trans. on AC.22, No.2, pp. 212-222*, 1977. [89](#)
- [143] S.V. Emelyanov. Theory of variable-structure control systems: inception and initial development. *Computational Mathematics and Modeling, Vol. 18, No. 4*, 2007. [89](#)
- [144] J.Y. Hung; W. Gao and J.C. Hung. Variable structure control: A survey. *IEEE Trans. on Industrial Electronics, Vol. 40, No. 1*, 1993. [89](#)
- [145] R.A. Decarlo; S.H. Zak and G.P. Matthews. Variable structure control of nonlinear multivariable systems: A tutorial. *Proc. of IEEE, Vol. 76, No. 3, pp. 212-232*, 1988. [89](#)
- [146] S.Z. Sarpturk; Y. Istefanopulos and O. Kaynak. On the stability of discrete-time sliding mode control systems. *IEEE Tran. on Automatic Control, Vol. AC-32, No. 1, pp. 930-932*, 1987. [89](#)
- [147] A. Sakurai. Sliding mode control of switched reluctance motor. *Phd Thesis, University of Toronto*, 2001. [90](#)
- [148] V. Utkin; J. Guldner and J. Shi. Sliding mode control in electromechanical system. *CRC Press*, 2000. [94](#), [103](#), [104](#)
- [149] J.J.E Slotine and W. Li. *Applied nonlinear control*. Printice-Hall, 1991. [95](#), [103](#)
- [150] *www.quanser.com*, 2014. [104](#)
- [151] Y. Xia; M. Fu; H. Yang and G.P. Liu. Robust sliding mode control for uncertain time-delay systems based on delta operator. *IEEE Transactions on Industrial Electronics, Vol. 56, No. 9, pp. 3646-3655*, 2009. [106](#)

- [152] H.P. Pang; C.J. Liu and A.Z. Liu. Sliding mode control for time-delay systems with its application to networked control systems. *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06)*, 2006. 106
- [153] 106
- [154] J. Hu; J. Chu and H. Su. Smvsc for a class of time-delay uncertain systems with mismatching uncertainties. *IEEE Proc. Control Theory Appl.*, Vol. 147, pp. 687-693, 2000. 106
- [155] Y. Niu; D.W.C. Ho and J. Lam. Robust integral sliding mode control for uncertain stochastic systems with time-varying delay. *Automatica*, Vol. 41, pp. 873-880, 2005. 106
- [156] C.N. Kien. Fingertip force control for grasping and in-hand manipulation. *HANDLE workshop Benicassim*, 2012. 109
- [157] N. Fukaya; S. Toyama; T. Asfour and R. Dillmann. Design of the tuat/karlsruhe humanoid hand. *IEEE/RSJ Intern. Conf. on Intelligent Robots and Systems Japan*, 2000. 132
- [158] D. Keymeulen and C. Assad. Harada electric industry inc. <http://www.h-e-i.co.jp/Products/e-m-g/ph-sh-2-004.html>, 2004. 134
- [159] J. Butterfass; M. Grebenstein and H. Liu. Dlr - hand ii: Next generation of a dexterous robot hand. *In Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 109-114, 2001. 135
- [160] T. Mouri; H. Kawasaki; K. Yoshikawa; J. Takai and S. Ito. The design and development of the dist-hand dextrous gripper. *In Proc. IEEE Int. Conf. On Robotics And Automation*, pp. 2075-2080, 1998. 135
- [161] <http://www.robotnik.es/es/productos/manos-roboticas/barrett-hand>. 137

BIBLIOGRAPHY
