Modeling of physical interactions between a human and a robot with adaptive compliance

Dmitrii Popov

COLLEGES BRETAGNE LOIRE | SCIENCES POUR L'INGENIEUR ET LE NUMERIQUE

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPÉRIEURE
MINES-TÉLÉCOM ATLANTIQUE BRETAGNE
PAYS-DE-LA-LOIRE - IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 648
Sciences pour l'Ingénieur et le Numerique
Spécialité : *Génie industriel, productique, automatique et robotique*

Par
Dmitrii POPOV

Modeling of Physical Interactions Between a Human and a Robot with Adaptive Compliance

Thèse présentée et soutenue à l'IMT Atlantique à Nantes, le 17 mai 2023
Unité de recherche : Laboratoire des Sciences du Numérique de Nantes (LS2N)
Thèse N° : 2023IMTA0357

Rapporteurs avant soutenance :

Andrea CHERUBINI — Professeur, Université Montpellier, LIRMM
Med Amine LARIBI — Maître de conferences, HDR, Université de Poitiers

Composition du Jury :

Président :	Abderrahmane KHEDDAR	Directeur de recherche, CNRS, LIRMM
Examinateurs :	Natalie SMITH-GUERIN	Maître de conferences, Université Bretagne Sud
	Damien CHABLAT	Directeur de recherche, CNRS, LS2N
	Andrea CHERUBINI	Professeur, Université Montpellier, LIRMM
	Med Amine LARIBI	Maître de conferences, HDR, Université de Poitiers
Dir. de thèse :	Anatol PASHKEVICH	Professeur, IMT-Atlantique
Co-dir. de thèse :	Alexandr KLIMCHIK	Associate Professor, University of Lincoln, UK

Invité :

Alexander MALOLETOV — Professor, Innopolis University, Russia

Acknowledgement

Before beginning this manuscript, I would like to express my gratitude to the people who have helped and guided me in my research and who have provided all possible assistance throughout the time of working on the thesis. It is a pleasure for me to thank those who made this thesis possible.

First and foremost I am sincerely thankful to my supervisors, professors Anatol Pashkevich and Alexandr Klimchik. In particular, it was Alexander Klimchik who sparked my interest in robotics and encouraged me to make my PhD. I am grateful to Anatol Pashkevich for directing my research in the right way, giving invaluable advises, and spending a lot of time to make my thesis better. Both of my supervisors taught me to write high-quality scientific works and carry out independent research. Without their assistance and dedicated involvement in every step throughout the process, this work would have never been accomplished. I would like to thank them very much for their support and understanding over these years.

I take this opportunity to express my gratitude to all of my colleagues from IMT-Atlantique and Laboratoire des Sciences du Numérique de Nantes, as well as Innopolis University and Center for Technologies in Robotics and Mechatronics Components for their help and support. I would like to express special gratitude to Dr. Damien Chablat and Stanislav Mikhel for their help with organizing this research and experimental study.

Finally, the last and most important thanks go to my spouse Albina and my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Abstract and keywords

The thesis deals with the modeling of physical interactions between a human and a robot, which is extremely important for the design and development of modern collaborative work cells for new industrial applications. It focuses on the development of interaction identification techniques in order to improve human safety and human-robot work cell performance. Special attention is paid to the accuracy issues caused by the noisy measurements data and the interaction parameters identification in singular cases, which can arise during the physical interaction because of limited measurements information provided by robot torque sensors. The thesis also presents new techniques for computing the interaction force and its application point using measurement data obtained from the robot internal joint torque sensors only. In contrast to existing approaches, the proposed methods are applicable for singular cases associated with an insufficient number of independent equations in a static equilibrium system, which produces non-unique solutions for the interaction parameters. For these numerically hard cases, a special singularity resolution technique was developed, based on some practice-inspired heuristics and on the interaction parameters estimation history. In addition, the adaptive interaction handling controller is developed by integrating the interaction identification in order to ensure human safety by changing the robot behavior mode. The validity of the developed approaches and their efficiency was confirmed by an experimental study involving collaboration between an operator and the KUKA LBR iiwa 14 robot.

Keywords: Physical Human-Robot Interaction, Safe Robotics, Human-Robot Collaboration, Interaction Parameters Identification, Singularity Resolution, Adaptive Interaction Handling.

Résumé et mots clés

La thèse porte sur la modélisation des interactions physiques entre un humain et un robot, ce qui est extrêmement important pour la conception et le développement de cellules de travail collaboratives modernes pour de nouvelles applications industrielles. Elle se concentre sur le développement de techniques afin d'améliorer la sécurité des personnes et les performances des cellules de travail homme-robot. Une attention particulière est accordée aux problèmes de précision causés par les données de mesures bruyantes et l'identification des paramètres d'interaction dans des cas singuliers, qui peuvent survenir lors de l'interaction physique en raison des informations de mesure limitées fournies par les capteurs de couple du robot. Elle présente de nouvelles techniques pour calculer la force d'interaction et son point d'application en utilisant uniquement les données de mesure obtenues à partir des capteurs de couple de l'articulation interne du robot. Contrairement aux approches existantes, les méthodes proposées sont applicables pour des cas singuliers associés à un nombre insuffisant d'équations indépendantes dans un système d'équilibre statique, qui produit des solutions non uniques pour les paramètres d'interaction. Pour ces cas numériquement difficiles, une technique spéciale de résolution de singularité a été développée, basée sur des heuristiques inspirées de la pratique et sur l'historique d'estimation des paramètres d'interaction. De plus, le contrôleur de gestion d'interaction adaptative est développé, celui-ci intègre l'identification d'interaction afin d'assurer la sécurité humaine en changeant le mode de comportement du robot. La validité des approches développées et leur efficacité ont été confirmées par une étude expérimentale impliquant une collaboration entre un opérateur et le robot KUKA LBR iiwa 14.

Mots clés: Interaction Physique Homme-Robot, Robotique Sûre, Collaboration Homme-Robot, Identification des Paramètres d'Interaction, Résolution de Singularité, Contrôle d'Interaction Adaptatif.

TABLE OF CONTENTS

List of Figures

List of Tables

List of Algorithms

List of Symbols

$\mathbf{p}$	Vector, interaction force application point
$\mathbf{p}^*$	Vector, arbitrary point
$\hat{\mathbf{p}}$	Vector, estimated interaction force application point
$\mathbf{p}^k$	Vector, interaction force application point expressed in the kth frame
$\mathbf{p}_i$	Vector, position of the ith joint origin
$\mathbf{p}_{prev}$	Vector, previously estimated force application point
$\mathbf{p}^0$	Vector, initial guess for force application point
$\delta\mathbf{p}$	Vector, small variation of force application point position
$\Delta\mathbf{p}$	Vector, shift of force application point position
$\{\mathbf{p}\}$	Set, set of all possible interaction force application points
$\mathbf{F}$	Vector, interaction force
$\hat{\mathbf{F}}$	Vector, estimated interaction force
$\delta\mathbf{F}$	Vector, small variation of interaction force
$\{\mathbf{F}\}$	Set, set of all possible interaction forces
$\mathbf{F}^\alpha$	Matrix, support vectors of friction cone approximation
$\mathbf{M}$	Vector, interaction moment
$\hat{\mathbf{M}}$	Vector, estimated interaction moment
$\mathbf{W}$	Vector, interaction wrench
$\hat{\mathbf{W}}$	Vector, estimated interaction wrench
$\boldsymbol{\mathcal{L}}$	Force action line
$\boldsymbol{\tau}$	Vector, torques in robot joints generated by interaction force
$\boldsymbol{\tau}^{(k)}$	Vector, torques in robot joints #1,...,k
$\tilde{\boldsymbol{\tau}}$	Vector, measured torques in robot joints
$\boldsymbol{\tau}_r$	Vector, torques in robot joints generated by estmated force
$\boldsymbol{\varepsilon}$	Vector, noise components of the torque measurments
$\mathbf{q}$	Vector, joint configuration angles of the manipulator
$\mathbf{q}^{(k)}$	Vector, joint configuration angles of the joints #1,...,k
$\mathbf{e}_i$	Vector, the ith joint roration axis direction
L_i	Length of the ith link
k	Index of the link to which the force is applied to

$\mathbf{J}$	Matrix, robot Jacobian
$\mathbf{J}_k$	Matrix, robot Jacobian for joints #1,...,k
$\mathbf{J}_k^w$	Matrix, extended robot Jacobian for joints #1,...,k
$\mathbf{J}_s$	Matrix, robot Jacobian for simplified model
$\mathbf{D}_k$	Matrix, linearized robot Jacobian for joints #1,...,k
$\mathbf{u}_p$	Matrix, nullspace of linearized Jacobin for force application point
$\mathbf{u}_F$	Matrix, nullspace of linearized Jacobin for force vector
$\boldsymbol{\mathcal{F}_\mu}$	general force direction constraint describing friction cone
$\mathbf{\Omega}^k$	Set, general surface constraint for the kth link
$\mathbf{s}_i^k$	Vector, coordinate of the ith node describing the kth link
$\mathbf{sv}^k$	Matrix, coordinate of trinagular face vertices describing the kth link surface
$\mathbf{sf}^k$	Matrix, list of connections describing the kth link surface
$\mathbf{n}^k$	Matrix, face noral vectors describing the kth link surface
$\mathbf{G}$	Set, graph repesenting robot surface
$\mathbf{E}$	Set, edges of graph nodes
$\mathbf{N}$	Set, coordinates of graph nodes
$\mathbf{nb}$	Set, list of node neighbors
μ	Friction coefficient
$\mathbf{\Sigma}_p$	Matrix, standart deviations for force application point estimation
$\mathbf{\Sigma}_\tau$	Matrix, standart deviations for torque measurments
δ_τ	Tolerance value for joint torques
δ_w	Tolerance value for squares residuals
t	Time
t_c	Current time
Δt	Sampling interval
$\mathbf{T}(\cdot)$	Homogeniuos translation matrix
$\mathbf{R}(\cdot)$	Homogeniuos rotation matrix
$\mathbf{S}(\cdot)$	Skew symmetric matrix
$(\cdot)^\#$	Pseudoinverse operation
$(\cdot)^T$	Transpose operation
$cov(\cdot)$	Covariance

GENERAL INTRODUCTION

Motivation

Human-robot interaction is one of the essential research topics in robotics with methods varying from visual and audio processing to direct physical contact identification. Although most of these methods were proposed decades ago, they gain more popularity in recent years due to the wide use of collaborative robots, which are usually equipped with force/torque sensors that allow them to detect physical interactions with an operator or environment [1], [2]. This type of robot is specially designed to be human-friendly and can be used for the creation of shared human-robot work cells, where human safety is a priority.

However, ensuring human safety is still a challenging issue, since most industrial robots could achieve rather high speeds and forces, which could be dangerous for a human [3], [4]. Although currently there exist a number of collision avoidance techniques, sometimes a collision between a human and a robot is unavoidable or even necessary for the manufacturing process. To ensure safety in practice, all physical interactions should be properly identified and handled by an appropriate robot reaction that can vary from a simple emergency stop to more complicated scenarios, like moving away from the obstacle or lowering robot joint stiffness. It is clear that to implement any of these scenarios, the human-robot interaction force should be well estimated, motivating the development of special techniques [5], [6].

Another important aspect, which arises during the physical interaction, is the performance of a human-robot work cell. Since collaborative robots are designed for safe human-robot interaction, their joints and/or links stiffness are essentially low. For these robots even small applied interaction forces can shift the robot tool trajectory from the desired path. Thus, to improve the robot accuracy and human safety during the physical interaction, the adaptive compliance behavior can be used, which changes the robot stiffness depending on the given task and identified interactions. It is clear that to implement any human-robot interaction scenarios or adaptive compliance behavior, the parameters of the human-robot interaction should be well estimated which motivates the development of special techniques presented in this thesis.

To identify the desired interaction force, different techniques can be applied, ranging from direct measurements to sophisticated numerical algorithms. In practice, different internal and external sensors could be used to extract parameters of interactions and

contact forces [7], [8]. In this work, we propose to use information from internal robot sensors only, i.e. to limit the measurement subsystem by one-axis joint torque sensors and angle encoders. By processing data from sensors, it is possible to extract the desired tactile information concerning the interaction, which includes the contact position on the robot surface and the interaction force amplitude and direction.

Most of the existing approaches for extracting the contact position and interaction parameters from the internal sensors are based on minimizing the squared residuals of static equilibrium equations, subject to some constraints arising from both the robot links geometry and a contact surface property [9]. For instance, the contact point should clearly be located on the robot link surface and the force direction be close to the surface normal. Corresponding numerical techniques for interaction parameters identification rely on straightforward nonlinear constrained optimization, which generally guarantees good accuracy, but provides rather low computation speed. The latter is caused by the iterative nature of the relevant optimization algorithms requiring numerous evaluations of the objective function for the interaction parameters search. So, in practice, it is required to find some reasonable compromise between the algorithm speed and accuracy. Other difficulties arising here are associated with an inability to handle interactions in singular or close-to-singular configurations, where multiple minima of the objective functions exist, and the interaction cannot be identified in a unique way. Thus, the existing techniques do not perfectly suit the requirements imposed by online implementation.

Principal goal and research problems

The *principal goal* of the thesis is the development of new techniques for modeling the physical interaction between a human and a robot, with adaptive compliance, in order to improve human safety and human-robot workcell performance. Special attention is paid to the accuracy issues and the interaction parameters identification in singular cases, which can arise during the physical interaction because of limited measurements information provided by robot torque sensors.

To achieve this goal, the following problems should be addressed:

Problem 1: Comparative study of the existing approaches for interaction parameters identification, which are used for the modeling of physical interaction between a robot and a human, in order to detect their weaknesses and propose appropriate enhancements.

Problem 2: Development of a new efficient method for the interaction parameters identification in the human-robot physical collaboration, which is capable of estimating the interaction force and its application point in real-time and for both singular and non-singular cases.

Problem 3: Development of the adaptive interaction handling strategy, which covers all phases of interaction starting from its initial detection, identification to classification, and reaction, allowing to change the robot behavior in order to improve the safety and efficiency of the collaborative work cell.

Problem 4: Experimental validation of the developed interaction identification method using a real collaborative robot and practical application of the proposed adaptive interaction handling strategy in a real-time environment.

Thesis structure

To present the main results and contributions of this thesis, it is divided in five chapters and is organized as follows.

Chapter 1. This chapter is devoted to the state-of-the-art and related literature review on collaborative robotics, human-robot collaboration, and modeling of physical interaction. Particular attention is paid to the basic rules of human safety during the collaboration process with a robot and the mathematical formalization of human-robot physical interaction. The main focus of this chapter is to review existing methods in modeling of physical interaction and highlight their advantages, disadvantages, and difficulties, which were not addressed yet and allow to define the principal goal.

Chapter 2. This chapter presents a comparative study of the existing approaches for interaction parameters identification, which are used for the modeling of physical interaction between a robot and a human, in order to detect their weaknesses and propose appropriate enhancements. In particular, it evaluates numerically and experimentally existing approaches for the interaction parameters identification that were implemented by the author of this thesis. In the frame of this comparison, three main criteria are used: accuracy, run-time, and algorithm ability to find multiple solutions.

Chapter 3. This chapter deals with interaction parameters identification for the 2D planar case. It proposes a new fast non-iterative technique for computing the interaction force and its application point using measurement data obtained from the internal joint torque sensors only. The proposed algorithm is based on a parametrized general solution of a static equilibrium equation allowing to find the desired parameters even in the case of essential ambiguity. In contrast to existing approaches, the proposed method is applicable for singular cases associated with an insufficient number of independent equations in a static equilibrium system producing non-unique parameters.

Chapter 4. This chapter deals with interaction parameters identification for the spatial case and presents techniques allowing to estimate the force and its application point in human-robot physical interaction. It is assumed that the desired parameters are estimated using data obtained from torque sensors embedded in the robot joints.

In practice, the measurement data are corrupted by noise, which causes identification errors. Based on the relevant analysis, enhancements were proposed, which improve the identification accuracy and robustness with respect to the measurement noise.

Chapter 5. This chapter focuses on the development of an adaptive technique for handling physical interactions between a human and a robot, as well as experimental validation of the developed identification algorithms. The main goals of this technique are to ensure the safety of a human operator and improve the performance of human-robot collaboration by implementing various scenarios. In the scope of this technique, the interaction parameters identification methods proposed in the previous chapter are used for the selection of an appropriate reaction strategy. These parameters define the interaction force and its application point on the robot surface that are used for interaction classification within the set of predefined categories. Based on these categories and the current robot state, the algorithm chooses an appropriate robot reaction. At the end of this chapter, the experimental results are presented that are obtained for the real collaborative robot KUKA LBR iiwa and validate the developed techniques.

Finally, *Conclusion* summarizes the main contributions of the thesis and defines perspectives for future research work.

Main theoretical contributions

Theoretical results presented in this work are in the area of the modeling of human-robot physical interactions. Among them, there are three contributions that can be treated as the most essential ones. Their brief summary is presented below.

1) The method for *interaction parameters identification*, which allows to find interaction force and its application point in both *singular* and *non-singular cases* using the joint torque measurements only.
2) The method for *accuracy improvement* of the interaction parameters identification and ambiguity resolution, which allows to reduce the parameters deviation caused by the measurements noise.
3) The method for *adaptive interaction handling*, which covers all possible interaction types and allows to adapt the robot behavior in order to improve the safety and efficiency of the collaborative workcell.

Novelty of the research results

The thesis contributes to the enhancement of numerical and analytical methods for the modeling of human-robot physical interaction, including accuracy improvement of interaction parameters identification in singular cases. In contrast to previous works in this area, the proposed methods:

- Can be used to identify the interaction parameters in singular cases. The previous works considered non-singular cases only, so the known techniques cannot be applied to arbitrary robot configurations, interaction force directions, and force application points.
- Can produce all possible solutions for the interaction force and its application point in case of ambiguity of relevant equations. In the previous works, only a single solution for the interaction force and its application point was provided, which is not necessarily a correct one.
- Allow to estimate robustness and identification accuracy with respect to the measurement noise. In previous works, this issue was out of the authors' attention. Besides, in contrast to the common practice, it was assumed that the noise is presented as unbiased and independent but not identically distributed random variables, which is more close to real-life applications.
- Provide high computational speed for the interaction parameters identification due to the non-iterative nature of relevant numerical algorithms, which allows efficient implementation in real-time.

Theoretical and practical significance

This thesis presents theoretical contributions in the field of human-robot physical interaction modeling and the identification of its parameters. In particular, it allows to estimate the interaction parameters (the interaction force and its application point) in the singular cases, which mathematically are related to the insufficient number of the static equilibrium equations. Consequently, the desired force and its application point should be found from an undetermined system with an infinite number of solutions. To resolve this ambiguity, it is proposed to apply relevant geometric constraints describing only physically possible interaction forces, which allows to reduce essentially the set of possible solutions in an analytical way. Besides, for both singular and non-singular cases, the obtained solutions are presented in closed form, while the previous techniques are based on numerical optimization, which does not provide the desired computational speed.

First of all, the practical contributions of the thesis include improved human safety in the presence of a collaborative robot. The developed interaction identification techniques allow to continuously monitor the interaction force vector amplitude for violating the safe interaction values. If it is violated, the adaptive interaction handling algorithm can change the robot behavior to prevent injury, by utilizing different scenarios and dynamical switching between them. In addition to the force vector amplitude, its application point can be used to further improve human safety by limiting robot movements in order to avoid human "jamming" between the robot links. Moreover,

the real-time capabilities of the developed techniques allow to almost instantly react to any potentially unsafe situations.

Second, practical contributions can be used to improve the efficiency of the human-robot work cell. On one hand, the identified interaction parameters can be used for communicating between the human and the robot by enabling special collaboration scenarios, depending on how and where the human touches the robot. On the other hand, the same parameters can be used for the compensation of compliance errors, which can increase the accuracy of the robot in the presence of the interaction force, improving the overall performance of the work cell. In such a way, the developed methods could not only improve the safety of a human in a robot workspace, but also increase the performance of the work cell by activating different collaboration scenarios.

Finally, practical contributions can be used in collaborative robot design. The relevant identification accuracy analysis can be used as a methodology for robot design, by providing the expected identification accuracy based on used sensor parameters and their locations.

Research methods

In this dissertation, various methods were used including the theory of kinematic and dynamic modeling of robotics systems, numerical optimization, as well as physical interaction modeling. Besides, artificial neural networks were applied to the interaction classification as well as to the identification of its parameters. For the interaction reaction module, the finite state machine was used. All models and algorithms were implemented in MATLAB and Python environments and tested on real hardware.

Validity of the obtained results

The validity of the main results presented in the thesis is confirmed by simulation results and dedicated experimental studies, which were conducted for industrial collaborative robots KUKA LBR iiwa 14 and Universal Robotics UR10e; as well as by correct use of related mathematical apparatus and by publications in peer-reviewed journals and conference proceedings.

Dissemination of the research results

The results of the thesis were presented to the scientific community and approved at 12 conferences, including: eight IEEE conferences such as International Conference on Intelligent Robots and Systems (IROS 2021, 2022), International Conference on Automation Science and Engineering (CASE 2021), International Conference on Robot and Human Interactive Communication (RO-MAN 2017), International Conference on Mechatronics (ICM 2019). They were also discussed at two IFAC conferences on Manufacturing Modelling, Management and Control (MIM 2022, 2019) and others.

Publications

The main results obtained in this thesis have been published in 14 papers and were presented at 12 conferences. Among them, there are 3 journal papers (IEEE Robotics and Automation Letters; Mechatronics, Automation, Control), 2 journal papers under review (Robotics and Computer-Integrated Manufacturing; Engineering Applications of Artificial Intelligence) and 10 conference papers indexed by Web of Science / Scopus: IEEE International Conference on Intelligent Robots and Systems; IEEE International Conference on Automation Science and Engineering; IEEE International Symposium on Robot and Human Interactive Communication; IEEE International Conference on Mechatronics; IFAC Conference on Manufacturing Modelling, and others.

Personal contribution of the author

The main results presented in this thesis are obtained by the author. In particular, the author was directly involved in all stages of research, including the comparative analysis, development of the new methods, algorithms for interaction identification, as well as their experimental validations and practical applications of the final results.

Chapter 1

Modeling of physical interaction for collaborative robots

Contents

This chapter is devoted to the state-of-the-art and related literature review on collaborative robotics, human-robot collaboration, and modeling of physical interaction. Particular attention is paid to the basic rules of human safety during the collaboration process with a robot and the mathematical formalization of human-robot physical interaction. The main focus of this chapter is to review existing methods and tools in modeling of physical interaction and highlight their advantages, disadvantages, and difficulties, which were not addressed yet, and allow us to define the principal goal of this thesis.

1.1 Principles of safe and efficient human-robot interaction

Nowadays, industrial robots are one of the most important parts of the automation process. They are highly versatile and found their place in a large spectrum of practical applications starting from assembling to welding and machining. However, with the increased complexity of manufacturing, some of the processes require complex cognitive skills unavailable for classical industrial robots. To address this issue, a collaborative work cell, consisting of a human and a robot was proposed. The latter obviously introduced additional requirements to the robot in terms of safety. This section briefly overviews the advantages and disadvantages of collaborative robots over classical ones as well as the safety aspects of the corresponding ISO standards for human-robot collaboration.

1.1.1 Collaborative robots in industry

Historically, humans always seek the tools and methods to extend their capabilities in terms of manufacturing. The invention of the manufactory and then the conveyor line allowed us to significantly boost our labor productivity previously unavailable to artisans. However, this industrialization process forced humans to work in a very unpleasant environment, including but not limited to doing a lot of repetitive and tedious tasks in dangerous and sometimes dirty workspaces. At some point, the advances in automation allowed us to replace humans in such places with robots. A typical example of such robots utilized in manufacturing can be found in the automotive industry, where industrial manipulators are utilized in assembly lines. Obviously, not all of the processes can be easily solved by industrial robots, sometimes human presence or even direct intervention is necessary. For such cases, a different type of industrial robot was proposed: collaborative robot [10]. In a broad sense, by the name of collaborative robots can be called any robot which is capable of safely working in a shared environment with humans including, for example, wearable robots like exoskeletons. Nonetheless, in this work, only collaborative manipulators will be considered that are similar to widely used industrial manipulators and can functionally resemble a human arm. The main difference between the collaborative and the typical industrial manipulator or robot is the availability of additional sensors and software, which can ensure the safety of a human in the proximity of a robot.

Safety improvement is not the only advantage of collaborative robots over the classical industrial ones. The ability to work safely in the shared workspace also allows for a human and a robot to work on the same task. On one hand, industrial robots

(a) Collaborative robot KUKA iiwa (b) Industrial robot KUKA Agilus

Figure 1.1: Typical examples of collaborative and classical industrial robots in manufacturing. Compared to the collaborative robot, the industrial one is enclosed in a safety perimeter preventing it from physical interaction with a human. ©KUKA AG(kuka.com)

are known for their high accuracy, repeatability, and speed, which are unmatched by humans, while they are only able to perform strictly defined preprogrammed tasks. Humans, on the other hand, can easily adapt to some unforeseen events and have unique cognitive skills required for some complex tasks. However, their motor skills are limited and depend on fatigue. By combining both of them, the human and the collaborative robot for the same task we can benefit from their strongest sides, essentially improving the overall efficiency of the production line [2].

It is clear that if collaborative robots are safe and can potentially work with humans on the same task, the whole production line working area can be essentially changed. The safety requirement on the typical industrial robot demand it to be completely separated from humans, by using different proximity sensors and safety fences. Since the reachable workspace of the robot can be quite large, all its borders should be guarded by such fences as shown in Fig. 1.1b. The latter increases the work-cell footprint which also increases costs. With collaborative robots, the safety perimeter is not required and humans can easily share the workspace with the robot because of its internal safety features (see Fig. 1.1a). Thus, the utilization of collaborative robots over industrial ones can benefit from more compact and flexible production lines.

In addition, compared to industrial robots, collaborative robots can be programmed in a more intuitive and human-friendly way. The classic approach for programming a robot arm is similar to traditional software programming which operates with several trajectory waypoints presented in some coordinate frame. The use of various simulation software is usually needed to understand how the robot will move in real life before running the code on the robot. All these obviously require special skills and education. Collaborative robots, on contrary, can be programmed by demonstration or walk-through by an operator without special training. Even more, some recent advances presented the robot programming process done with augmented or mixed reality

Table 1.1: Popular models of commercially available collaborative robots

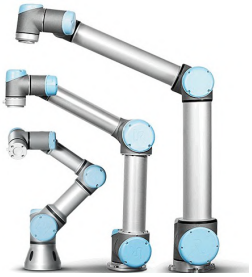
	Robot	**KUKA LBR iiwa**
	DoF	7
	Payload	7/14 kg
	Repeatability	$\pm$0.05 mm
	Weight	22/30 kg
	Max velocity	2000 m/s
	Force sensing	Joint torque sensors
	Robot	**Universal Robots UR 3e/5e/10e**
	DoF	6
	Payload	3/5/10 kg
	Repeatability	$\pm$0.05 mm
	Weight	11/19/30 kg
	Max velocity	1000 m/s
	Force sensing	6 DoF Force/torque in EE + current-based estimation of joint torques
	Robot	**Franka Emika Panda**
	DoF	7
	Payload	3 kg
	Repeatability	$\pm$0.1 mm
	Weight	18 kg
	Max velocity	2000 m/s
	Force sensing	Joint torque sensors
	Robot	**Kinova Jaco2**
	DoF	6/7
	Payload	2.5 kg
	Repeatability	$\pm$0.15 mm
	Weight	5 kg
	Max velocity	2000 m/s
	Force sensing	Joint torque sensors
	Robot	**ABB YuMi IRB 14000**
	DoF	7+7
	Payload	0.5 kg
	Repeatability	$\pm$0.02 mm
	Weight	38 kg
	Max velocity	1500 m/s
	Force sensing	Current-based estimation of joint torques with observers

Table 1.2: Comparison of typical classical industrial and collaborative robots

Classical industrial robots	**Collaborative robots**
High accuracy	Mediocre accuracy
Very high operational speed	Low operational speed
High repeatability	Mediocre repeatability
High load capacity up to several tons	Very low load capacity, 35 kg max
Only joint position sensors	Additional force/torque sensing
No safety features	Embedded emergency routines
Separated workspace with human	Shared workspace with human
Fixed installation	Fixed or mobile installation
Heavy weighted	Light weighted
Offline programming	Offline and online programming
Interaction with operator forbidden	Frequent interactions with operator

tools, which help simplify and visualize this task. Consequently, collaborative robots allow them to program online or introduce some quick adjustments of the trajectory on the fly, whereas industrial robots are mostly limited to offline approaches.

Despite the before-mentioned advantages, collaborative robots currently are not so popular in the industry. The first reason for that is their price, which is several times greater than that of an analogous industrial robot. The second reason is their mediocre characteristics in terms of accuracy, repeatability, and load, which arise due to the implementation of all safety features in both, robot software and hardware. The general summary and comparison between collaborative robots and classical industrial ones are presented in Table 1.2.

Typical examples of collaborative robots, popular in the industry and research with their technical characteristics are presented in Table 1.1. As follows from this table, collaborative manipulators usually have a redundant number of degrees of freedom, which makes them more flexible. By looking at their values of the maximum linear velocity of the end-effector, and weight, it can be assumed that impact force during accidental contact with a human can be harmful. The latter justifies the high requirements for the safety algorithms during the direct physical interaction with the operator. In addition to typical characteristics that are common for all manipulators, the collaborative ones also have force-sensing capabilities. Depending on the robot design, most of the manufacturers choose direct torque estimation in the robot joints axis or estimation of torque based on the secondary values like for example motor current. For this reason, in this work, a typical setup for the collaborative robot with joint torque sensors will be considered.

Thus, collaborative robots provide greater flexibility and safety than classical industrial robots. Their main feature is the ability to safely interact with a human, which can be used to improve human-robot work-cell efficiency. In this work, the human-robot physical interaction will be the focus to further improve the ability of such a collaborative unit.

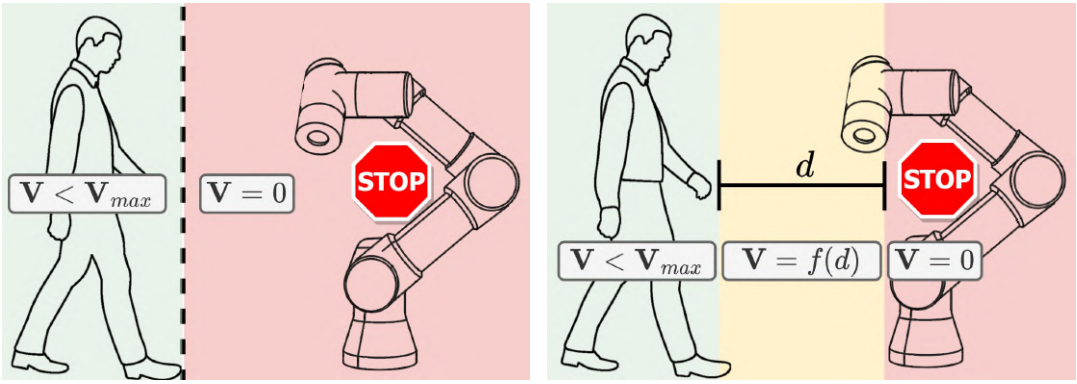
1.1.2 Human-robot interaction modes and human safety

Recently, different steps were taken to somehow categorize the human-robot collaboration scenarios into levels. Each level basically refers to how much of the collaboration is in the human-robot collaboration. Consequently, the safety requirements can be different for each level, with its maximum at full collaboration. Despite some of the inconsistency in the applied terminology [11]–[13], three main groups can be formulated: coexistence, cooperation, and collaboration.

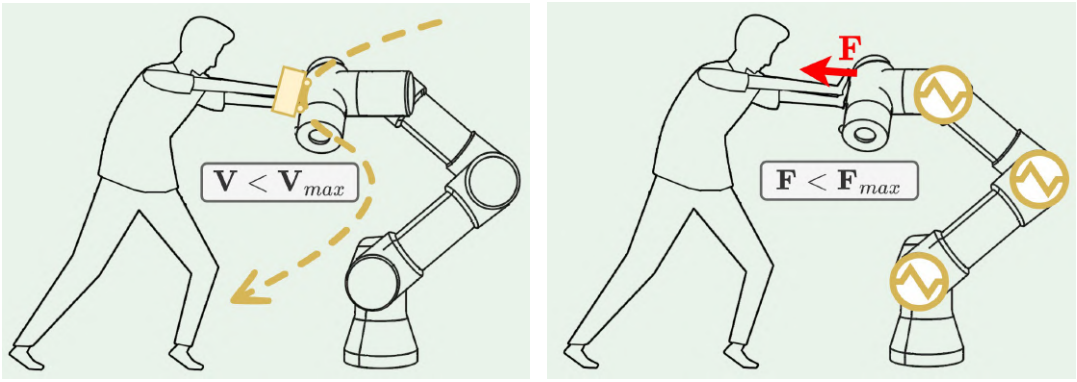
- The basic level of human-robot collaboration is "Coexistence". At this level a human and a robot can share the same workspace, however, both of them execute different tasks. There is no direct physical interaction between them except for accidental or emergency contact.
- The next level of collaboration is "Cooperation". At this level, a human and a robot share the same workspace and the task but they do not work on the task at the same time. The task itself does not require human assistance, however, occasional interventions are possible. So, this level assumes accidental and purposeful interaction between a human and a robot.
- "Collaboration" is the most advanced level of interaction. At this level, a human and a robot share the same workspace and execute the task simultaneously. It also assumes accidental and purposeful interaction between a human and a robot but compared to cooperation, the interaction includes regular, frequent, and active task sharing.

The safety standards for collaborative robots and human-robot interaction is regulated by the family of ISO 10218 standards. It is a dedicated Type C standard, which regulates the safety of industrial robots in general with its two parts, where its first part ISO 10218-1 [14] deals with requirements for the robot and its controller, and the second part ISO 10218-2 [15] sets the requirements for robot and its additional devices. Here the first one is intended to be used for robot manufacturers while the second one is dedicated to system integrators. The additional technical specification ISO TS 15066 [16] provides additional information and guidance on collaborative robot operations.

The standards explicitly specify only four collaborative modes that can be used under one of the before-mentioned levels. Fig. 1.2 presents an illustrative summary.



(a) Safety-rated Monitored Stop (b) Speed and Separation Monitoring



(c) Hand Guiding (d) Power and Force Limiting

Figure 1.2: Human-robot collaboration modes defined by ISO 10218 standard [14], [15].

The first collaborative mode is "Safety-rated Monitored Stop"(Fig. 1.2a). This is the simplest mode that refers to a more classical approach in industrial robotics where the robot completely stops when the human enters the safety perimeter. In the case of a collaborative robot, this perimeter could be virtual, e.g. using cameras, lidars, or other non-contact detection methods. Here, a robot and a human can work on the same task at the same workspace but not at the same time, similar to "coexistence".

The second mode is "Speed and Separation Monitoring"(Fig. 1.2b). In this mode, the robot's speed is calculated based on the distance to a human. When a human is outside of the robot workspace, the robot can operate at the full speed but decrease in proximity up to a complete stop when a collision is inevitable. Obviously, implementation of this mode requires continuous monitoring of the robot workspace using some additional sensors and prediction of the human/robot movement to ensure safety including transient processes before the robot completely stops.

The third mode is "Hand Guiding"(Fig. 1.2c), which enables one of the most intuitive solutions for collaborative robot programming. In this mode, the operator grabs the robot by the special tool and guide it through the set of waypoint required for the task. Once the human release the tool robot switch to the 'Speed and Separation Mon-

itoring" mode. Although this mode relies on the direct physical interaction between a human and a robot this case can not be considered at the "collaboration" level.

The last, fourth mode is "Power and Force Limiting"(Fig. 1.2d). The main idea of this mode is to limit the possible interaction forces to some safe values. The exact values of contact forces, for static and transient types of contact, are described in ISO TS 15066 which also presents the admissible pressures and forces for 29 areas of the human body. This mode is the only one that satisfies the "collaboration" level of interaction and ensures that any purposeful or accidental contact will prevent any harm to a human.

Indeed, the described modes can be used to implement a few basic robot reactions to accidental or purposeful interaction with a human. The most simple one is the robot stop when certain monitoring signal conditions are met. The more complex behavior is robot avoidance where the robot tries to move away from the object with which it has collided. And the most advanced compliance robot behavior resembles the spring-damper system in each of the robot joints, effectively limiting the forces.

In this work, it is proposed to use so-called "Adaptive compliance" to actively change the robot's behavior based on the interaction properties. It is clear that in practice the robot must be safe for a human, although, at the same time, task requirements in terms of path tracking and accuracy should be also satisfied. Human safety during the collaboration is dictated by "Power and Force Limiting" mode, which requires the robot to be compliant with low stiffness values in its joints, including some post-collision reactions as was studied in [5]. The latter obviously contradicts with accurate trajectory execution that is better when the robot is perfectly stiff. So, adaptive compliance allows to actively change the robot stiffness values based on the current task and the presence or absence of interaction.

It is worth mentioning that the proposed "Adaptive compliance" technique should not be confused with low-level adaptive compliance control law. Despite a similar name and goal, "Adaptive compliance" in this work corresponds to more high-level control that uses advanced interaction properties as well as the current robot state and its task to choose the robot behavior. Most of the collaborative robots manufacturers do not allow the user to interfere with low-level robot control, so the high-level "Adaptive compliance" technique only uses robot control methods, which are expected to be used by the manufacturer. By working as an Add-on, it also allows to use all internal safety features of the robot and does not violate its certification.

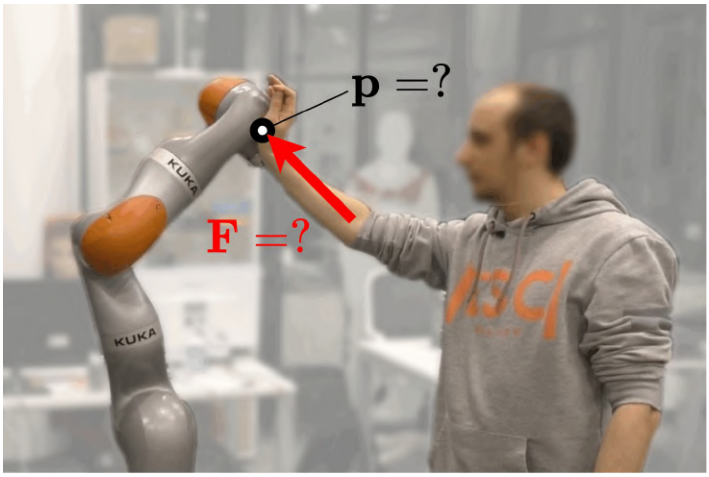


Figure 1.3: Identification of physical human-robot interaction parameters: the contact point **p** coordinates and interaction force vector **F** are unknown.

1.2 State of the art in modeling of physical human-robot interaction

In collaborative robotics, the safety of a human and the environment is the highest priority. To ensure this, a multi-stage handling procedure is required, which is based on the model of physical interaction. Indeed, such a procedure implies the presence of special sensors that can directly or indirectly measure some parameters of physical interaction, additional processing that can extract the force and other interaction parameters based on its model, and finally software routines and control mechanisms that can alter the robot behavior. This section presents the interaction handling procedure and a review of hardware and software approaches used for the modeling of physical human-robot interaction.

1.2.1 Generalized presentation of interaction handling procedure

Generally, the problem of collision event handling in collaborative robotics is treated in several steps, which include collision detection, isolation, identification, classification, and reaction. In addition, there are also some post and pre-collision steps that deal with collision avoidance and post-collision recovery. At the first of these stages, the presence of the interaction force acting on the robot is detected. In the second stage the force application point is estimated; this can be done with different precision levels: from determining a colliding link number to the exact point location on the link surface (see Fig. 1.3). In the third stage, the force amplitude and direction are computed, assuming

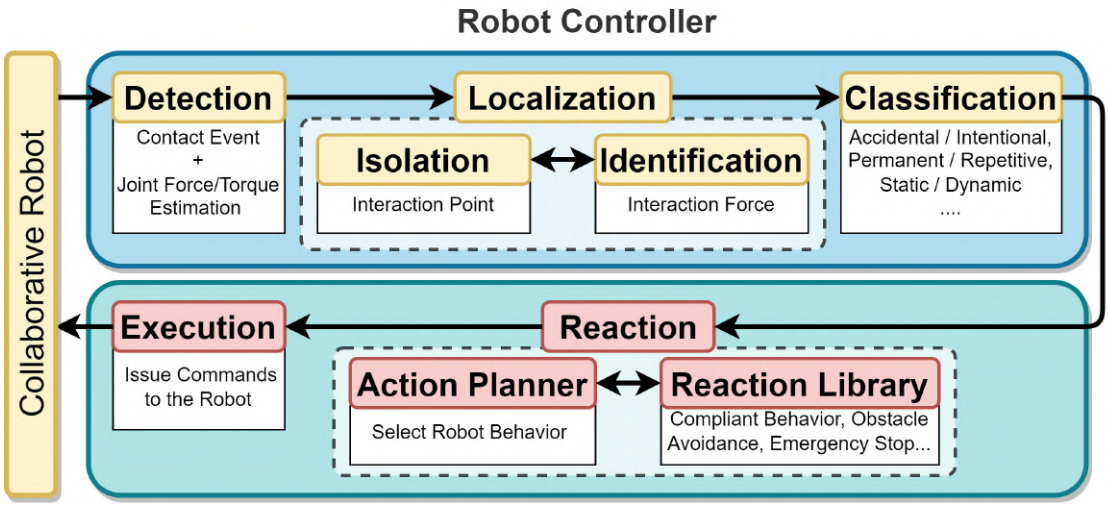
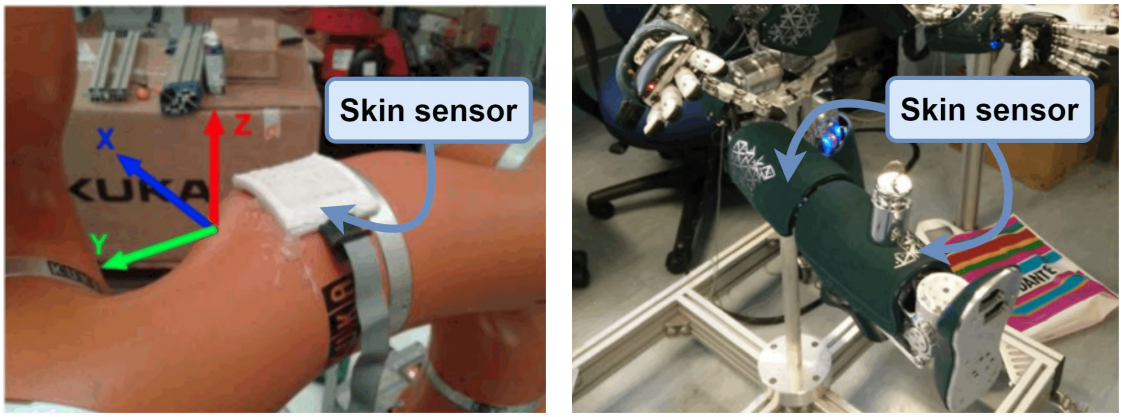


Figure 1.4: Basic steps of the interaction handling in human-robot collaboration.

that the force is applied to the isolated point found in the previous stage. Further, at the collision classification stage, some additional properties of the interaction are evaluated, such as accidental or intentional, permanent or repetitive, static or dynamic, etc. And finally, at the reaction stage, an appropriate robot behavior is selected based on the interaction force properties in order to provide safe and efficient human-robot collaboration. More details concerning the above-listed stages can be found in the works of A. De Luca, S. Haddadin, A. Albu-Schaffer and A. Kheddar [9], [17]–[20]. A typical example of a such handling procedure is presented in Fig. 1.4.

Obviously, before any physical interaction identification, the fact that interaction is present should be known by the appropriate detection algorithms. The main demands for these algorithms are response time (that passes from the interaction event started to the moment when the algorithm detects it), which should be as small as possible, and detection precision that minimizes the number of false-positive classifications. Since it is a binary problem, most of the methods for collision detection use a comparison of the input signals with some threshold. The simplest way is to monitor changes in motor current or torque values [21], [22]. These methods do not require a robot model. Another approach is based on the nominal robot model and comparing the model to the actual joint torques. This technique is similar to direct torque estimation or inverse dynamics approach [5]. The interaction detection based on deep neural networks was proposed in [23], which showed a reliable detection for cyclic operations. More complicated algorithms allow not only to detect of a collision but also to give an estimation of external torque (external torque here is referring to the part of the joint torque which was caused by the external force in the contact event), and they are based on disturbance observers.



(a) Skin sensor for KUKA LWR robot [31] (b) Skin sensor in iCub robot [32]

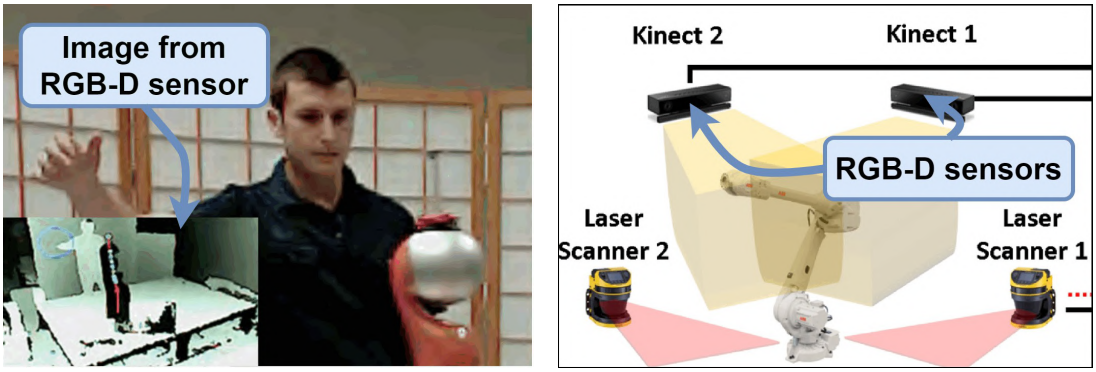
Figure 1.5: Examples of tactile skin sensors used in collaborative robotics. ©2015,2018 IEEE

By evaluating the characteristics of the collision it is possible to attribute it to one of the predefined classes. This allows us to understand the context of the contact event and make a decision for the reaction. In [24], [25] contacts with a human are divided into intentional and unintentional. Parusel et al. [26] designed a control architecture for realizing human-friendly behaviors and intuitive state-based programming with four control modes. The first mode is autonomous in case of human absence, the second mode is compliance with human presence, the third is a collaborative mode with a human in the loop and the last mode is the case of a fault. Magrini and De Luca in their work [27] implemented a finite state machine with 3 basic robot states: idle state, null space redundancy state, and high compliance mode.

It is worth mentioning that this work concentrates on the interaction isolation and identification steps which are combined in a single localization/identification step. The main reason for such merging is that the procedure of isolation/identification is very time-consuming. Besides, it is rather difficult to separate these stages in computations. It should be stressed that in spite of numerous works in this area [9], [28]–[30] its solution (i.e., computing the interaction force amplitude, direction, and application point) is still challenging. One of the related difficulties is the presence of multiple minima for the used objective function, which will be carefully considered below.

1.2.2 Sensors and observers used for human-robot interaction

In practice, performance of the localization algorithms essentially depends on the set of available sensors providing data for the interaction detection and evaluation. It is clear that the localization problem could be significantly simplified by the right choice of sensors. Currently, a large variety of sensors are used in collaborative robotics. The most attractive of them are the tactile skin sensors [31]–[33], which directly yield



(a) Setup with Kinect sensor [36] (b) Setup with Kinects and lidars [35]

Figure 1.6: Examples of RGB-D sensors setup for human-robot collaboration. ©2012 IEEE

both the interaction force and its application point, making the localization step in contact handling trivial (see Fig. 1.5). This type of sensor can be based on different physical phenomena that can vary from magnetic to light-sensing [34]. However, most of them have the same principal drawbacks: their size, bulkiness, price, and calibration difficulty, which outperform their advantages. Another popular type of external sensor is vision/depth cameras [35]–[38], which can be useful for isolation only and in the absence of occlusions (see Fig. 1.6). It is also worth mentioning some quite uncommon sensors like accelerometers or sound sensors, which are used for the collision handling as well [39]–[41].

In industrial robotics, the most popular sensors for the contact localization are the force/torque measurement devices, which are divided into external and internal ones. The external six-axis force/torque sensors could be integrated into either the robot end-effector or the robot base. They provide a complete wrench (force and torque) describing robot interaction with the environment. For the end-effector mounted sensors, the interaction between the robot tool and the environment could be detected or identified directly. But it is clear that collisions in the intermediate points (between the robot base and end-effector) are undetectable in this case, thus making this type of sensor unattractive for a shared human-robot work cell [42], [43]. In contrast, for the base-mounted sensors, the desired interaction parameters can be easily found assuming that the torque is equal to zero at the contact point [44], [45]. However, there is an essential difficulty here related to the separation of the sensed force and torque into two components: generated by the robot dynamics and caused by the external force.

In spite of obvious advantages, the external six-axis sensors are rarely used in the collaborative robotics. In most practical applications, only internal robot sensors are used. Currently, most of robot manufactures prefer one degree-of-freedom torque sensors embedded in the manipulator joints, for example: KUKA LBR iiwa, Kinova Jaco,

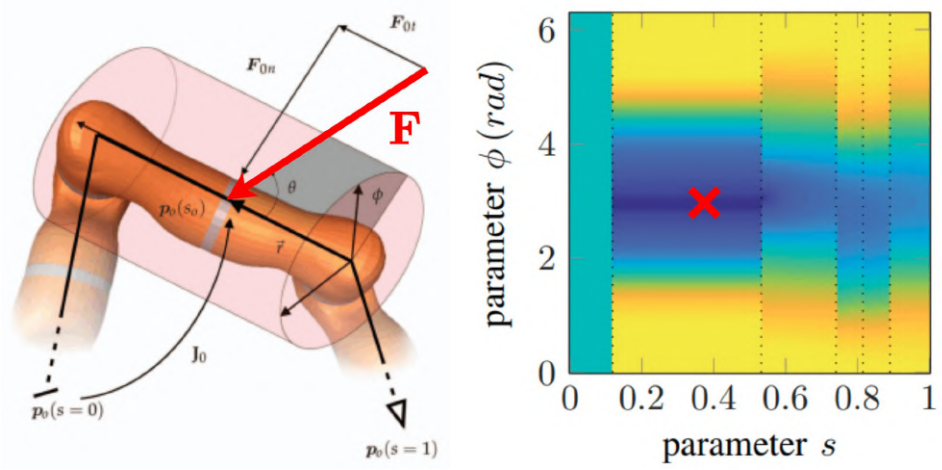
and Franka Emika Panda [46] as was shown in Table 1.1. Such design obviously reduces the robot cost, but it creates some complexity for the interaction force estimation and localization. In particular, here the identification algorithm efficiency depends essentially on the robot kinematical configuration and the interaction force direction. Moreover, there exist an infinite number of force/configuration combinations for which the interaction parameters are undetectable. This issue will be discussed in detail in the following chapters.

It should be noted that the identification of human-robot interaction considered in this work requires some preprocessing of the measurement data obtained from the manipulator joint sensors. In particular, it is necessary to eliminate force/torque components generated by the robot dynamics and obtain pure values caused by the interaction only. It is clear that direct estimation from the robot dynamic model is impractical here due to the necessity of the second-order numerical differentiation of the robot generalized coordinates. For this reason, state observers are usually used to estimate the dynamic force/torque components in the measurement data. For example, in the work [47] a momentum observer was used, which does not require the inertia matrix inversion as well as joint accelerations computing. This algorithm was later improved in [48] by using a second-order sliding mode technique, which ensures better response characteristics and lower noise sensitivity. Other relevant algorithms based on the state observers for the considered application could be found in [49]–[54]. All of them allow to obtain the joint forces/torques estimates that further are used for the interaction parameter identification, but their efficiency depends on particular applications.

Currently, the momentum observer proposed by De Luca [55] is one of the most popular solutions for the estimation of external load, not only for the robotics arm but also for other kinds of robotic systems like drones [56]. The momentum observer does not require inertia matrix inversion and joint acceleration values, but at the same time provides a first-order filtered version of the external torques and, as a result, has asymptotic convergence for the constant external load only. A survey on the detection and estimation methods can be found in [9].

1.2.3 Current approaches for identification of interaction parameters

The most difficult stage in modeling of human-robot interaction is its localization/identification, at which it is necessary to find the interaction force application point as well as its amplitude and direction. It is clear that the desired solution must satisfy both the manipulator static equilibrium equations as well as some geometric constraints describing the manipulator shape and admissible force directions. At present there exist



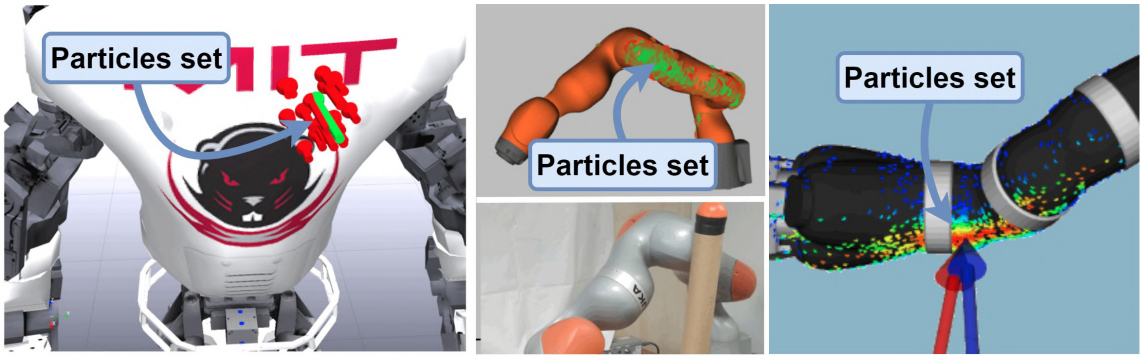
(a) Cylindrical model [28] (b) Relevant objective function [57]

Figure 1.7: Interaction parameters identification using a simplified cylindrical approximation of robot link surfaces and straightforward optimization in 2D space. ©2018 IEEE

a number of techniques allowing to solve this problem. Most of them transform the problem into constraint minimization of the residuals derived from the static equilibrium equations. The common practice is to apply general iterative search algorithms both gradient and non-gradient ones. However, this straightforward approach requires rather high computational resources. Besides, the presence of multiple minima of the objective function can hardly be resolved here.

To speed up computations, several improvements of this technique were proposed based on different approximations of the robot surface. In particular, Likar et. al. used a cylindrical approximation and assumed that the interaction force direction is strictly normal to the robot surface [28]. The latter allowed to simplify computations essentially and reduce the human-robot interaction identification to a rather fast two-dimensional search. A similar idea later was also used by other authors [57], [58]. It was assumed that this direction is restricted by the so-called friction cone created around the surface normal, which adds more physical meaning in the considered model as shown in Fig. 1.7. However, in spite of obvious advantages, these approximation-based techniques do not ensure accuracy, which is required by many practical applications, especially for human safety.

The considered problem can be also solved by means of different Monte Carlo search techniques, in which potential solutions are presented as a set of samples. One of such methods, a particle filter, is widely used in robotics for autonomous vehicle localization. For identification of human-robot interaction, this technique was first used by Manulli and Tedrake [29], where the force application point was described by a set of particles, which are moved randomly on the robot surface (motion model).

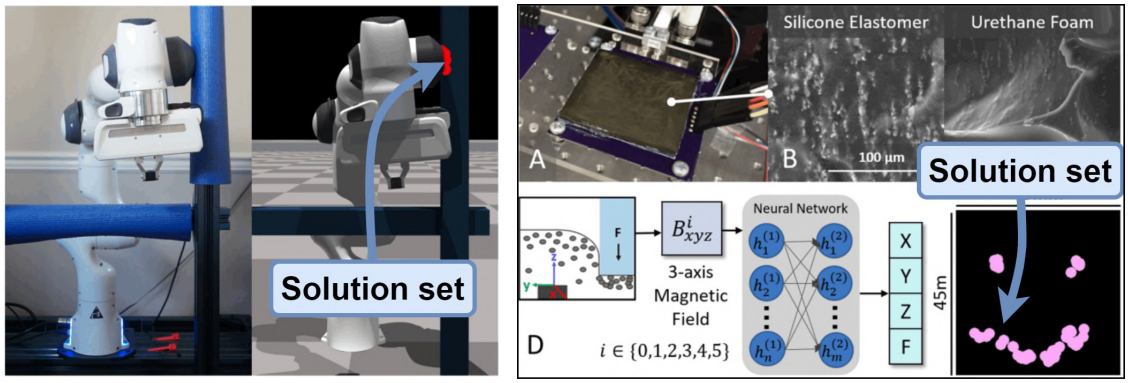


(a) Method from [29] (b) Method from [59] (c) Method from [30]

Figure 1.8: Interaction parameters identification using Monte Carlo-based approaches.

For each of these particles, the probability of the interaction is computed using the residual between measured and modeled joint forces/torques, that could be created by the interaction force in the corresponding surface point (measurement model). In the beginning, the particle locations are generated randomly. Further, the new particle locations are obtained via resampling based on the current probability distribution. The essential advantage of this technique is the ability to identify multiple interactions when several particle sets are running in parallel. However, despite the high accuracy of this method, its runtime is very high, mostly due to the complexity of the resampling, where new particle positions are projected onto the robot surface. Besides, the interaction force estimation step is also computationally difficult here since it requires solving the quadratic programming problem for each particle.

Several authors concentrated on improvements of particle filter-based algorithms. For instance, Bimbo et al. proposed to use small exploratory robot movements to increase the identification accuracy and improve interaction identifiability [59]. Some other enhancements were proposed by Zwiener who tested three different methods for contact location extraction from the particle set [30]. The main idea of these methods was to present particle sets as clusters with their means and then select the contact point as the closest one between the cluster means and a set of predefined surface points. This allowed to simplify resampling but increase the setup time by including additional experimental steps required for training. Typical outputs of such algorithms are presented in Fig. 1.8, which also shows the positions of particles on the robot surface. Summarizing the advantages and disadvantages of the Monte Carlo-based techniques employed in the related works, it is worth mentioning that they provide a rather high accuracy, but their computational-intensive nature is limiting them to be utilized in real collaborative applications. Besides, such algorithms always converge to a single solution, while in some singular cases, the solution is not unique.

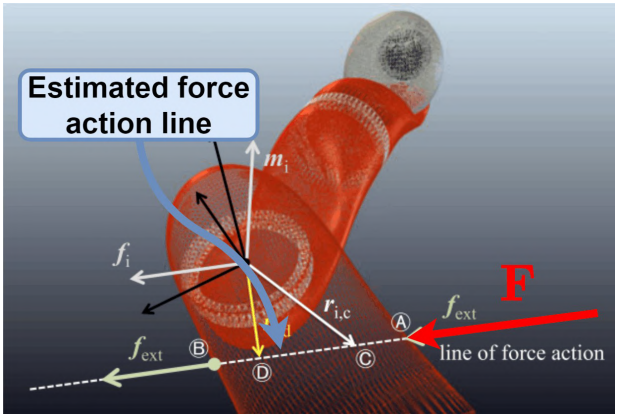


(a) Identification using DL method [61] (b) Identification using NN method [62]

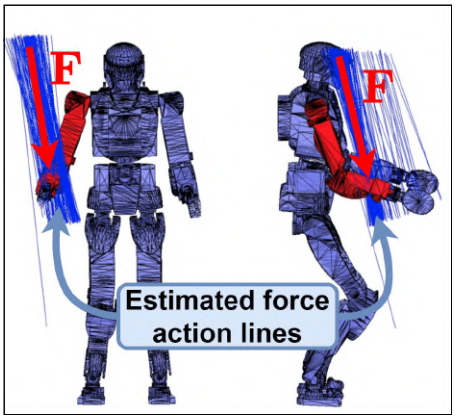
Figure 1.9: AI-based approaches for interaction parameters identification. ©2021,2020 IEEE

As follows from the dedicated analysis, the best identification efficiency of the human-robot interaction is provided by custom (specially designed) algorithms, which take into account all particularities of the considered problem. Another technique capable of handling multiple minima problem was developed by Pang et al., who used a combination of rejecting sampling and gradient descent [60]. However, the algorithm running time was rather high and the reported frequency is about 10 times per second. In general, in spite of some successful case studies, the existing algorithms are hard to apply in practice because of their iterative nature resulting in a variable running time.

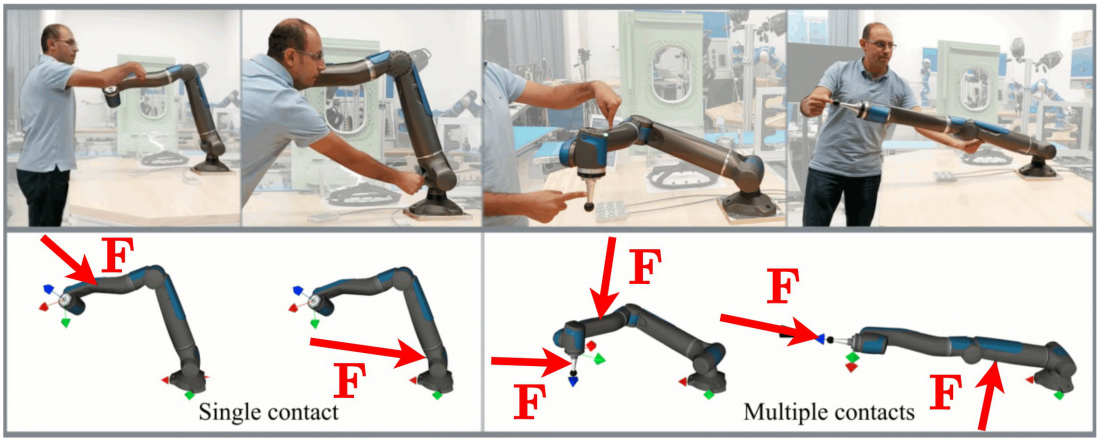
Recently, AI-based approaches became also popular for the identification of human-robot interaction. A typical application here is interaction detection, where the binary output (contact/no contact) is provided by analyzing internal sensors data using machine learning [23], [63], [64]. The full-scale identification, which includes the estimation of the interaction force amplitude/direction and application point location, machine learning was applied in [24], [62], [65], [66]. The authors used this technique for data fusion from internal and external devices such as joint torque gauges, inertia sensors, tactile sensors, etc. For the case of joint torque sensors only, various neural networks were successfully applied for interaction point localization using a cylindrical approximation of the robot surface [57]. Another approach was used by several other authors who presented the human-robot interaction problem as a classification task, where the force application point was selected from a limited number of predefined locations on the robot surface [30], [57], [61], [67]. Some graphical examples are presented in Fig. 1.9. In general, the main difficulty in the application of the above techniques is the necessity of a learning step, which is rather time-consuming and must be executed in advance. It is worth mentioning that this step could be partially reduced by using transfer learning techniques, where training on the real robot is combined with the simulation. Another point that complicates the application of these methods is the



(a) Interaction with KUKA LWR robot [9]



(b) Interaction with humanoid [68]



(c) Interaction with DLR SARA robot [69]

Figure 1.10: Examples of case studies that use analytical approaches for interaction parameters identification. ©2017,2019,2021 IEEE

ambiguity of the provided solution in some singular cases not included in the training sets, which is unacceptable for safe human-robot collaboration.

In spite of the popularity of the above-presented techniques, they are not well suited for the real-time requirements of industrial applications. For this reason, non-iterative techniques based on partial closed-form solutions are the focus of many current research. Within these methods, the problem is also solved in two steps, similar to the isolation and identification from Fig. 1.4. At the first of them, the interaction force amplitude and its action line are calculated by applying the least squares technique to the static equilibrium equations written for some arbitrary point on the robot surface [9], [71]. It is worth mentioning that currently there is no clear methodology for this point selection and its influence on the final solution. In this thesis, this problem is discussed in detail, with careful mathematical proof. Then, in the second step, the intersections between the force action line and robot surfaces are computed, defining a set of potential contact points. In fact, not all obtained intersections correspond to

Table 1.3: Summary of existing methods for interaction parameters identification

Ref.	Robot	Core idea	Advantages	Disadvantages
Surface approximation approaches				
[28] 2014	KUKA LWR	Cylindrical approximation of link surfaces	The number of interaction parameters is reduced to two	Low accuracy, global optimization algorithm is required
[58] 2017	UR 5	Line approximation of planar robot links	Two parameters to identify, real-time implementation	Low accuracy, can be used only for planar robots
[57] 2018	Kinova Jaco 2	Cylindrical approximation of link surfaces	Two parameters to identify, slightly more accurate than [28]	Low accuracy due to surface approximation errors
Monte Carlo based approaches				
[29] 2016	Atlas humanoid	Contact particle filter on the robot link surfaces	High accuracy, can be used for multiple interactions	Very slow updates, can converge to wrong local minima
[59] 2019	KUKA iiwa	Particle filter + special Bayesian motion model	High accuracy, additionally outputs contact stiffness values	Slow updates, only normal to surface force directions are estimated
[30] 2019	Kinova Jaco 2	Particle filter + various clustering methods	Accurate and robust to noise, option to apply friction cone constraint	Slow updates, only normal to surface force directions are estimated
[60] 2021	KUKA iiwa	Rejection sampling with gradient descent	High accuracy, possible to find multiple minima of objective function	Requires motions to resolve ambiguity, slow updates
Machine learning approaches				
[57] 2018	Kinova Jaco 2	Classification for fixed points using NN and RF	Faster than conventional optimization-based approaches	Requires training, Tends to overfit to configurations in dataset
[61] 2021	Franka Emika Panda	Deep learning using unwrapped robot surface	Does not require torque data, can output multiple solutions	Requires training and transfer learning. Problems with unwrapping
Analytical approaches				
[9] 2017	KUKA iiwa	Analytical approach using joint torque sensors	Fast and accurate closed form solution	Works only for the sixth and following links, low robustness to noise
[70] 2021	KUKA LWR	Analytical approach + base velocities	Fast and accurate closed form solution for any robot link	Only for floating base robots with additional velocity sensors
[69] 2021	DLR SARA	Analytical approach + internal F/T sensors	Fast and accurate closed form solution for any robot link	Requires several additional six-axis force/torque sensors

physically possible solutions, since in practice feasible force direction must be restricted by the friction cone created around the surface normal. Moreover, because of sensor measurement errors, sometimes there are even no intersections of the obtained force action line with the robot surfaces. It means that such an algorithm is not able to produce any solution in this case, in contrast to the straightforward optimization-based techniques, which may be critical in real-life industrial applications.

It is worth mentioning that the first of the above steps is rather fast, due to pure closed-form expressions. In contrast, the second step dealing with intersections finding usually requires exhaustive enumeration of the robot surface elements [9], [71]. In the case of 3D models, the surfaces are typically represented by polygon mesh (collection of vertices, edges, and faces), while 2D models operate with a collection of straight-line segments. It is clear that the computing speed of the second step essentially depends on the graphical model precision associated with the basic element number. To speed up this step some special techniques from computer graphics could be applied allowing to reduce the enumeration complexity [72]–[74].

Another important issue arising here is related to the rank deficiency of the identification equations, which may occur when the number of active joint torque sensors is insufficient. For instance, only three independent scalar equations can be extracted from the static equilibrium conditions if the interaction force is applied to the 3rd link. In this case, all joint torque sensors starting from the fourth one provide zero measurements, so corresponding equations cannot be used for the identification. Thus, when the number of unknowns is larger than the number of equations and the desired force and its application point cannot be estimated uniquely. A similar problem arises if the force is applied to the sixth link, but the corresponding kinematical Jacobian is singular. It should be mentioned that the above problem could be easily overcome by using additional sensors, which provide six-dof force/torque data [69]. Such technique is widely used in humanoid robots [68], [75], [76] and in space robotics applications [70] (see Fig. 1.10). However, for typical industrial robots equipped with one-dof joint torque sensors, the rank deficiency problem is rather important. Nevertheless, as follows from the presented literature analysis, it was not carefully studied yet.

As it follows from the presented review, summarized in Table 1.3, in spite of numerous works in the considered area, there are still several important issues to be investigated for the identification algorithms in more detail. In particular, it is required to increase the computation speed of relevant identification algorithms allowing their efficient real-time utilization. Besides, particular attention should be paid to the physical feasibility of the obtained solutions as well as to the identification in the case of insufficient measurement data. These issues will be the focus of this thesis.

1.3 Mathematical model of physical interaction and its parameters

The main problem considered in this thesis is to model and identify the parameters of human-robot interaction: force and its application point using data obtained from internal torque sensors embedded in the robot joints. The modeling was divided into two parts described in this section. The first part deals with a two-dimensional planar model, which allowed us to simplify the computations and focus on the essential properties of the considered problem. The second part presents a more complex three-dimensional spatial model, which is close to the real system and includes some uncertainties like measurement noise.

1.3.1 Problem formalization for planar case

It is assumed that an operator interacts with a general n-dof planar serial manipulator, which consists of a fixed base, end-effector, and a number of links connected by n revolute joints. Besides, it is also assumed that the interaction force could be applied to an arbitrary point on the manipulator link surface (see Fig. 1.3). It should be mentioned that only the point contact between the operator and the robot is considered here, implicitly assuming that any surface-to-surface contact (or more complicated one) can be efficiently approximated by point-to-point interaction.

The input data for the considered identification algorithm are provided by one-axis torque sensors whose outputs also may include additional components caused by robot dynamics. But it is assumed here that the dynamic component is already excluded from the measurement data by means of a dedicated robot state observer. So further, the pure interaction components will be referred to as the joint torques for simplicity.

For this problem, the manipulator geometric model must include two types of parameters. The first of them defines the link lengths, similar to the conventional robot description. The second one describes the true link shapes giving a more realistic robot presentation in contrast to the classical approach that operates with wire frame models. These lead to the following system of equations describing the robot geometry in the case when the interaction force is applied to the kth link

$$\begin{aligned} x &= \sum_{i=1}^{k-1}\left(L_i \cos\left(\sum_{j=1}^{i} q_j\right)\right) + x^k \cos\left(\sum_{j=1}^{k} q_j\right) + y^k \sin\left(\sum_{j=1}^{k} q_j\right) \\ y &= \sum_{i=1}^{k-1}\left(L_i \sin\left(\sum_{j=1}^{i} q_j\right)\right) - x^k \sin\left(\sum_{j=1}^{k} q_j\right) + y^k \cos\left(\sum_{j=1}^{k} q_j\right) \end{aligned} \qquad k \le n \qquad (1.1)$$

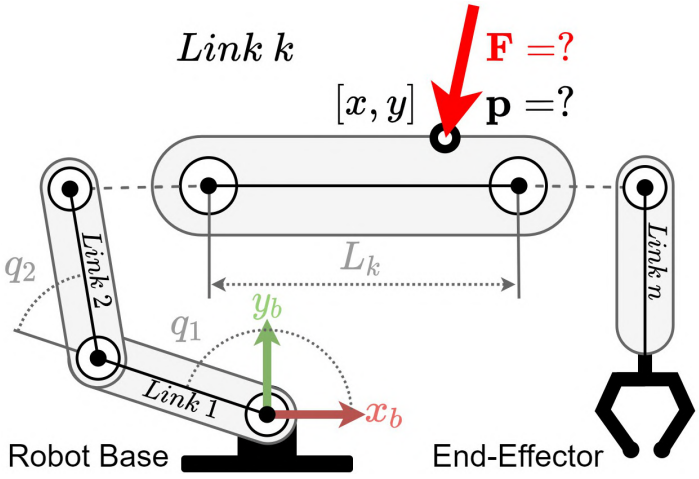


Figure 1.11: The interaction parameters for planar serial manipulator: the force $\mathbf{F} = [F_x,\ F_y]^T$ is applied at the contact point $\mathbf{p} = [x,\ y]^T$ on surface of the kth link.

Here x^k, y^k are the coordinates of the contact point w.r.t. at the kth link local frame, L_i is the length of the ith link, and q_j is the joint rotation angle. It should be noted that the above model is suitable for a rather general case, when the interaction force is applied to any intermediate manipulator link ($k < n$), not necessary to its end-effector ($k = n$). The latter is graphically shown in Fig. 1.11.

In such manipulator description, the shape of the kth link is convenient to define as a closed polyline $\mathbf{\Omega}_k$ composed of connected line segments. This yields the following geometric constraint

$$\mathbf{p} \in \mathbf{\Omega}_k \tag{1.2}$$

that must be further considered jointly with the basic equation (1.1). Here $\mathbf{p}$ is the vector of contact point with its coordinates x, y. The polyline $\mathbf{\Omega}_k$ can be described by the set of its m consecutive nodes

$$\mathbf{\Omega}_k = hull\left(\left\{\mathbf{s}_1^k, \ldots, \mathbf{s}_m^k\right\}\right) \tag{1.3}$$

which gives the following equations for the link shape description

$$\mathbf{p} = \alpha \cdot \mathbf{s}_i^k + (1-\alpha)\,\mathbf{s}_{i+1}^k, \quad \forall i = 1, \ldots m, \quad 0 \le \alpha \le 1 \tag{1.4}$$

where $\mathbf{s}_i^k$ is the coordinate of the ith node describing the kth link.

To take into account the static equilibrium condition associated with the considered interaction, the basic relation $\boldsymbol{\tau} = \mathbf{J}(\mathbf{q}, \mathbf{p})^T\mathbf{F}$ between the interaction force $\mathbf{F}$ and the

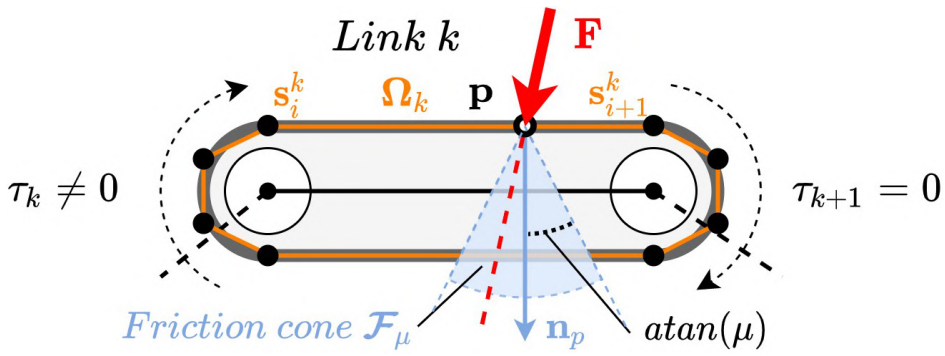


Figure 1.12: Planar case: graphical representation of the constraints $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ and $\mathbf{p} \in \boldsymbol{\Omega}_k$ for kth link. The first constraint is imposed on the direction of the force $\mathbf{F}$, which must be inside of the friction cone. The second constraint is applied to the location of the contact point $\mathbf{p}$ that must belong to the link surface.

joint torques vector $\boldsymbol{\tau}$ via the Jacobian $\mathbf{J}(\mathbf{q}, \mathbf{p})$ should be presented in the form

$$\left[\mathbf{J}_k(\mathbf{q}, \mathbf{p})^T\right]_{k\times 2} \cdot \begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \dots \\ \tau_k \end{bmatrix} \tag{1.5}$$

which includes only k significant components of the joint torques vector $\boldsymbol{\tau}$. This index k can be easily found from the original joint torques vector $\boldsymbol{\tau} = (\tau_1, \dots, \tau_n)$ by eliminating the last zero values, i.e., by applying the following rule

$$\tau_k \neq 0 \quad \& \quad \tau_i = 0, \quad \forall i > k \tag{1.6}$$

It is clear that in practice this rule should be applied with some tolerances depending on the measurement noise.

In the above equation (1.5) the joint coordinate vector $\mathbf{q} = [q_1, \dots, q_k]^T$ and torque vector $\boldsymbol{\tau} = [\tau_1, \dots, \tau_k]^T$ are known, while the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$ should be found. Related Jacobian $\mathbf{J}_k(\mathbf{q}, \mathbf{p})$ of size $2 \times k$ can be directly obtained from the geometric model (1.1), but it can be easily proved that for the considered problem this matrix can be presented in a more convenient form

$$\mathbf{J}_k(\mathbf{q}, \mathbf{p}) = \begin{bmatrix} -(y - y_1) & \dots & -(y - y_k) \\ (x - x_1) & \dots & (x - x_k) \end{bmatrix}_{2\times k} \tag{1.7}$$

where $x_i,\ y_i,\ i = 1, ..k$ define the joint axes locations that are computed as follows

$$x_i = \sum_{j=1}^{i-1} L_j \cos\left(\sum_{l=1}^{j} q_l\right), \qquad y_i = \sum_{j=1}^{i-1} L_j \sin\left(\sum_{l=1}^{j} q_l\right) \tag{1.8}$$

Another particularity of the considered problem that must be obligatorily taken into account is related to the possible directions of the interaction force $\mathbf{F}$, which are physically bounded. In fact, this force is composed of two principal components, the normal and tangential ones that are related to the Coulomb friction law. Geometrically, the possible directions can be presented in the form of a so-called friction cone, which is built around the link surface normal $\mathbf{n}_p$ at the contact point $\mathbf{p}$, as shown in Fig. 1.12. Assuming that the friction coefficient is equal to μ and $\mathbf{n}_p$ is an inward-pointing normal (i.e. pointing towards the interior of the link), the related constraint can be presented in the following way

$$\measuredangle(\mathbf{n}_p,\ \mathbf{F}) \ \leq\ \operatorname{atan}(\mu) \tag{1.9}$$

Further, this friction cone constraint will be referred to as $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. It is worth mentioning that in the above expressions both vectors $\mathbf{n}_p$ and $\mathbf{F}$ must be expressed with respect to the same coordinate frame (either local or global).

Thus, the desired solution $(\mathbf{F}, \mathbf{p})$ of the considered identification task must satisfy static equilibrium equations (1.5) as well as constraints on the force direction $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ and the contact point location $\mathbf{p} \in \boldsymbol{\Omega}_k$. It is worth mentioning that depending on the number of equations k, the system of static equilibrium equations can be either over- or under-determined. Besides, in practice, some terms in these equations can be corrupted by the measurement noise that may cause the equations inconsistency in the strong sense. For this reason, it is meaningful to present the considered identification task as the following non-linear constrained optimization problem

$$\begin{gathered} \left\| \boldsymbol{\tau} - \mathbf{J}_k(\mathbf{q}, \mathbf{p})^T \mathbf{F} \right\| \rightarrow \min_{\mathbf{F},\mathbf{p}} \\ \mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu \end{gathered} \tag{1.10}$$

where the index k denotes the link to which the interaction force is applied, which is determined using expressions (1.6). It is clear that this is a specific non-linear optimization problem which cannot be solved in a straightforward way and requires the development of a dedicated technique proposed in the following sections.

1.3.2 Problem formalization for spatial case

It is assumed that the operator interacts with a general n-dof serial manipulator, which consists of a fixed base, an end-effector, and several links connected by n revolute joints. It is also assumed that the interaction force could be applied to an arbitrary point on the manipulator link surface as shown in Fig. 1.3. It should be mentioned

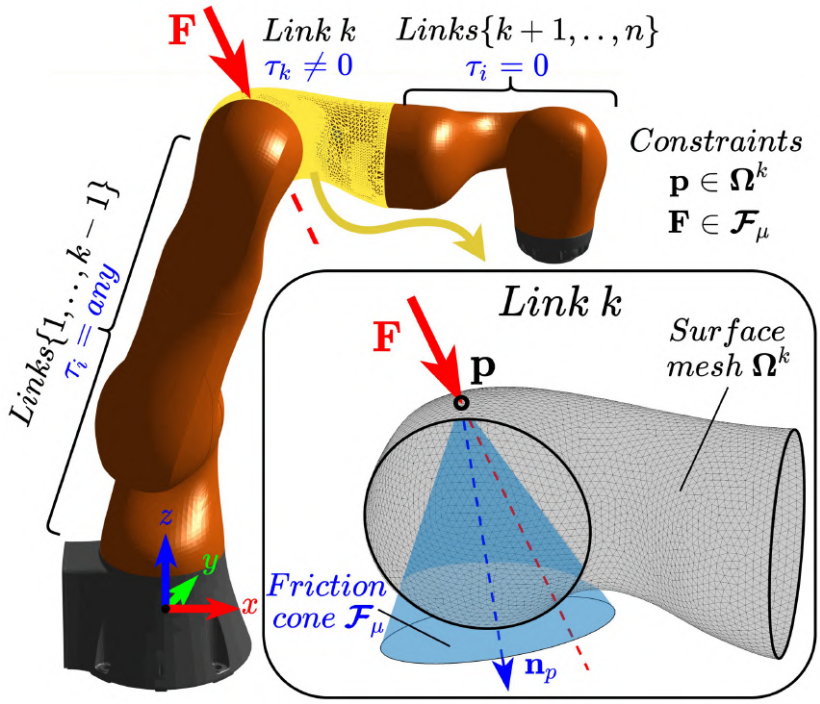


Figure 1.13: Spatial case: graphical representation of the geometric constraints for the kth link. The first constraint $\mathbf{p} \in \mathbf{\Omega}_k$ is applied to the contact point $\mathbf{p}$ location that must belong to the link surface. The second constraint $\mathbf{F} \in \mathcal{F}_\mu$ is imposed on the force $\mathbf{F}$ direction, which must be inside of the friction cone.

that only the so-called point contact between the robot and operator is considered here, implicitly assuming that any surface-to-surface contact (or more complicated one) can be efficiently approximated by the point-to-point interaction.

The input data for the considered identification algorithm are provided by one-axis torque sensors whose outputs usually include some additional components caused by robot dynamics. But it is assumed here that the dynamic component is already excluded from the measurement data by means of a dedicated robot state observer [9], [77] and the interaction between the human and the robot is already detected. So further, only pure interaction components will be referred to as the joint torques for simplicity. The output data is the interaction parameters, which include the identified interaction force magnitude, its direction, and its application point.

For the considered interaction, the basic relation (static equilibrium condition) between the force $\mathbf{F} = [F_x, F_y, F_z]^T$ and joint torques vector $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)^T$ can be written using the manipulator Jacobian $\mathbf{J}(\mathbf{q}, \mathbf{p})$, where $\mathbf{p}$ is the force application point and q is the joint coordinate vector. It is worth mentioning that here the interaction force may be not generally applied to the end-effector but to any manipulator kth link. So, the reduced Jacobian $\mathbf{J}_k(\mathbf{q}, \mathbf{p})$ should be used, which is obtained from the conventional one $\mathbf{J}(\mathbf{q}, \mathbf{p})$ by extraction of its first k columns. Using these notations,

the desired static equilibrium equations can be written as

$$\mathbf{J}_k\left(\mathbf{q}^{(k)},\mathbf{p}\right)^T\cdot\begin{bmatrix}F_x\\F_y\\F_z\end{bmatrix}=\begin{bmatrix}\tau_1\\\dots\\\tau_k\end{bmatrix}+\begin{bmatrix}\varepsilon_1\\\dots\\\varepsilon_k\end{bmatrix} \tag{1.11}$$

where $\mathbf{p}$ is the force application point, $\mathbf{q}^{(k)}=(q_1,\dots,q_k)^T$ is the reduced joint coordinate vector and only k significant components of the joint torque vector $\boldsymbol{\tau}$ are used. In addition, it is assumed here that all torque measurements τ_i are disturbed by some noise components ε_i that are further treated as unbiased independent random values but their distribution is not assumed to be identical. The latter allows us to present the practically observable torque measurements as $\tilde{\tau}_i=\tau_i+\varepsilon_i$. From these data, the k-index defining the interacting link can be found using the following decision rule [9]

$$||\tilde{\tau}_k||\geq\delta_\tau\quad\&\quad||\tilde{\tau}_i||<\delta_\tau,\quad\forall i>k \tag{1.12}$$

where δ_τ is some tolerance value allowing to distinguish significant and non-significant components in the measured torque vector $\tilde{\boldsymbol{\tau}}=(\tilde{\tau}_1,\dots,\tilde{\tau}_n)$.

In the above equation (1.11) both, the joint coordinate vector $\mathbf{q}^{(k)}$ and measured torque vector $\tilde{\boldsymbol{\tau}}^{(k)}=(\tilde{\tau}_1,\dots,\tilde{\tau}_k)^T$ are known, while the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$ should be found. Related Jacobian $\mathbf{J}_k(\mathbf{q},\mathbf{p})$ of size $3\times k$ can be obtained in conventional way. It can be easily proved that here the desired Jacobian matrix can be presented as

$$\mathbf{J}_k\left(\mathbf{q}^{(k)},\mathbf{p}\right)=\left[\mathbf{e}_1\times(\mathbf{p}-\mathbf{p}_1)\quad..\quad\mathbf{e}_k\times(\mathbf{p}-\mathbf{p}_k)\right] \tag{1.13}$$

where $\mathbf{p}_i$ defines the joint origin location and the unit vector $\mathbf{e}_i$ defines the rotation axis direction.

As follows from the physical nature of the considered problem, the above static equilibrium equation should be solved in conjunction with some constraints describing the manipulator link surfaces. In practice, it is convenient to describe these surfaces using the conventional triangle meshes, which are widely used in 3D CAD modeling. This allows to present the basic geometric constraints in the following form

$$\mathbf{p}\in\boldsymbol{\Omega}_k \tag{1.14}$$

where the superscript k denotes the link number and the triangle mesh $\boldsymbol{\Omega}_k$ is defined by a set of its vertices $\mathbf{sv}^k$, list of connections $\mathbf{sf}^k$ forming 3D faces and corresponding

face normal vectors $\mathbf{n}^k$, i.e.

$$\boldsymbol{\Omega}_k = mesh\left(\mathbf{sv}^k,\ \mathbf{sf}^k,\ \mathbf{n}^k\right). \tag{1.15}$$

Using this notation, each of the above 3D faces can be described by the equation

$$\begin{gathered}\boldsymbol{\Omega}_k^j = \alpha_1 \cdot \mathbf{sv}_1^{k,j} + \alpha_2 \cdot \mathbf{sv}_2^{k,j} + \alpha_3 \cdot \mathbf{sv}_3^{k,j} \\ 0 \leq \alpha_i \leq 1 \quad \sum \alpha_i = 1\end{gathered} \tag{1.16}$$

where j-index denotes the number of triangular face $\mathbf{sf}_j^k$ from $\boldsymbol{\Omega}_k$ with $\mathbf{sv}_1^{k,j}$, $\mathbf{sv}_2^{k,j}$, $\mathbf{sv}_3^{k,j}$ as its vertices. It is worth mentioning that in the above equation, the index j is unknown and should be found taking into account the static equilibrium equation (1.11).

Another constraint that must be obligatorily taken into account is related to the feasible directions of the interaction force $\mathbf{F}$, which are obviously bounded. Taking into account the Coulomb friction law, the possible directions can be presented in the form of a so-called 'friction cone', which is built around the link surface normal $\mathbf{n}_p$ at the contact point $\mathbf{p}$, as shown in Fig. 1.13. Assuming that the friction coefficient is equal to μ and $\mathbf{n}_p$ is an inward-pointing normal (i.e. pointing towards the interior of the link), the related constraint can be presented in the following way

$$|\angle\left(\mathbf{n}_p,\ \mathbf{F}\right)| \leq atan\left(\mu\right) \tag{1.17}$$

Although, some of the human-robot interactions, like grasping, do not follow Coulomb friction law, it is still possible to use (1.17) with $\mu \longrightarrow \infty$, which will correspond to the enlarged friction cone with angle up to 180 deg. Further, this constraint will be referred to as $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. Thus, the desired solution $(\mathbf{F}, \mathbf{p})$ of the considered identification problem must satisfy the static equilibrium equations (1.11) as well as constraints on the force direction $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ and the contact point location $\mathbf{p} \in \boldsymbol{\Omega}_k$. It is worth mentioning that depending on the number of equations k, the system of static equilibrium equations can be either over- or under-determined. Besides, in practice, some terms in these equations can be corrupted by the measurement noise that may cause the equations inconsistency in the strong sense. For this reason, it is meaningful to present the identification task as the following non-linear constrained optimization problem

$$\begin{gathered}\left\|\tilde{\boldsymbol{\tau}}^{(k)} - \mathbf{J}_k\left(\mathbf{q}^{(k)}, \mathbf{p}\right)^T \mathbf{F}\right\| \rightarrow \min_{\mathbf{F},\mathbf{p}} \\ \mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu\end{gathered} \tag{1.18}$$

where the torque measurements $\tilde{\boldsymbol{\tau}}^{(k)}$ are assumed to be corrupted by the measurement

noise and the index k is determined using expressions (1.12). It is clear that the practical application of such identification technique (1.18) requires careful investigation of its noise sensitivity, which will be considered in this work.

1.3.3 Modeling difficulties and singularities: typical case studies

As mentioned before, the considered problem does not always have a unique and straightforward solution, which is connected to certain singularities. To illustrate singular cases, consider a planar 4-degree-of-freedom robot with link length $\mathbf{L} = [1, 1, 1, 1]$. The interaction force $\mathbf{F} \in \mathbb{R}^{2\times 1}$ applied in the contact point $\mathbf{p}$ and vector of external torque $\boldsymbol{\tau} \in \mathbb{R}^{4\times 1}$ caused by it. This force will generate the following torque

$$\boldsymbol{\tau} = \begin{bmatrix}\tau_1 & \tau_2 & \tau_3 & \tau_4\end{bmatrix}^T, \ \mathbf{F} = \begin{bmatrix}F_x & F_y\end{bmatrix}^T \tag{1.19}$$

The solutions of system (1.5) can be found by numerically estimating the residual $r(\mathbf{p})$ for all points on the link surface:

$$r(\mathbf{p}) = \|\boldsymbol{\tau}_{\boldsymbol{r}} - \boldsymbol{\tau}\| \tag{1.20}$$

where $\boldsymbol{\tau}_r = \mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p})^T\mathbf{F}$ found from (1.10) for an arbitrary contact point $\mathbf{p}$ on the i-th link using pseudo-inverse. The residual value of 0 means the exact recreation of input torque $\boldsymbol{\tau}$, while other values mean the inability to find a force vector. The presence of multiple solutions will result in multiple minima in $r(\mathbf{p})$.

Let us examine a few cases where multiple minima can be detected. In the first stage, consider a planar robot with rod links, where the contact point can be placed on the axis of the link. The summary of test cases is presented in Table 1.4.

Case #1: The robot is in a configuration $\mathbf{q} = [0^\circ \ 45^\circ \ -90^\circ \ 45^\circ]^T$, has the interaction in point $i = 4$ $\mathbf{p} = [0.5, 0]^T$, interaction force is $\mathbf{F} = [0, 1]^T$. This robot configuration is not singular kinematically and from the interaction force point of view. As result, only one point with zero residual and one corresponding force is present.

Case #2: The robot is in a configuration $\mathbf{q} = [0^\circ \ 45^\circ \ -90^\circ \ 45^\circ]^T$ has the interaction in point $i = 4$ $\mathbf{p} = [0.5, 0]^T$, interaction force is $\mathbf{F} = [1, 0]^T$. From the kinematic point of view, this configuration is not singular, but the direction of $\mathbf{F}$ intersects the 4th joint axis and produces a zero torque τ_4 making it singular to the given force. As a result, any point on the 4th link can be used to describe an interaction. The estimated force vector will be equal to the original contact force at any point of the 4th link.

Table 1.4: Singular and non-singular cases for rod presentation of robot links

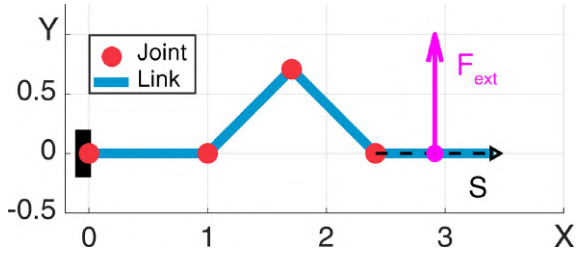
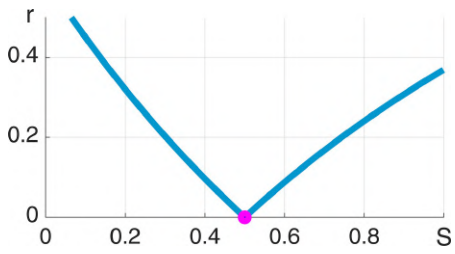
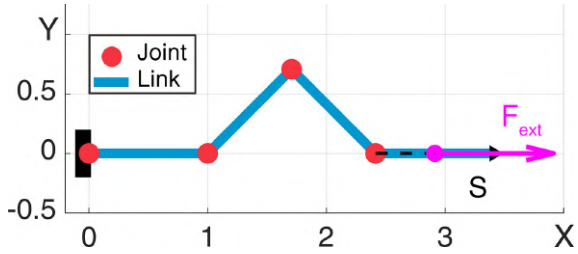
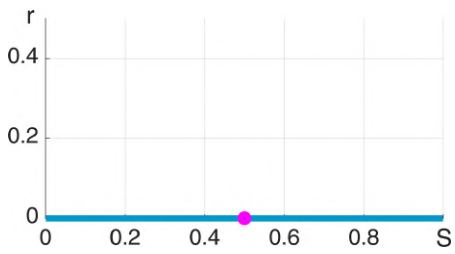
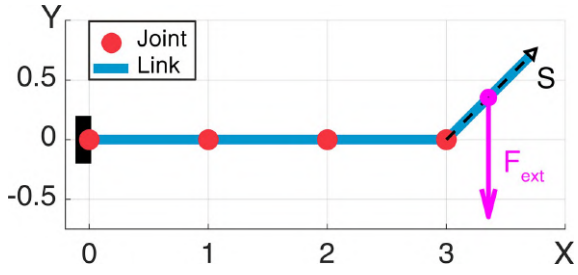
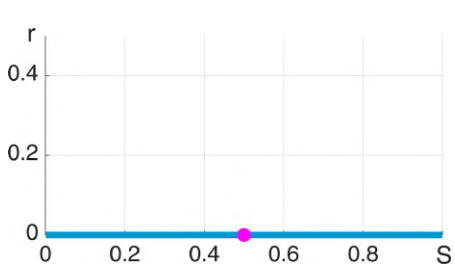
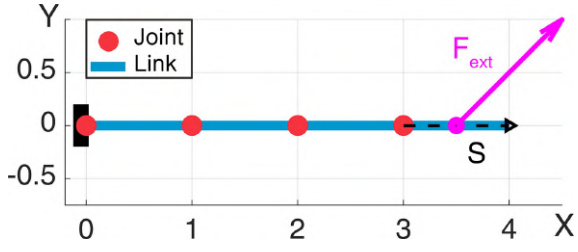
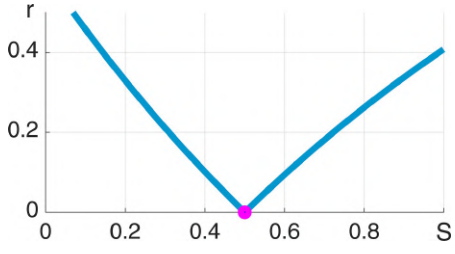
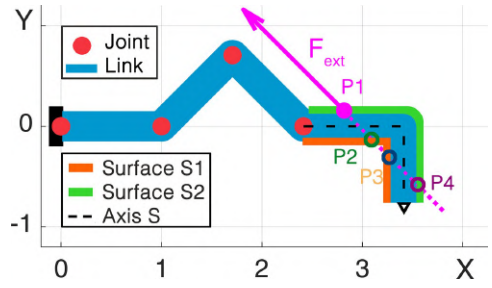
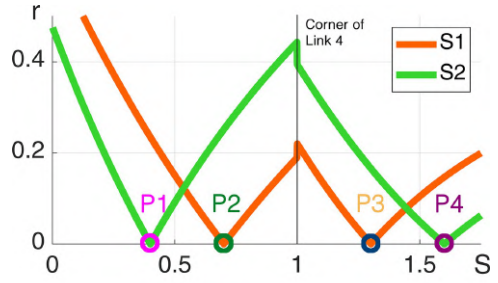
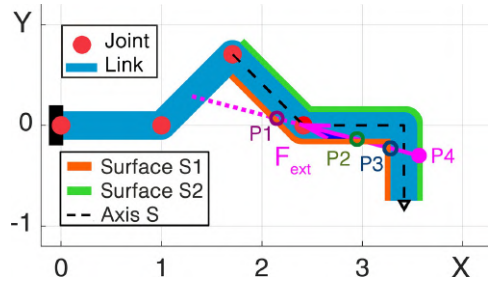
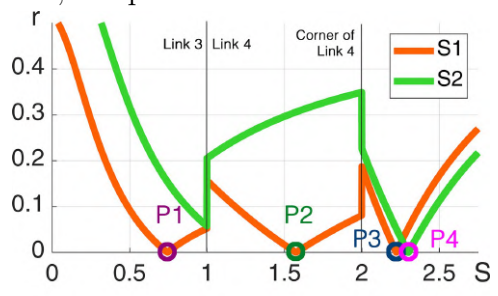
Robot configuration	Cost function
Case #1: Non-singular configuration. $rank(\mathbf{J}_k) = 2$. Non-aligned joint centers. Force does not intersect the last joint axis. One solution: unique $\mathbf{p}$, unique $\mathbf{F}$.	
	
Case #2: Singular configuration. $rank(\mathbf{J}_k) = 2$. Non-aligned joint centers. Force intersects the last joint axis. Multiple solutions: multiple $\mathbf{p}$, unique $\mathbf{F}$.	
	
Case #3: Singular configuration. $rank(\mathbf{J}_k) = 2$. Aligned joint centers. Force does not intersect the last joint axis. Multiple solutions: multiple $\mathbf{p}$, multiple $\mathbf{F}$.	
	
Case #4: Singular configuration. $rank(\mathbf{J}_k) = 1$. Aligned joint centers. Force does not intersect the last joint axis. Multiple solutions: unique $\mathbf{p}$, multiple $\mathbf{F}$.	
	

Table 1.5: Non-singular cases for surface presentation of robot links

Robot configuration	Cost function
Case #5: $rank(\mathbf{J}_k) = 2$. Non-aligned joint centers. Force does not intersect the last joint axis. Multiple solutions: multiple $\mathbf{p}$, same link, unique $\mathbf{F}$	
	
Case #6: $rank(\mathbf{J}_k) = 2$. Non-aligned joint centers. Force intersects the last joint axis. Multiple solutions: multiple $\mathbf{p}$, different links, unique $\mathbf{F}$	
	

Case #3: The robot is in a configuration $\mathbf{q} = [0° \ 0° \ 0° \ 45°]^T$ has the interaction in point $i = 4$ $\mathbf{p} = [0.5, 0]^T$, interaction force is $\mathbf{F} = [0, -1]^T$. The robot Jacobian rank $rank(\mathbf{J}_k) = 2$, configuration is not kinematically singular, force action line does not intersect joints axis, but has multiple minima. The main reason for singularity here is the position of all joint axis, where all of them lies on the line. In this situation, at any point of the 4th link, an infinite number of interaction forces exist will result in a zero residual.

Case #4: The robot is in a configuration $\mathbf{q} = [0° \ 45° \ -90° \ 45°]^T$ has the interaction in point $i = 4$ $\mathbf{p} = [0.5, 0]^T$, interaction force is $\mathbf{F} = [1, 1]^T$. Here the Jacobian rank $rank(\mathbf{J}_k) = 1$ and robot pose is kinematically singular, all joint centers lie on the line. This case has one point of collision and an infinite number of force vectors which can be applied to this point to achieve a zero residual.

For the before-mentioned cases, the multiple minima behavior is mostly observable in a singular configuration. Here the term singular corresponds to not only kinematic singularity but also to the instances with aligned joint centers and when the force intersects the last joint.

So far, multiple solutions in a rod link robot were observed, but in the real world, the robot's link could have a complex shape. In addition, the interaction force could be applied only to the surface. The presence of a surface could increase the number of possible solutions even in non-singular cases.

Let us now consider the same 4 DoF planar robot with a thicker link, where the contact point can lie only on its right or left surface and the Γ shaped last link. The summary of test cases is presented in Table 1.5.

Case #5: The robot in a configuration $\mathbf{q} = [0° \ 45° \ -90° \ 45°]^T$ has the interaction in the point $i = 4$ $\mathbf{p} = [0.4, 0.15]^T$, the interaction force is $\mathbf{F} = [-1, 1]^T$. The robot pose is not singular kinematically and from the force point of view, since the force vector direction does not intersect the last joint axis. The line of force action intersects the 4th link surfaces in 4 points $P1, P2, P3, P4$, where $P1$ is an original point of contact. This scenario is similar to **Case #1**, but the existence of link surface increased the number of potential solutions from 1 to 4. All potential intersection points have a zero residual value. Therefore the same interaction force could be applied in any of these points without any possibility to distinguish between them in static.

Case #6: The robot in a configuration $\mathbf{q} = [25° \ 45° \ 25° \ 45°]^T$ has the interaction in the point $i = 4$ $\mathbf{p}_{=}[1.15, -0.3]^T$, the interaction force is $\mathbf{F} = [1.15, 0.3]^T$. The robot is in the same configuration as in **Case #5**, but the force vector action line now intersects the 4th joint axis and produces a zero torque in $\tau_4 = 0$. Points $P2, P3, P4$ lies on the 4th link and $P1$ on the 3rd. This case is similar to the **Case #2**, but because of the link surface, it is no longer possible to have an infinite number of potential contact points. Now, it is impossible to distinguish even the link of the robot with a collision.

The presented case study shows that even for a simple planar case the unique solution for interaction parameters does not exist. The source of such ambiguity can be caused by different factors, for example kinematically singular configuration of the robot, the singular direction of the interaction force, or multiple intersections of the robot surface. The latter motivates to carefully study possible singular cases in the physical interaction between a human and a robot and propose a technique for ambiguity resolution.

1.4 Summary of related research and the principal goal of the thesis

As follows from the above-presented analysis, in spite of numerous works in the related area, there are still several important issues to be investigated in more detail. Primarily, it is required to increase the computation speed of relevant identification algorithms allowing their efficient real-time utilization. Besides, particular attention should be paid to the physical feasibility of the obtained solutions as well as to the interaction identification in the case of insufficient measurement data. These issues will be the focus of this thesis and considered in the following chapters.

In particular, it was shown that none of the existing approaches deals with the uncertainties and singularities of the interaction identification when only single-axis torque sensors are available. These methods are able to find only a single contact position, while often multiple solutions are possible depending on the robot pose and an external force direction. In practice, the obtained location of the force application point should define the robot reaction to the detected interaction. For example, in the case of real contact with a human at the robot elbow, the existing algorithm may wrongly identify contact with the robot end-effector. Consequently, instead of increasing the robot compliance and limiting the robot interaction force that is required for the elbow contact, the interaction handling algorithm will produce an opposite reaction that increases the interaction force in order to compensate for human intervention. Thus, in this case, the wrong robot reaction may harm the human operator. Another example related to the weakness of the existing techniques can be found in the generation of collision-avoidance reactions that are obviously different for collision with the robot elbow and its end-effector.

For these reasons, the *principal goal* of the thesis is the development of new techniques for modeling the physical interaction between a human and a robot with adaptive compliance in order to improve human safety and human-robot workcell performance. Special attention should be paid to the accuracy issues and the interaction parameters identification in singular cases, which can arise during the physical interaction because of limited measurements information provided by robot torque sensors.

To achieve this goal, the following problems should be addressed:

Problem 1: Comparative study of the existing approaches for interaction parameters identification, which are used for the modeling of physical interaction between a robot and a human, in order to detect their weaknesses and propose appropriate enhancements.

Problem 2: Development of a new efficient method for the interaction parameters identification in the human-robot physical collaboration, which is capable of estimation of the interaction force and its application point in real-time and for both singular and non-singular cases.

Problem 3: Development of the adaptive interaction handling strategy, which covers all phases of interaction starting from its initial detection, identification to classification and reaction, allowing to change the robot behavior in order to improve the safety and efficiency of the collaborative work cell.

Problem 4: Experimental validation of the developed interaction parameters identification method using a real collaborative robot and practical application of the proposed adaptive interaction handling strategy in a real-time environment.

To address the above-mentioned problems, the remainder of this work is organized as follows. Chapter 2 deals with a detailed comparative study of the existing techniques. Chapter 3 focuses on the development of the enhanced interaction identification technique for the planar case. Chapter 4 is devoted to the development of the enhanced interaction identification technique for the spatial case and the robustness analysis of the proposed methods. Finally, Chapter 5 presents a practical example of the complete interaction event handling technique developed in this thesis.

Chapter 2

Comparative analysis of existing methods for interaction parameters identification

Contents

This chapter presents a comparative study of the existing approaches for interaction parameters identification, which are used for the modeling of physical interaction between a robot and a human, in order to detect their weaknesses and propose appropriate enhancements. In particular, it evaluates numerically and experimentally existing approaches for the interaction parameters identification that were implemented by the author of this thesis. In the frame of this comparison, three main criteria are used: (i) accuracy, (ii) run-time, and (iii) algorithm ability to find multiple solutions.

2.1 Introduction

In the previous chapter, several approaches for the interaction parameters identification were presented, which are based on different software and hardware techniques. Since most modern collaborative robots can directly measure joint torques with one-degree-of-freedom torque sensors or estimate torque with relevant observers using the joint electrical current, in this work, as well as in comparison analysis, only the methods based on internal torque measurements will be considered. To compare performance and detect weaknesses of the interaction parameters identification approaches, it is important to implement and test them in practice with real hardware. To do so, five approaches were selected for the comparison analysis: the first two are based on a simplified robot model [28], the third one is based on machine learning [57], the fourth one is based on a particle filter [29] and the last one is based on custom straight-forward optimization.

It is important to note here that the implemented methods may differ from the original authors' techniques. In most cases, some improvements were proposed to deal with algorithm speed and accuracy in order to achieve better results.

2.2 Approach based on simplified robot model

The main idea of the method is to approximate the complex shape of the robot link surfaces with cylinders. This allows representing the interaction parameters by only two values: one for interaction force direction and one for normalized force application point. Then, the reduced set of the interaction parameters can be found with conventional straightforward optimization algorithms.

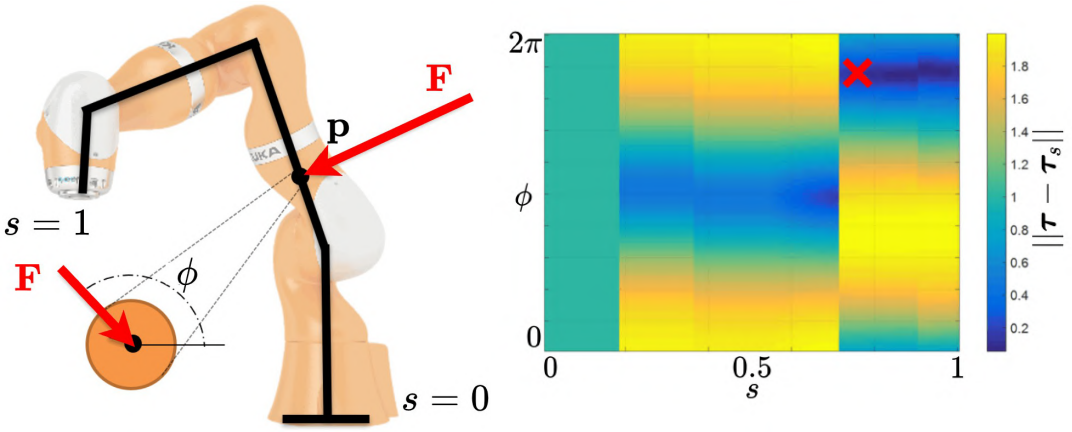
2.2.1 Robot model: cylindrical approximation

The cylinder approximation approach uses the main idea presented in the work [78]. In general, the physical interaction is described with the static equilibrium equations using robot Jacobian:

$$\boldsymbol{\tau} = (\mathbf{J}(\mathbf{q}, \mathbf{p}))^T \mathbf{F} \tag{2.1}$$

where $\mathbf{q}$ is robot joint angle vector, $\mathbf{F} = [F_x \ F_y \ F_z]^T$ is an interaction force, $\boldsymbol{\tau}$ is an external torque from the robot sensors. By using the cylindrical approximation shown in Fig. 2.1, the force application point is presented by only two values s and ϕ, instead of its actual x, y, z coordinates on the robot surface

$$\boldsymbol{\tau} = (\mathbf{q}, \mathbf{J}_s(s, \phi))^T \mathbf{F} \tag{2.2}$$



(a) Cylyndrical approximation (b) Typical objective function values

Figure 2.1: Example of interaction parameters identification using simplified robot model and straight-forward optimization.

where ϕ is angle, $s = l/L$ is a point in the normalized S-space of the robot (0 is the base of the robot, 1 is its end-effector), l is the distance from the base of the robot to the contact point, L is the total length of the robot, $\mathbf{J}(\mathbf{q}, s, \phi)$ is Jacobian from the base of the robot to point s.

To find the second interaction parameter, the interaction force $\mathbf{F}$, some approximations can be also used. Here, it is convenient to assume that the force is directed only by the normal to the cylinder surface in such a way that the force action line intersects the internal cylinder axis at the point s. The approximated force $\mathbf{F}_s$ is equal to

$$\mathbf{F}_s = \mathbf{F}_n \cdot [cos(\phi)\ sin(\phi)\ 0]^T \tag{2.3}$$

where $\mathbf{F}_n$ is the force magnitude, and angle ϕ defines the direction of interaction force. By substituting (2.3) to (2.2) the modified static equilibrium equations of the system

$$\boldsymbol{\tau}_s = (\mathbf{J}(\mathbf{q}, s, \phi))^T \mathbf{F}_s(\phi) \tag{2.4}$$

where angle ϕ and value s are unknown parameters of the interaction. It should be noted here, that different values of s correspond to the relevant link index k and size of the corresponding system.

2.2.2 Implementation of identification procedure: optimization-based

The straightforward optimization can be applied to identify the interaction parameters s, ϕ. The objective function for such optimization is a classical second norm of the

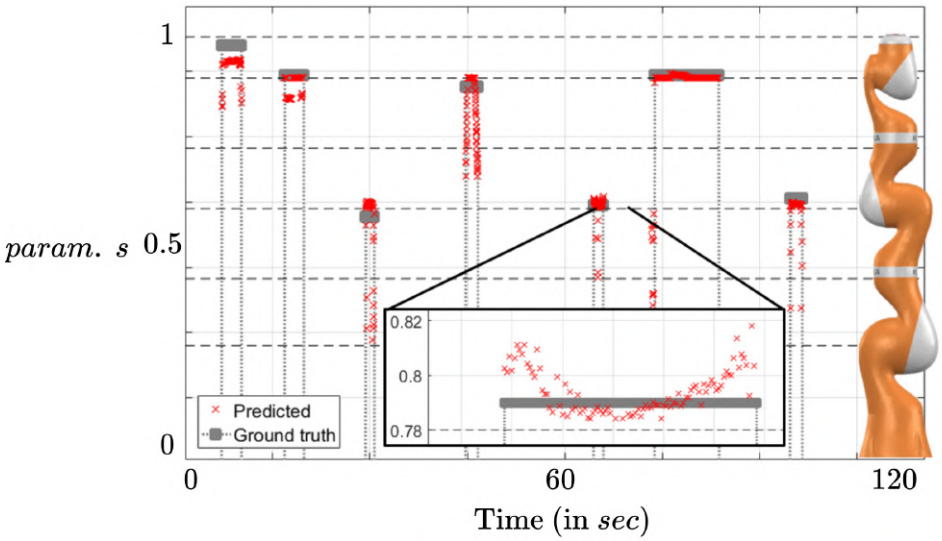


Figure 2.2: Experimental validation of the algorithm based on simplified robot model for contact point identification. During the experiments, the interaction force was applied in different locations. The red markers show the identified contact points, while the gray markers show the "true" force application points.

difference between measured joint torques and recreated torques using the model

$$\left\| \boldsymbol{\tau} - \mathbf{J}_s(\mathbf{q}, s)^T \mathbf{F}_s(\phi) \right\| \rightarrow \min_{s,\phi} \tag{2.5}$$

The example of the objective function (2.5) is presented in Fig. 2.1 and shows that it is noncontinuous due to Jacobian and can have multiple local minima. It should be noted that the result of gradient descent-based approaches started from some random initial point most likely end up in local minima, so several algorithm runs are required to find the global one. Original technique [78] deals with this minimization problem in two steps:

- Global optimization. For global optimization, the DIRECT [79] method is used. DIRECT method divides task space into rectangles and evaluates objective function in the centers of these rectangles, then the rectangles with the lowest objective function values are sub-sampled again.
- Local optimization. The DIRECT method requires a large number of iterations in order to estimate the exact minimum. To reduce the number of DIRECT iterations, local optimization is used at the final stage of optimization, where DIRECT is not so efficient. The local optimizer takes the output of DIRECT as an initial value. For local optimization, *matlab* function *Fmincon* is used.

It allows to roughly estimate the global minima and then refine it by the local optimizer. It is clear that in the case of multiple global minima, this approach provides only one solution.

2.2.3 Performance of simplified approach

The typical example of the interaction parameters identification is presented in Fig. 2.1, with the normalized parameters s and ϕ. Values with $s < 0.2$ correspond to the first link, where localization of the contact point basically is not possible.

The parameters identification for s during several interactions in time are shown in Fig. 2.2, where the identified values of s are also supplemented with the "true" interaction forces application points. The algorithm was tested for 250 interaction events and showed an average accuracy of 5.5 cm. It should be noted that in these experiments the forces were applied to the robot parts, which shapes are resemble cylinders. Otherwise, the accuracy of the identification quickly degrades due to the surface approximation errors. The average run-time of the algorithm is 52 ms, so it is hard to use in real-time applications. In addition, it is not possible to estimate multiple potential solutions, since it outputs only a single minimum point.

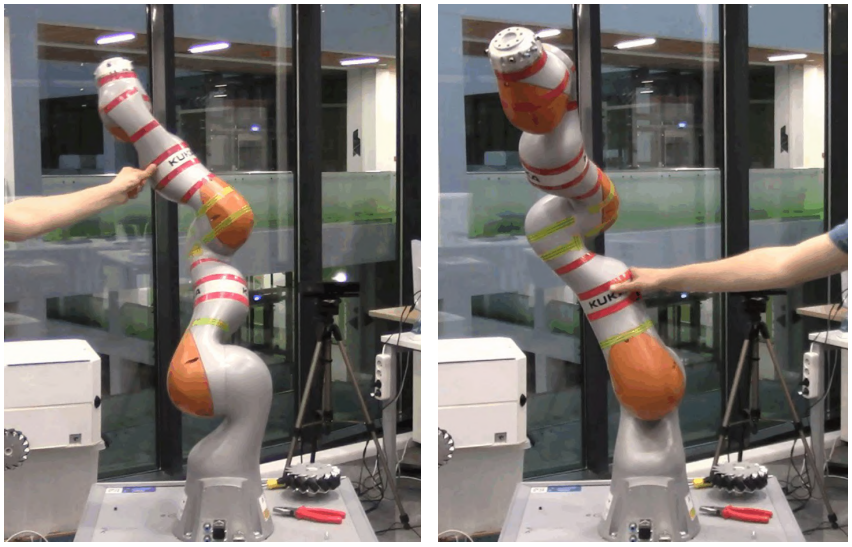
2.3 Approaches based on neural networks

In these approaches, machine learning-based techniques are utilized to solve the identification problem of human-robot physical interaction. One of the main disadvantages of the existing approaches based on straightforward optimization is their runtime, which due to a large number of computations makes them impossible to use in real-time systems. In this section, it is proposed to solve this problem with multi-player perceptrons or neural networks, ensuring better runtime of the identification procedure, that were generally inspired by [57]. In addition, the transfer learning paradigm is used to improve the performance of neural networks in practice. In contrast to conventional learning, the neural networks were trained on two datasets, the first one is generated in simulation, and the other one is captured from the real robot. Obtained results show that using a pre-trained network allows to increase the overall accuracy of the interaction parameters identification.

2.3.1 Contact point identification using NN trained on simplified model

The estimation of the force application point in general is a complex task since the surface constraint is quite hard to describe. For this reason, the simplified robot model was used, which is similar to the one described in the previous section.

Instead of trying to estimate three components of the interaction force vector and three coordinates of the force application point, the simplified model allows to reduce



(a) Contact at the 5th link (b) Contact at the 3rd link

Figure 2.3: Examples of interactions during the dataset creation. The force is applied only at marked points with a known location.

the number of parameters to only two, described in the previous section and graphically shown in Fig. 2.1. The first parameter (s) is the distance from the base of the robot to the contact point projected on the internal link axis in the way that $s = 0$ is the base and $s = 1$ is the end-effector. The second parameter (ϕ) corresponds to the angle of the applied interaction force.

The identification problem was presented in the form of a regression task. As the approximator the feed-forward neural network was used, which consist of 2 hidden layers with 28 and 14 neurons each. The input of such network is joint angles $\mathbf{q}$ (7 values) with external torque values τ (7 values), and the output is two parameters of interaction (s and ϕ). The training was done using Levenberg-Marquardt backpropagation with a MSE performance measure, the data division was 70% train, 15% for validation, and 15% for the test.

The simulated dataset was created by applying random force inside of the friction cone ($\mu = 0.5$) in a random robot configuration. Each sample stores information about contact point location, joint angles, and external torque values.

To collect the dataset from the real robot, the collaborative robot Kuka LBR iiwa 14 R820 was used in this work. In order to control the robot and read the current robot state, ROS package *iiwa_stack* [80] allowing to read joint torque, external torque, current position, desired position, etc. with some modifications was used. External PC with ROS master recorded *.bag* files with information about the robot's current joint position and external torque with 100 Hz framerate. The contact event was captured by the camera in order to label the contact position during the ground truth annotation.

The robot surface was marked with 20 points and 12 level marks with approximately 10 mm accuracy. Robot joint stiffness was set to 200 Nm/rad in order not to harm a human or trigger an internal robot safety mechanism. During the dataset creation, the robot was executing random movements, and the operator randomly touches marked points on the robot surface applying external force. The total number of collision events is around 250, which means that at least 12 collision samples for each marked point. Fig. 2.3 shows examples of collision events from the camera.

2.3.2 Contact point identification using NN trained on fixed-node model

The main source of errors in the previous approach was mostly caused by the cylindrical approximation, which in some cases poorly describes the true robot link shapes. Using this approximation, some information about the robots surface is basically lost. To overcome this and make the estimation part as simple as possible another approach based on the fixed-node model was proposed. The main idea is to fix several points on the robot's surface and then select only one of them as the force application point during physical interaction. Once the contact point is estimated, the interaction force can be found by using the least squares solution.

The robot surface is discretized by sampling points in order to formulate the classification problem. For simplicity, 20 points on the 6th link surface of the KUKA iiwa R820 robot were chosen. The simulated dataset is created by applying random force inside the friction cone ($\mu = 0.5$) in a random robot configuration. Each sample stores information about the contact point (class), joint angles (7 values), Jacobian of the corresponding link (21 values), and external torque values (7 values).

For classification, the feedforward neural network was used, where the multi-layer perceptron had 3 hidden layers with 128, 64, and 32 neurons each. The training was done using the scaled conjugate gradient backpropagation with cross-entropy performance measure. Data division was 70% for training, 15% for validation and testing.

2.3.3 Performance of NN-based approaches

The performance of both neural networks was evaluated using the dataset obtained from the real robot, however, the training set was different. In practice, it is quite complicated to collect a large dataset from the real experiment, and quite easy to do the same in simulation. Therefore two sets were collected, which are different in size. The first training set was obtained only from the real robot, its size is small and includes 10^4 samples, while the second was obtained in simulation with 10^5 samples.

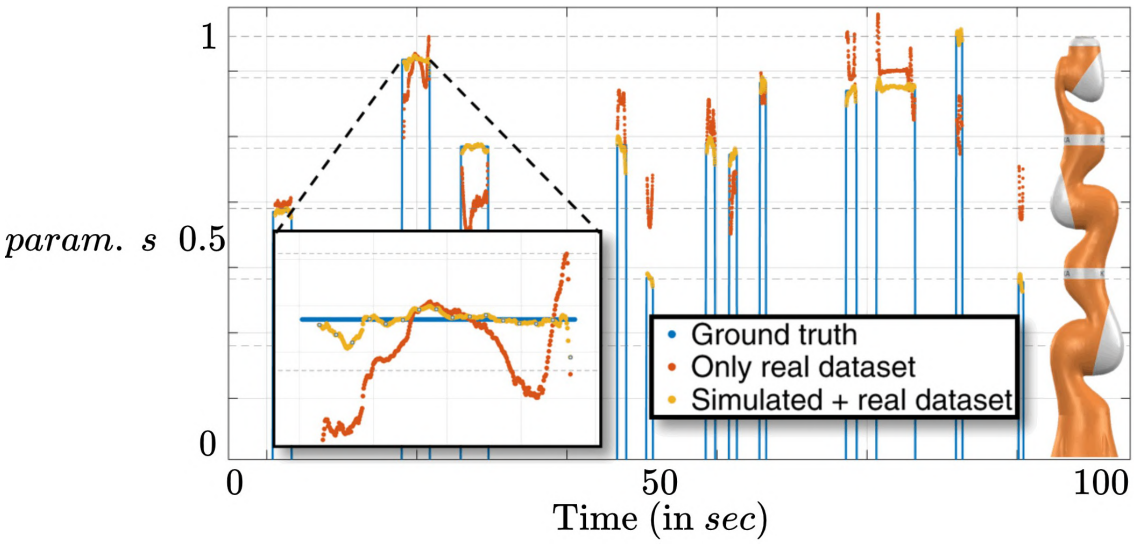
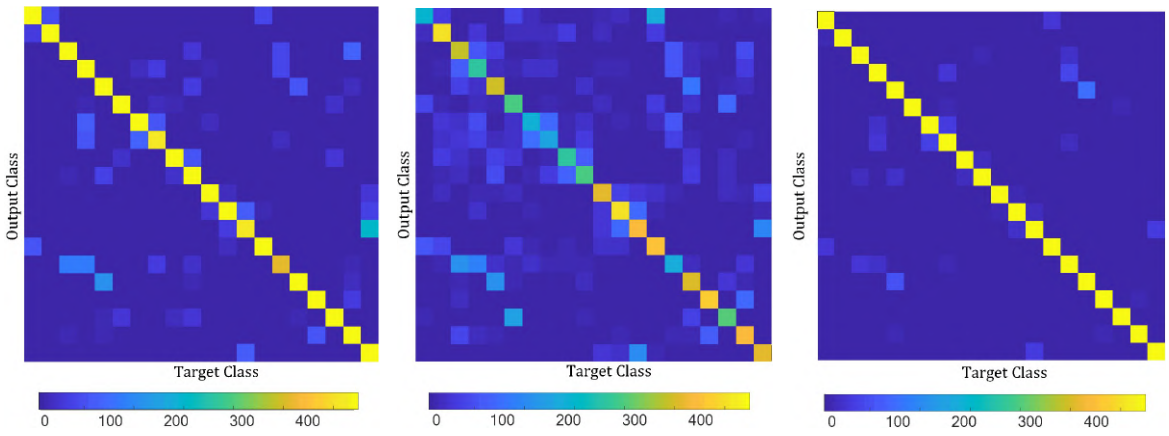


Figure 2.4: Experimental validation of the neural networks trained on simplified robot model. During the experiments, the interaction force was applied in different locations. The red and yellow markers show the identified contact points, while the blue markers show the "true" force application points.



(a) Simulated (b) Real robot (c) Transfer learning

Figure 2.5: Experimental validation of the neural networks trained on fixed-node robot model for contact point identification. The confusion matrices show the performance of neural networks trained on different datasets. Each class predicted by the neural network corresponds to one node point fixed on the robot surface.

The results of experimental validation for the neural network based on the simplified model are shown in Fig. 2.4 and for the network presenting fixed node model in Fig. 2.5. In the case of networks trained only on the small dataset from the real robot, the accuracy is 8.4 *cm* for the first network and 49.2% for the fixed-node based model. By using only the simulated dataset the accuracy is 12.2 *cm* and 71.3% for the first and the second network correspondingly.

By using transfer learning and firstly training the networks with the large simulated dataset and then carefully adjusting the last layer weights with the small dataset from a real robot, the identification accuracy can be improved. The first network showed an accuracy of 6.4 *cm*, while the second achieved 87.7%. So, with transfer learning, it is possible to improve performance by more than 15% compared with the only simulated dataset and more than 36% for the only dataset from the real robot for the classification task. For the regression task in the first approach using the pre-trained network with a real dataset over only a real dataset gives a 25% reduction in mean squared error. Since the networks are small-sized and light-weighted, the interaction parameters can be identified more than 180 times per second, which is approximately 10 times faster than the approach based on a simplified model and DIRECT optimization.

2.4 Approach based on sphere mapping of the robot surface

The main idea of the method is to transform the complex shape of the robot surface into a sphere, find the interaction parameters in the sphere space and using the inverse transformation obtain the parameters in the initial space.

In order to simplify the computation of the force direction constraint 1.17, some properties of the considered problem can be used. It is important to note that the problem (1.18) is convex for the fixed interaction force application point $\mathbf{p}$ since the values of $\boldsymbol{\tau}$ and $\mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p})$ are constants, and friction cone is a convex set. It is possible to solve this optimization problem for the set of fixed surface points lying on $\boldsymbol{\Omega}_k$, then the point with the minimum of the objective function is most likely the "true" force application point, and the corresponding force is the interaction force. From a computational point of view, it is useful to approximate friction cone with a polyhedron, for example, with the tetrahedral pyramid, as shown in Fig. 2.6:

$$\mathbf{F} = \sum_{i=1}^{4} \alpha_i \mathbf{F}_i^{\alpha} \tag{2.6}$$

where α_i is the weight of each support vector $\mathbf{F}_i^{\alpha}$. It allows us to rewrite the optimiza-

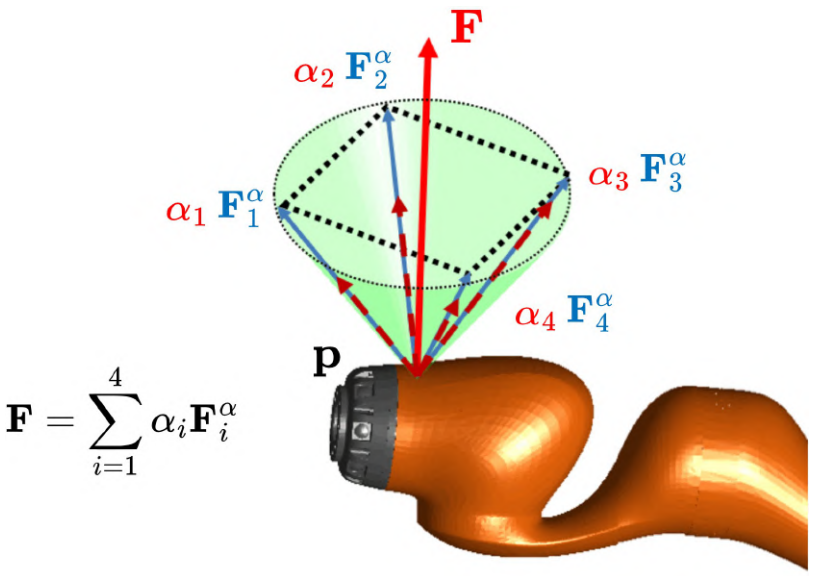


Figure 2.6: Friction cone approximation with a polyhedron. The interaction force is located inside of the polyhedron defined by four support vectors.

tion problem (1.18) in terms of quadratic programming (QP) for a fixed contact point

$$\min_{\boldsymbol{\alpha} \geq 0} \left\| \boldsymbol{\tau} - \mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p}, \mathbf{F}^\alpha)^T \boldsymbol{\alpha} \right\| \tag{2.7}$$

Here $\mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p}, \mathbf{F}^\alpha)$ is Jacobian with size $k \times 4$ written for the fixed arbitrary point $\mathbf{p}$ at the robot surface and the corresponding support vectors $\mathbf{F}^\alpha$. Since $\boldsymbol{\tau}^T \boldsymbol{\tau}$ is a constant, it is sufficient to solve:

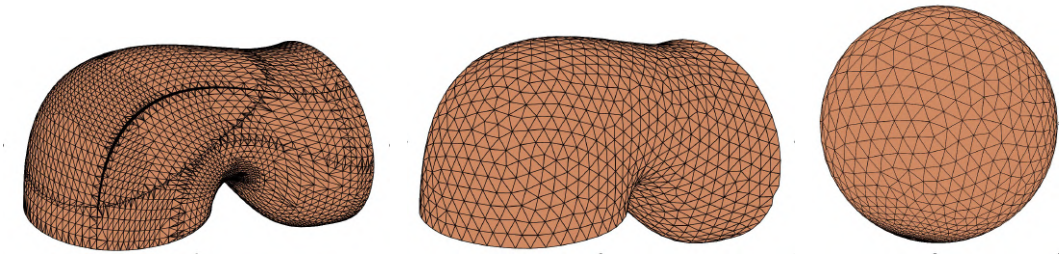
$$QP = \min_{\boldsymbol{\alpha} \geq 0} \left\| \boldsymbol{\alpha}^T \mathbf{H} \boldsymbol{\alpha} + \mathbf{g}^T \boldsymbol{\alpha} \right\| \tag{2.8}$$

where $\mathbf{H} = \mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p}, \mathbf{F}^\alpha) \cdot \mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p}, \mathbf{F}^\alpha)^T$ is a square of Jacobians, $\boldsymbol{\alpha}$ includes all weights of support vectors with size 4×1, $\mathbf{g} = -2\mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p}, \mathbf{F}^\alpha)\boldsymbol{\tau}$.

The minimal value of QP is zero, which corresponds to complete coincidence, and the maximal value depends on the model error. For further analysis, it can be useful to consider the residuals between the measured torque values and the ones found by QP for some arbitrary fixed point as $||\boldsymbol{\tau}_r - \boldsymbol{\tau}||$, where $\boldsymbol{\tau}_r = \mathbf{J}_k(\mathbf{q}^{(k)}, \mathbf{p}, \mathbf{F}^\alpha)^T \boldsymbol{\alpha}$. It is convenient to map the result into the interval from 0 to 1:

$$w = e^{-||\boldsymbol{\tau}_r - \boldsymbol{\tau}||} \tag{2.9}$$

The same scaling will be also used for all validation graphs in this thesis, since it provides a clear representation of local minima, without saturating the image.



(a) Initial (b) Trimesh (c) Final

Figure 2.7: Process of link mesh transformation for spherical mapping of the robot surface. The initial mesh is transformed into the trimesh with equally sized triangles and then is presented as a sphere.

2.4.1 Robot model: spherical representation of link surfaces

Typical manipulator arm surfaces can be very complex due to mechanical issues like optimal motor placement, heat dissipation, minimal deflections, etc. Even for collaborative robots like KUKA LBR iiwa that are designed to have a smooth surface for safety, it has a complex shape that is hard to approximate with simple geometric forms like cylinder or parallelepiped. In the case of real link shape, constraint equation for (1.18) ensures that the point belongs to a surface set and it becomes too complex and can seriously affect localization accuracy and computational effort.

The surface constraint is essentially simple for a sphere surface, where the constraint is basically described by the unit length of the contact point coordinate vector. To achieve this, it is necessary to map complex link surface mesh to a simple geometrical shape like a genus-0 unit radius sphere. In this case, one can find a point on the sphere's surface and then map it back to a real shape link. By using this sphere, it is possible to simplify the surface constraint by the relevant parametrization of the link mesh. In the case of such parameterization, the surface constraint is replaced by the simple unit vector constraint.

The identification can be described with the modified optimization problem

$$\min_{\boldsymbol{\alpha}\geq 0,\ \|\mathbf{s}\|=1} \left\| \boldsymbol{\tau} - \mathbf{J}_k(\mathbf{q}^{(k)}, tfs(\mathbf{s}), \mathbf{F}^{\alpha})^T \boldsymbol{\alpha} \right\| \tag{2.10}$$

where tfs is a mapping function from coordinate $\mathbf{s}$ on the sphere $\boldsymbol{\Omega}_s$ to a coordinate $\mathbf{p}$ on real link mesh $\boldsymbol{\Omega}_k$. Compared to the original objective function 1.18, both of the constraints were simplified in order to provide better runtime. For this particular optimization problem, it is convenient to divide the optimization process into two steps. The first step deals with estimating the residual for some sampled points on the robot surface and the corresponding friction cone. The second step is focused on picking the next point on the robot's surface.

The mapping between typical robot link meshes and their spherical representation can be described as follows. The initial mesh provided by Matlab Robotics System Toolbox (Fig. 2.7a) is simplified and remeshed as trimesh with equally distributed points across its surface (Fig. 2.7b). Spherical parametrization itself is done using FLASH [81] algorithm with triangle area preserving total energy minimization feature. During the parametrization, no overlapping triangles were detected (Fig. 2.7c). It should be noted that some areas of the resulting spherical mesh will be not achievable for the interaction and should be ignored.

Transformation function tfs, which maps sphere coordinates $\mathbf{s}$ to robot link coordinates $\mathbf{p}$ could be found by using the following algorithm:

- Find the triangle with the intersection of a ray vector $\mathbf{s}$ with the sphere $\mathbf{\Omega}_s$
- Find barycentric coordinates w, v, u of the intersection for this triangle
- Find the corresponding triangle in the link mesh and by using w, v, u, find the exact point $\mathbf{p}$ in robot link coordinates.

The first two steps of this algorithm can be implemented by using the approach presented in [72]. The barycentric coordinates uniquely define a point in a triangle, so the distortion caused by mesh parameterization does not affect the final accuracy of the proposed method. It is important to note here that link mesh $\mathbf{\Omega}_k$ and its spherical variant $\mathbf{\Omega}_s$ differ only in their vertex coordinates, however, their connections as well as the total number of vertices and triangles are the same.

2.4.2 Implementation of identification procedure: two-step technique

To find the interaction force and its application point, it is required to find the minima of the optimization problem 2.10. Various straight-forward optimization techniques can be used but most of them require a good starting point in order to converge to the right local minima of the objective function. As the starting point, an approximate location of the force application point can be used, which is found as a point with the lowest residual (2.9) from a set of points distributed across the robot surface. Despite of many ways of how points can be distributed across the robot surface, the simplest one is to use the trimesh vertex points (Fig. 2.7b), which are equally distributed. Unfortunately, solving QP for each vertex point and then solving local optimization is not possible in real time for any significant number of the starting point. But for the starting point, evaluating the exact residuals is not always required, it is sufficient to find an approximate location. The approximate residuals can be evaluated for the points without solving QP and considering force direction constraint, simply evaluating for the force directed along the surface normal.

Graphically, the estimation of residuals with different constraints is presented in Fig. 2.8, which shows the position of the "true" force application point and its approximation. In the case of Fig. 2.8a, the hard constraint is used, where the interaction force direction matches the surface normal vector. Obviously, the maximum weight (2.9) can only achieve a value of 0.7, so the found minima can not perfectly describe actual interaction. By using the soft constraint (Fig. 2.8b), where the interaction force direction is located inside of the friction cone, the location of obtained minima is very close to the "true" force application point and has a weight value close to 1. Depending on the computational capabilities of the identification system both of these methods can be used to estimate the starting point of the optimization algorithm in order to find the global minima.

Besides the selection of the initial point, the execution time of the algorithm is also reduced by the pre-calculation of configuration-independent variables. It includes the mesh transformation maps, its trimeshes as well as corresponding spherical meshes of each link. All of them can be found offline, before the interaction identification.

In addition, some of the computation is repeated for all surface vector points and can be unified to further reduce the computation time. For example, the robot Jacobian is calculated for each mesh vertex point in order to find the residuals, however, this includes a lot of similar operations for points of the same link. To overcome this, the temporal Jacobian for each link origin and one particular robot configuration $\mathbf{q}$ can be found using the following equation for each link

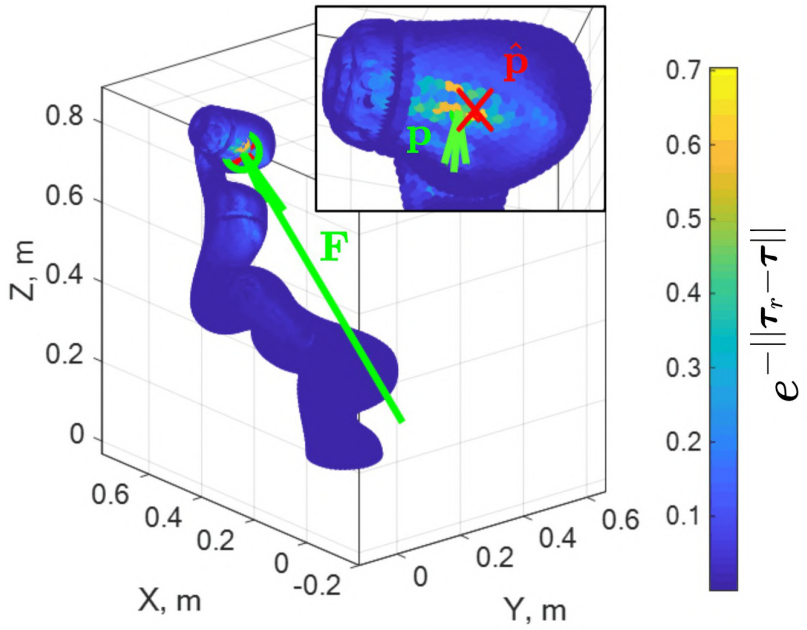
$$\mathbf{J}_{temp,k} = \left[\prod_{k-1}^{j=1} \mathbf{T}_{link,j}\mathbf{R}_{joint,j}(q_j)\right] \mathbf{T}_{link,k} \frac{\delta \mathbf{R}_{joint}(\mathbf{q}_k)}{\delta \mathbf{q}_k} \tag{2.11}$$

where $\mathbf{T}_{link}$ is a homogeneous transformation matrix of the link, $\mathbf{R}_{joint}$ is a homogeneous transformation matrix of the joint, i is a current link. The size of this temporal Jacobian matrix is 4×4. This calculation is done once for every robot configuration. To find the Jacobian for some particular point $\mathbf{p}^k$ presented in the kth link local frame, a simple linear offset can be used

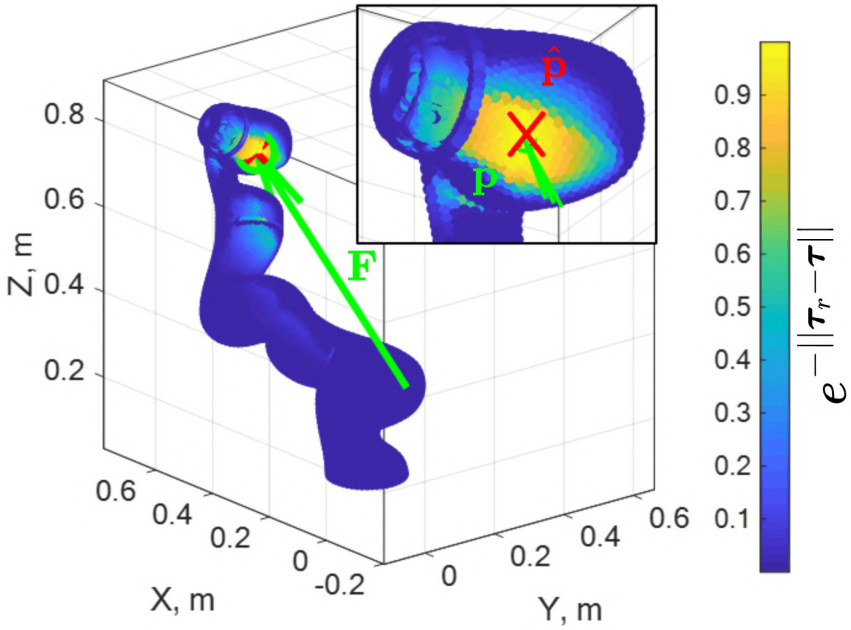
$$\mathbf{J}_{temp,\mathbf{p}^k} = \mathbf{J}_{temp,k}\mathbf{T}(\mathbf{p}^k) \tag{2.12}$$

where $\mathbf{T}$ is a homogeneous transformation matrix to point $\mathbf{p}^k$ in the link coordinate system. After that, the linear components of the Jacobian can be extracted

$$\mathbf{J}_{\mathbf{p}^k} = \begin{bmatrix} \mathbf{J}_{temp,\mathbf{p}^k}(1,4) \\ \mathbf{J}_{temp,\mathbf{p}^k}(2,4) \\ \mathbf{J}_{temp,\mathbf{p}^k}(3,4) \end{bmatrix} \tag{2.13}$$



(a) Hard constraint: force is constrained by the normal vector



(b) Soft constraint: force is constrained by the friction cone

Figure 2.8: Estimation of the interaction force application point using different constraints. In the first step, the hard constraint is used to find an initial estimation, which is further replaced by the soft constraint to obtain the desired interaction parameters. The green arrow denotes the real interaction force; the red cross shows the obtained force application point.

Algorithm 1: Interaction parameters estimation using sphere mapping

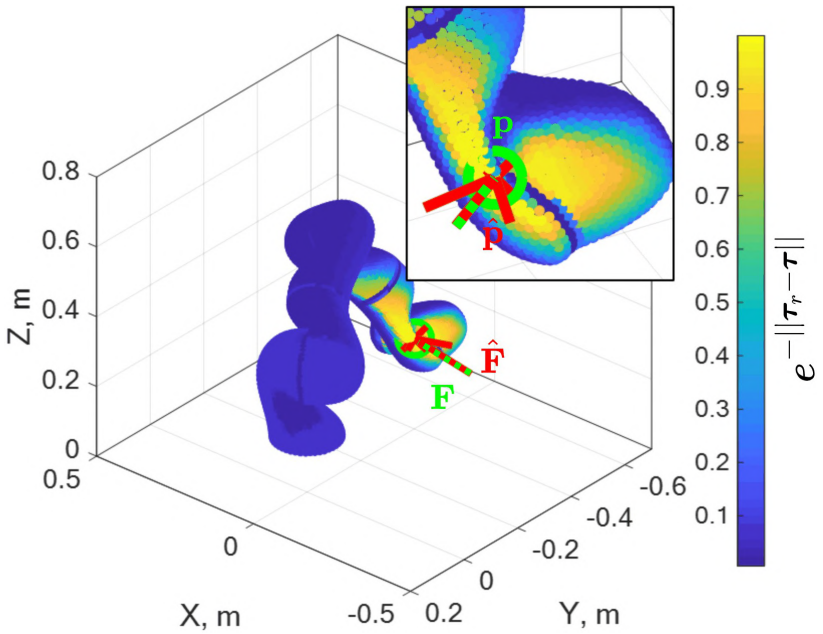
Function `FindContact`$(\boldsymbol{\tau}, \mathbf{p}_{prev})$**:**
  Input : Joint torques $\boldsymbol{\tau}$, previous contact location $\mathbf{p}_{prev}$
  Output: Force application point $\mathbf{p}$, interaction force $\mathbf{F}$
  if $\boldsymbol{\tau} > \boldsymbol{\epsilon}$ **then**
    if $\mathbf{p}_{prev} = \varnothing$ **then**
      Find $\mathbf{J}_{temp,i},\ \mathbf{FK}_{temp,i}$;
      for $\mathbf{p} \in \boldsymbol{\Omega}_s$ **do**
        $\mathbf{F}_{n,p} = \mathbf{n}_p$;
        $\mathbf{w} = \exp(-\left\|\boldsymbol{\tau} - \mathbf{J}_k^T \mathbf{F}_{n,p}\right\|)$;
      $\mathbf{p}_{max} = \mathbf{p}(\max(\mathbf{w}))$;
      $\mathbf{s} = \text{tfs}^{-1}(\mathbf{p}_{max})$;
      $\mathbf{p},\ \mathbf{F} = \text{SQP}(\text{SphereCost}, \mathbf{s})$;
      $\mathbf{p}_{prev} = \mathbf{p}$;
    else
      $\mathbf{s} = \text{tfs}^{-1}(\mathbf{p}_{prev})$;
      $\mathbf{p},\ \mathbf{F} = \text{SQP}(\text{SphereCost}, \mathbf{s})$;
      $\mathbf{p}_{prev} = \mathbf{p}$;
  else
    $\mathbf{p}_{prev} = \varnothing$;

Algorithm 2: Cost function for sphere mapping technique

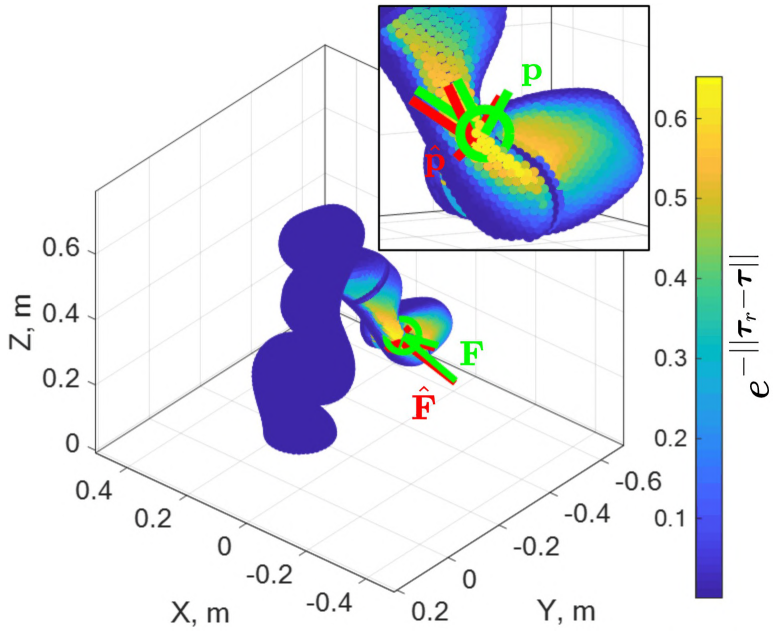
Function `SphereCost`$(\boldsymbol{\tau}, \mathbf{c}_{sph})$**:**
  Input : Joint torques $\boldsymbol{\tau}$, location on sphere $\mathbf{s}$
  Output: Interaction force $\mathbf{F}$, application point $\mathbf{p}$
  intersection $= (0, \vec{s}) \cap \mathbf{sph\_map}_i$;
  Extract barycentric coordinates w, v, u;
  $\mathbf{p} = \text{tfs}(\mathbf{s})$;
  $\boldsymbol{\alpha} = \min_{\boldsymbol{\alpha} \geq 0} \left\|\boldsymbol{\tau} - \mathbf{J_k}(\mathbf{q}, \mathbf{p}, \mathbf{F}^{\alpha})^T \boldsymbol{\alpha}\right\|$;
  $\mathbf{F} = \sum_{i=1}^{4} \alpha_i \mathbf{F}_i^{\alpha}$;

In the same way, the forward kinematics for any point at the robot link could be found. This allows to speed up the evaluation process for a large number of points scattered across the robot surface (up to 10 000) and get a more precise location of the global minima.

Optimization for (2.7) is executed using CVX library [82] compiled with CVXGEN [83]. For the second stage optimization (2.10), a local optimizer based on Sequential Quadratic Programming (SQP) was used. The corresponding technique is described in Algorithm 1 and Algorithm 2.



(a) Without torque measurements noise



(b) With torque measurements noise

Figure 2.9: Experimental validation of the algorithm based on the spherical mapping. The green arrow denotes the real interaction force and its application point; the red arrow shows the obtained interaction force and its application point.

Table 2.1: Identification accuracy for spherical mapping technique: with and without torque measurements noise

Robot link with interaction	**7**	**6**	**5**	**4**	**3**
Case without noise					
Force application error, $[cm]$	0.19	0.23	0.36	0.63	0.82
Force amplitude error, [%]	0	0	0.3	3	11.7
Case with $N(0, 0.3)$ Nm noise					
Force application error, $[cm]$	0.83	1.57	2.2	5.1	10.4
Force amplitude error, [%]	3.9	4.1	6.3	9.1	16.6
Case with $N(0, 0.5)$ Nm noise					
Force application error, $[cm]$	1.05	1.89	3.1	6.4	11.6
Force amplitude error, [%]	5.2	7.2	5.2	12.1	16.1

2.4.3 Performance of approach based on sphere mapping

In order to test the approach, 1000 random robot configurations were chosen with 1000 random interaction forces applied at each robot link. The interaction force directions were selected from the friction cone at the force application point with the coefficient $\mu = 0.5$. Information on previous collision location was not used, i.e. assuming that each contact is the first one in the sequence. To make the simulation result closer to the real robot Gaussian noise with different amplitude was added to torque measurements. Corresponding simulation results are presented in Table 2.1 for the noise magnitude 0, 0.3, 0.5 Nm respectively. The mean position estimation error for 0 Nm noise is 0.44 cm and 0.26 cm for the last 3 links. With additional measurement noise, this error increases to 4 cm and 4.8 cm for 0.3 and 0.5 Nm noise, while the mean error for the last 3 links is 1.5 cm and 2 cm correspondingly. Localization errors for the case without measurement noise can be explained with high tolerance values in SQP optimization and since all configurations and forces are random, some of them could be singular configurations. In this case, multiple solutions exist, and it is not possible to find the correct one by using the presented algorithm.

The example of interaction parameters identification is shown in Fig. 2.9a. Here green and red arrows are colliding as a result of zero estimation error for the force and its application point. Fig. 2.9b shows the same configuration, contact point, and force, but with 0.5 Nm magnitude Gaussian noise in torque sensing. The difference in reference and the estimated contact point is essential, the highest residual weight here is about 0.65, compared to 1 in the case without noise.

The developed algorithm was tested on Intel Core i5-4210H 3GHz CPU, 8Gb RAM PC as a Matlab single-thread program. The typical execution time of the algorithm is about 9-11 ms or 90-110 Hz. One evaluation of the cost function (SphereCost) was done in approximately 35-40 us, and called about 150 times during the SQP optimization procedure including gradient calculation.

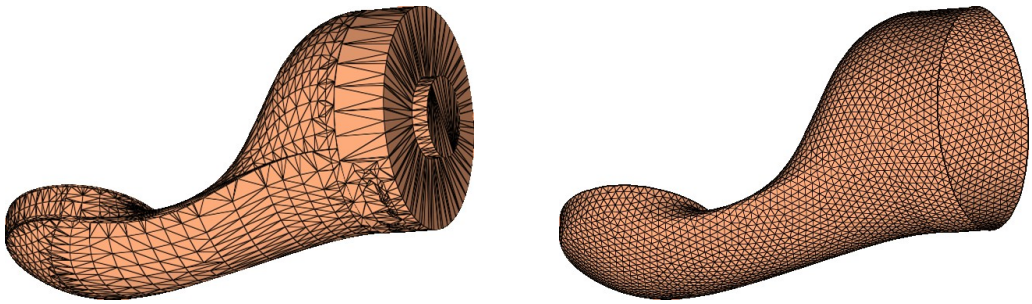
2.5 Approach based on particle filter on graph

This section describes the approach based on graph theory and particle filter. The approach utilizes basic ideas of the contact particle filter proposed by Manuelli and Tedrake [29] but with improved run-time. The reduction of computational cost is achieved by representing the robot surface as a graph, which essentially replaces the most time-consuming step of particle projection from the original method.

2.5.1 Robot model: graph representation of robot link surfaces

Although the surface of the robot is usually approximated with primitive shapes like cylinders, the interaction identification algorithm will produce a poor result, especially for the robot with a complex shape like KUKA iiwa. To overcome this, it is proposed to use a modified surface of the robot links extracted from the .STL model. The robot .STL model is widely used for visualization purposes, but its accuracy is good enough for the localization task. The initial model, imported from the .STL file presented in Fig. 2.10a. It has an internal area of the link and some surfaces, unavailable during physical interaction with a robot. Also, it is more convenient and computationally easier to work with a finite number of points, sampled on the robot surface, than to find a point on the facets of the link. To provide sufficient accuracy in this case, the points should be equally distributed with the distance between them less than the desired accuracy. The modified mesh of the robot link is shown in Fig. 2.10b, which was obtained by removing unreachable regions and isotropically resampling the initial robot mesh.

The remeshed link could be represented in the form of a graph, where its vertices represent the nodes ($\mathbf{N}$) of the graph and faces correspond to the edges ($\mathbf{E}$) as $\mathbf{G} = (\mathbf{N}, \mathbf{E})$. Each node $\mathbf{N}_i = \langle \mathbf{p}_i, \mathbf{n}_i, \mathbf{nb}_c \rangle$ includes information about its position in a joint coordinate frame $\mathbf{p}_i$, $\mathbf{n}_c$ is a normal vector at this point and supplemented with a list of its neighbors $\mathbf{nb}_c$. This graph is obtained offline and does not change during the run-time of the algorithm.



(a) Initial mesh (b) Modified mesh

Figure 2.10: Process of link mesh transformation for particle filter on graph technique. In mesh (a), there are some areas unfeasible for interaction, which are removed in mesh (b). Besides, the modified mesh (b) is isotropically resampled in order to increase the identification accuracy.

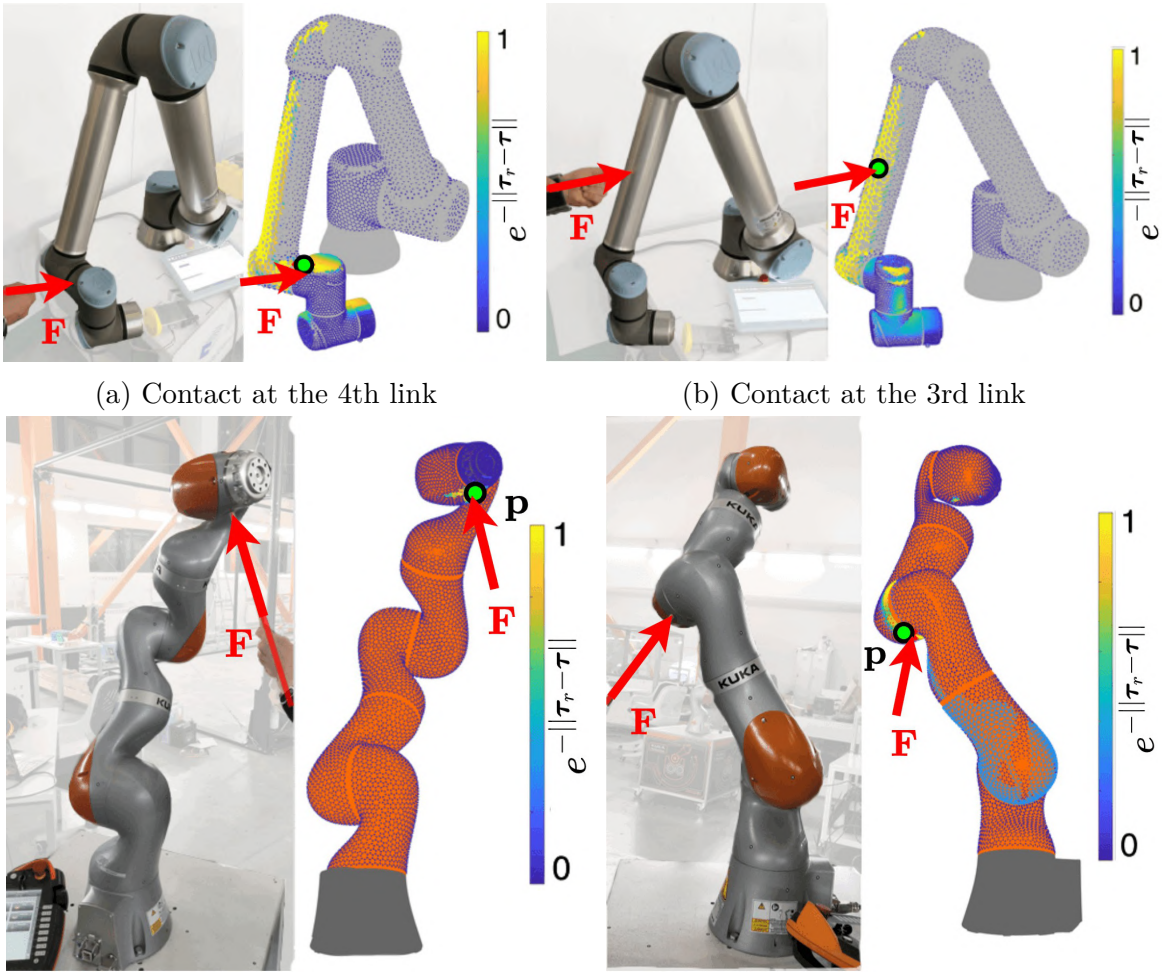
2.5.2 Implementation of identification procedure: particle filter

The exact interaction parameters are estimated using particle filter [29], which in contrast to the original work operates on the robot surface graph $\mathbf{G}$. This allows to avoid the computational complexity of a motion model update, where new positions of the particle are projected on the surface of the robot by finding ray-mesh intersections. To describe the particle filter motion and measurement model are presented below.

The motion model is used for updating the particle location during the prediction step of the particle filter. Due to the unpredictable behavior of the interaction point on the robot surface, it is not possible to define a model with good generalization. For example, assuming contact point motion on the surface will have good results when a human grabs the robot. At the same time, robot collision with a wall will benefit from a model with a contact point in the base frame. Here, a random walk policy is used as the motion model, which will generate reasonable results for most cases.

The random walk policy on the graph consists of choosing a random neighbor of the current node and switching to it. The number of jumps is defined by a random integer between 0 to max_steps. This variable depends on the weight of the current particle, so the particle with a high weight will have a lower maximum walk distance.

During the measurement/update step, the weights are applied to the particles. The weight of each particle is equal to the residual between the measured torque in the robot joints and recreated torque for this particle by using QP from the previously described approach 2.7. For further analysis, it is useful to map the result into the interval from 0 to 1, i.e weight is equal to $w \sim exp(-\alpha QP)$, where α is the scaling coefficient.



(a) Contact at the 4th link

(b) Contact at the 3rd link

(c) Contact at the 6th link

(d) Contact at the 4th link

Figure 2.11: Experimental validation of the algorithm based on particle filter on graph: the identification of interaction parameters for KUKA LBR iiwa and UR10e collaborative robots. The green circle denotes the obtained collision location, the red arrow shows the real interaction force.

2.5.3 Performance of particle filter approach

For the experimental study, two collaborative robots were used: KUKA LBR iiwa 14 and Universal Robots UR 10e. The obtained results were tested in simulation and hardware. Without noise in torque measurements, the particle filter shows accuracy similar to the method based on spherical approximation and provided almost zero error in the identification of the interaction force and its application point. With added noise, these algorithms showed a 4-5 cm mean error in position estimation and a 16-17 % error in interaction force estimation. With the real robot, the particle filter on the graph showed slightly better localization than the approach based on the spherical model but had a slower update loop, 60 Hz vs. 100 Hz. The examples of identification using particle filter on graph estimation for real KUKA iiwa and UR10e robots are presented

in Fig. 2.11. The accuracy of contact localization is 4.2 cm, for the particle filter with 200 particles. The run-time is about 16 ms, which is significantly better than the original approach with the reported 100 ms run-time. However, the accuracy of the graph-based method is lower due to the estimation for a set of fixed points, where the original approach does not have such limitations. The localization of multiple solutions is not possible in both cases, since the particle filter tends to converge to one point.

2.6 Approach based on clustered robot surface representation

This section presents another way of finding a solution for the system (1.11). The original system of static equilibrium equations has two unknowns, $\mathbf{p}$ as a contact point and $\mathbf{F}$ as an interaction force. In particular, the point of contact $\mathbf{p}$ is constrained by the known surface $\mathbf{\Omega}_k$ and an interaction force $\mathbf{F}$ by the normal vector in the contact point $\mathbf{n}_p$ and a friction coefficient μ. Since the robot surface is usually formulated as a triangular mesh, it already includes information about the link surface and link normal vectors. Thus, by analyzing the robot mesh, one can reduce the search space of the problem and as result reduce computational complexity, enabling real-time execution. However, robot mesh is mostly used for visualization, therefore should be modified.

One way to avoid iteration through all points on the link surface is to introduce a tree-like structure. The idea behind that is to divide the surface into segments, where each segment will be represented by the reference point. One significant requirement for the reference point and all segment points is that they should have similar properties in a sense of position and orientation. According to that, points in one segment should be clustered by their position and their normal orientations, so classical 3D clustering algorithms like k-means are not applicable. In this way, the robot will be the root of a tree, the first layer of leaves are links, each link has leaves as segments and each segment has bottom leaves as points on the surface. By isolating the link with a potential collision, and evaluating the residual (1.18) in the reference points of this link, the segment with the lowest residual value can be chosen to finally search the contact point only in this segment. As result, the number of evaluations can be greatly decreased, but only if the right segment is chosen.

The problem of multiple solutions is solved by estimating the interaction force in the last link with non-zero torque in its joint and then recursively checking for an intersection between this force line of action and the next joint axis. If the line of force action intersects the axis, then this force could also appear on the next link.

2.6.1 Robot model: link representation by hierarchical clusters

Let us assume that the surface is described by a discrete triangular mesh which consists of a finite number of vertices $\mathbf{v} \in \mathbb{R}^{3\times m}$ and a list of facets $\mathbf{f} \in \mathbb{R}^{3\times n}$. Each facet consists of three edges between three points from $\mathbf{v}$. In addition, each vertex has a normal vector $\mathbf{n} \in \mathbb{R}^{3\times m}$ that locally follows the surface normal.

The segmentation or clustering allows to group a large number of vertices and represents them by one reference point. The first approach for making k clusters is to sample k uniformly distributed points on the robot surface and then use a Voronoi segmentation for all vertices. This can be achieved by the Farthest Point Sampling algorithm and following Lloyds relaxation. The algorithm for this is presented in Algorithm 3. Below, this method will be described in more detail.

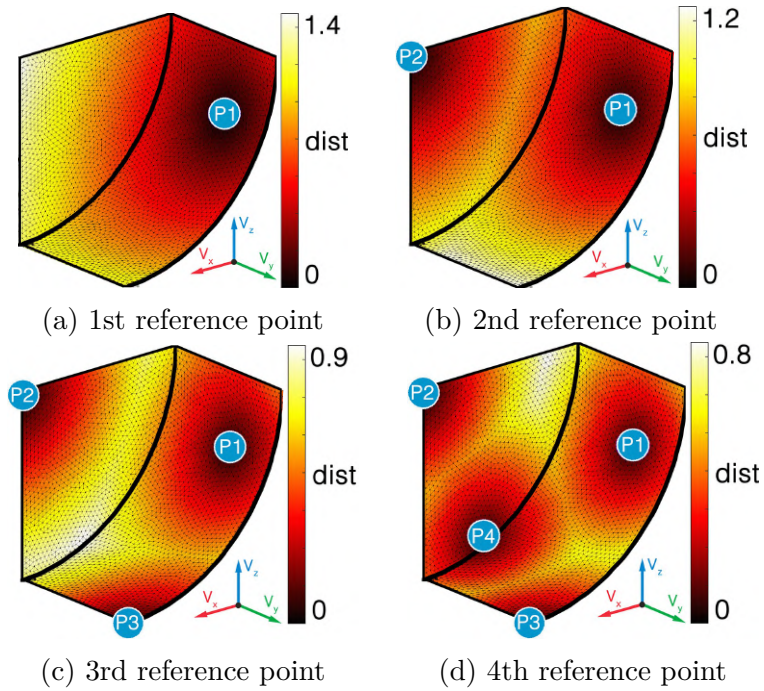
Suppose there is a triangular mesh that is divided into four segments. The first step is to sample the first segment reference point by randomly choosing it from the vertices list and then find a distance to all other vertices $\mathbf{v}$ (Fig. 2.12a). The distance function here is not a Euclidean distance, but a geodesic distance that shows the distance on the mesh surface. The presented algorithm is implemented using the heat method [84] as a part of *gptoolbox* [85] software.

In the second step after measuring the distance to all points, select the farthest one as a second cluster reference point (Fig. 2.12b). Now the distance to some arbitrary point $\mathbf{p}$ on the surface can be estimated as the minimum distance from existing reference points $dist = \min(\text{geoDist}(\mathbf{P1}, \mathbf{p}), \text{geoDist}(\mathbf{P2}, \mathbf{p}))$. This step is repeated until the number of segments reference points is equal to the desired number of clusters, as shown in Figs. 2.12c and 2.12d.

The next step is to apply the Voronoi partition and assign a cluster to all points on the surface. It can be done to any surface point by finding the closest reference point and assigning the same class. The clustering result is shown in Fig. 2.13a. Since the reference points were fixed during the previous step, obtained clusters are unbalanced, and reference points are sometimes located in the corners of its segments (**P2** and **P3**).

To balance the clusters and move the reference points Lloyds relaxation is used. By iteratively moving the reference points to the center of its segments, acquire balanced segments with almost the same size and reference points in their centers (Fig. 2.13b). It should be noted that the choice of initial point does not show any significant difference in the final cluster size or quality.

If we look at the first and the fourth cluster with reference points **P1**, **P4**, they include two kinds of points: one of them lies on the vertical wall of a mesh, and the second on the curved surface. Vertices normals in these two groups will be orthogonal,



(a) 1st reference point (b) 2nd reference point

(c) 3rd reference point (d) 4th reference point

Figure 2.12: The typical example of the Farthest Sampling Point algorithm. The color shows the geodesic distance (the distance on the surface of the mesh) between reference points and all mesh vertices.

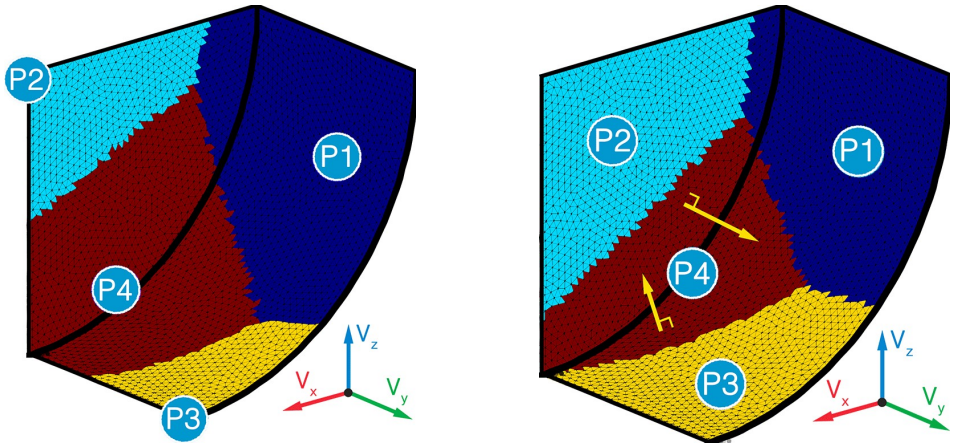
thus the reference point is capable of describing a position of neighborhood vertices, but not able to describe their orientation or normals. Although it is possible to increase the number of clusters, it will not guarantee that two points in the same cluster will have orthogonal normal vectors.

To overcome this problem, one can repeat the same steps of Algorithm 3, but for the normal vectors $\mathbf{n}$. The result for cluster segmentation in the normal vector space is shown in Fig. 2.14a. By mapping clusters back to the vertices positions $\mathbf{v}$, the clusters are now grouped according to their normals, all vertices from the vertical wall are in the same cluster and a curved surface is divided by other three clusters (Fig. 2.14b).

The robot mesh preprocessing procedure is executed offline and allows to obtain the tree-like structure for each link. In this procedure, all robot link meshes are clusterized in a similar way that was described in a previous section.

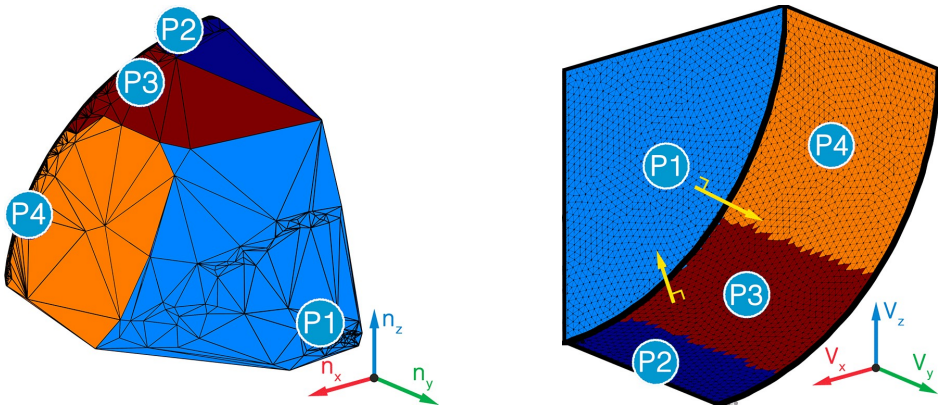
Usually, link mesh files are derived in a *.STL* format with a list of vertices coordinates $\mathbf{v}$, a list of facets $\mathbf{f}$, and a list of normal vectors for each facet. The image of such mesh for the second link of the KUKA iiwa robot is shown in Fig. 2.15a.

The standard optimization routine for localization assumes a continuous function for the potential contact point $\mathbf{p}$, so in the case of a triangular mesh, it means that the contact point could be in a vertex or a facet of a mesh. The searching process implies a gradient descent-like method to identify the exact location of the local minimum



(a) Initial partition (b) After relaxation

Figure 2.13: Example of clusters normalization process. (a) shows segments for the reference points located as in Fig. 2.12d, clusters size is unbalanced, reference points are static. (b) shows segments after Lloyds relaxation, where reference points iteratively moved to the center of each cluster.



(a) Clustering in $\mathbf{n}$ (b) Mapping to $\mathbf{v}$

Figure 2.14: Clusters in a normal vector space. Applying farthest Point Sampling and Lloyds relaxation for $\mathbf{n} = (n_x,\ n_y,\ n_z)$ (a) and projecting assigned clusters into Cartesian space for $\mathbf{v} = (v_x,\ v_y,\ v_z)$ (b).

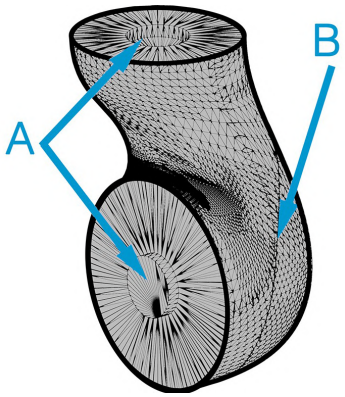
Algorithm 3: Link mesh clustering

```
Function geoCluster(v, f, num_cl):
    Input  : Mesh geometry (v, f), number of clusters num_cl
    Output: List of clusters c
    1st step: Farthest Point Sampling;
    g = random(v);
    for 1 : num_cl do
        D = geoDist(v, f, g);
        n_ind = find(max(D));
        g.add(n_ind);
    c = voronoi(v, f, g);
    2nd step: Lloyd's Relaxation;
    while c ≠ c_prev do
        for each c do
            vi, fi = c(v, f);
            g = findGeoCenter(vi, fi);
        c = c_prev;
        c = voronoi(v, f, g);
```

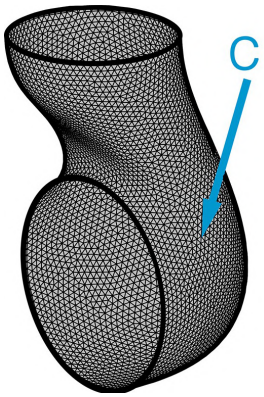
on the surface $\mathbf{\Omega}_k$ and a lot of computational power. At the same time, the noise in external torque measurements will limit the accuracy of localization. Thus, in this approach, a set of equally distributed points are scattered on the robot surface and estimate contacts only in these points. The average distance between scattered points is set to a value, which is several times lower than the desired localization accuracy. To obtain a new mesh with equally scattered vertices, an isotropic remesh algorithm from [86] can be used. In addition to remeshing, the original link mesh file (Fig. 2.15a) was manually edited to remove internal areas of the link (A). The resulting mesh is presented in Fig. 2.15b. The area marked with (B) had very inconsistent triangles sizes compared to the same region of the remeshed link (C).

Mesh clustering could be done using Algorithm 3 for a vertices coordinates $\mathbf{v}$. The output of this algorithm is shown in Fig. 2.15c. Again, as was described previously, two points of the same segment (shown in the zoomed image of Fig. 2.15c) have almost orthogonal normals directions. This will result in a bad representation of a cluster with a reference point. To overcome this, Algorithm 3 was used in a vertices normals space $\mathbf{n}$ (Fig. 2.15d). The same zoomed region now shows that there are two separate clusters. Despite that, obtained clusters are unbalanced in size (D) and can have unconnected parts. Therefore it is proposed to use two-step clustering.

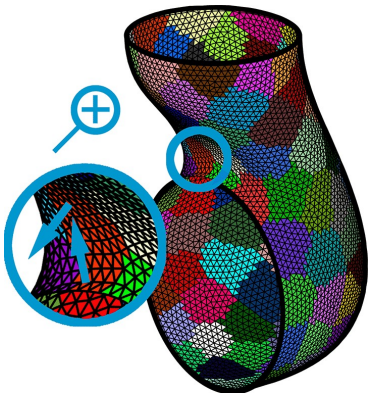
The stages of a two-step clustering, used for the link mesh preprocessing are demonstrated in Algorithm 4. The first step is the clustering of a link in a vertices normals



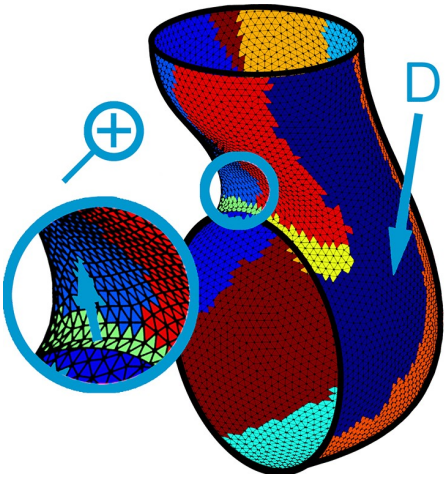
(a) Initial triangular link mesh imported from *.stl*.



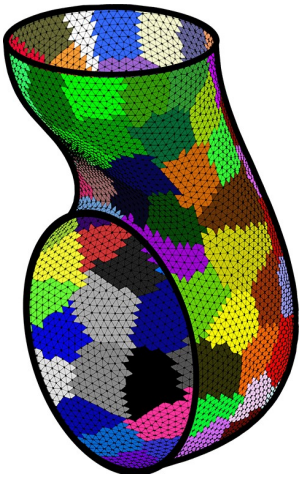
(b) Isotropic remesh of the (a)



(c) Clustering result for vertices coordinates ($v_x,\ v_y,\ v_z$)



(d) Clustering result for vertices normals ($n_x,\ n_y,\ n_z$)



(e) Two step clustering

Figure 2.15: Link mesh clustering steps. Imported mesh (a) from *.STL* model has unreachable areas (A) and inconsistent vertex distribution (D). This mesh is preprocessed (b) by removing unnecessary areas and remeshed by uniformly placed vertex points (C). In (c) Algorithm 3 it was applied for vertices coordinates, a zoom-up image shows that two points in the same segment have almost perpendicular normals. This can be avoided by applying the same algorithm to vertices normals (d), but one can expect clusters with inconsistent size (D) and unconnected areas. As an alternative, a two-step clustering algorithm (e) is proposed (Algorithm 4) which eliminates the disadvantages of (c, d).

Algorithm 4: Mesh preprocessing

```
1  Function clusterMesh(file):
       Input  : Mesh file in .STL format
       Output: Data structure link
2      d_size desired size of a cluster;
3      v_stl, f_stl, fn_stl = importSTL(file);
4      v, f, fn = isoRemesh(v_stl, f_stl, fn_stl);
5      vn = calcNormals(v, f, fn);
6      cn = geoCluster(vn, f, n_cl);
7      for each cn do
8          c_i = connectedComponents(cn);
9          for each c_i do
10             if size(c_i ≥ d_size) then
11                 n_cl = ⌊size(v(c_i))/d_size⌋;
12                 c = geoCluster(v(c_i), f(c_i), n_cl);
13             else
14                 c = c_i;
15             store c data in a link;
```

space, and after that, the clustering of all obtained clusters on subclusters. In this way, one can achieve clusters of the desired size and approximately the same orientation. The result could be evaluated in Fig. 2.15e.

Compared to traditional clustering with a weighted objective function that considers vertices' position and normals, this approach better describes small details. Traditional clustering with a weighted coefficient between vertices positions and normals (separated optimization is required to find optimal weights) will not give clusters with a fixed maximum size and most likely miss-classify minor parts of the surface. The two-stage algorithm generates clusters with limited maximum size to ensure algorithm run-time and is not limited to minimum size to preserve smaller surface elements and features.

After the segmentation of mesh vertices, data about each segment is stored in a **Robot** structure. Information about stored values is presented in Fig. 2.16, where the robot is represented as a structure with several links and corresponding segments.

2.6.2 Implementation of identification algorithm for clustered structure

In contrast with other available approaches, the algorithm is capable of estimating multiple solutions for the cases where the force vector intersects the last joint axis. Algorithm 5 consists of two parts, the first part (*localization_link*) deals with local-

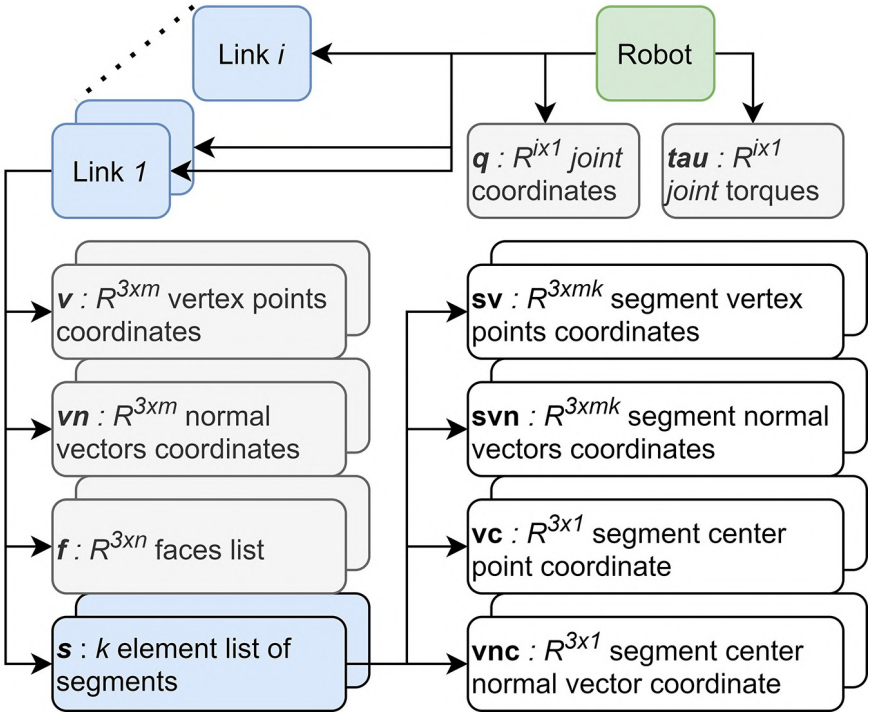


Figure 2.16: Data structure for the robot state and internal surface representation.

ization on the robot link, and the second part (*localization_robot*) with the whole robot. Let us consider the first part in more detail.

As was mentioned previously, it is convenient to exploit the proposed tree-like structure of the robot by estimating residual $\mathbf{r} = ||\boldsymbol{\tau}_r - \boldsymbol{\tau}||$ in the cluster reference points. For the reference points, their location in the global and local frame is known, the normal vector is given, and joints position and torque are known from the robot state. Consequently, the interaction force vector $\mathbf{F}$ could be estimated using pseudo inversion. Typically there will be at least two points where the line of external force action intersects the link surface, but only one of them is possible due to friction cone constraint. Here the initial residuals are set as a large value for the points where the estimated interaction force is out of the friction cone. Next, pick only $\mathbf{n}$ segments with the lowest residual $\mathbf{r}$ in their reference points and find the residual for the vertices of the segments. In this way, there is no need to iterate through all link vertices, which obviously reduces computation requirements. It should be noted that instead of pseudo inversion, a quadratic program (2.8) with a friction cone approximation can be solved.

At the last stage, it is important to check the possibility of multiple solutions. The multiple solutions on the same link are basically multiple minima on the already estimated residual $\mathbf{r}$, so it is sufficient to output the local minima. However, the external force, estimated in the current link could intersect the following link joint axis and generate zero torque in it. To address this issue we introduced a simple metric of

Algorithm 5: Localization on the robot surface

```
1  Function localization_robot(Robot):
       Input  : Structure Robot, the previous contact point p_prev
       Output: Interaction force F, its application point p,
2      if isEmpty(p_prev) then
3          isolate last link with nonzero torque;
4          link = lastNonZero(abs(τ) > ϵ);
5      else
6          link = link with p_prev;
7          link.c = cluster with p_prev;
8      F, p, r, d = localization_link(link, Robot);
9      while abs(d) < ϵ_d & link < link_max do
10         link++;
11         F, p, r, d = localization_link(link, Robot);
12     p = find(min(r));
13     p_prev = p;
```

```
1  Function localization_link(link, Robot):
       Input  : Link index link, Structure Robot
       Output: Link interaction force F_l, application point p_l, residual r,
               dist. between F_l and joint axis d
2      for each Robot.Links(link).vc do
3          F, r = estimate_F(ind, Robot);
4      pick top n clusters c_top with min r;
5      for each c_top do
6          for each Robot.Links(link).c(c_top).v do
7              F, r = estimate_F(ind, Robot);
8      p_l = find(min(rp));
9      F_l = F(p_l);
10     check for joint sensor axis intersection;
11     O = link + 1-th joint axis;
12     d = dist(F, O);
```

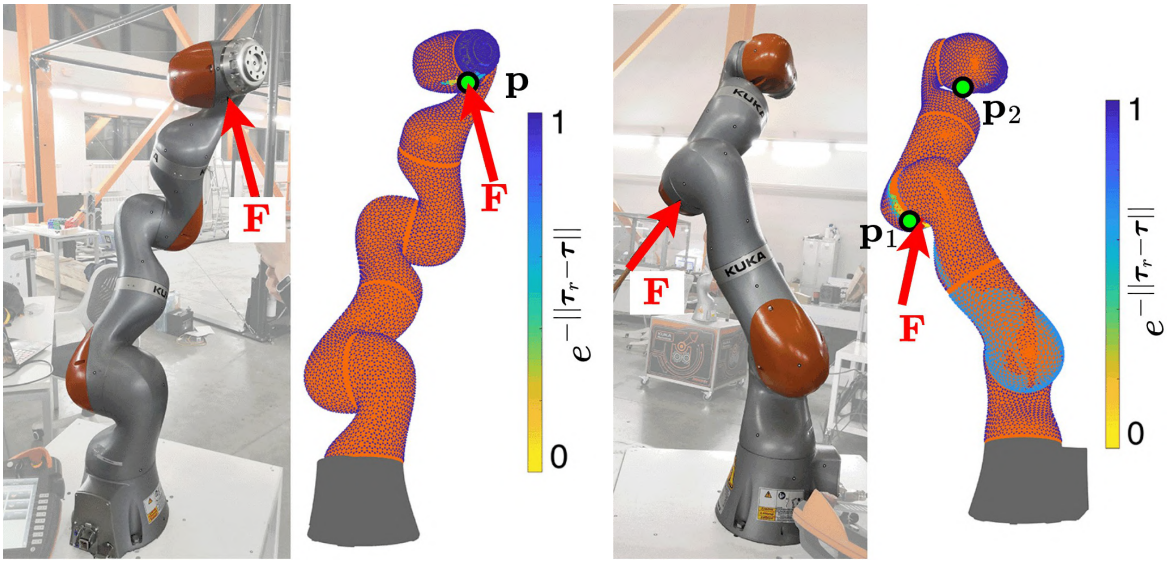
```
1 Function estimate_F(ind, Robot):
      Input  : Surface point index ind, structure Robot
      Output: Interaction force F, residual r
2     J = J_ind(Robot.Links(link).v(ind));
3     F = (J J^T)^-1 J τ;
4     if ∠(F, Robot.Links(link).vn(ind)) < atan(μ) then
5         τ_t = J^T F;
6         r = ‖τ − τ_t‖;
7     else
8         r = Inf;
```

a distance **d** between the estimated external force and the following joint axis. If there is an intersection, then the next link is evaluated for a collision too.

The localization for the whole robot starts with link isolation. Firstly, the link following the last non-zero torque is isolated. In the practice, of course, there are some threshold values that depend on the joint torque noise. The isolated link also could be extracted from the estimated collision location from the previous timestamp (**p_prev**) if there is one. This isolated link is evaluated for a collision in the *localization_link* routine and the following multiple solution evaluation. In this evaluation, compare the distance between the estimated line of force action and the joint axis. In other words, if there is an intersection, start to iteratively check the adjacent links until there will be no intersection, until the last link. In this way, one can acquire multiple solutions, one from the considered link and additional solutions from the following links.

Consider the example in Fig. 2.17a. The KUKA iiwa robot has a collision at the 6th link. The last joint with non-zero torque is the 6th joint, so isolate the 6th link and try to estimate a contact point on it. There are two possible solutions, one of them is where the actual external force is applied and the second one is on the other side of the link, which represents the same force but in the opposite direction. The second potential contact point is ignored due to the friction cone constraint. The estimated force for the first potential contact point does not intersect the axis of the 7th joint. Thus, the robot in a given pose and interaction has only one solution.

Another example in Fig. 2.17b shows multiple minima behavior. The real contact is at the 4th link and the 4th joint is the last joint with non-zero torque. The external force, estimated after localization on the 4th link will intersect the axis of the 5th and 6th joint. By searching for another collision point for these links, there is another solution at the 6th link. From the current robot state, it is not possible to decide which of them is a true contact location. So, both of the points are marked as the potential solution until changes in robot pose or force will leave only one of them.



(a) Contact at the 6th link (b) Contact at the 4th link

Figure 2.17: Experimental validation of the algorithm based on clustered structure for KUKA iiwa robot. The green circle is the identified interaction force application point, the red arrow shows the applied interaction force.

2.6.3 Performance of clustered structure approach

The developed algorithm was tested on Intel Core i5-4210H 3GHz CPU, 8Gb RAM PC as a Matlab program. The result of virtual and real experiments is presented in Table 2.2. For the test, 5 different algorithm parameters were compared with one straightforward case, which assumed simple iteration through all vertices of robot mesh. The vertices on the mesh were located with a 0.5 cm mean distance between them. As a parameter, the desired size of a cluster d_size from Algorithm 4 was set as 10, 25, 50, 100 vertices, which give approximately 500, 200, 100, 50 segments per link. In

Table 2.2: Mesh parameters influence on the algorithm accuracy for simulated and real robot KUKA iiwa 14

Mesh parameters	**Accuracy,** $[cm]$		**Loop run-time**		
	Sim.	**Real**	$[ms]$	$[Hz]$	**IF***
All points	2.32	2.17	75	13.3	-
Cluster $d\_size = 100$	3.19	3.04	1.8	555.6	41.6
Cluster $d\_size = 50$	2.56	2.34	1.6	625.1	46.9
Cluster $d\_size = 25$	2.48	2.33	1.5	666.7	50.0
Cluster $d\_size = 10$	2.51	2.31	1.7	588.2	44.1

*Improvement Factor

the simulation scenario, $N(0, 0.5Nm)$ noise was added to the torque values to emulate dynamic model inaccuracies in external torque estimation. The accuracy metrics in this table correspond to the average distance between the ground truth point of contact and the closest estimated local minimum from a set of possible contacts. The run-time evaluates the average time to estimate the force application point.

According to the results, the presented algorithm allows to achieve 50x speedup for the localization, without significant loss of accuracy. The model with $d\_size = 100$ shows worse performance due to a large cluster that is hard to represent with one reference point, which leads to picking the wrong segments and results in the loss of accuracy. Using a very small cluster $d\_size = 10$ requires evaluating more segments and is slower due to increased cluster initialization time. The most optimal parameter is a $d\_size = 25$ since it gives a good balance between run-time and accuracy.

It should be noted that the accuracy of an algorithm also depends on the link with a collision. The last links tend to have better accuracy due to the larger number of sensors. With $d\_size = 25$, estimated accuracy in simulation with $N(0, 0.5Nm)$ noise is 0.9, 1.2, 2.2, 4.2 cm for the 7th, 6th, 5th, and 4th link respectively.

2.7 Summary of the chapter: comparison results

This chapter presents a comparative study of the existing approaches for interaction parameters identification, which are used for the modeling of physical interaction between a robot and a human. In order to detect their weaknesses, several techniques mentioned in Section 1.2 were implemented and carefully studied via simulation and real-life experiments. For the **simulation study**, 1000 random robot poses were generated and external force was applied at 22 different points, which gives 22000 samples in total. All examined algorithms were tested on Intel i5-4210H 3GHz CPU, 8Gb RAM PC in a Matlab environment. The obtained results are summarized below.

The first technique (Section 2.2) is based on the simplified robot model. Its main idea is to represent the complex shape of the robot surfaces as a set of cylinders. This allows parameterizing the identification problem using two variables only. In such simplification, the interaction force is assumed to be co-linear to the normal at the contact point. As follows from the simulation study, this technique provides not only relatively low accuracy because of very rough robot shape approximation but also a rather high run-time despite of apparent simplicity.

The second group of techniques (Section 2.3) is based on feed-forward neural networks, which are trained for the interaction force application point identification. One of them uses cylindrical parameterization and predicts the point based on measured

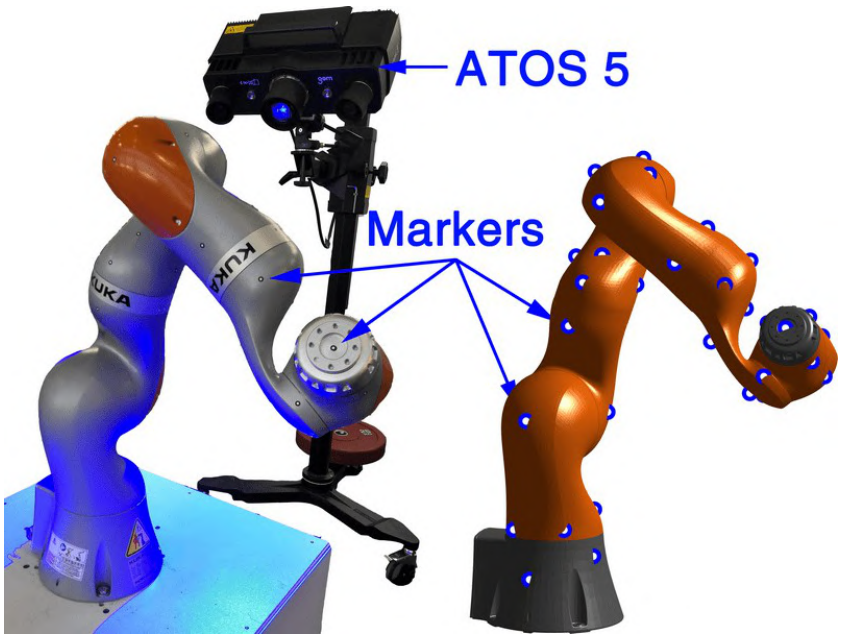


Figure 2.18: Experimental setup: real and simulated KUKA iiwa with ATOS5 markers. By measuring markers coordinate in the robot base frame, it is possible to match markers position in the real robot and its virtual model, to obtain ground truth.

torque values and the robot pose. Another one presents the original identification problem as a classification task, where each class corresponds to some fixed point on the robot surface. To apply these two approaches to real case studies, the basic methodology of transfer learning was used. As follows from relevant numerical results, the run-time for such a technique is rather low but the accuracy is very limited.

The third technique (Section 2.4) is based on sphere mapping of the robot surface. The main idea of the method is to transform the complex shape of the robot surface into a sphere, find the interaction parameters in the sphere space and obtain the desired parameters in the initial space using the inverse transformation. In the frame of this technique, a real-time algorithm based on a two-step optimization procedure was implemented. In the first step, an approximate position of the contact point is estimated, while in the second step, a local optimization is applied to find interaction parameters more accurately. As follows from the relevant simulation study, this technique provides a rather mediocre accuracy and run-time, which is not good enough for real-time applications.

The fourth technique (Section 2.5) is based on the particle filter on the graph. It utilizes basic ideas of the Contact Particle Filter proposed by Manuelli and Tedrake but with some enhancements allowing to reduce the run-time. This improvement was achieved by representing the robot surface as a graph, which leads to a reduction of computational complexity for one of the most time-consuming steps from the original

Table 2.3: Experimental results on accuracy and runtime of existing approaches

Approach	**Robot**	**Δp**	**T_c**
Surface approximation approaches			
Cylinder approx. [Section 2.2]	KUKA iiwa	5.4	1/19
Sphere mapping [Section 2.4]	KUKA iiwa	4.2	1/100
DIRECT [78]	KUKA iiwa	6.0	1/30
DIRECT-link [57]	Kinova Jaco2	12.0	1/20
Monte Carlo based approaches			
PF on graph [Section 2.5]	KUKA iiwa	4.0	1/60
Contact particle filter [87]	KUKA iiwa	2.4	1/10
PF with FC binning [88]	Kinova Jaco2	11.0	1/63
PF with binning [88]	Kinova Jaco2	12.0	1/71
PF with nearest neighbor [88]	Kinova Jaco2	11.0	1/125
PF with weighted means [88]	Kinova Jaco2	11.0	1/159
Machine learning based approaches			
Feed-forward NN [Section 2.3]	KUKA iiwa	6.4	1/180
Random forest [57]	Kinova Jaco2	8.0	1/200
Multilayer perceptrons [57]	Kinova Jaco2	4.0	1/200
Clustered surface approach			
Clustered surface [Section 2.6]	KUKA iiwa	2.3	1/600

Note: **Δp** - estimation error of the force application point $[cm]$;
T_c - cycle time of the estimation algorithm $[1/Hz]$.

method. The simulation results demonstrated the high accuracy of this technique but its run-time is still rather high, which makes it unattractive for real-time applications.

Finally, the fifth technique (Section 2.6) is based on clustered robot surface representation. Its main idea is to transform the robot surface from an initial mesh to a hierarchical structure obtained by robot link surfaces preprocessing. This preprocessing includes isotropic remeshing and two-step clustering in the space of normal vectors and position of the vertices. As follows from the simulation study, this technique allows achieving high accuracy and very low run-time, which makes it attractive for application in real-time. Moreover, it is able to detect multiple solutions for desired interaction parameters for some specific singular cases, where the force action line intersects several joint axes. Nevertheless, more general singular cases with multiple solutions are not treated by this technique.

It is worth mentioning that all the above techniques were also carefully studied in **real-life experiments** with the KUKA LBR iiwa 14 collaborative robot. For these experiments, 10 random manipulator configurations were used; for each of them the interaction force was applied to one of 22 marked points located at the robot links #4,...,7, which gives 220 experimental records in total. In order to estimate the coordinates of the force application points directly, the ATOS 5 measurement system was used, which provides a precision of about ± 0.03 mm. The general experimental setup including the KUKA iiwa robot and ATOS 5 system is shown in Fig. 2.18. During the experiments, the operator applied the interaction force only at these marked points with a known location. It should be noted that direct measurements of the force amplitude and direction were not included in the experimental study, since the force application point is more essential for the interaction handling in collaborative robotics.

The obtained experimental results are presented in Table 2.3, which includes the force application point errors $\Delta\mathbf{p}$ and the algorithm cycle time $\mathbf{T}_c$. It should be stressed that this table also contains some experimental data from relevant literature obtained for both the same robot KUKA LBR iiwa and the collaborative robot Kinova Jaco. It is clear that for the cases from the literature, the run-time performance corresponds to different hardware providing different computational power. Nevertheless, they also give some useful information for comparison analysis.

Finally, the comparison study of the existing techniques presented in this section can be summarized as follows:

1) **Accuracy and robustness**. For most of the existing algorithms, the force application point estimation error is in the range of 4 ... 12 cm. Only a few algorithms provide a lower value of this error $\Delta\mathbf{p} < 4$ cm. It is clear that high force position errors make the methods unattractive for human-robot interaction handling since it is comparable to the size of a human hand. Moreover, the joint torque measurements are usually corrupted by noise, which also affects accuracy. However, none of the studied approaches takes into account this issue and provides the desired robustness with respect to the measurements noise.

2) **Real-time performance**. The cycle time of the studied approaches varies from 1/10 to 1/600 second, depending on the computational complexity of the corresponding algorithm. As follows from relevant analysis, only a single approach from the considered ones is suitable for real-time applications. It allows to estimate the desired interaction parameters 500+ times per second, which is comparable with a control loop frequency of a typical collaborative robot. Other approaches, cannot be used for human-robot interaction handling because of significant reaction time, critical for the human safety.

3) **Singularities and multiple solutions**. As follows from the presented study, none of the existing approaches is able to identify the interaction parameters in singular cases, where the solution of the corresponding system of static equilibrium equations is non-unique. In such cases, the existing techniques usually provide a single solution from the set of possible ones, which may be essentially different from the real one. In particular, this singular solution may even correspond to the wrong link number, which is clearly useless for proper human-robot interaction handling. Although, it is worth mentioning that this issue was partially addressed in the clustered surface-based approach that can provide multiple solutions but only for a limited type of singularities.

Therefore, as follows from the comparison study presented in this chapter, none of the existing approaches completely satisfy requirements of the real-time human-robot interaction handling, especially in some singular cases and in the presence of the measurement noise essentially influencing the identification accuracy. These issues motivate the development of a new technique presented in the following chapters.

Chapter 3

Development of enhanced interaction parameters identification technique: 2D planar case

Contents

This chapter deals with interaction parameters identification for the 2D planar case. It proposes a new fast non-iterative technique for computing the interaction force and its application point using measurement data obtained from the internal joint torque sensors only. The proposed algorithm is based on a parametrized general solution of a static equilibrium equation allowing to find the desired parameters even in the case of essential ambiguity. In contrast to existing approaches, the proposed method is applicable for singular cases associated with an insufficient number of independent equations in a static equilibrium system, which produces non-unique solutions for the interaction parameters.

3.1 Introduction

The chapter deals with the parameters estimation in the human-robot collaboration scenario. It presents an analytical algorithm for computing the interaction force and its application point using measurement data obtained from the internal joint torque sensors only. The proposed algorithm is based on a specific extension of static equilibrium equations allowing to find the desired interaction force and action line. Further, this general solution is combined with geometric constraints describing manipulator surfaces and corresponding friction cones. Particular attention is paid to singular cases, which arise when there is an insufficient number of independent equations in relevant static equilibrium system. The developed technique was carefully evaluated via the simulation study with a planar robot involving measurements noise.

As was shown in the previous chapter, most of the existing approaches for estimation of human-robot interaction parameters are based on minimizing the squared residuals of the static equilibrium equations, subject to some constraints arising from both the robot shape and the contact surface property. For instance, the contact point should be clearly located on the robot link surface and the force direction should be close to the surface normal. In recent years several different techniques for estimation of the interaction parameters were developed, which are based on either straightforward nonlinear constrained optimization (see [28]), or Monte-Carlo algorithms (see [29], [30]), or other methods (see [60], [75]). Generally, these approaches provide good accuracy but possess rather low computation speed. The latter is caused by the iterative nature of the relevant numerical algorithms requiring numerous evaluations of the objective function for the interaction parameters search. Other difficulties arising here are associated with an inability to handle interactions of singular or close-to-singular manipulator configurations, where multiple minima of the objective functions exist, and the interaction cannot be identified in a unique way. Thus, the existing techniques do not perfectly suit the engineering requirements imposed by online implementation.

The main problem considered in this chapter is to estimate the human-robot interaction force and its application point using data obtained from internal torque sensors embedded in the robot joints. It is assumed that an operator interacts with a general n-dof planar serial manipulator, which consists of a fixed base, end-effector, and a number of links connected by n revolute joints. Besides, it is also assumed that the interaction force could be applied to an arbitrary point on the manipulator link surface. It should be mentioned that only the point contact between the operator and the robot is considered here, implicitly assuming that any surface-to-surface contact (or more complicated one) can be efficiently approximated by point-to-point interaction.

3.2 Identification of interaction parameters for non-singular case

As follows from a number of related works, the above optimization problem (1.10) may have either a unique or non-unique solution. Let us start from a non-singular case when there exists a unique solution for both the force $\mathbf{F}$ and its application point $\mathbf{p}$. In this case, all static equilibrium equations are completely satisfied and the considered objective function reaches its minimum possible value that is obviously equal to zero. To simplify the process, the problem will be solved in two steps, where the first one ignores the geometric constraints and the second one takes them into account in order to obtain the desired solution.

3.2.1 Computing of interaction force and its action line

Let us consider first the static equilibrium equations only, omitting the constraints on the unknown variables $\mathbf{F}, \mathbf{p}$. This allows us to obtain a general solution of these equations that will be further integrated with the constraints $\mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. The desired solution will be found in a closed form using a specially developed technique based on a proposed extension of the original system and its subsequent reduction.

Before focusing on the general solution, let us concentrate on some important properties of the considered algebraic system (1.5). It is worth mentioning that the total number of equations here is equal to k, which depends on the link index where the interaction force is applied to. At the same time, the number of unknowns in this system is always equal to four; they include the components of the force (F_x, F_y) and the coordinates of the contact point (x, y). Generally, if such a system is consistent, it may have either single or multiple solutions. However, for our particular system, the unique solution is not possible, since physically the application point $\mathbf{p}$ can be moved along the force action line without violating the static equilibrium equations. Mathematically this property can be formulated in the following way.

Statement 1. *If the force vector* $\mathbf{F}$ *and its application point* $\mathbf{p}$ *satisfy the static equilibrium equations (1.5), then the set of solutions with the same* $\mathbf{F}$ *and the application point* $\mathbf{p}$ *belonging to the line* $\mathbf{p}^* = \mathbf{p} + \alpha \cdot \mathbf{F}$, $\forall \alpha \in \Re$ *also satisfy these equations.*

Proof. This statement can be easily proved by straightforward substitution of the shifted point $x^* = x + \alpha \cdot F_x$; $y^* = y + \alpha \cdot F_y$ into system (1.5), which after simple transformation yields an additional component of the form $\alpha \cdot 0$ for each equation that does not violate the original equality.

□

To find a desired closed-form solution of the considered system (1.5), let us introduce an extended system of static equilibrium equations of the following form

$$\left[\mathbf{J}_k^w(\mathbf{q},\mathbf{p})^T\right]_{k\times 3}\cdot\begin{bmatrix}F_x\\F_y\\M_z\end{bmatrix}=\begin{bmatrix}\tau_1\\\dots\\\tau_k\end{bmatrix} \tag{3.1}$$

where the unknown force vector $\mathbf{F} = [F_x, F_y]^T$ is replaced by the wrench $\mathbf{W} = [F_x, F_y, M_z]^T$ including the torque M_z. Consequently, the original $2 \times k$ Jacobian (1.7) is replaced by the extended matrix of size $3 \times k$

$$\mathbf{J}_k^w(\mathbf{q},\mathbf{p})=\begin{bmatrix}-(y-y_1) & \dots & -(y-y_k)\\ (x-x_1) & \dots & (x-x_k)\\ 1 & \dots & 1\end{bmatrix}_{3\times k} \tag{3.2}$$

Let us also assume that the system is consistent, and the force application point $\mathbf{p}$ is known. Then the extended system can be easily solved for $\mathbf{W}$ using the conventional least squares technique, which yields

$$\mathbf{W}=\left[\mathbf{J}_k^w(\mathbf{q},\mathbf{p})^T\right]^{\#}\cdot\boldsymbol{\tau} \tag{3.3}$$

where $\boldsymbol{\tau} = [\tau_1, \dots, \tau_k]^T$ is torque vector and $[.]^{\#}$ denotes the MoorePenrose matrix pseudoinverse. Moreover, it can be proved that, for any arbitrary $\mathbf{p}$, the obtained wrench $\mathbf{W}$ always provides us with desired components of the interaction force F_x, F_y from the original system (1.5). Mathematically this result could be formulated in the following way.

Statement 2. *If the original system (1.5) includes $k \geq 3$ consistent static equilibrium equations and exists some point $\mathbf{p}^*$ providing full rank $rank(\mathbf{J}_k^w(\mathbf{q},\mathbf{p}^*)) = 3$ for the extended system, then the solution of the extended system (3.1) $\mathbf{W} = [F_x, F_y, M_z]^T$ is unique and yield the desired force vector F_x, F_y for the original system.*

Proof. Let us assume that $\mathbf{W} = [F_x, F_y, M_z]^T$ is solution of the extended system for some guess point $\mathbf{p}^0 = [x + \Delta x, y + \Delta y]^T$ which differs from the real force application point x, y by some shift $\Delta x, \Delta y$. It means that for all $i \leq k$ the following equations are satisfied

$$\tau_i = -(y + \Delta y - y_i)\cdot F_x + (x + \Delta x - x_i)\cdot F_y + M_z, \qquad \forall i \leq k \tag{3.4}$$

that can be further transformed into

$$\tau_i = -(y - y_i) \cdot F_x + (x - x_i) \cdot F_y + [M_z - \Delta y \cdot F_x \ + \Delta x \cdot F_y] \tag{3.5}$$

The latter shows that there is strict correspondence between the solution of the original and the extended systems

$$\{\ (F_x, F_y), \quad (x + \Delta x, y + \Delta y)\ \} \ \longleftrightarrow\ \{\ (F_x, F_y, M_z), \quad (x, y)\ \} \tag{3.6}$$

where the force components F_x, F_y are exactly the same and the torque M_z depends on the initial point $\mathbf{p}^0 = [x + \Delta x, y + \Delta y]^T$ as follows

$$M_z = \Delta y \cdot F_x - \Delta x \cdot F_y \tag{3.7}$$

Hence, if the initial guess $\mathbf{p}^0 = [x^0, y^0]^T$ coincides with the real force application point $\mathbf{p} = [x, y]^T$, then the extended system (3.1) provides us with the desired force components F_x, F_y and zero torque $M_z = 0$. □

Therefore, the above statement allows us to find the interaction force vector $\mathbf{F}$ using an arbitrary initial guess $\mathbf{p}^0$ for the force application point. Besides, it also allows to find the force action line that must obviously satisfy the equation (3.7) with $\Delta x = x^0 - x; \quad \Delta y = y^0 - y$. The latter yields the following equation of the interaction force action line

$$F_y \cdot x - F_x \cdot y = F_y \cdot x^0 - F_x \cdot y^0 + M_z \tag{3.8}$$

which will be further combined with the geometric constraint $\mathbf{p} \in \mathbf{\Omega}_k$ describing the shape of the kth manipulator link, where the interaction force is applied to.

In more detail, the desired closed-form solution can be obtained by expanding the matrix expression (3.3), which takes the following form

$$\begin{bmatrix} F_x \\ F_y \\ M_z \end{bmatrix} = \begin{bmatrix} S_{yy} & -S_{xy} & -S_y \\ -S_{xy} & S_{xx} & S_x \\ -S_y & S_x & k \end{bmatrix}^{-1} \cdot \begin{bmatrix} -S_{y\tau} \\ S_{x\tau} \\ S_\tau \end{bmatrix} \tag{3.9}$$

where

$$S_{xx} = \sum_{i=1}^{k} (x^0 - x_i)^2; \qquad S_{yy} = \sum_{i=1}^{k} (y^0 - y_i)^2; \qquad S_{xy} = \sum_{i=1}^{k} (x^0 - x_i)(y^0 - y_i);$$

$$S_x = \sum_{i=1}^{k} (x^0 - x_i); \qquad S_y = \sum_{i=1}^{k} (y^0 - y_i);$$

$$S_{x\tau} = \sum_{i=1}^{k} (x^0 - x_i) \cdot \tau_i; \qquad S_{y\tau} = \sum_{i=1}^{k} (x^0 - x_i) \cdot \tau_i; \qquad S_\tau = \sum_{i=1}^{k} \tau_i$$

Also, it can be easily seen that the above expression (3.9) can be essentially simplified by shifting the origin of the base coordinate system to the mean point of the set $\{x_i, y_i\}$, i.e. by using the following initial guess

$$x^0 = \overline{x}; \qquad y^0 = \overline{y} \tag{3.10}$$

where

$$\overline{x} = \frac{1}{k} \sum_{i=1}^{k} x_i; \qquad \overline{y} = \frac{1}{k} \sum_{i=1}^{k} y_i \tag{3.11}$$

which ensures that $S_x = 0$ and $S_y = 0$. Consequently, the system is reduced down to

$$\begin{bmatrix} F_x \\ F_y \\ M_z \end{bmatrix} = \begin{bmatrix} S_{yy} & -S_{xy} & 0 \\ -S_{xy} & S_{xx} & 0 \\ 0 & 0 & k \end{bmatrix}^{-1} \cdot \begin{bmatrix} -S_{y\tau} \\ S_{x\tau} \\ S_\tau \end{bmatrix} \tag{3.12}$$

where the analytical matrix inversion can be applied, yielding the following expressions for the force and torque components

$$\begin{aligned} F_x &= (S_{x\tau} \cdot S_{xy} - S_{y\tau} \cdot S_{xx})/D_F \\ F_y &= (S_{x\tau} \cdot S_{yy} - S_{y\tau} \cdot S_{xy})/D_F \\ M_z &= S_\tau / k \end{aligned} \tag{3.13}$$

where

$$D_F = S_{xx} \cdot S_{yy} - S_{xy}^2 \tag{3.14}$$

Summarizing the above results, it is worth mentioning that the above-presented technique, which is based on extending the original system of static equilibrium equations (1.5), allows us to estimate both the interaction force $\mathbf{F}$ and its action line using an arbitrary initial guess $\mathbf{p}^0$. However, by choosing $\mathbf{p}^0$ as the mean of the set $\{x_i, y_i, \;\; i = 1, ..k\}$ it is possible to reduce computational efforts essentially by applying the closed-form expressions (3.13).

3.2.2 Computing of force application point

In the previous subsection, the general solution of the static equilibrium equations was obtained providing us with the force vector $\mathbf{F}$ and its action line (3.8) while ignoring the geometric constraints $\mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. For this reason, this subsection focuses on applying these constraints to the obtained solution, which finally allows us to identify the desired interaction parameters $\mathbf{F}, \mathbf{p}$.

Let us assume that the interaction force $\mathbf{F}$ is applied to the kth link of the manipulator, where the k-index is estimated by using expression (1.6). According to the adopted robot model, the shape of this link is described by a closed polyline $\boldsymbol{\Omega}_k = hull\left(\left\{\mathbf{s}_1^k, \ldots, \mathbf{s}_m^k\right\}\right)$ composed of m line segments connecting the consecutive nodes $\mathbf{s}_i, \mathbf{s}_{i+1}$ as shown in Fig. 1.12. It is worth mentioning that further for simplicity the superscript k denoting the interacting link is omitted. Besides, for computational convenience, it is also assumed that all the nodes $\mathbf{s}_i$ are presented in the local frame of the kth link. Thus, the first geometric constraint $\mathbf{p} \in \boldsymbol{\Omega}_k$ is written in the form

$$\boldsymbol{\Omega}_k = \{\mathbf{s} \ \mid \mathbf{s} = \alpha \ \cdot \mathbf{s}_i + (1-\alpha) \cdot \mathbf{s}_{i+1}, \qquad \forall i = 1, ..m, \qquad 0 \leq \alpha \leq 1\} \tag{3.15}$$

The second geometric constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ associated with the friction cone can be also presented using the above notations. Assuming that the nodes are listed in clockwise order, it is easy to compute inward-pointing normal vectors (i.e. pointing towards the interior of the link) for each segment

$$\mathbf{n}_i = \mathbf{R}_{\pi/2} \cdot (\mathbf{s}_{i+1} - \mathbf{s}_i), \qquad \forall i = 1, ..m \tag{3.16}$$

where $\mathbf{R}_{\pi/2}$ is the conventional 2×2 orthogonal matrix representing the counterclockwise rotation by $\pi/2$. It should be mentioned that here the normal vectors $\mathbf{n}_i$ are expressed in the local coordinate frame of the kth link. For this reason, the interaction force vector $\mathbf{F}$ should be also presented in the kth link frame. Such transformation of the force vector from the global to the local coordinate frame can be easily achieved by using the appropriate 2×2 rotation matrix defining the mutual orientations of these frames.

Further, let us integrate the first geometric constraint $\mathbf{p} \in \boldsymbol{\Omega}_k$ describing the link shape with the force action line (3.8) from the previous subsection. It is clear that geometrically this is equivalent finding the intersection of a line and a segment on a plane. So, the desired force application point $\mathbf{p} = [x, y]^T$ can be easily found as line-line intersection, where the second line is obtained by extending the considered segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$. Using the adopted notation, it can be proved that the coordinates of this

point can be found from the following system of equations

$$\begin{cases} n_x \cdot x + n_y \cdot y = C_\Omega \\ F_y \cdot x - F_x \cdot y = C_F \end{cases} \tag{3.17}$$

where n_x, n_y are the components of the normal vector $\mathbf{n}_i$ (with the subscript i omitted for simplicity) and F_x, F_y are the force vector components. The right-hand side of this system is expressed as follows

$$\begin{aligned} C_F &= M_z + F_y \cdot x^0 - F_x \cdot y^0 \\ C_\Omega &= n_x \cdot s_x + n_y \cdot s_y \end{aligned} \tag{3.18}$$

where s_x, s_y are the components of the vertex $\mathbf{s}_i$. Thus, an analytical solution of this system can be presented as follows

$$\begin{aligned} x &= (C_\Omega \cdot F_x + C_F \cdot n_y)/D_p \\ y &= (C_\Omega \cdot F_y - C_F \cdot n_x)/D_p \end{aligned} \tag{3.19}$$

where

$$D_p = F_x \cdot n_x + F_y \cdot n_y \tag{3.20}$$

Since the above solution was obtained from the line-line intersection, it should be verified if it belongs to the considered segment, i.e. $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$. As known from analytical geometry, this condition can be presented in the following way

$$\left(F_y \cdot s_x^i - F_x \cdot s_y^i - C_F\right) \cdot \left(F_y \cdot s_x^{i+1} - F_x \cdot s_y^{i+1} - C_F\right) < 0 \tag{3.21}$$

where s_x^i, s_y^i and s_x^{i+1}, s_y^{i+1} are the coordinates of the vertices $\mathbf{s}_i$ and $\mathbf{s}_{i+1}$ defining the segment ends. It should be noted that this verification can be executed before computing the line-line intersection point (3.19), to ensure the line-segment intersection.

Finally, it is also necessary to verify the second constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ ensuring that the force $\mathbf{F}$ direction is feasible with respect to the friction cone at the interaction point $\mathbf{p}$ belonging to the segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$. Using the adopted notation, the relevant condition expressed by inequality (1.9) can be rewritten as

$$\left|F_x \cdot n_y^i - F_y \cdot n_x^i\right| \leq \mu \left(F_x \cdot n_x^i + F_y \cdot n_y^i\right) \tag{3.22}$$

where n_x^i, n_y^i are the coordinates of the normal vector $\mathbf{n}_i$ for the ith segment and μ is the friction coefficient between the manipulator link surface and the interacting object.

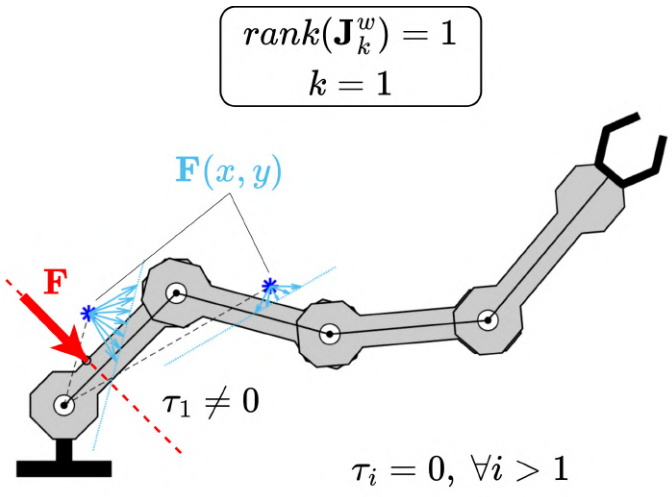


Figure 3.1: Example of singularity in static equilibrium equations for $k = 1$. The force is applied to the first link providing a single relation between $(F_x,\ F_y)$ and $(p_x,\ p_y)$.

Thus, the above-presented expressions allow us to find the coordinates of the interaction point $\mathbf{p}$ corresponding to the force $\mathbf{F}$ and its action line from the previous subsection, which satisfy both constraints $\mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. It is clear that relevant computations should be executed for each of m segments describing the kth link shape. Also to speed up the process, it is reasonable to perform the computations in the following order: (i) Verification of the friction cone constraint; (ii) Verification of the line-segment intersection existence; (iii) Computing of the line-segment intersection point (if exists). The latter allows us to exclude some unnecessary steps if at least one of the constraints is violated.

3.3 Identification of interaction parameters for singular cases

Let us concentrate now on singular cases when either the interaction force $\mathbf{F}$, its action line or the force application point $\mathbf{p}$ cannot be computed in a unique way. It is clear that these cases are associated with the rank deficiency of the extended Jacobian (3.2) which here is less than three, in contrast to the previous section. Besides, in the singular cases the objective function (1.10) does not reach its global minimum at a single isolated point, which yields multiple solutions of the considered problem that obviously must include the "true" one corresponding to the real physical interaction. Consequently, both the force vector, its action line, and the force application point may be expressed in a parametric form, which contains some arbitrary values.

3.3.1 Singularities in computing of interaction force and its action line

First, let us investigate in detail the singularities in the force and its action line computing, which arise when the solution for F_x, F_y, M_z is provided by expression (3.12) includes a non-invertible matrix. It is clear that in this case, the determinant D_F is equal to zero, which is possible when the extended Jacobian $\mathbf{J}_k^w$ is rank-deficient, i.e. when either $rank(\mathbf{J}_k^w) = 1$ or $rank(\mathbf{J}_k^w) = 2$. Typical examples of such singular cases are presented in Figs. 3.1 and 3.2, where the singularities arise because the interaction force is applied to the lower manipulator link resulting in an insufficient number of the static equilibrium equations $k < 3$. However, as follows from the relevant study, such singularities may exist even for $k \geq 3$ for particular manipulator configurations presented at the end of this subsection.

It is clear that the case $rank(\mathbf{J}_k^w) = 1$ is marginal since it provides us with a single independent equation for four unknown variables F_x, F_y and x, y. Since the original system (1.5) is consistent, any of the k equation can be used, i.e

$$(x - x_i) \cdot F_y - (y - y_i) \cdot F_x = \tau_i, \quad i = 1...k \tag{3.23}$$

where (x_i, y_i) are the ith sensor locations and τ_i is the corresponding torque. This equation obviously has multiple solutions for the force componentsF_x, F_y even if the remaining variables x, y describing the contact point location are known or can be obtained from the geometric constraint. In fact, for each given x, y the above equations yield a pencil of possible force vectors

$$\begin{aligned} F_x(x, y, \alpha) &= \frac{\tau_i}{(x - x_i) \cdot \sin\alpha - (y - y_i) \cdot \cos\alpha} \cdot \cos\alpha \\ & \qquad\qquad \alpha \in [\alpha_0, \alpha_0 + \pi] \\ F_y(x, y, \alpha) &= \frac{\tau_i}{(x - x_i) \cdot \sin\alpha - (y - y_i) \cdot \cos\alpha} \cdot \sin\alpha \end{aligned} \tag{3.24}$$

that are expressed via an arbitrary parameter α, with $\alpha_0 = \text{sign}(\tau_i) \cdot \text{atan2}(y - y_i, x - x_i)$. Here, the angle α defines the interaction force directions that obviously are limited by a half circle in accordance with the sign of τ_i. Geometrical interpretation of the above expression is presented in Fig. 3.1 where two different points (x, y) are shown with corresponding half-pencils of the possible forces, while the real interaction force is applied to the first link of the robot. It can be proved, that this singular case is possible if the force is applied to the first robot link only, which can mathematically be formulated as follows.

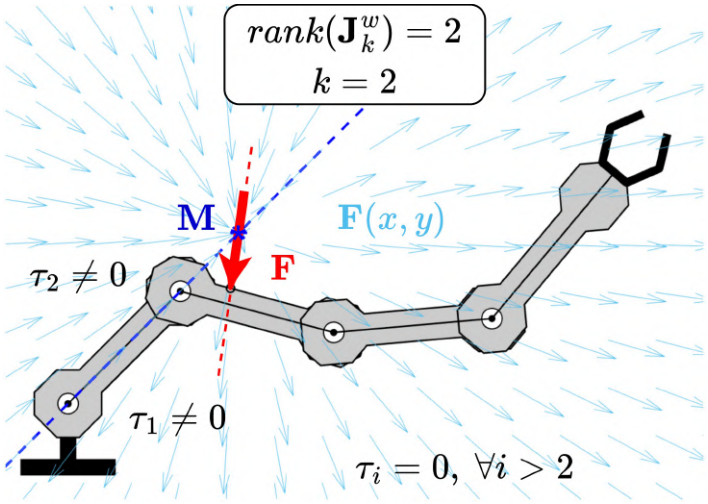


Figure 3.2: Example of singularity in static equilibrium equations for $k = 2$. The force is applied to the second link providing two relations between $(F_x,\ F_y)$ and $(p_x,\ p_y)$.

Statement 3. *If the rank of the extended system* $rank(\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})) = 1$ *then all sensors are located at the same point* $\mathbf{p}_1 = \mathbf{p}_2 = ... = \mathbf{p}_k$*, which is physically possible only if* $k = 1$*. Corresponding solutions for the force vector* $\mathbf{F}$ *and its application point* $\mathbf{p}$ *are described by parametric expressions (3.24) which geometrically defines the set of straight lines pencils for each point* (x, y) *on the plane.*

Proof. If the rank of $3 \times k$ size extended Jacobian matrix (3.2) is equal to one, i.e. $rank(\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})) = 1$, then all of its columns are proportional to each other. In addition, since the 3rd elements of all columns are equal to 1, the remaining elements should be also equal to each other, which leads to $x - x_1 = x - x_2 = ... = x - x_k$ and $y - y_1 = y - y_2 = ... = y - y_k$ and consequently to $x_1 = x_2 = ... = x_k$ and $y_1 = y_2 = .. = y_k$. It is clear that it is physically possible if $k = 1$ only. □

More interesting from a practical point of view is the case $rank(\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})) = 2$, when there are at least two independent static equilibrium equations. For simplicity, let us consider the case when $k = 2$ and the interaction force is applied to the second link (see Fig. 3.2) and two static equilibrium equations are used to estimate the force components and the contact point coordinates

$$\begin{aligned} (x - x_1) \cdot F_y - (y - y_1) \cdot F_x &= \tau_1 \\ (x - x_2) \cdot F_y - (y - y_2) \cdot F_x &= \tau_2 \end{aligned} \tag{3.25}$$

Here $F_x,\ F_y$ are unknown components of the interaction force, $x,\ y$ are unknown coordinates of the contact point. Let us assume first that the contact point $x,\ y$ is given and the corresponding determinant of size 2×2 is not equal to zero. Then the

above system can be uniquely solved for F_x and F_y

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} -(y-y_1) & x-x_1 \\ -(y-y_2) & x-x_2 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{3.26}$$

providing us with the vector field $\mathbf{F}(x, y)$ describing the general solution of the system (3.25) for this singular case. The geometric interpretation of such a general solution is presented in Fig. 3.2 which shows a set of possible force vectors $\mathbf{F}$ arranged as a pencil of straight lines. Such arrangement is based on the Statement 1 that allows the force shifting along its direction without violating the considered static equilibrium system. It is worth mentioning that the non-zero determinant assumption is valid only for those (x, y), which do not belong to the line passing through the points $(x_1,\ y_1)$ and $(x_2,\ y_2)$, see dashed line in Fig. 3.2.

To find the vertex $\mathbf{M}$ coordinates (x_m, y_m) of the above-mentioned pencil it is necessary to take into account both the zero-determinant condition and also the consistency of the system (3.25). This leads to the following similar relation

$$\frac{x-x_1}{x-x_2} = \frac{y-y_1}{y-y_2} = \frac{\tau_1}{\tau_2} \tag{3.27}$$

which yields the desired coordinates of the vertex (see Fig. 3.2)

$$x_m = \frac{\tau_1 \cdot x_2 - \tau_2 \cdot x_1}{\tau_1 - \tau_2}; \qquad y_m = \frac{\tau_1 \cdot y_2 - \tau_2 \cdot y_1}{\tau_1 - \tau_2} \tag{3.28}$$

that obviously belongs to the line passing through the sensor locations. Using such notation, the set of possible solutions of the system of static equilibrium equations for $k = 2$ can be presented in parametric form as follows

$$\begin{aligned} x(\alpha, \eta) &= x_m + \eta \cdot \cos\alpha \\ y(\alpha, \eta) &= y_m + \eta \cdot \sin\alpha \end{aligned} \qquad \eta \in \Re \tag{3.29}$$

and

$$\begin{aligned} F_x(\alpha) &= \frac{(\tau_1 - \tau_2)}{(y_1 - y_2)\cdot\cos\alpha - (x_1 - x_2)\cdot\sin\alpha} \cdot \cos\alpha \\ F_y(\alpha) &= \frac{(\tau_1 - \tau_2)}{(y_1 - y_2)\cdot\cos\alpha - (x_1 - x_2)\cdot\sin\alpha} \cdot \sin\alpha \end{aligned} \qquad \alpha \in [\alpha_0, \alpha_0 + \pi] \tag{3.30}$$

where the parameter α is the force direction angle and the parameter η allows the application point shifting along the force action line. It is clear that in this case there

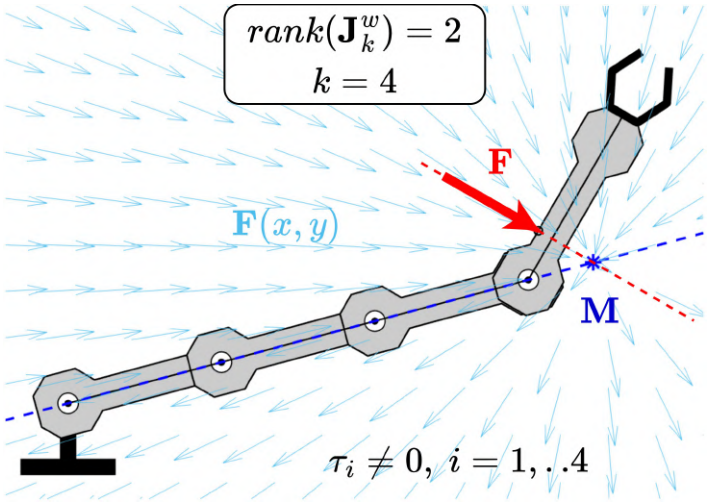


Figure 3.3: Example of singularity in static equilibrium equations for $k = 4$. The interaction force is applied to the fourth link, but all sensors are located inline, which provides only two independent relations between $(F_x,\ F_y)$ and $(p_x,\ p_y)$.

is only one pencil of solutions and the angle $\alpha_0 = \text{sign}(\tau_1 - \tau_2) \cdot \text{atan2}(y_2 - y_1, x_2 - x_1)$ depends only on the robot configuration.

The above result can be easily generalized for more complex cases with $\mathbf{J}_k^w(\mathbf{q}, \mathbf{p}) = 2$ and $k > 2$. One of such cases is presented in Fig. 3.3, where the interacting force $\mathbf{F}$ is applied to the link with $k = 4$, but manipulator all joint torque sensors are located inline. For this reason, there are only two linear-independent equations among k ones

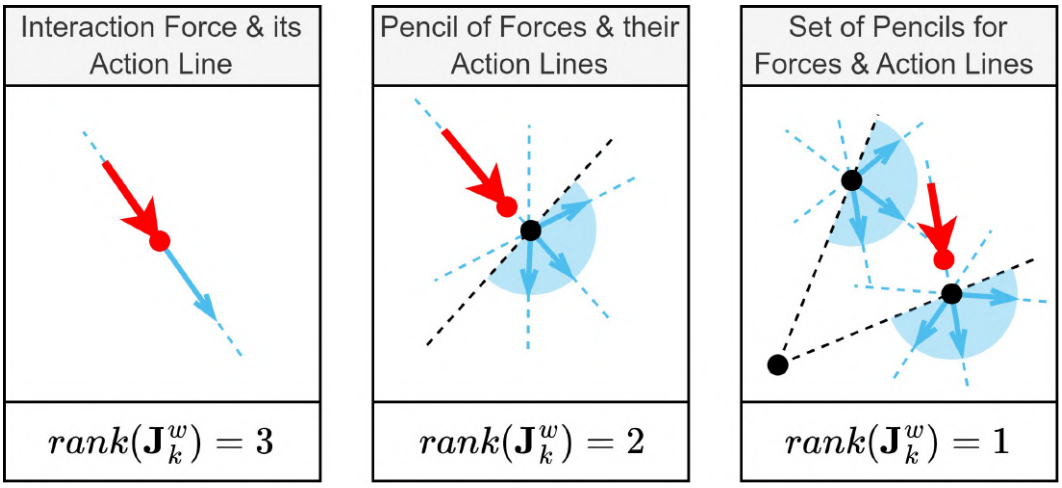
$$(x - x_i) \cdot F_y - (y - y_i) \cdot F_x = \tau_i, \quad i = 1, ...k \tag{3.31}$$

describing the static equilibrium condition, which is equivalent to the case $k = 2$ considered before. However, by taking into account the redundancy of the system (3.31), here the expression (3.26) for the force vector $\mathbf{F}(x, y)$ should be replaced by

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} S_{yy} & -S_{xy} \\ -S_{xy} & S_{xx} \end{bmatrix}^{-1} \cdot \begin{bmatrix} -S_{y\tau} \\ S_{x\tau} \end{bmatrix} \tag{3.32}$$

which can be obtained by using the least squares technique. It is clear that this expression can be applied if the relevant matrix is invertible only, i.e. for (x, y) satisfying the inequality $S_{xx} \cdot S_{yy} \neq S_{xy}^2$. It can be also proved that the latter condition can be violated if (i) all joint torque sensors locations (x_i, y_i) are inline and (ii) the considered point (x, y) belongs to this line. Moreover, it can be also proved that here the force vector field $\mathbf{F}(x, y)$ is also arranged as a pencil of straight lines, similar to the case $k = 2$. Relevant expressions for the vertex point (x_m, y_m) and force components (F_x, F_y) can be easily obtained from (3.28) and (3.30) by replacing the link indexes

Table 3.1: Possible solutions for the interaction force $\mathbf{F}$ and its action line: three possible cases depending on $rank(\mathbf{J}_k^w)$



Note: Real force is shown in red, possible solutions are shown in blue

$(1, 2)$ with arbitrary (i, j), such that $i \neq j$ and $i, j \leq k$. In a more strict way, this result for the singular case with $rank(\mathbf{J}_k^w) = 2$ could be formulated as follows.

Statement 4. *If the rank of the extended system* $rank(\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})) = 2$ *then either* $k = 2$ *or* $k \geq 3$ *with the joint sensor locations* $\mathbf{p}_i$ *belonging to the same line. Corresponding solutions for the force vector* $\mathbf{F}$ *and its application point* $\mathbf{p}$ *are described by parametric expressions (3.28)-(3.30) which geometrically defines the pencil of straight lines.*

Proof. The case $k = 2$ is obvious. If $k \geq 3$ and $rank(\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})) = 2$ then any sth column of the extended Jacobian (3.2) can be expressed as a linear combination of two other columns with some indices (i, j), i.e.

$$\begin{bmatrix} -(y - y_s) \\ (x - x_s) \\ 1 \end{bmatrix} = \alpha \cdot \begin{bmatrix} -(y - y_i) \\ (x - x_i) \\ 1 \end{bmatrix} + \beta \cdot \begin{bmatrix} -(y - y_j) \\ (x - x_j) \\ 1 \end{bmatrix} \tag{3.33}$$

where α, β are some scalar coefficients. Further, after equating line by line one can get that $\alpha + \beta = 1$ and consequently,

$$x_s = \alpha \cdot x_i + \beta \cdot x_j \quad \text{and} \quad y_s = \alpha \cdot y_i + \beta \cdot y_j \tag{3.34}$$

which proves that all joint torque sensors locations $\mathbf{p}_i$ belong to the same line. □

Hence, in the considered singular cases the static equilibrium equations do not allow us to obtain a unique solution for the interaction force and its direction, which can be further used in conjunction with the geometrical constraints (1.2) and (1.9). Summary

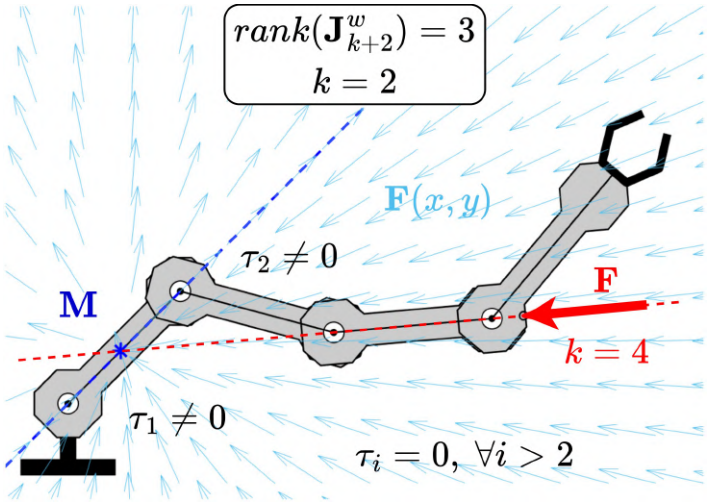


Figure 3.4: Examples of singularity in static equilibrium equations for "false" $k = 2$. The force is applied to the fourth link, but the force action line crosses the axes of sensors #3,4, which provides two independent relations between $(F_x,\ F_y)$ and $(p_x,\ p_y)$.

for all possible cases (both singular and non-singular ones) are presented in Table 3.1. As follows from this table, if $rank(\mathbf{J}^w_k) = 3$, there exists a unique solution for both the force vector and its action line. In contrast, if $rank(\mathbf{J}^w_k) = 1$, there are infinite number of solutions $\mathbf{F}$ for any considered point (x, y). And if $rank(\mathbf{J}^w_k) = 2$, the set of possible solutions is also infinite, but it can be represented as a single pencil of straight lines for the possible force directions.

Nevertheless, this subsection does not cover all possible singularities in the interaction force and its action line computing. In fact, the above-presented decision rule (1.6) may provide us with a "false" interacting link index k yielding the wrong number of the static equilibrium equations, which consequently leads to undesired rank deficiency of the extended Jacobian $\mathbf{J}^w_k$. A typical example of such a situation is presented in Fig. 3.4 where the actual value of this index $k = 4$ and the corresponding extended Jacobian is not singular, i.e. $rank(\mathbf{J}^w_k(\mathbf{q}, \mathbf{p})) = 3$. In fact, here the interaction force is applied to the fourth link, but its action line passes through the last two torque sensor locations (x_3, y_3) and (x_4, y_4) resulting in $\tau_3 = 0$ and $\tau_4 = 0$. The latter leads to degeneration of the above decision rule (1.6), which yields $k = 2$ that corresponds to the reduced number of static equilibrium equations. Consequently, this case can be handled by applying Statement 4 that provides an extended set of possible solutions in the form of a straight-line pencil shown in this figure. However, since the real interaction force is applied to the fourth link, the correct number of the static equilibrium equations should be equal to $k = 4$. So, according to Statement 2, here the equilibrium condition must uniquely define the force and its action line that obviously passes through the torque sensor locations x_3, y_3 and x_4, y_4 as well as the pencil vertex $\mathbf{M}$. Also, it is clear

that this force action line obtained for $k = 4$ belongs to the straight line pencil found using the reduced number of static equilibrium equations $k = 2$.

Therefore, in some situations, the basic decision rule (1.6) may provide us with the wrong value of the index k defining the link number where the interaction force is applied. For this reason, the following subsection concentrates on the enhancement of the k-decision rule, which was previously based on the joint torques analysis only.

3.3.2 Singularities related to k-index of interacting link

As follows from the previous subsection, in some singular cases the basic decision rule (1.6) is incapable of producing the correct index k defining the link where the interaction force is applied. In fact, this decision rule defines the lowest k value where the interaction can happen, while in reality for some given measurements $\tau_1, ...\tau_n$ the force can be applied to one of the subsequent links with the indices $k+1,\ k+2,\ \ldots$ This situation is clearly shown in Fig. 3.4 where there are only two non-zero torque measurements $\tau_{1,2} \neq 0$, but the force is applied to the fourth link. Here, the force action line intersects axes of the joints three and four, which causes zero torques $\tau_{3,4} = 0$. So, the basic decision rule (1.6) produces $k = 2$ instead of $k = 4$. Therefore, to overcome this difficulty, the decision rule for the k-index should be modified in order to produce the range $k \in \{k_{min}, ...,\ k_{max}\}$ and consider all possible links where the interaction force can be applied for the given torque measurements. Otherwise, there exists a risk that real interacting parameters $\mathbf{F}, \mathbf{p}$ will be not included in the set of potential ones.

To find the desired range of k-indices for the given torque measurements $\{\tau_i,\ i = 1, ..., n\}$ the following technique can be applied. First, the lower bound of the k-index is estimated using the basic decision rule (1.6). This yields k_{min} such as $\tau_{k_{min}} \neq 0$ and $\tau_i = 0,\ \forall\ i > k_{min}$. Then, the upper bound of k is obtained by successively increasing the number of considered static equilibrium equations and verifying their consistency. It is worth mentioning that the consistency test can be based on the residual analysis of the corresponding least square solution of the considered system. It is clear that some tolerances should be also defined to take into account the measurement noise. In a more formal way, this technique for computing the k-index range can be presented as:

$$\begin{aligned} &\textbf{\textit{Lower bound:}} \quad k_{\min} = \arg\max_i \ (\|\tau_i\| \geq \delta_\tau) \\ &\textbf{\textit{Upper bound:}} \quad k_{\max} = \arg\max_i \ \left(\left\|\mathbf{A}_i\mathbf{A}_i^{\#} \cdot \boldsymbol{\tau}_i - \boldsymbol{\tau}_i\right\| \leq \delta_w\right) \end{aligned} \tag{3.35}$$

where δ_τ, δ_w are the tolerances for joint torques and least square residuals respectively. Here, the second expression includes the least square solution of the linear system

composed of the static equilibrium equations $\{1, ..., i\}$ with the left-hand side matrix

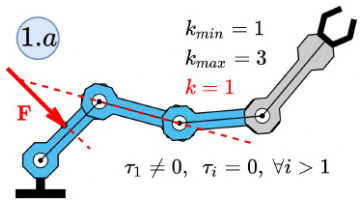
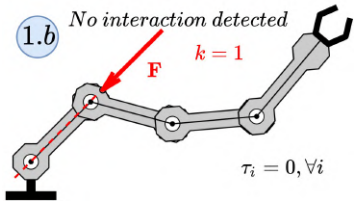
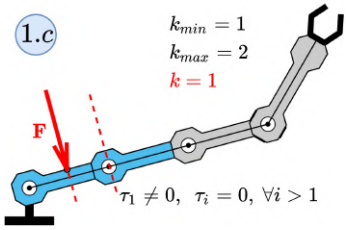
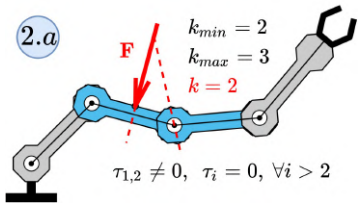
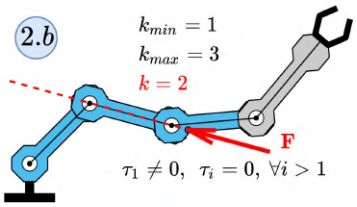
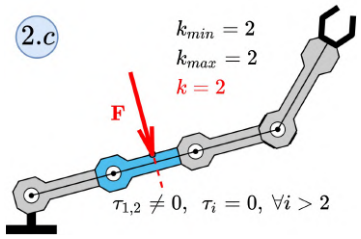
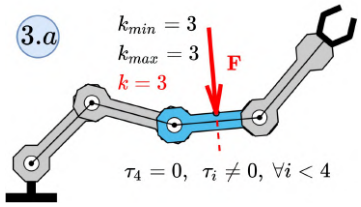
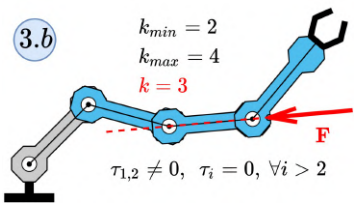
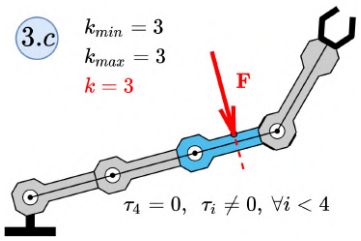
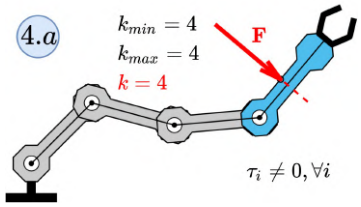
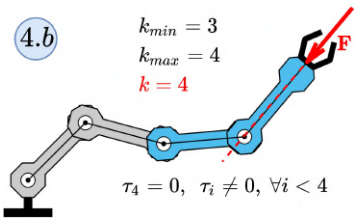
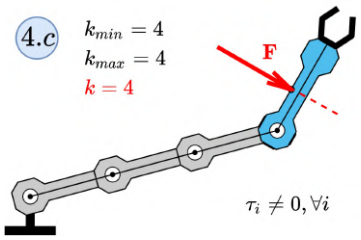
$$\mathbf{A}_i = \left(\mathbf{J}_i^w\left(\mathbf{q}, \mathbf{p}\right)\right)^T \tag{3.36}$$

To illustrate the application of the proposed k-decision rule for different robot poses and interaction forces, several typical cases were studied and are presented in Table 3.2. The first column of this table shows the cases where both the robot configuration and the interaction force direction are non-singular, while the second and third ones deal with some singular cases. As follows from the table, there is only one extremely hard case $(1.b)$ where the interaction is undetectable because all torques are equal to zero. However, from a practical point of view, this case is almost insignificant since even a small deviation in the applied force direction or the robot configuration moves it from the singularity. Other difficult cases arise when $k_{min} \neq k_{max}$, so there exist several links with potential interaction satisfying given torque measurements. The case where both the robot configuration and the direction of the interaction force are singular is not directly presented in Table 3.2 because it is similar to $(1.b)$.

Thus, in practice, the torque measurements $\tau_1, ...\tau_n$ may produce a non-unique k-index defining the links that may be involved in the considered interaction. In particular, if the force is applied to the first link as shown in $(1.a)$, there is rather high uncertainty in k-index $k \in \{1,\ 2,\ 3\}$ and it is impossible to identify the link with the interaction uniquely. A similar problem exists in the case $(1.c)$, where the force is also applied to the first link, but because of the "inline" manipulator configuration, the k-index range is smaller $k \in \{1,\ 2\}$. If the force is applied to the second link, the uncertainty exists even for the kinematically non-singular case $(2.a,\ 2.b)$ where the k-index range can be either $k \in \{2,\ 3\}$ or $k \in \{1,\ 2,\ 3\}$. In contrast, for the kinematically singular case $(2.c)$, the k-index is computed uniquely and $k = 2$. It should be also mentioned that the cases $(2.b)$ and $(1.a)$ are equivalent from the torque measurements point of view ($\tau_1 \neq 0$ & $\tau_i = 0,\ \forall\ i > 1$), which gives a good example of multiple potential solutions for the interacting link. Finally, if the force is applied to the third or fourth link, the k-index estimation usually produces a unique result, as for the cases $(3.a)$, $(4.a)$ and $(3.c)$, $(4.c)$. The problem arises only when the force action line intersects at least one of the joint sensor axis as in cases $(3.b)$ and $(4.b)$ resulting in the k-index uncertainties $k = \{2,\ 3,\ 4\}$ and $k = \{3,\ 4\}$ respectively. It is clear that a similar conclusion can be drawn if the force is applied to the fifth or any other subsequent link, where the k-index can be found in both unique and not unique ways.

It is worth mentioning that in some cases, such as $(1.a)$ and $(2.b)$ with $k \in \{1,\ 2,\ 3\}$, the general solution of the static equilibrium equations is extremely ambiguous. In particular, for $k = 1$ corresponding to the assumption that the force is applied to the

Table 3.2: Ambiguity in the k-index estimation for typical manipulator configurations and interaction forces

Non-singular config., No force-joints inters.	**Non-singular config., Force-joints inters.**	**Singular config., No force-joints inters.**
		
		
		
		

Note: The interaction force direction is shown with a red arrow, potential links to which the force can be applied are highlighted with blue color, potential force action lines are shown with the red dashed line.

first link, there are a set of pencils describing possible forces $\mathbf{F}$ and their action lines (see Table 3.1). Further, for $k = 2$, when the real force is applied to the second link, possible forces, and their action lines are limited by a single pencil. And finally, for $k = 3$ when the real force is assumed to be applied to the third link, the desired force vector $\mathbf{F}$ and its action line can be computed uniquely. Thus, the ambiguity in the k-index leads to an infinite number of possible solutions for the force vector $\mathbf{F}$ and its action line, which obviously include the "true" interaction parameters.

Generally, the above presented case studies from Table 3.2, which are related to the ambiguity in computing of the k-index, can be summarized as follows:

- If the interaction force $\mathbf{F}$ is applied to the first or second link, then the desired value of the k-index cannot be identified uniquely, except for a special case $(2.c)$ when the manipulator configuration is singular and the force direction is non-singular.
- If the interaction force $\mathbf{F}$ is applied to the third or subsequent link, then the desired value of the k-index can be identified uniquely, except for a special case $(3.b)$ when the manipulator configuration is non-singular but the force direction is singular.

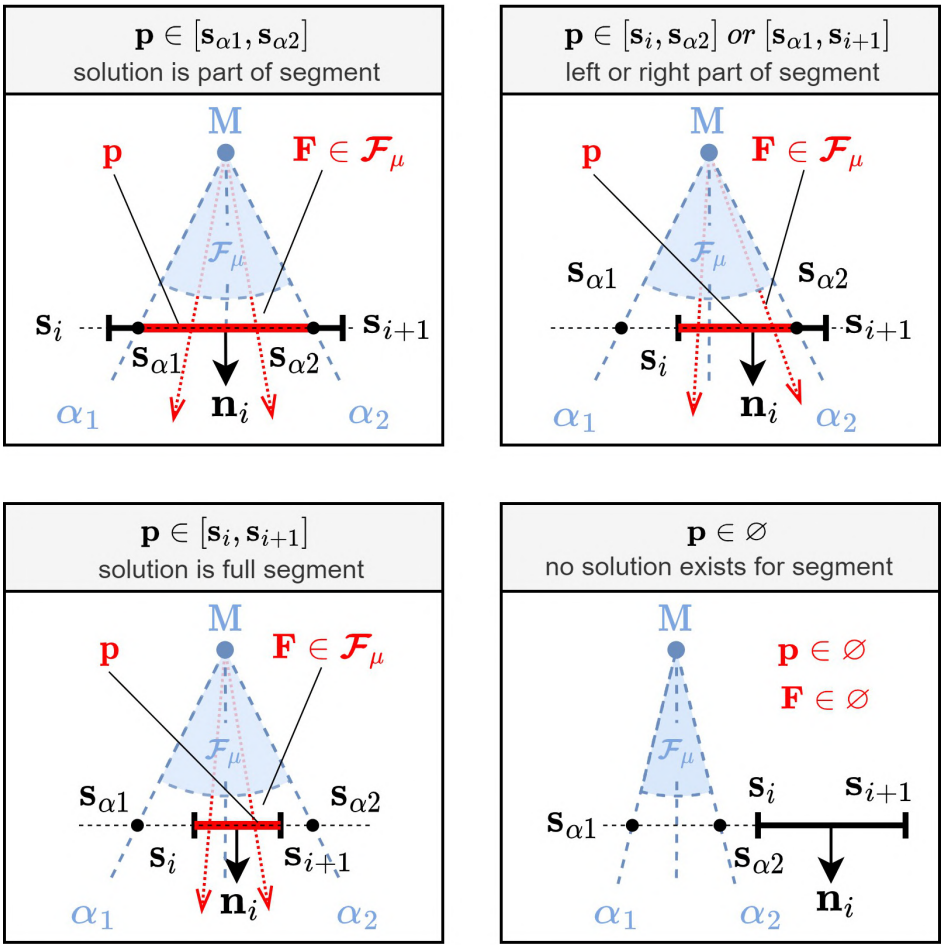
Hence, the joint torque measurements $\tau_1, ...\tau_n$ allow us to identify potential links with the index $k \in \{k_{min}, ..., k_{max}\}$ that may be involved in the interaction as well as to define sets of k corresponding static equilibrium equations yielding the potential interaction forces $\mathbf{F}$ and their action lines. It is clear that it is insufficient to identify the desired interaction parameters definitely. So, further analysis must be enhanced by including the geometrical constraints describing the manipulator link shape(s) and feasible force direction(s) limited by the friction cone. These issues are considered in the following subsection.

3.3.3 Singularities in computing the force application point

In the previous subsection, the solution of the static equilibrium equations was obtained providing us with the force vector $\mathbf{F}$ and its action line $\mathbf{p} \in \boldsymbol{\mathcal{L}}$ for two singular cases $rank(\mathbf{J}_k^w) = 1$ or $rank(\mathbf{J}_k^w) = 2$. In contrast to the non-singular case, here the obtained solution $(\mathbf{F}, \boldsymbol{\mathcal{L}})$ is not unique and presented in a parametric form. Further, to identify the desired interaction parameters $(\mathbf{F}, \mathbf{p})$, it is necessary to apply to $(\mathbf{F}, \boldsymbol{\mathcal{L}})$ the geometric constraints $\mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$.

Let us consider the **first case** with $rank(\mathbf{J}_k^w) = 2$. From the computational point of view, it is convenient to start with applying the friction cone constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ and take into account the surface constraint $\mathbf{p} \in \boldsymbol{\Omega}_k$ afterward. Let us assume that the solution for the interaction force $\mathbf{F}$ is parametrized with a single arbitrary variable α as

Table 3.3: Singularity with $rank(\mathbf{J}_k^w) = 2$: possible solutions for the force application point $\mathbf{p}$ depending on the friction cone constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ and constraint $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$



Note: Possible locations for the force application point $\mathbf{p}$ are shown in red.

in (3.30), which geometrically describes a pencil of straight lines shown in the second column of Table 3.1. Also, for each segment $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$ representing the surface of the kth link, let us define the normal direction $\mathbf{n} = [n_x, n_y]$ and corresponding orientation parameter α computed as $\phi_n = \text{atan2}\,(n_y, n_x)$.

Using the above definitions, the boundary values for the angle α satisfying the friction cone constraint $\mathbf{F}(\alpha) \in \boldsymbol{\mathcal{F}}_\mu$ can be computed for some friction coefficient μ as $\phi_1 = \phi_n + \text{atan}\,(\mu)$ and $\phi_2 = \phi_n - \text{atan}\,(\mu)$. In addition, by taking into account only feasible force directions from the pencil (3.30), the allowable α-range should be further limited to

$$[\alpha_1, \alpha_2] = [\phi_1, \phi_2] \cap [\alpha_0, \alpha_0 + \pi] \tag{3.37}$$

where the interval intersection operation is performed considering the 2π-periodicity of the angle α. It is clear that the above operation can also produce an empty interval

$[\alpha_1, \alpha_2] = \varnothing$, which means that there is no solutions for the considered segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$. It should be noted that these boundary values α_1 and α_2 correspond to the border force action lines defined by force vectors $\mathbf{F}(\alpha_1)$ and $\mathbf{F}(\alpha_2)$, which are shown in Table 3.3. Similar to the non-singular case, the surface constraint $\mathbf{p} \in \boldsymbol{\Omega}_k$ can be applied by computing the intersection points between the possible force action lines and the line passing through the considered surface segment $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$. The desired intersection coordinates (x, y) can be computed by solving the following system

$$\begin{cases} n_x \cdot x + n_y \cdot y = C_\Omega \\ S_\alpha \cdot x - C_\alpha \cdot y = C_M \end{cases} \tag{3.38}$$

where $S_\alpha = \sin(\alpha)$, $C_\alpha = \cos(\alpha)$ and right hand side values C_Ω and C_M are defined as

$$C_\Omega = n_x \cdot s_x + n_y \cdot s_y; \qquad C_M = S_\alpha \cdot x_m - C_\alpha \cdot y_m \tag{3.39}$$

with the coordinates (x_m, y_m) of the vertex point $\mathbf{M}$ computed from (3.28). So, an analytical solution of this system, which is parametrized for some arbitrary angle $\alpha \in [\alpha_1, \alpha_2]$, can be presented in the following way

$$\begin{aligned} x(\alpha) &= (C_\Omega \cdot C_\alpha + C_M \cdot n_y)/D_\alpha \\ y(\alpha) &= (C_\Omega \cdot S_\alpha - C_M \cdot n_x)/D_\alpha \end{aligned} \tag{3.40}$$

where $D_\alpha = C_\alpha \cdot n_x + S_\alpha \cdot n_y$. This yields the intersection points of the friction cone borders and the line passing through the segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$

$$\mathbf{s}_{\alpha 1} = [x(\alpha_1), y(\alpha_1)]\,; \quad \mathbf{s}_{\alpha 2} = [x(\alpha_2), y(\alpha_2)] \tag{3.41}$$

Further, to combine both constraints $\mathbf{p} \in \boldsymbol{\Omega}_k$ and $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ together, it is necessary to find the intersection of two intervals

$$\mathbf{p} \in \left[\mathbf{s}_i, \mathbf{s}_{i+1}\right] \cap [\mathbf{s}_{\alpha 1}, \mathbf{s}_{\alpha 2}] \tag{3.42}$$

where the first one describes the considered segment of the robot link surface and the second interval takes into account the friction cone constraint. For computation convenience, the above operation can be executed in a scalar manner. Since both of these intervals belong to the same line, a simple projection on this line can be used, which yields $s \in [0, s_{max}]$ and $s \in [s_1, s_2]$, where

$$s_{\max} = (\mathbf{s}_{i+1} - \mathbf{s}_i)^T \cdot \mathbf{e}_i; \quad s_1 = (\mathbf{s}_{\alpha 1} - \mathbf{s}_i)^T \cdot \mathbf{e}_i; \quad s_2 = (\mathbf{s}_{\alpha 2} - \mathbf{s}_i)^T \cdot \mathbf{e}_i \tag{3.43}$$

and $\mathbf{e}_i$ is a unit vector corresponding to the segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$. This allows us to present the intersection in the form $[\rho_1, \rho_2]$ with

$$\rho_1 = \max\{0, s_1\}; \qquad \rho_2 = \min\{s_{max}, s_2\} \tag{3.44}$$

So, the final result combining the general solution of the static equilibrium equations with the geometric constraints can be presented as $\mathbf{p} \in [\mathbf{p}_1, \mathbf{p}_2]$, where $\mathbf{p}_1 = \mathbf{s}_i + \rho_1 \cdot \mathbf{e}_i$ and $\mathbf{p}_2 = \mathbf{s}_i + \rho_2 \cdot \mathbf{e}_i$. The latter defines a set of potential points where the interaction force can be applied to the considered segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$ of the robot link. It should be noted that in some cases this set can be empty, i.e, $\mathbf{p} = \varnothing$, which is possible if $s_1 > s_{max}$ or $s_2 < 0$. It is also worth mentioning, that it is assumed here that $s_1 \leq s_2$, otherwise s_1 and s_2 are swapped. The geometric interpretation of the above technique is given in Table 3.3, which clearly shows that the solution for the interaction force application point $\mathbf{p}$ depends on the relative position of the friction cone $\boldsymbol{\mathcal{F}}_\mu$ and the considered segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$. As result, several alternatives are possible, which include a solution as a part of the segment, a whole segment, or no solution.

It is obvious that the above-described procedure should be repeated for each segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$, $i = 1, .., m$ describing the considered kth link of the manipulator and the relevant results should be merged to obtain a complete solution. The typical example of such an operation is presented in Fig. 3.5, which clearly shows that the final solution for $\mathbf{p}$ may consist of several isolated intervals. Hence, in the considered singular case with $rank(\mathbf{J}_k^w) = 2$, both the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$ can be estimated with some uncertainty.

In the **second case** with $rank(\mathbf{J}_k^w) = 1$, a similar technique can be used to find the force application point $\mathbf{p}$. Although this rank deficiency increases the ambiguity of the solution $(\mathbf{F}, \boldsymbol{\mathcal{L}})$, the geometric constraints can be used to reduce the uncertainty. By using the same notation as above, the boundary values for the angle α can be denoted as α_1 and α_2. However here, the feasible force directions, which are defined by the pencil $\alpha \in [\alpha_0, \alpha_0 + \pi]$ are not constant and depend on the considered point location (x, y). The latter modifies the α-range expression (3.37) as

$$[\alpha_1(\mathbf{p}), \alpha_2(\mathbf{p})] = [\phi_1, \phi_2] \cap [\alpha_0(\mathbf{p}), \alpha_0(\mathbf{p}) + \pi] \tag{3.45}$$

where $\phi_1 = \phi_n + \operatorname{atan}(\mu)$, $\phi_2 = \phi_n - \operatorname{atan}(\mu)$, $\alpha_0(\mathbf{p}) = \operatorname{sign}(\tau_1) \cdot \operatorname{atan2}(y - y_1, x - x_1)$ and $\mathbf{p} = (x, y)$. In practice, it is sufficient to evaluate the segment endpoints to verify the existence of the solution. For convenience, the range of feasible solutions for the α angle can be denoted as $\boldsymbol{\alpha}(\mathbf{p}) = [\alpha_1(\mathbf{p}), \alpha_2(\mathbf{p})]$. Particularly, if the $\boldsymbol{\alpha}(\cdot)$-set is non-empty for both segments ends, i.e. $\boldsymbol{\alpha}(\mathbf{s}_i) \neq \varnothing$ and $\boldsymbol{\alpha}(\mathbf{s}_{i+1}) \neq \varnothing$, then the whole

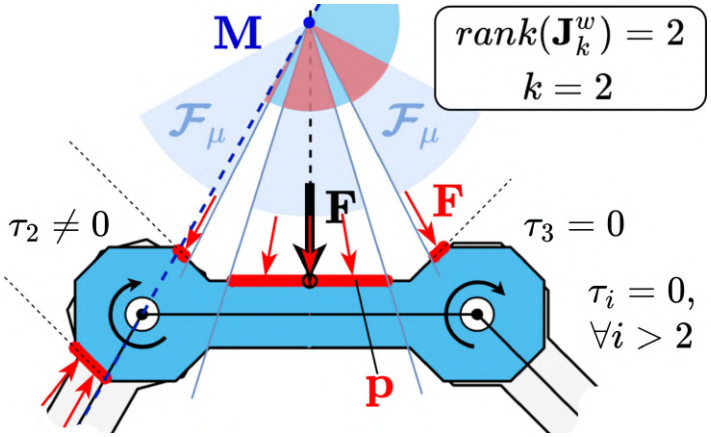


Figure 3.5: A typical example of uncertainty in the interaction parameters estimation for the singular case with $k = 2$. The real interaction force is shown with a black arrow, the estimated forces $\mathbf{F}$ and their application points $\mathbf{p}$ are shown with red color.

segment can be marked as the potential solution, yielding $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$. In contrast, if $\boldsymbol{\alpha}(\cdot)$-set is empty for both ends, i.e. $\boldsymbol{\alpha}(\mathbf{s}_i) = \varnothing$ and $\boldsymbol{\alpha}(\mathbf{s}_{i+1}) = \varnothing$, no solution for the considered segment exists, yielding $\mathbf{p} \in \varnothing$. These rules can be easily proved by the α angle monotony combined with the assumption that the considered segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$ is a geometric line. It is worth mentioning that such a technique allows to reduce computational demands of the algorithm since it replaces the solution of the linear system (3.38) by simple interval intersection operation. The remaining cases dealing with the combination of empty and non-empty $\boldsymbol{\alpha}(\cdot)$-sets are considered in detail below.

It is clear that some combinations of the interaction force $\mathbf{F}$ and the segment $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$ can produce combinations of empty and non-empty $\boldsymbol{\alpha}(\cdot)$-sets, i.e. $\boldsymbol{\alpha}(\mathbf{s}_i) = \varnothing$, $\boldsymbol{\alpha}(\mathbf{s}_{i+1}) \neq \varnothing$ or $\boldsymbol{\alpha}(\mathbf{s}_i) \neq \varnothing$, $\boldsymbol{\alpha}(\mathbf{s}_{i+1}) = \varnothing$. In this case, the intermediate point on the segment must be found to define the possible locations of the force application points $\mathbf{p}$. As follows from the above, there are two points of interest on the segment connected with the friction cone boundaries. Particularly, these two points are the intersection points $\mathbf{s}_{\alpha 1}, \mathbf{s}_{\alpha 2}$ between the force action lines formed by the friction cone boundaries and the line passing through the considered surface segment $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$. They can be found by solving the following system for (x, y) using α_1 and α_2

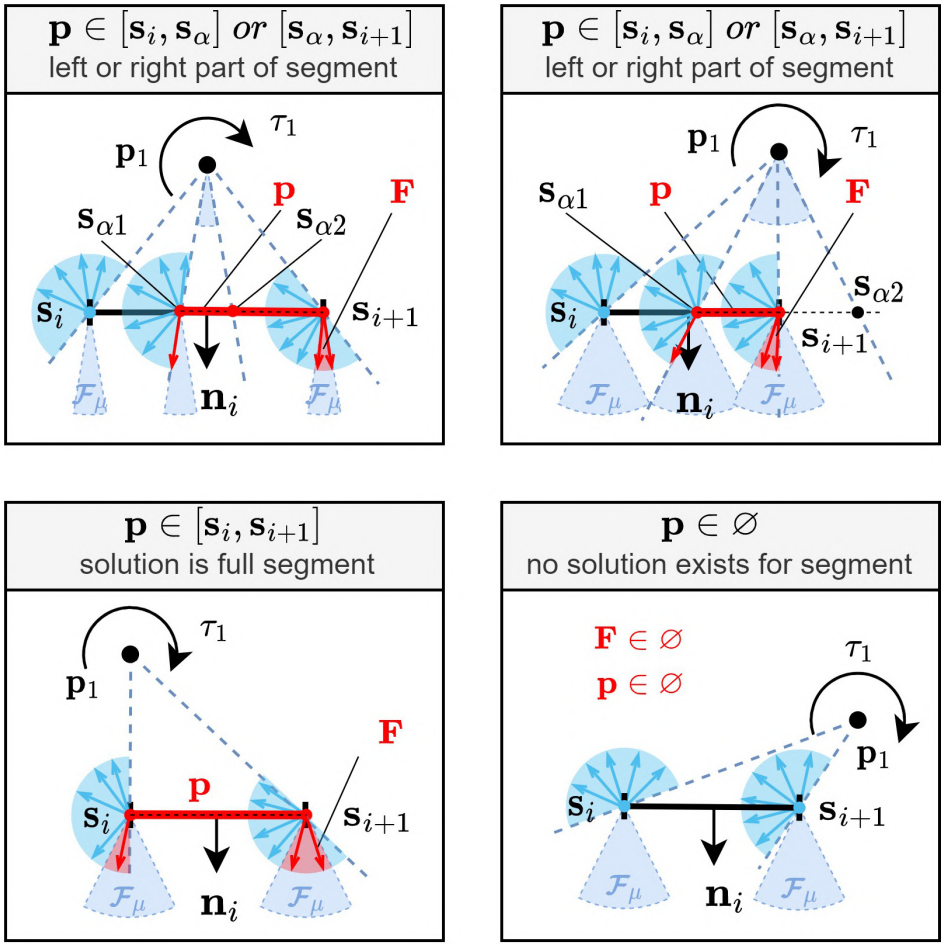
$$\begin{cases} n_x \cdot x + n_y \cdot y = C_\Omega \\ S_\alpha \cdot x - C_\alpha \cdot y = C_0 \end{cases} \tag{3.46}$$

where $S_\alpha = \sin(\alpha)$, $C_\alpha = \cos(\alpha)$ and right hand side values C_Ω and C_0 are defined as

$$C_\Omega = n_x \cdot s_x + n_y \cdot s_y \tag{3.47}$$

$$C_0 = S_\alpha \cdot x_1 - C_\alpha \cdot y_1 \tag{3.48}$$

Table 3.4: Singularity with $rank(\mathbf{J}_k^w) = 1$: possible solutions for the force application point $\mathbf{p}$ depending on the friction cone constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ and constraint $\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}]$



Note: Possible locations for the force application point $\mathbf{p}$ are shown in red.

It should be mentioned that at least one of the obtained points (x, y) $\mathbf{s}_{\alpha 1}, \mathbf{s}_{\alpha 2}$ belongs to the segment $[\mathbf{s}_i, \mathbf{s}_{i+1}]$ and defines either left or right end of the reduced segment for the possible locations of the force application points. Similar to (3.41) these points will be further denoted as $\mathbf{s}_{\alpha 1}$ and $\mathbf{s}_{\alpha 2}$. In more formal way, if α-range is non-empty at the point $\mathbf{s}_i$, i.e $\boldsymbol{\alpha}(\mathbf{s}_i) \neq \varnothing$ and $\boldsymbol{\alpha}(\mathbf{s}_{i+1}) = \varnothing$, the desired solution for $\mathbf{p}$ can be presented as follows

$$\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}] \cap ([\mathbf{s}_i, \mathbf{s}_{\alpha 1}] \cup [\mathbf{s}_i, \mathbf{s}_{\alpha 2}]) \tag{3.49}$$

otherwise, if $\boldsymbol{\alpha}(\mathbf{s}_{i+1}) \neq \varnothing$ and $\boldsymbol{\alpha}(\mathbf{s}_i) = \varnothing$, the possible locations of the force application points for the considered interval can be expressed as follows

$$\mathbf{p} \in [\mathbf{s}_i, \mathbf{s}_{i+1}] \cap \left([\mathbf{s}_{i+1}, \mathbf{s}_{\alpha 1}] \cup [\mathbf{s}_{i+1}, \mathbf{s}_{\alpha 2}]\right) \tag{3.50}$$

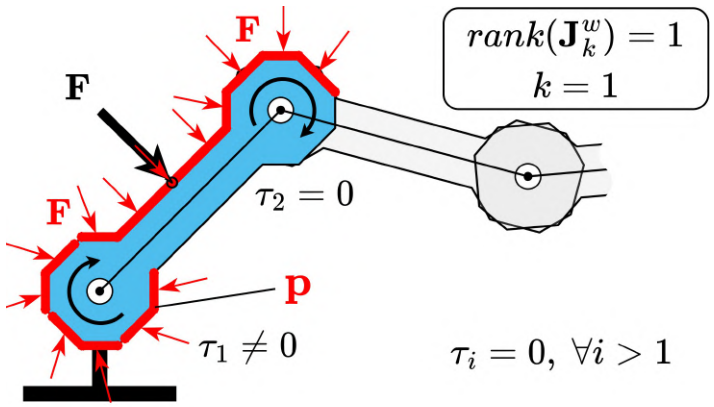


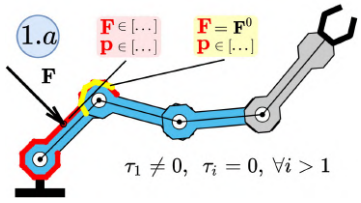
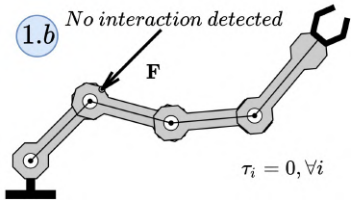
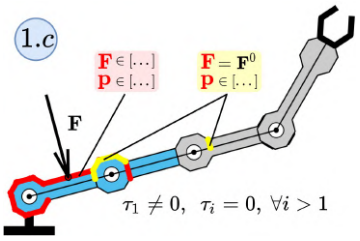
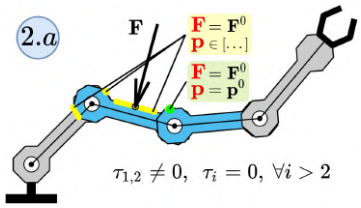
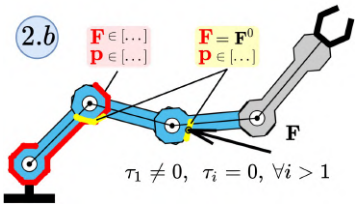
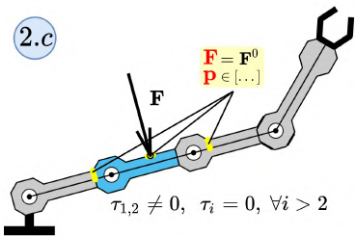
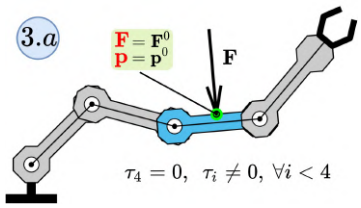
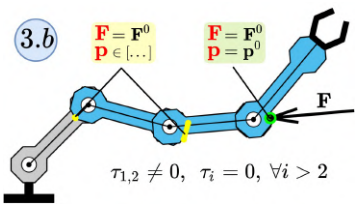
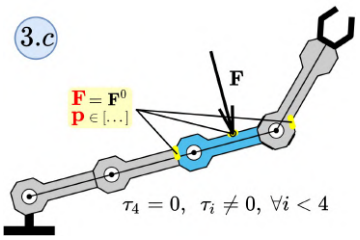
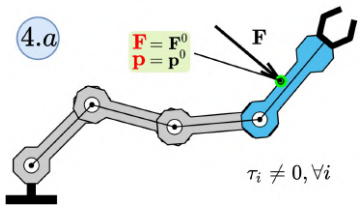
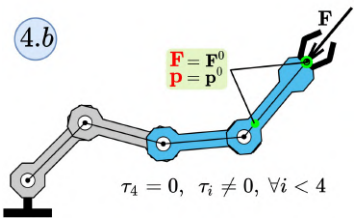
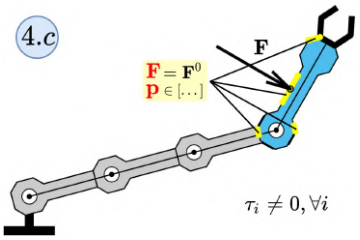
Figure 3.6: A typical example of uncertainty in the interaction parameters estimation for the singular case with $k = 1$. The real interaction force is shown with a black arrow, the estimated forces $\mathbf{F}$ and their application points $\mathbf{p}$ are shown with red color.

All of these cases dealing with combinations of empty and non-empty $\boldsymbol{\alpha}(\cdot)$-sets are clearly shown in Table 3.4, where the first two instances present a part of segment solution, the third shows a full segment solution and the last has no solution for the force application point at all. However, it should be noted that in contrast to the case with $rank(\mathbf{J}_k^w) = 2$, when both points $\mathbf{s}_{\alpha 1}, \mathbf{s}_{\alpha 2}$ are lying on the segment, the solution is not limited by the corresponding interval $[\mathbf{s}_{\alpha 1}, \mathbf{s}_{\alpha 2}]$ but the one extended to the end of the segment $[\mathbf{s}_{\alpha 1}, \mathbf{s}_{i+1}]$ as shown in the first instance of Table 3.4. The typical example of obtained interaction parameters in case of $rank(\mathbf{J}_k^w) = 1$ is presented in Fig. 3.6, which clearly shows that the final solution for $\mathbf{p}$ is rather ambiguous and may consist of several isolated intervals covering more than a half of the link surface.

Thus, the above-presented methods allow us to find the solution for the force application point $\mathbf{p}$ in singular cases with $rank(\mathbf{J}_k^w) = 2$ and $rank(\mathbf{J}_k^w) = 1$. In contrast to the non-singular case, the solution is represented by not a point but an interval on the link surface. Obviously, the proposed procedure should be applied to all m segments describing the kth link shape to achieve the complete solution. Summary for all possible cases, including both singular and non-singular ones, are presented in Table 3.6, which together with Table 3.1 cover all types of the solution for both, the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$ correspondingly.

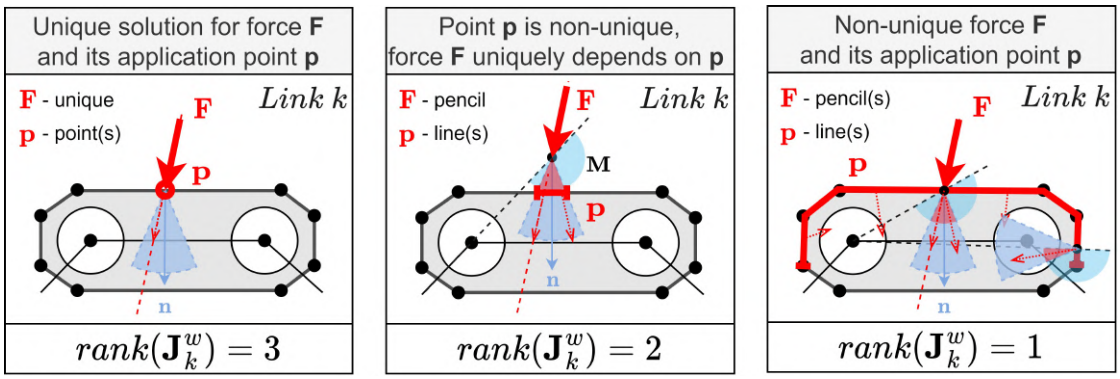
To illustrate the complete solutions and its ambiguity for the serial manipulator in singular and non-singular configurations, the proposed techniques were applied for the same configurations as in Table 3.2 and shown in Table 3.5. In contrast to the previous table, which highlights the links with the potential interaction, this information is supplemented with the interaction force application points that can be used to recreate the same torque values in joints. Because of the used constraints and the rank of the

Table 3.5: Ambiguity of the complete solutions for the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$ for typical manipulator configurations and interaction forces

Non-singular config., No force-joints inters.	**Non-singular config., Force-joints inters.**	**Singular config., No force-joints inters.**
		
		
		
		

Note: The real interaction forces are shown by black arrows, the obtained solutions ($\mathbf{F}$, $\mathbf{p}$) are highlighted by red, yellow or green color depending on its ambiguity. In some cases, the estimation algorithm produces a unique solution $\mathbf{F}^0$ and $\mathbf{p}^0$ highlighted in green, while in other ones there is a non-unique interval solution [...] for the obtained force vector and its action line.

Table 3.6: Summary of possible solutions for the interaction force application point **p**: three possible cases depending on the rank of extended Jacobian $\mathbf{J}_k^w$



corresponding system, the potential solution for **F** and **p** can be presented in the form of exact value (F^0 and $\mathbf{p}^0$) or some interval [...] (see Tables 3.1 and 3.6).

For example, the exact solution for the force and its application point is only possible for the third and following links in non-singular configuration (see 3.a, 4.a, 3.b, 4.b) and can be obtained even if the real interaction force is applied to the previous links (see 2.a). The interval solution for **p** can be expected in the singular configurations for any link, since the rank of the considered system is never full i.e. $rank(\mathbf{J}_k^w) < 3$ (see 1.c, 2.c, 3.c, 4.c). The most ambiguous case with $rank(\mathbf{J}_k^w) = 1$ where both **F** and **p** are represented by the intervals is valid only when the torque detected in the first sensor $\tau_1 \neq 0,\ \tau_i = 0,\ \forall i > 1$. (see 1.$a$, 2.$b$, 1.$c$). It should be noted that similar to Table 3.2, it is not possible to detect any interactions at all for the case (1.b).

Despite the k-index range defined as (3.36), the link from this set is not obligatory and will have a feasible solution. Such an example is demonstrated in (1.a), where the third link is marked as the potential link with the interaction, but the force direction is violating the friction cone constraint in its application point. In the example (4.b) the complete solution for the third and the fourth link provide us with two identical forces $\mathbf{F} = \mathbf{F}^0$, but after applying the appropriate constraints, the estimated force $\mathbf{F}^0$ can only be applied to one of two non-identical points $\mathbf{p}^0$ on third or fourth link respectively.

Summarizing this section, the identification of interaction force application point was provided for all three cases which are possible for the planar serial manipulator. The utilization of geometric constraints allowed us to significantly reduce the number of possible solutions for the given system of static equilibrium equations even in the singular cases. Nonetheless, there is some ambiguity in the complete solution which still persists, so the relevant resolution techniques will be the focus of the next section.

3.4 Singularity resolution technique

In this section, a practice-oriented heuristic approach is proposed for singularity resolution. The core idea of the developed technique is to use several previous estimates ($\mathbf{F},\ \mathbf{p}$) and to interpolate from them in the case of singularity. It is clear that this technique is useful for the so-called *singularity crossing*, when the series of non-singular cases are interrupted by the singular one.

3.4.1 Basic hypothesizes and practice-inspired heuristics

As follows from previous sections, in some cases there is an infinite number of possible solutions for the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$. This can be caused by three main reasons: (i) kinematic singularity of robot configuration; (ii) specific force direction when it intersects some joint axes; (iii) specific location of the force application point leading to an insufficient number of the static equilibrium equations. However, for practical application, a single solution ($\mathbf{F},\ \mathbf{p}$) should be chosen from possible ones, since in reality there is only one physical interaction. The latter motivates us to develop a special singularity resolution technique for the human-robot interaction parameters computing, allowing to overcome the above-mentioned ambiguity problem.

It is clear that in the frame of the considered strictly mathematical model, it is not possible to emphasize one of the possible solutions, which are mathematically equivalent. Obviously, there is no unique and strict approach for such ambiguity resolution and it is reasonable to take into account additionally some practical considerations related to the nature of the human-robot physical interaction. In particular, the interaction usually occurs at the robot end-effector or its last links, so the singularity resolution technique should prioritize contact point locations $\mathbf{p}$ that are closer to the robot tool. Besides, as a rule, this point cannot "jump" on the robot surface, while some small movements of $\mathbf{p}$ are quite possible in practice.

To resolve the ambiguity of the considered identification problem several hypotheses are used here, which are based on practical experience. These hypotheses are presented below and are related to the ambiguity in computing the k-index, the force direction $\mathbf{F}$, and the force application point $\mathbf{p}$ as well as the utilization of the estimation history to resolve the singular cases.

Hypothesis 1. *If the human-robot interaction model does not allow to identify uniquely the link to which the interacting force is applied, then the closest to the end-effector link is selected from possible options, corresponding to the largest k-index from the obtained range $[k_{min},\ k_{max}]$.*

Justification. The priority to the furthest potential link with the index k_{max} can be supported by the fact that in real-life a human operator usually interacts with the robot end-effector or its tool, which correspond to the largest values of k. In addition, the workspace of the collaborative work cell is usually designed in such a way that the robot base is out-of-reach for a human, so the interaction probability is higher for the links that are located closer to the end-effector. □

Hypothesis 2. *If the human-robot interaction model does not allow to identify uniquely the force direction, then the force* $\mathbf{F}$ *that is directed closer to the orthogonal of the* k*th link axis is selected from possible options.*

Justification. Since modern robotics manipulators are usually actuated by revolute motors, it is more likely that the interaction force is directed similarly to the velocity vectors of the considered links surface points. For simplicity, these directions are convenient to approximate by the orthogonal of the corresponding link axis. Another supporting point of such an assumption is the alternative force directions that are co-linear to the link axis are usually unfeasible because of robot mechanical structure (except for the end-effector). □

Hypothesis 3. *If the human-robot interaction model does not allow to identify uniquely the force application point* $\mathbf{p}$*, then the center point of the current interval* $[\mathbf{s}_i,\ \mathbf{s}_{i+1}]$ *is selected from possible options.*

Justification. In this work it is assumed that the robot surface model is extracted from a high-resolution CAD presentation, which provides us with a large number of line segments $[\mathbf{s}_i,\ \mathbf{s}_{i+1}]$ of a rather small size. For this reason, it is rational to reduce the computational expenses by analyzing the segment center point only. It is clear that such simplification does not influence significantly the identification accuracy. □

Hypothesis 4. *If previous estimations of the interaction parameters* $\{\mathbf{F}_t,\ \mathbf{p}_t \mid t < t_c\}$ *are available and the current robot configuration does not allow to identify uniquely the force and its application point, then the desired parameters* $(\mathbf{F}_{t_c},\ \mathbf{p}_{t_c})$ *are computed by interpolating* $\mathbf{F}_t,\ \mathbf{p}_t$ *for* $t = t_c$ *taking into account all geometric constraints.*

Justification. In practice, during the physical human-robot interaction the human usually grasps the robot, so the force application point is almost fixed. Also, some sliding contacts are possible, when the force application point is slowly moving on the robot surface. Since the joint torque measurements are obtained with high frequency, the current location of the force application point should be close to the previous ones. The latter allows us to compute the desired interaction parameter $\mathbf{p}$ using an enhanced interpolation technique, which takes into account relevant geometric constraints. □

3.4.2 Implementation of developed singularity resolution technique

Using the above-presented Hypotheses 1 to 4, it was developed a dedicated singularity resolution technique, which is formalized in Algorithms 6 to 8. This technique consists of two main steps. At the first of them, the k-index of the interacting link is estimated, while the second step deals with the computing of the interaction parameters $\mathbf{F},\ \mathbf{p}$. In more detail, this technique is described below.

The *first step* is based on Hypothesis 1 that is implemented in Algorithm 6. Here, at the beginning the largest possible k-index of the interacting link k_{max} is estimated using the modified decision rule (3.36). Further, if the corresponding set of possible solutions for the force application points $\{\mathbf{p}\}$ is empty, then lower k-indices from the range $[k_{min},\ k_{max}]$ are verified sequentially for the solution existence. Thus, the largest possible k-index is obtained for which the set of possible $\{\mathbf{F},\ \mathbf{p}\}$ is not empty. The necessity of such sequential computations is clearly illustrated by examples $1.a$ and $2.b$ from Table 3.5, where the k-index range is $\{1,\ 2,\ 3\}$ but because of the geometric constraints there is no solution for $k_{max} = 3$. Another particularity of Algorithm 6 is connected with the availability of the interaction parameters estimation history. It is clear that the estimation history does not exist at the initial time before the interaction and the history is deleted after the interaction force is no longer applied to the robot.

The *second step* of the proposed singularity resolution technique is implemented in Algorithms 7 and 8 but only one of them is executed depending on the estimation history availability. Both algorithms start with the estimation of the solution type, which depends on $rank(\mathbf{J}_k^w)$. Here, three cases are possible as shown in Table 3.6. The first case corresponds to $rank(\mathbf{J}_k^w) = 1$, when there is no unique solution for both the interaction force and its application point. In the second case of $rank(\mathbf{J}_k^w) = 2$ there is also no unique solution for the application point $\mathbf{p}$ but the force uniquely depends on $\mathbf{p}$. Finally, if $rank(\mathbf{J}_k^w) = 3$, then both parameters $\mathbf{F},\ \mathbf{p}$ are estimated uniquely for the considered robot link segment. In both algorithms, a full set of all possible solutions for $\mathbf{F},\ \mathbf{p}$ is obtained but different techniques are applied for the selection of a single one in the case of the singularity.

Algorithm 7 is executed if there is no estimation history, it relies on Hypothesis 2 and Hypothesis 3 to extract a single solution from the set of possible ones. According to them, the interaction force is selected as the most orthogonal one to the kth link axis $(\mathbf{O}_{k+1} - \mathbf{O}_k)$, where $\mathbf{O}_k$ and $\mathbf{O}_{k+1}$ define the manipulator joint locations. Here, the force application point $\mathbf{p}$ is computed as the center of the corresponding solution segment. If the interaction force $\mathbf{F}$ is obtained uniquely, the force application point $\mathbf{p}$ is selected as the closest one to the end-effector.

Algorithm 6: Singularity resolution for k

Function `InteractionIdentification(`**History**, $\boldsymbol{\tau}$, **Robot**`):`
- **Input :** **History** is a list of parameters identified on the previous steps, $\boldsymbol{\tau}$ is torque, **Robot** is a structure with robot parameters.
- **Output:** **F**, **p** are unique interaction force and its application point.
- $[k_{min},\ k_{max}] = \text{find\_k}\,(\boldsymbol{\tau},\ \textbf{Robot})$;
- **if** isEmpty(**History**) **then**
  - $[\mathbf{F},\ \mathbf{p}] = \text{IdentifyParamsInitial}\,(\boldsymbol{\tau},\ k_{max},\ \textbf{Robot})$;
  - **while** isEmpty(**p**) **do**
    - $k_{max} = k_{max} - 1$;
    - $[\mathbf{F},\ \mathbf{p}] = \text{IdentifyParamsInitial}\,(\boldsymbol{\tau},\ k_{max},\ \textbf{Robot})$;
- **else**
  - $[\mathbf{F},\ \mathbf{p}] = \text{IdentifyParamsHistory}\,(\textbf{History},\ \boldsymbol{\tau},\ k_{max},\ \textbf{Robot})$;
  - **while** isEmpty(**p**) **do**
    - $k_{max} = k_{max} - 1$;
    - $[\mathbf{F},\ \mathbf{p}] = \text{IdentifyParamsHistory}\,(\textbf{History},\ \boldsymbol{\tau},\ k_{max},\ \textbf{Robot})$;
- **History**.add(**F**, **p**);

Algorithm 7: Singularity resolution for **F**, **p** without estimation history

Function `IdentifyParamsInitial(`$\boldsymbol{\tau}$, k, **Robot**`):`
- **Input :** $\boldsymbol{\tau}$ is torque measurements, k is a link index, **Robot** is a structure with robot geometric and kinematic parameters.
- **Output:** **F**, **p** are identified force and its application point for the kth link.
- **switch** $rank(\mathbf{J}_k^w)$ **do**
  - **case** 1 **do**
    - $[\{\mathbf{F}\},\ \{\mathbf{p}\}] = \text{find\_set}\,(\boldsymbol{\tau},\ k,\ \textbf{Robot})$;
    - $[\mathbf{F},\ i] = \arg\min_i(\perp(\{\mathbf{F}_i\},\ \mathbf{O}_{k+1} - \mathbf{O}_k))$;
    - $\mathbf{p} = (\{\mathbf{p}_i\} + \{\mathbf{p}_{i+1}\})/2$;
  - **case** 2 **do**
    - $[\{\mathbf{F}\},\ \{\mathbf{p}\}] = \text{find\_pencil}\,(\boldsymbol{\tau},\ k,\ \textbf{Robot})$;
    - $[\mathbf{F},\ i] = \arg\min_i(\perp(\{\mathbf{F}_i\},\ \mathbf{O}_{k+1} - \mathbf{O}_k))$;
    - $\mathbf{p} = (\{\mathbf{p}_i\} + \{\mathbf{p}_{i+1}\})/2$;
  - **case** 3 **do**
    - $[\mathbf{F},\ \{\mathbf{p}\}] = \text{find\_line}\,(\boldsymbol{\tau},\ k,\ \textbf{Robot})$;
    - $\mathbf{p} = \arg\min_i(||\mathbf{O}_{last} - \{\mathbf{p}_i\}||)$;
  - **otherwise do**
    - $\mathbf{F} = [\,],\ \mathbf{p} = [\,]$

Algorithm 8: Singularity resolution for $\mathbf{F}$, $\mathbf{p}$ with estimation history

Function `IdentifyParamsHistory(`**History**, $\boldsymbol{\tau}$, k, **Robot**`):`
  Input : **History** is a list of parameters identified on the previous steps, $\boldsymbol{\tau}$ is torque measurements, k is a link index, **Robot** is a structure with robot geometric and kinematic parameters.
  Output: $\mathbf{F}$, $\mathbf{p}$ are identified force and its application point for the kth link.
  switch $rank(\mathbf{J}_k^w)$ **do**
    case 1 **do**
      $[\{\mathbf{F}\},\ \{\mathbf{p}\}] = \text{find\_set}\,(\boldsymbol{\tau},\ k,\ \mathbf{Robot})$;
      $\mathbf{p} = \text{interpolate}(\mathbf{History}.\mathbf{p}_{prev}, \{\mathbf{p}\})$;
      $\mathbf{F} = \mathbf{F}(\mathbf{p})$;
    case 2 **do**
      $[\{\mathbf{F}\},\ \{\mathbf{p}\}] = \text{find\_pencil}\,(\boldsymbol{\tau},\ k,\ \mathbf{Robot})$;
      $\mathbf{p} = \text{interpolate}(\mathbf{History}.\mathbf{p}_{prev}, \{\mathbf{p}\})$;
      $\mathbf{F} = \mathbf{F}(\mathbf{p})$;
    case 3 **do**
      $[\mathbf{F},\ \{\mathbf{p}\}] = \text{find\_line}\,(\boldsymbol{\tau},\ k,\ \mathbf{Robot})$;
      $\mathbf{p} = \arg\min_i(||\mathbf{History}.\mathbf{p}_{prev} - \{\mathbf{p}_i\}||)$;
    otherwise do
      $\mathbf{F} = [\,],\ \mathbf{p} = [\,]$;

Algorithm 8 is executed if the estimation history is available, it relies on Hypothesis 4. According to it, in the case of the interaction parameters ambiguity, the desired single solution is obtained by interpolating the previous estimates. It should be noted that in contrast to the standard interpolation technique, here the geometric constraints are also taken into account. Also, in the full rank case, the interpolation is replaced by selecting the closest point to the previous estimates.

Thus, the developed singularity resolution technique based on several hypotheses can be used to select a single solution in case of the interaction parameters ambiguity.

3.5 Simulation study

The efficiency of the developed approach was confirmed by a number of experiments simulating human-robot collaboration in a real-life environment. These experiments utilize KUKA LBR iiwa industrial manipulator, whose model was reduced to a planar one by proper selection of the force direction. The goal of this simulation is to test the algorithm in terms of interaction parameters estimation accuracy, its robustness to the measurements noise, and its ability to work in singular cases as well as computation efficiency compared to the straightforward optimization techniques.

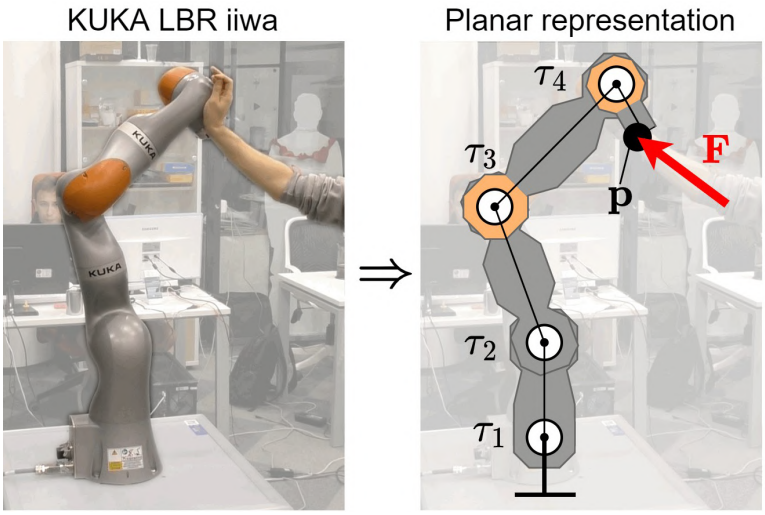


Figure 3.7: Planar representation of KUKA iiwa manipulator for modeling of physical human-robot collaboration and the interaction parameters identification. The human applies the force $\mathbf{F}$ at the point $\mathbf{p}$, generating the joint torques $\tau_1, ..., \tau_4$ that are used for estimation of this force and its application point.

Table 3.7: Computing time comparison for proposed and conventional approaches

Estimation method	**Computing time, $[\mu s]$**		
	min	**avg**	**max**
Proposed analytical technique	0.19	0.21	0.78
Straightforward optimization	286	414	1065

3.5.1 Performance of developed enhanced identification technique

The developed technique was tested in the frame of human-robot collaboration experiments involving KUKA iiwa industrial manipulator presented in Fig. 3.7, whose model was reduced to a planar one by proper selection of the force direction. For this robot, the link surfaces were approximated by polylines composed of up to 20 segments, and the friction coefficient was assumed to be equal $\mu = 0.5$. The interaction force was applied to the surfaces of the 3rd or 4th link, its amplitude was about $|\mathbf{F}| = 100\ N$, and its direction always belonged to the friction cone.

The computational efficiency of the developed technique was also evaluated via simulation in Matlab environment, where the proposed analytical method was compared with the straightforward optimization based on a Matlab function $fmincon(\cdot)$. The computing time for these two techniques is presented in Table 3.7, which shows the essential advantage of the proposed one that is about 2000 times faster.

Table 3.8: Accuracy of proposed technique for different measurement noise level

Noise level, $[N \cdot m]$	**Error in force amplitude,** $[N]$	**Error in force direction,** $[deg]$	**Error in contact position,** $[cm]$	**Non-intersection rate,** [%]
0.0	10^{-13}	10^{-14}	10^{-13}	0
0.1	0.41	0.33	0.93	0
0.5	2.08	1.53	4.63	0.51
1.0	4.23	2.89	8.23	6.86
2.0	8.31	5.96	12.42	25.51

Note: Simulation parameters: force amplitude 100 N;
Manipulator: planar model of KUKA LBR iiwa.

The robustness of the proposed algorithm was evaluated via simulation, by applying some noise to the torque measurements. For each noise level, 10^4 virtual experiments were conducted, where both the robot pose and force application point were selected randomly. The measurement noise was generated using the normal distribution with the mean value of $0.1 - 2.0$ $N \cdot m$, which corresponds to experimental data from the real robot. Relevant simulation results are presented in Table 3.8. It is clear that in the case of zero measurement noise, the estimation error is negligible and is caused by computation round-offs. However, even for rather a realistic measurement noise of 0.5 $N \cdot m$, some particularities may arise where the algorithm is incapable to find the intersection between the force action line and the robot link surfaces. The number of such cases is presented in the last column of Table 3.8. To find the desired solution in such particular cases it is reasonable to combine the proposed analytical and straightforward optimization techniques, where the incomplete analytical estimation is used as an initial guess for the optimization step. Another way to overcome this difficulty is to use the measurement data from several subsequent timestamps and implement a moving window approach.

3.5.2 Performance of proposed singularity resolution technique

The performance of the proposed singularity resolution technique was considered during the experimental study. Different scenarios were evaluated and one of them is shown in Fig. 3.8 as an example. In this figure, the collaborative robot is presented in three different timestamps, where t_c is a current time and Δt is a sampling interval.

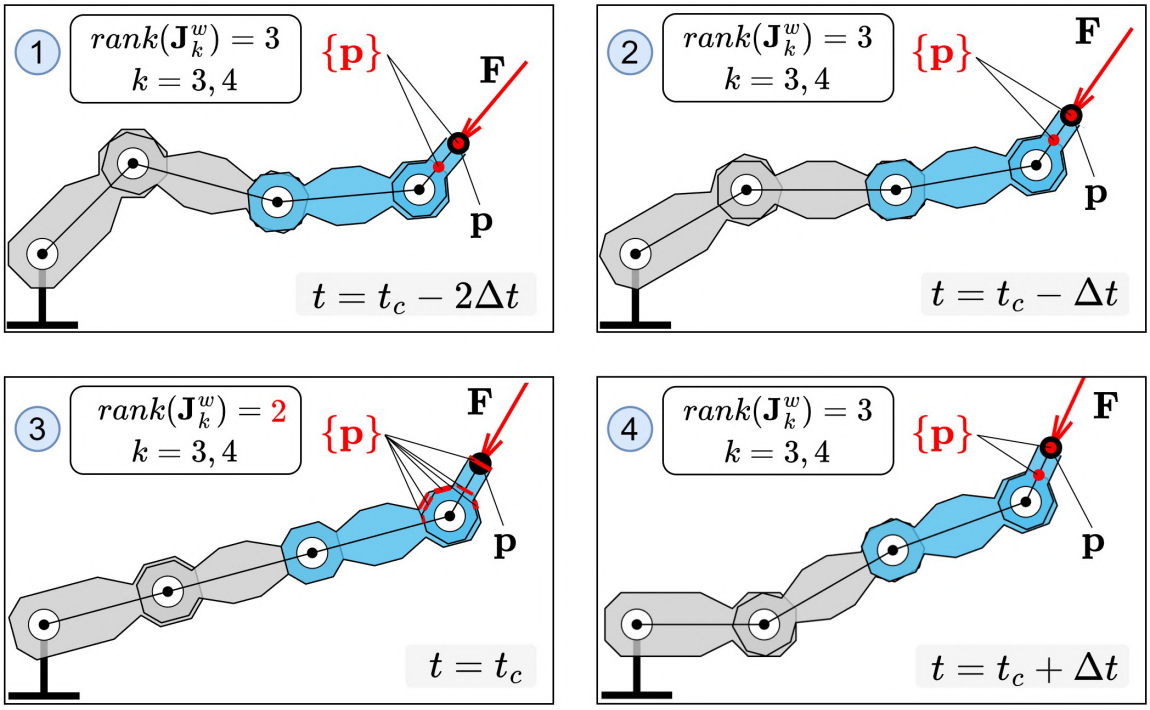


Figure 3.8: Example of the singularity resolution using the estimation history. The red arrow corresponds to the real interaction force, the red points/lines show all potential force application points $\{\mathbf{p}\}$ provided by the estimation algorithm, while the black circle shows the parameter $\mathbf{p}$ chosen by the developed singularity resolution technique.

It should be noted that for the singularity resolution, the presented example is rather complex since all robot configurations have ambiguity in k-index estimation. In fact, here k can be either 3 or 4 because the force action line is intersecting the 4th joint axis. In addition, the last configuration corresponding to $t = t_c$ has also a kinematic singularity since the Jacobian $\mathbf{J}_k^w$ is rank deficient. Assuming that the estimation history is not available for $t = t_c - 2\Delta t$, the interaction parameters for the first configuration were estimated by applying Algorithm 7, which prioritized the force application point location at the last link with $k = 4$.

For the second timestamp with $t = t_c - \Delta t$, Algorithm 8 is applied since the previous estimate of the interaction parameters for $t = t_c - 2\Delta t$ is already available. Here, the set of all potential solutions for force application point $\{\mathbf{p}\}$ consists of two points, corresponding to $k = 3$ and $k = 4$. In this case, the point $\mathbf{p}$ located on the last link with $k = 4$ is selected since it is closer to the previous estimate. It should be noted that for both timestamps $t = t_c - 2\Delta t$ and $t = t_c - \Delta t$ the corresponding static equilibrium systems have full rank, so the solution for interaction force $\mathbf{F}$ is unique.

For the third timestamp with $t = t_c$, Algorithm 8 is also applied, but here the estimation history includes two estimates of $(\mathbf{F},\ \mathbf{p})$ corresponding to $t = t_c - 2\Delta t$ and $t = t_c - \Delta t$. In contrast to the previous cases, the robot configuration here

is kinematically singular and the Jacobian $\mathbf{J}_k^w$ is rank-deficient. So, the set of all potential solutions for force application point $\{\mathbf{p}\}$ is represented by the multiple line intervals, which yields a pencil of possible forces $\{\mathbf{F}\}$. Thus, the ambiguity for both interaction parameters ($\mathbf{F}$, $\mathbf{p}$) is high, but it is resolved by applying the developed interpolation technique of two previous estimates that provides a single solution. It should be noted that in this interpolation, the set of all potential solutions is used as constraints, ensuring that estimated the point $\mathbf{p}$ is located on the robot surface and the force $\mathbf{F}$ is inside of the friction cone.

Thus, as follows from this experimental study, the proposed technique is capable of resolving the singularity problem in the identification of the human-robot interaction parameters. It is clear that in industrial applications, it is challenging to reduce the ambiguity of multiple solutions produced by the identification algorithm presented in the previous sections. Besides, the industry-oriented singularity resolution technique should be simple, reliable, and rather fast. So, the developed technique is based on several hypotheses obtained from practical experience, which allows us to easily integrate the identification algorithm into real collaborative work cells.

It is also worth mentioning that in spite of obvious advantages the proposed technique has some limitations. In particular, it requires the full rank of the extended Jacobian that may be violated in practice if the manipulator geometric configuration is close to a singular one. A similar difficulty arises if the interaction force is applied to the lower links with $k < 3$ and the system of static equilibrium is underdetermined.

3.6 Summary of the chapter

This chapter proposes a practice-oriented enhanced technique for the identification of the human-robot interaction parameters for the 2D planar case. The developed algorithms require information from the internal robot sensors only, i.e., the joint encoders and one-axis torque sensors. In contrast to other works, the identification process is based on the closed-form solution of the system of static equilibrium equations, which also takes into account relevant geometrical constraints. Besides, compared to other existing techniques, the developed one is capable to estimate the desired interaction parameters even in some difficult singular cases. The latter arises when the interaction force is applied to lower manipulator links, resulting in a reduced rank of the static equilibrium system, or when the force action line intersects some joint axes, leading to zero torque measurements from the corresponding sensors. In both cases, it is not possible to obtain a unique solution for the interaction parameters. To overcome this difficulty and select a single solution from the set of possible ones, the special singu-

larity resolution technique is proposed. This technique is based on practice-inspired heuristics and interpolation of the interaction parameters obtained at the previous timestamps. The validity of the developed methodology was confirmed by a simulation study.

The **main advantages** of the interaction parameters identification technique can be summarized as follows:

1) The **accuracy** of the proposed technique is rather high. It obviously allows to obtain zero estimation error in the absence of the measurement noise. In a more realistic case of 0.5 $N \cdot m$ torque measurement noise, the developed algorithm provided the position error of about 4.6 cm with 99.5 % success identification ratio. It is clear that such accuracy is suitable for practical applications.
2) The **real-time performance** of the developed algorithm is also better than that of the existing ones. For the considered case study, it allowed to find the desired interaction parameters in 0.2 μsec on average. Such high performance is due to the analytical nature of the technique, which allowed to achieve essential speed up (about 2000 times), compared to the conventional approaches based on straightforward optimization. So, it well suits to real-time applications, where the response time is critical because of safety reasons.
3) The ability of **singularity resolution**: the developed technique allows to obtain the desired interaction parameters even in the case of essential ambiguity of relevant mathematical equations. By using the identification history, the set of possible solutions is reduced to a single one, which is closer to the real interaction parameters. As follows from the relevant case studies, only two previous history timestamps are sufficient to cross the singularity without losing the real interaction parameters. Thus, such a simple singularity resolution method providing a unique solution for the interaction parameters essentially contributes to the safe and efficient human-robot interaction handling.

It is worth mentioning that despite the above-mentioned advantages, there is a **weak point** of the proposed method. It is a rather **low robustness** to the measurement noise, which was observed in the simulation study. In particular, for the high torque measurements noise of 2 $N \cdot m$ that may be observed in some cases, the identification success ratio is rather low (less than 75 %) that is unacceptable in practice. The latter motivates further enhancements of the proposed technique and its robustness improvements. Besides, in order to be applied in a real-life industrial environment, the proposed technique should be generalized for the 3D spatial case. These issues are the focus of the following chapter of this thesis.

Chapter 4

Development of enhanced interaction parameters identification technique: 3D spatial case

Contents

This chapter deals with interaction parameters identification for the spatial case and presents techniques allowing to estimate the force and its application point in human-robot physical interaction. It is assumed that the desired parameters are estimated using data obtained from internal torque sensors embedded in the robot joints. In practice, the measurement data are corrupted by noise, which causes identification errors. Based on the relevant analysis, enhancements were proposed, which improve the identification accuracy and robustness with respect to the measurement noise.

4.1 Introduction

The chapter deals with the interaction parameters identification in the human-robot collaboration scenario for the 3D spatial case. It presents a technique for computing of the interaction force and its application point using measurement data obtained from the internal joint torque sensors only. The proposed algorithm is based on a specific extension of static equilibrium equations allowing to find the desired interaction force and action line. Further, this general solution is combined with geometric constraints describing manipulator surfaces and corresponding friction cones. Particular attention is paid to singular cases arising when the an insufficient number of independent equations in a static equilibrium system. Compared to the previous chapter, the algorithm was extended to the more general case of the spatial 3D robot.

The main problem considered in this chapter is to estimate the human-robot interaction force and its application point using data obtained from internal torque sensors embedded in the robot joints. Here, the torque measurements can be corrupted by the measurement noise, so its influence on the final identification accuracy should be also studied. It is assumed that the operator interacts with a general n-dof serial manipulator, which consists of a fixed base, an end-effector, and several links connected by n revolute joints. It is also assumed that the interaction force could be applied to an arbitrary point on the manipulator link surface. It should be mentioned that only the so-called point contact between the robot and operator is considered here, implicitly assuming that any surface-to-surface contact (or more complicated one) can be efficiently approximated by the point-to-point interaction.

The above-described problem of interaction parameters identification in the spatial case was studied in a number of works, where the authors applied different numerical techniques, such as machine learning [30], [57], [61], [67] or modern optimization methods based on particle filters [29], Monte Carlo simulation [30], [59], or other approaches [28], [60]. Although these techniques demonstrated some promising results, the required computing time is usually rather high, which makes their real-time application difficult. In contrast, the more conventional approach proposed by De Luca [18], [55] and finalized by Haddadin [9] looks more attractive for industrial applications. It offers an analytical solution in two steps, where the interaction force vector with its action line is obtained first, followed by the contact point estimation that is found as the intersection between the robot surface and the force action line. However, it should be noted that this conventional technique can be used only if the considered problem is not singular.

4.2 Identification of interaction parameters for non-singular case

As follows from a number of related works and the presented study of the planar case from the previous chapter, the optimization problem (1.18) may have either a unique or non-unique solution. Let us start from a non-singular case when there exists a unique solution for both the force $\mathbf{F}$ and its application point $\mathbf{p}$. In this case, all static equilibrium equations are completely satisfied and the considered objective function reaches its minimum possible value that is obviously equal to zero. To simplify the process, the problem will be solved in two steps, where the first one ignores the geometric constraints and the second one takes them into account in order to obtain the desired solution.

4.2.1 Computing of interaction force and its action line

Let us present first the identification algorithm which incorporates basic ideas of De Luca and Haddadin [9] but also includes some enhancements allowing to improve its robustness and accuracy. This algorithm is based on the usual assumption that the system of static equilibrium equations (1.11) is consistent, which is obviously valid for the noise-free case, i.e. when $\varepsilon_i = 0,\ \forall i$. The latter allows us to find the desired solution in a closed-form using a two-step procedure, where the first step focuses on the static equilibrium equations only, omitting the geometric constraints on the unknown variables $\mathbf{F}, \mathbf{p}$ but extending these equations by including additional variables. Here, the general solution of these equations is obtained and integrated with the constraints $\mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ at the second step giving the desired interaction parameters.

Before focusing on the general solution, let us mention some important properties of the considered algebraic system (1.11). Here, the number of unknowns is equal to six; they include the components of the force $\mathbf{F} = [F_x, F_y, F_z]^T$ and the coordinates of the contact point $\mathbf{p} = [x_p, y_p, z_p]^T$. Generally, if such a system is consistent, it may have either single or multiple solutions. However, for our particular case, the unique solution is not possible, since physically the application point $\mathbf{p}$ can be moved along the force action line without violating the static equilibrium equations. Mathematically this property can be formulated in the following way.

Statement 5. *If the force vector* $\mathbf{F}$ *and its application point* $\mathbf{p}$ *satisfy the static equilibrium equations* (1.11)*, then the set of solutions with the same force vector* $\mathbf{F}$ *and the application point* $\mathbf{p}$ *belonging to the line* $\mathbf{p}^* = \mathbf{p} + \beta \cdot \mathbf{F},\ \beta \in \Re$ *also satisfy these equations.*

Proof. This statement can be easily proved by straightforward substitution of the shifted point $\mathbf{p}^*$ into system (1.11)

$$\begin{gathered}\tau_i = [\mathbf{e}_i \times (\mathbf{p} - \mathbf{p}_i)] \cdot \mathbf{F} + \beta \cdot [\mathbf{e}_i \cdot (\mathbf{F} \times \mathbf{F})] \\ \forall i \leq k\end{gathered} \tag{4.1}$$

which after simple transformation yields an additional component of the form $\beta \cdot \begin{bmatrix}0 & \dots & 0\end{bmatrix}^T$ that does not violate the original equality.

□

To find a closed-form solution of the system (1.11), let us introduce an extended system of the static equilibrium equations of the following form

$$\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})^T \cdot \begin{bmatrix}\mathbf{F} \\ \mathbf{M}\end{bmatrix} = \begin{bmatrix}\tau_1 \\ \dots \\ \tau_k\end{bmatrix} \tag{4.2}$$

where the unknown force vector $\mathbf{F} = [F_x, F_y, F_z]^T$ is supplemented by the torque vector $\mathbf{M} = [M_x, M_y, M_z]^T$ and the original $3 \times k$ Jacobian (1.13) is replaced by the extended Jacobian matrix of size $6 \times k$

$$\mathbf{J}_k^w(\mathbf{q}, \mathbf{p}) = \begin{bmatrix}\mathbf{e}_1 \times (\mathbf{p} - \mathbf{p}_1) & .. & \mathbf{e}_k \times (\mathbf{p} - \mathbf{p}_k) \\ \mathbf{e}_1 & .. & \mathbf{e}_k\end{bmatrix} \tag{4.3}$$

Further, let us also assume that the extended system (4.2) is consistent, $rank(\mathbf{J}_k^w) = 6$ and the force application point $\mathbf{p}$ is known. Then this system can be easily solved for the wrench $\mathbf{W} = [\mathbf{F}, \mathbf{M}]^T$ using the following expression

$$\mathbf{W} = \left[\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})^T\right]^{\#} \cdot \boldsymbol{\tau} \tag{4.4}$$

where $[.]^{\#}$ denotes the Moore-Penrose pseudoinverse. It is worth mentioning that the expression (4.4) gives the exact solution even for an over-determined case when $k > 6$. Moreover, it can be proved that for any arbitrary $\mathbf{p}$, the estimated wrench $\mathbf{W}$ always provides us with desired components of the interaction force $\mathbf{F}$ satisfies the original system (1.11). Mathematically this result could be formulated in the following way.

Statement 6. *If the original system (1.11) includes $k \geq 6$ consistent static equilibrium equations and exists some point $\mathbf{p}^*$ providing full rank $rank(\mathbf{J}_k^w(\mathbf{q}, \mathbf{p}^*)) = 6$ for the extended system (4.2), then the solution of the extended system $\mathbf{W} = [\mathbf{F}, \mathbf{M}]^T$ is unique and yield exact force vector $\mathbf{F}$ for the original system.*

Proof. Let us assume that $\mathbf{W} = [\mathbf{F}, \mathbf{M}]^T$ is solution of the extended system for some $\mathbf{p}^0 = [\mathbf{p} + \Delta\mathbf{p}]$ which differs from the real force application point $\mathbf{p}$ by some shift $\Delta\mathbf{p}$.

It means that $\forall i \leq k$ following equations are satisfied

$$\tau_i = (\mathbf{e}_i \times (\mathbf{p} + \Delta\mathbf{p} - \mathbf{p}_i)) \cdot \mathbf{F} + \mathbf{e}_i \cdot \mathbf{M} \tag{4.5}$$

that can be further transformed into

$$\tau_i = (\mathbf{e}_i \times (\mathbf{p} - \mathbf{p}_i)) \cdot \mathbf{F} + \mathbf{e}_i \cdot (\Delta\mathbf{p} \times \mathbf{F} + \mathbf{M}) \tag{4.6}$$

The latter shows that there is strict correspondence between the solution of the original and the extended systems

$$(\mathbf{F},\ \mathbf{p},\ \mathbf{M}) \longleftrightarrow (\mathbf{F},\ \mathbf{p} + \boldsymbol{\Delta}\mathbf{p}) \tag{4.7}$$

where the force components $\mathbf{F}$ are exactly the same and the torque depends on the initial point $\mathbf{p}^0$ as

$$\begin{gathered} \mathbf{e}_i \cdot (\Delta\mathbf{p} \times \mathbf{F} + \mathbf{M}) = 0 \\ \mathbf{M} = -\Delta\mathbf{p} \times \mathbf{F} \\ \forall \mathbf{e}_i,\ i \leq k \end{gathered} \tag{4.8}$$

It is clear that if the point $\mathbf{p}^0$ coincides with the real force application point $\mathbf{p}$, then the torque vector $\mathbf{M}$ is equal to zero. □

Using the above property one can obtain a closed-form solution for the interaction force $\mathbf{F}$, which is found by solving the extended system (4.2) for any arbitrary point $\mathbf{p}^0$. Furthermore, it also allows to find the force action line that must obviously be directed along $\mathbf{F}$ and pass through the point $\mathbf{p} = \mathbf{p}^0 + \Delta\mathbf{p}$, where $\Delta\mathbf{p}$ satisfies equation (4.8), for instance

$$\Delta\mathbf{p} = [\mathbf{S}(\mathbf{F})]^{\#} \cdot \mathbf{M} \tag{4.9}$$

with $\mathbf{S}(\cdot)$ denoting transformation of the vector to the skew-symmetric matrix. Using these notations, the force action line can be presented as

$$\mathbf{p} = \mathbf{p}^0 + \Delta\mathbf{p} + \beta \cdot \mathbf{F}, \quad \beta \in \mathbb{R} \tag{4.10}$$

or in alternative form as

$$\mathbf{S}(\mathbf{F}) \cdot (\mathbf{p} - \mathbf{p^0}) = \mathbf{M} \tag{4.11}$$

Summarizing the above results, it is worth mentioning that the above-presented technique is an extended version of the identification algorithm for the planar case described in Chapter 3. It is based on the extending of the original system of static equilibrium equations (1.11), allowing us to estimate both the interaction force $\mathbf{F}$ and its action line using an arbitrary initial guess $\mathbf{p}^0$.

4.2.2 Computing of force application point

To complete identification, i.e. to find the desired force application point $\mathbf{p}$ the above obtained general solution of the original static equilibrium equations should be integrated with the *geometric constraints* describing the robot links shape, where the force may be applied to. At this step, most of the authors apply the straightforward geometric approach based on ray-triangle intersection search [72], using the force action line parametric equation (4.10). However, taking into account the particularities of the considered problem, we propose to use the action line equation (4.11) and solve it simultaneously with the surface constraint (1.14), which is equivalent to finding the intersection point of the force action line and the plane defined by the jth 3D face

$$\begin{cases} \mathbf{S}(\mathbf{F}) \cdot (\mathbf{p} - \mathbf{p^0}) = \mathbf{M} \\ \mathbf{n}^{k,j} \cdot (\mathbf{p} - \mathbf{sv}^{k,j}) = 0 \end{cases} \tag{4.12}$$

where $\mathbf{n}^{k,j}$ is the surface normal vector of the jth face of the kth link and $\mathbf{sv}^{k,j}$ is one of the face vertex. Obviously, some additional verification is required to estimate if the point $\mathbf{p}$ is located inside of the triangular face (1.16) and if the friction cone constraint (1.16) is satisfied. The second geometric constraint (1.16), which bounds the angle between the interaction force $\mathbf{F}$ and the normal vector $\mathbf{n}^{k,j}$, can be presented in the following way

$$\left|\mathbf{n}^{k,j} \times \mathbf{F}\right| \leq \mu \left(\mathbf{n}^{k,j} \cdot \mathbf{F}\right) \tag{4.13}$$

where μ is the friction coefficient between the manipulator link surface and the interacting object.

To obtain an integrated solution, let us combine the above constraints with the general solution of the static equilibrium equations. This problem (1.11) can be solved by applying Algorithm 9 for the kth link obtained from the decision rule (1.12). At the

Algorithm 9: Conventional identification method for non-singular cases

1 **Function** `BasicMethod(`$\mathbf{q}$, $\boldsymbol{\tau}$, k, **Robot**`):`

Input : $\mathbf{q}$ is joints positions, $\boldsymbol{\tau}$ is torque measurements, k is link index, **Robot** is a structure with robot geometric and kinematic parameters.

Output: **Solution** is a list of all potential solutions, $\mathbf{F}$ is estimated interaction force, $\mathbf{p}$ is estimated force application point.

2 $[\mathbf{F}, \mathbf{M}]^T = \left[\left(\mathbf{J}_k^w(\mathbf{q}, \mathbf{p}^0)\right)^T\right]^{\#} \cdot \boldsymbol{\tau}$;

3 **for** j=*1 : m* **do**

4 **if** $\left|\mathbf{n}^{k,j} \times \mathbf{F}\right| \leq \mu\left(\mathbf{n}^{k,j} \cdot \mathbf{F}\right)$ **then**

5 $d = \mathbf{n}^{k,j} \cdot (\mathbf{sv}^{k,j} - \mathbf{p}^0)$;

6 $\delta\mathbf{p} = \begin{bmatrix} \mathbf{S}(\mathbf{F}) \cdot \mathbf{S}(\mathbf{F})^T & \mathbf{n}^{k,j^T} \\ \mathbf{n}^{k,j} & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{S}(\mathbf{F}) \cdot \mathbf{M} \\ d \end{bmatrix}$;

7 $\mathbf{p} = \mathbf{p}^0 + \delta\mathbf{p}$;

8 **if** $\mathbf{p} \in \mathbf{sf}^{k,j}$ **then**

9 **Solution**.*add*($\mathbf{p}$);

first step of this algorithm, the interaction wrench is estimated by solving the extended system (4.4). It should be noted here that in practice, it is convenient to use the kth link joint coordinates as initial guess i.e. $\mathbf{p}^0 = \mathbf{p}_k$. It is worth mentioning that such $\mathbf{p}^0$ selection does not have an influence on the computing time and estimation accuracy. Further, in the second step, each 3D face representing the robot link surface is verified for the friction cone constraint. If this constraint is satisfied then the intersection point is obtained by solving the system (4.12). Finally, the obtained point is considered as a possible solution only if it belongs to the jth 3D face $\mathbf{sf}^{k,j}$.

Thus, the above-presented expressions allow us to find the coordinates of the interaction point $\mathbf{p}$ corresponding to the force $\mathbf{F}$ and its action line from the previous subsection, which satisfy both constraints $\mathbf{p} \in \boldsymbol{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. It is clear that relevant computations should be executed for each of m 3D faces describing the kth link shape. It is worth mentioning that Algorithm 9 may produce either a single or multiple solutions satisfying both the static equilibrium equations and the considered geometric constraints. However, it may not yield any solutions at all if the measured torques $\tilde{\tau}$ are essentially corrupted by the noise.

4.3 Identification of interaction parameters for singular cases

Let us concentrate now on singular cases when either the interaction force $\mathbf{F}$, its action line or the force application point $\mathbf{p}$ cannot be computed in a unique way. It is clear that these cases are associated with the rank deficiency of the extended Jacobian $\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})$, which in the previous section was assumed to be full-rank. Similar to the planar case considered in Chapter 3, here the related constraint optimization problem (1.18) also yields multiple solutions for the desired interaction parameters. However, in contrast to the planar case, it is rather hard to obtain closed-form or parametric expressions describing sets of possible interaction forces and their application points. For this reason, the main focus of this section is on the geometrical properties of these sets and their numerical computation.

4.3.1 Possible solutions sets for force and its application point

First, let us assume that the "true" interaction parameters $\mathbf{F}, \mathbf{p}$ are known and consider a simplified version of the original static equilibrium system $\mathbf{J}_k(\mathbf{q}, \mathbf{p})^T \cdot \mathbf{F} = \boldsymbol{\tau}$, which is linearized in its neighborhood. It can be easily proved that the desired linearized system can be presented as

$$[\mathbf{e}_i \times (\mathbf{p} - \mathbf{p}_i + \delta\mathbf{p})]^T \cdot [\mathbf{F} + \delta\mathbf{F}] = \tau_i, \quad \forall i = 1, \ldots, k \tag{4.14}$$

where $\delta\mathbf{F}$, $\delta\mathbf{p}$ are small variations of "true" interaction parameters $\mathbf{F}, \mathbf{p}$. After neglecting second-order small values this system can be reduced to

$$[\mathbf{e}_i \times (\mathbf{p} - \mathbf{p}_i)]^T \cdot \delta\mathbf{F} + [\mathbf{F} \times \mathbf{e}_i]^T \cdot \delta\mathbf{p} = 0, \quad \forall i = 1, \ldots, k \tag{4.15}$$

For further convenience, it also can be represented in the matrix form

$$\begin{bmatrix} [\mathbf{e}_1 \times (\mathbf{p} - \mathbf{p}_1)]^T & [\mathbf{F} \times \mathbf{e}_1]^T \\ \cdots & \cdots \\ \cdots & \cdots \\ \cdots & \cdots \\ [\mathbf{e}_k \times (\mathbf{p} - \mathbf{p}_k)]^T & [\mathbf{F} \times \mathbf{e}_k]^T \end{bmatrix}_{k \times 6} \cdot \begin{bmatrix} \delta\mathbf{F} \\ \delta\mathbf{p} \end{bmatrix}_{6 \times 1} = \begin{bmatrix} 0 \\ \cdots \\ \cdots \\ \cdots \\ 0 \end{bmatrix}_{k \times 1} \tag{4.16}$$

where the left hand matrix can be denoted as $\mathbf{D}_k$.

As known from linear algebra, the above homogeneous system always has the trivial solution $[\delta\mathbf{F},\ \delta\mathbf{p}]^T = \mathbf{0}$ and in some cases, when the matrix $\mathbf{D}_k$ is rank deficient, the

number of solution is infinite. In the general case, the set of possible variations $\delta\mathbf{F},\ \delta\mathbf{p}$ satisfying the linearized static equilibrium system (4.16) can be presented as

$$\begin{bmatrix}\delta\mathbf{F}\\ \delta\mathbf{p}\end{bmatrix}_{6\times1} = \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}_{6\times1} + [null(\mathbf{D}_k)]_{6\times d}\cdot\begin{bmatrix}\mu_1\\ \cdots\\ \mu_d\end{bmatrix}_{d\times1} \tag{4.17}$$

where $null(\mathbf{D}_k)$ denotes the nullspace of $\mathbf{D}_k$ matrix, $[\mu_1, ...\mu_d]^T$ is some arbitrary d-dimensional vector, and d is the rank deficiency of $\mathbf{D}_k$ computed as $d = 6 - rank(\mathbf{D}_k)$. For further convenience, let us present the $6 \times d$ matrix $null(\mathbf{D}_k)$ in the splitted form

$$null(\mathbf{D}_k) = \begin{bmatrix}\mathbf{u}_1^F & \cdots & \mathbf{u}_d^F\\ \mathbf{u}_1^p & \cdots & \mathbf{u}_d^p\end{bmatrix}_{6\times d} \tag{4.18}$$

where the upper $3 \times d$ block $\mathbf{u}_p$ corresponds to force application point variation $\delta\mathbf{p}$, and the lower $3 \times d$ block $\mathbf{u}_F$ corresponds to interaction force variation $\delta\mathbf{F}$.

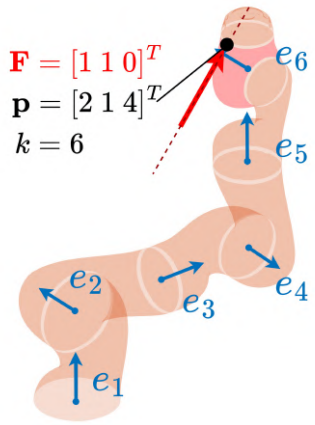
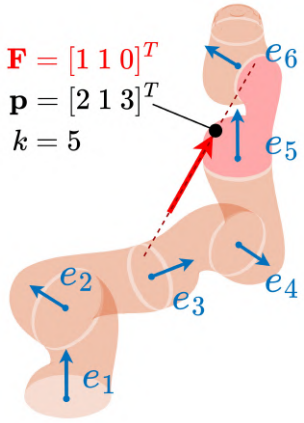
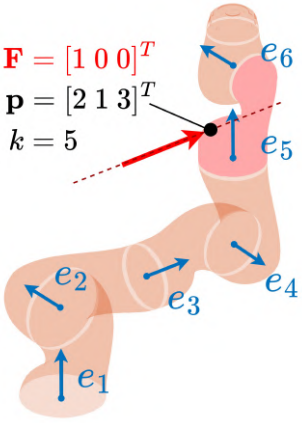
Obviously, the nullspace dimension d depends on the number of static equilibrium equations k, which gives the trivial lower and upper bounds $d \geq 6 - \min(6, k)$ and $d \leq 5$. However here, because of special properties of the considered matrix $\mathbf{D}_k$, the above-mentioned rank deficiency d cannot be lower than 1 even for k sufficiently large $k \geq 6$. This particularity is described by the following statement.

Statement 7. *For any number of static equilibrium equations k, the rank deficiency d of the linearized system (4.16) is always greater than one, i.e. $d \geq 1$.*

Proof. In the case of an insufficient number of static equilibrium equations $k < 6$ this statement is obvious, it follows from the trivial lower bound $d \geq 6 - \min(6, k)$. In the alternative case when $k \geq 6$, it is necessary to use Statement 5 from the previous section, which proves that even for an over-determined original system of consistent static equilibrium equations any solution $(\mathbf{F}, \mathbf{p})$ allows us to generate the set of other possible solutions $(\mathbf{F},\ \mathbf{p} + \mu\mathbf{F})$, $\mu \in \mathbb{R}$ corresponding to the translation of the force application point $\mathbf{p}$ along the force $\mathbf{F}$ action line. It is clear that this property is also valid for the linearized system (4.16) and can be easily verified by simple substitution $\delta\mathbf{p} = \mu\mathbf{F}$ and $\delta\mathbf{F} = \mathbf{0}$ that yields zero triple products $[\mathbf{F} \times \mathbf{e}]^T \cdot \mu\mathbf{F} \equiv 0$. □

Hence, the linearized system of static equilibrium equations (4.16) always has an infinite number of possible solutions for the variations of the desired interaction parameters $\delta\mathbf{F},\ \delta\mathbf{p}$. It is clear that the dimension and geometric properties of the corresponding solution set depend on rank deficiency d. Let us consider in detail the most important cases of $d = 1,\ 2,\ 3$.

Table 4.1: Examples of rank deficiency $d = 1$ in systems of the static equilibrium equations: comparison of linearized and extended systems with matrices $\mathbf{D}_k$ and $\mathbf{J}_k^w$

Case 1: $d = 1,\ k = 6$	**Case 2:** $d = 1,\ k = 5$	**Case 3:** $d = 2,\ k = 5$
		
$rank(\mathbf{D}_k) = 5$	$rank(\mathbf{D}_k) = 5$	$rank(\mathbf{D}_k) = 4$
$\mathbf{D}_k =$ $\begin{bmatrix} -1 & 2 & 0 & 1 & -1 & 0 \\ 3 & 0 & -2 & 0 & 0 & 1 \\ 0 & -3 & 1 & 0 & 0 & -1 \\ -3 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{6\times6}$	$\mathbf{D}_k =$ $\begin{bmatrix} -1 & 2 & 0 & 1 & -1 & 0 \\ 2 & 0 & -2 & 0 & 0 & 1 \\ 0 & -2 & 1 & 0 & 0 & -1 \\ -2 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 1 & -1 & 0 \end{bmatrix}_{5\times6}$	$\mathbf{D}_k =$ $\begin{bmatrix} -1 & 2 & 0 & 0 & -1 & 0 \\ 2 & 0 & -2 & 0 & 0 & 1 \\ 0 & -2 & 1 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}_{5\times6}$
$rank(\mathbf{J}_k^w) = 6$	$rank(\mathbf{J}_k^w) = 5$	$rank(\mathbf{J}_k^w) = 5$
$(\mathbf{J}_k^w)^T =$ $\begin{bmatrix} -1 & 2 & 0 & 0 & 0 & 1 \\ 3 & 0 & -2 & 0 & 1 & 0 \\ 0 & -3 & 1 & 1 & 0 & 0 \\ -3 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}_{6\times6}$	$(\mathbf{J}_k^w)^T =$ $\begin{bmatrix} -1 & 2 & 0 & 0 & 0 & 1 \\ 2 & 0 & -2 & 0 & 1 & 0 \\ 0 & -2 & 1 & 1 & 0 & 0 \\ -2 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{5\times6}$	$(\mathbf{J}_k^w)^T =$ $\begin{bmatrix} -1 & 2 & 0 & 0 & 0 & 1 \\ 2 & 0 & -2 & 0 & 1 & 0 \\ 0 & -2 & 1 & 1 & 0 & 0 \\ -2 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{5\times6}$
$null(\mathbf{D}_k) =$ $\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}_{6\times1}$	$null(\mathbf{D}_k) =$ $\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}_{6\times1}$	$null(\mathbf{D}_k) =$ $\begin{bmatrix} 0 & -1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{bmatrix}_{6\times2}$

Note: In these examples the manipulator kinematics is similar to KUKA iiwa robot. Besides, in order to improve the clarity of numerical expressions, the lengths of the robot links were set to 1, and the manipulator joint angles to either 0 or $\pi/2$, which yields the following vectors $\mathbf{e}_i$ and $\mathbf{p}_i$:

$$\mathbf{e}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{e}_4 = -\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_5 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_6 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\mathbf{p}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{p}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_3 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_4 = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_5 = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix};\quad \mathbf{p}_6 = \begin{bmatrix} 2 \\ 0 \\ 3 \end{bmatrix}$$

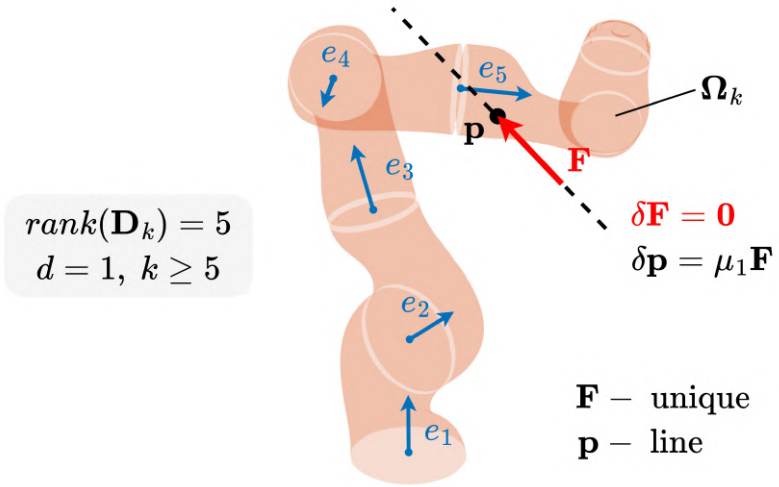


Figure 4.1: Graphical representation of general solution of static equilibrium equations for the case $d = 1$. The "true" interaction force $\mathbf{F}$ is shown in red, while the black line shows all possible force application points $\mathbf{p}$, satisfying the static equilibrium equations.

Case $d = 1$. This case is trivial and as follows from Statements 5 and 7, it yields the one-dimensional nullspace

$$\begin{bmatrix} \mathbf{u}^F \\ \mathbf{u}^p \end{bmatrix}_{6\times1} = \begin{bmatrix} \mathbf{0} \\ \mathbf{F} \end{bmatrix}_{6\times1} \tag{4.19}$$

which geometrically corresponds to a straight line passing through the "true" force application point $\mathbf{p}$ that is directed along the "true" interaction force $\mathbf{F}$. This line is described by the following parametric expressions

$$\delta\mathbf{F} = \mathbf{0}; \quad \delta\mathbf{p} = \mu_1\mathbf{F} \tag{4.20}$$

where the parameter μ_1 is any real number, i.e. $\mu_1 \in \mathbb{R}$. It is worth mentioning that the rank deficiency $d = 1$ may arise in many cases. It is clear that for $k \geq 6$ it is true because of Statement 7. However, even for $k = 5$ the same rank deficiency is generally observed, except for some singular cases when the force action line intersects the kth joint axis. Relevant examples are presented in Cases 1 and 2 in Table 4.1, which also clearly shows the relation between the ranks of the linearized system matrix $\mathbf{D}_k$ and extended Jacobian matrix $\mathbf{J}_k^w$. In particular, the expressions for the extended Jacobian (4.3) do not include the interaction force. For this reason, relevant $rank(\mathbf{J}_k^w)$ depends on the manipulator kinematic configuration only. In contrast, because of the force-related components $\mathbf{F} \times \mathbf{e}_i$ in the first column of the matrix $\mathbf{D}_k$, its rank deficiency d also depends on the interaction force direction. So, depending on the interaction force, for $k \geq 5$ the rank deficiency d of the matrix $\mathbf{D}_k$ may be either $d = 1$ or $d \geq 2$.

Summarizing the case $d = 1$, it is worth mentioning that here the interaction force amplitude and direction are unique, while its application point is non-unique and may belong to any point of the force action line. This case is graphically presented in Fig. 4.1, which clearly shows that further application of relative geometric constraints $\mathbf{p} \in \boldsymbol{\Omega}_k$ describing the robot link surfaces allows to reduce the continuous set of the potentials solutions for $\mathbf{p}$ to the finite one. The latter is then reduced by application of the friction cone constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$, which usually allows to obtain a unique solution for force application point $\mathbf{p}$. Hence, the case of $d = 1$ is not really a singular one, it provides the same solution as in the previous section with $rank(\mathbf{J}_k^w) = 6$.

Case $d = 2$. This case generally arises for $k = 4$ and also in some specific cases with $k > 4$ if the force action line intersects the last joint axes (see Table 4.2). It yields a two-dimensional nullspace

$$\begin{bmatrix} \mathbf{u}^F \\ \mathbf{u}^p \end{bmatrix}_{6\times 2} = \begin{bmatrix} \mathbf{u}_1^F & \mathbf{u}_2^F \\ \mathbf{u}_1^p & \mathbf{u}_2^p \end{bmatrix}_{6\times 2} \tag{4.21}$$

where the first column is the same as in the previous case

$$\begin{bmatrix} \mathbf{u}_1^F \\ \mathbf{u}_1^p \end{bmatrix}_{6\times 1} = \begin{bmatrix} \mathbf{0} \\ \mathbf{F} \end{bmatrix}_{6\times 1} \tag{4.22}$$

Geometrically, such nullspace gives a plane for all possible force application points $\mathbf{p} \in \mathbb{R}^3$. This plane is formed by the "true" force vector $\mathbf{F}$ and the nullspace vector $\mathbf{u}_2^p$. It is clear that for each point $\mathbf{p}$ belonging to this plane the force vector $\mathbf{F}$ is unique, which follows from the parametric expressions

$$\delta\mathbf{F} = \mathbf{0} + \mu_2\mathbf{u}_2^F; \qquad \delta\mathbf{p} = \mu_1\mathbf{F} + \mu_2\mathbf{u}_2^p \tag{4.23}$$

where μ_1 and μ_2 are any real numbers, i.e. $\mu_1, \mu_2 \in \mathbb{R}$. The latter means that the parametric equations (4.23) define a two-dimensional vector field $\{\mathbf{F} + \delta\mathbf{F}(\mathbf{p} + \delta\mathbf{p}),\ \mathbf{p} + \delta\mathbf{p}\}$ in the neighborhood of the "true" solution $(\mathbf{F}, \mathbf{p})$. However, it essentially differs from the pencil of solutions observed in the 2D case (see Fig. 3.2 from the previous chapter). This difference is caused by the linearization of the original static equilibrium equations (4.16), where the second order variations $\delta\mathbf{p}^T \cdot \delta\mathbf{F}$ were neglected. In particular as follows from expressions (4.23), the linearized system does not allow to shift the force along its action line, i.e. this locally simplified system violates the global principal property of the considered mechanical system formulated in Statement 5.

Typical examples of the rank deficiency $d = 2$ are presented in Table 4.2, which also shows a relation between the ranks of linearized system matrix $\mathbf{D}_k$ and the extended

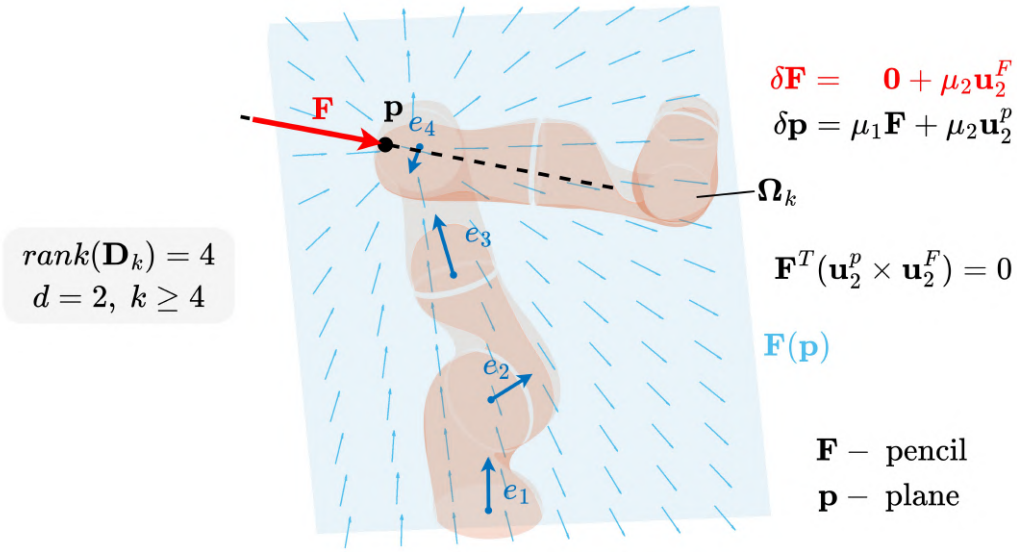


Figure 4.2: Graphical representation of general solution of static equilibrium equations for the case $d = 2$. The "true" interaction force $\mathbf{F}$ is shown in red with its application point $\mathbf{p}$ in black, while the blue arrows show all possible force directions and the blue plane shows all possible application points satisfying the static equilibrium equations.

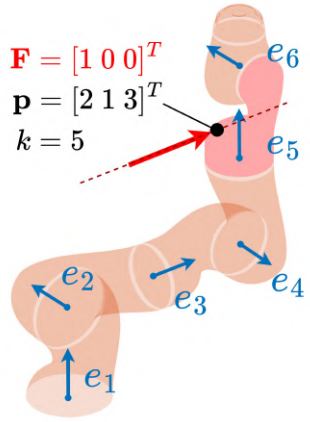
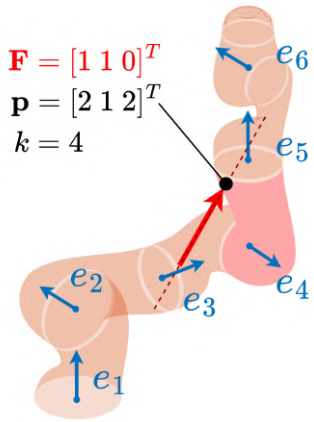
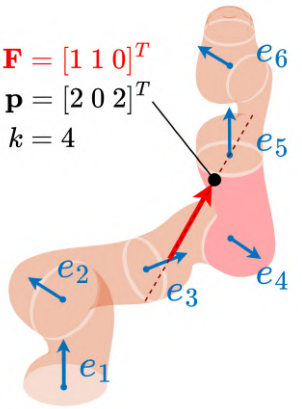
Jacobian $\mathbf{J}_k^w$. It is clear that this case generally arises when $k = 4$ (see Case 4), but it can also appear for $k = 5$ (see Case 3). On the other hand, some specific directions of the interaction force may cause higher rank deficiency $d = 3$ for $k = 4$ (see Case 5), which obviously increases ambiguity in the interaction parameters identification.

Summarizing the case $d = 2$, it is worth mentioning that here the solution for both the interaction force and its application point are non-unique, but they are mutually dependent and constrained by a plane. The latter directly follows from the nullspace of the linearized system (4.18), which describes the geometric properties locally. However, it can be proved that this plane-related geometry of the general solution is also valid for the original non-linear system of the static equilibrium equations (1.11). Moreover, as shown in Fig. 4.2 generated for the KUKA iiwa manipulator, the general solution for the interaction parameters is described by a vector field $\mathbf{F}(\mathbf{p})$ that can be treated as the pencil of straight lines belonging to the plane.

Case $d = 3$. This case generally occurs when the interaction force is applied to the third link, i.e. $k = 3$. Also, it can be observed for $k > 3$ if the force is applied to the following links in such a way that the corresponding linearized system of static equilibrium consists of three independent equations only. It yields the three-dimensional nullspace

$$\begin{bmatrix} \mathbf{u}^F \\ \mathbf{u}^p \end{bmatrix}_{6\times 3} = \begin{bmatrix} \mathbf{u}_1^F & \mathbf{u}_2^F & \mathbf{u}_3^F \\ \mathbf{u}_1^p & \mathbf{u}_2^p & \mathbf{u}_3^p \end{bmatrix}_{6\times 3} \tag{4.24}$$

Table 4.2: Examples of rank deficiency $d = 2$ in systems of the static equilibrium equations: comparison of linearized and extended systems with matrices $\mathbf{D}_k$ and $\mathbf{J}_k^w$

Case 3: $d = 2,\ k = 5$	**Case 4:** $d = 2,\ k = 4$	**Case 5:** $d = 3,\ k = 4$
		
$rank(\mathbf{D}_k) = 4$	$rank(\mathbf{D}_k) = 4$	$rank(\mathbf{D}_k) = 3$
$\mathbf{D}_k = \begin{bmatrix} -1 & 2 & 0 & 0 & -1 & 0 \\ 2 & 0 & -2 & 0 & 0 & 1 \\ 0 & -2 & 1 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}_{5\times6}$	$\mathbf{D}_k = \begin{bmatrix} -1 & 2 & 0 & 1 & -1 & 0 \\ 1 & 0 & -2 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}_{4\times6}$	$\mathbf{D}_k = \begin{bmatrix} 0 & 2 & 0 & 1 & -1 & 0 \\ 0 & 0 & -2 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}_{4\times6}$
$rank(\mathbf{J}_k^w) = 5$	$rank(\mathbf{J}_k^w) = 4$	$rank(\mathbf{J}_k^w) = 4$
$(\mathbf{J}_k^w)^T = \begin{bmatrix} -1 & 2 & 0 & 0 & 0 & 1 \\ 2 & 0 & -2 & 0 & 1 & 0 \\ 0 & -2 & 1 & 1 & 0 & 0 \\ -2 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{5\times6}$	$(\mathbf{J}_k^w)^T = \begin{bmatrix} -1 & 2 & 0 & 0 & 0 & 1 \\ 1 & 0 & -2 & 0 & 1 & 0 \\ 0 & -1 & 1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}_{4\times6}$	$(\mathbf{J}_k^w)^T = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 & 1 \\ 0 & 0 & -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}_{4\times6}$
$null(\mathbf{D}_k) = \begin{bmatrix} 0 & -1 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{bmatrix}_{6\times2}$	$null(\mathbf{D}_k) = \begin{bmatrix} 0 & -1 \\ 0 & -1 \\ 0 & 0 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}_{6\times2}$	$null(\mathbf{D}_k) = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}_{6\times3}$

Note: In these examples the manipulator kinematics is similar to KUKA iiwa robot. Besides, in order to improve the clarity of numerical expressions, the lengths of the robot links were set to 1, and the manipulator joint angles to either 0 or $\pi/2$, which yields the following vectors $\mathbf{e}_i$ and $\mathbf{p}_i$:

$$\mathbf{e}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{e}_4 = -\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_5 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_6 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\mathbf{p}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{p}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_3 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_4 = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_5 = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix};\quad \mathbf{p}_6 = \begin{bmatrix} 2 \\ 0 \\ 3 \end{bmatrix}$$

where the first column is the same as in the previous cases with rank deficiency $d = 1$ and $d = 2$, presented in (4.22).

Generally, for $d = 3$, both the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$ can not be found in a unique way. Moreover, compared to $d = 2$, the force application point $\mathbf{p}$ is not constrained by a plane, it may be located at any point of $\mathbb{R}^3$. The latter directly follows from the relevant parametric equations

$$\delta\mathbf{F} = \mathbf{0} + \mu_2\mathbf{u}_2^F + \mu_3\mathbf{u}_3^F; \qquad \delta\mathbf{p} = \mu_1\mathbf{F} + \mu_2\mathbf{u}_2^p + \mu_3\mathbf{u}_3^p \tag{4.25}$$

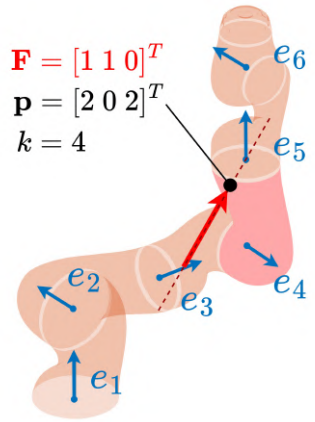
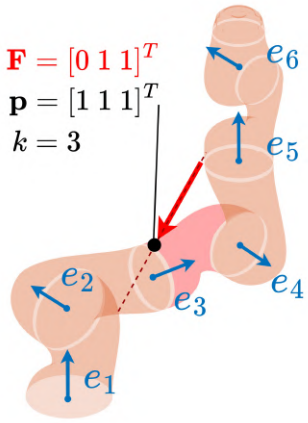
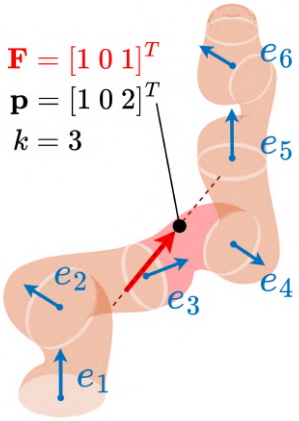
where the parameters μ_1, μ_2 and μ_3 are any real numbers, i.e. $\mu_1, \mu_2, \mu_3 \in \mathbb{R}$. Hence, the ambiguity for $d = 3$ is so high that even further applications of the force direction $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ and surface constraints $\mathbf{p} \in \boldsymbol{\Omega}_k$ do not allow significantly reducing the solution sets, which may include all points of the kth robot link surfaces.

However, it is important to note that for some robots with specific kinematics, such as KUKA iiwa, the case $d = 3$ yields very particular nullspace, where the vectors $\mathbf{u}_2^p$ and $\mathbf{u}_3^p$ are co-linear or at least one of them is equal to zero. This particularity leads to the redaction of the $\mathbf{p}-$solution set from $\mathbf{p} \in \mathbb{R}^3$ to $\mathbf{p} \subset \mathbb{R}^2$, which geometrically is similar to the case $d = 2$, where all possible force application points must belong to a plane. Nevertheless, the $\mathbf{F}-$solution set is not reduced here, i.e. $\mathbf{F} \in \mathbb{R}^2$.

Typical examples of the rank deficiency $d = 3$ are presented in Table 4.3. The first example (see Case 5) corresponds to $k = 4$, where the force is applied to the fourth link but its action line intersects the kth joint axis. The latter causes an additional rank deficiency in the 4×6 matrix $\mathbf{D}_k$. The second example (see Case 6) demonstrates a more typical case, where the interacting force is applied to the third link ($k = 3$) but there are no intersections between the force action line and joint axes. The last example (see Case 7) shows the worst case when the force is applied to the third link ($k = 3$) but its action line intersects the third joint axis, which leads to an increased rank deficiency $d = 4$.

Summarizing the case $d = 3$, it is worth mentioning that here the solutions for both the interaction force $\mathbf{F}$ and its application point $\mathbf{p}$ are non-unique and highly ambiguous. However, it was detected that for some specific robot kinematics, where the first three joint axes are intersecting at the same point, the $\mathbf{p}-$solution set is constrained by a plane. Also, it can be proved that the obtained solution set for the linearized system is similar to one for the original non-linear system of static equilibrium equations (1.11). For the KUKA iiwa manipulator, this solution set is shown in Fig. 4.3, where the obtained force application points $\{\mathbf{p}\}$ belong to the plane and all possible force vectors $\{\mathbf{F}\}$ create the pencil of straight lines for any point on this plane.

Table 4.3: Examples of rank deficiency $d = 3$ in systems of the static equilibrium equations: comparison of linearized and extended systems with matrices $\mathbf{D}_k$ and $\mathbf{J}_k^w$

Case 5: $d = 3,\ k = 4$	**Case 6:** $d = 3,\ k = 3$	**Case 7:** $d = 4,\ k = 3$
		
$rank(\mathbf{D}_k) = 3$	$rank(\mathbf{D}_k) = 3$	$rank(\mathbf{D}_k) = 2$
$\mathbf{D}_k = \begin{bmatrix} 0 & 2 & 0 & 1 & -1 & 0 \\ 0 & 0 & -2 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}_{4\times 6}$	$\mathbf{D}_k = \begin{bmatrix} -1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & -1 \end{bmatrix}_{3\times 6}$	$\mathbf{D}_k = \begin{bmatrix} 0 & 1 & 0 & 0 & -1 & 0 \\ 1 & 0 & -1 & -1 & 0 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 \end{bmatrix}_{3\times 6}$
$rank(\mathbf{J}_k^w) = 4$	$rank(\mathbf{J}_k^w) = 3$	$rank(\mathbf{J}_k^w) = 3$
$(\mathbf{J}_k^w)^T = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 & 1 \\ 0 & 0 & -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}_{4\times 6}$	$(\mathbf{J}_k^w)^T = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}_{3\times 6}$	$(\mathbf{J}_k^w)^T = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 \end{bmatrix}_{3\times 6}$
$null(\mathbf{D}_k) = \begin{bmatrix} 0 & 2 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}_{6\times 3}$	$null(\mathbf{D}_k) = \begin{bmatrix} 0 & 1 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 1 \end{bmatrix}_{6\times 3}$	$null(\mathbf{D}_k) = \begin{bmatrix} 1 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{6\times 4}$

Note: In these examples the manipulator kinematics is similar to KUKA iiwa robot. Besides, in order to improve the clarity of numerical expressions, the lengths of the robot links were set to 1, and the manipulator joint angles to either 0 or $\pi/2$, which yields the following vectors $\mathbf{e}_i$ and $\mathbf{p}_i$:

$$\mathbf{e}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{e}_4 = -\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_5 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_6 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\mathbf{p}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{p}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_3 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_4 = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_5 = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix};\quad \mathbf{p}_6 = \begin{bmatrix} 2 \\ 0 \\ 3 \end{bmatrix}$$

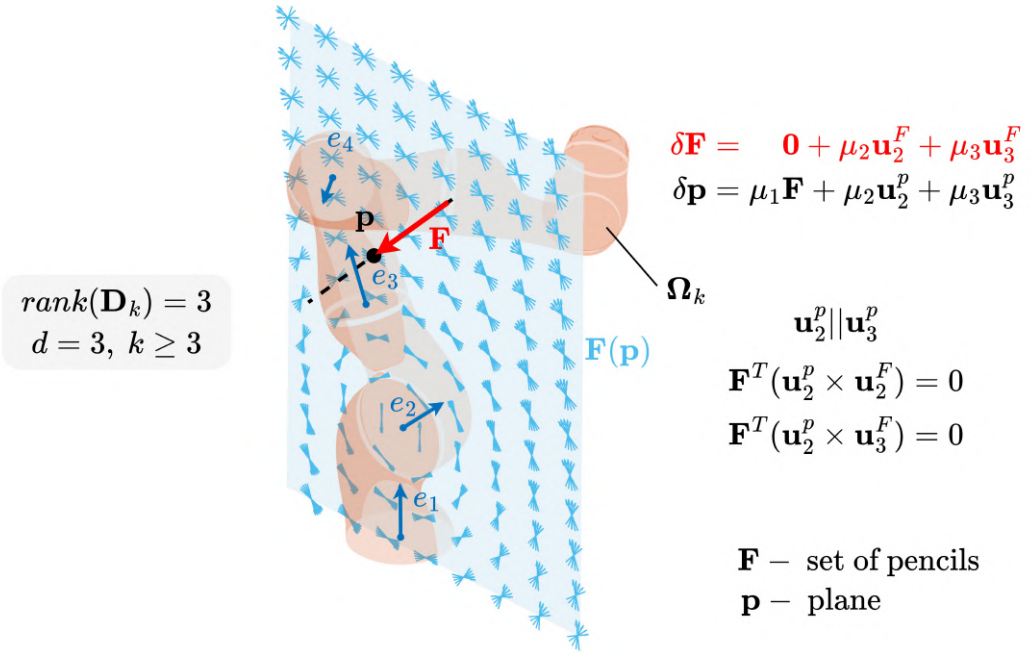
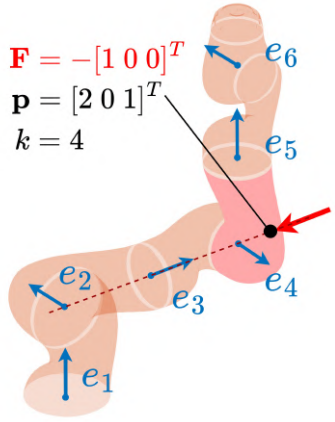
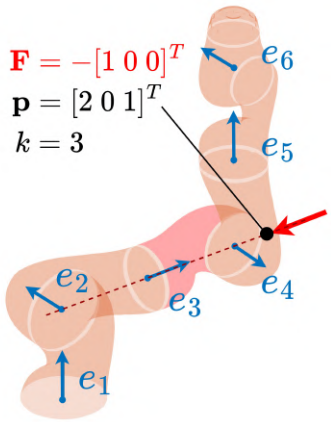
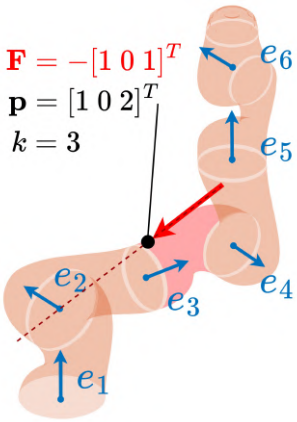


Figure 4.3: Graphical representation of general solution of static equilibrium equations for the case $d = 3$. The "true" interaction force $\mathbf{F}$ is shown in red with its application point $\mathbf{p}$ in black, while the blue arrows show all possible force directions and the blue plane shows all possible application points, satisfying the static equilibrium equations.

Case $d \geq 4$. This case is extremely ambiguous and arises when the interaction force is applied to the lower links of the robot, i.e. $k = 1$ or $k = 2$. Here, the relevant system of static equilibrium equations consists of single or two independent equations and the identification process generally does not allow to estimate the interaction parameters certainly. The main difficulty, in this case, is related to the nullspace dimension, which allows a very high variety of possible interaction forces and their application points, which satisfy the static equilibrium equations. Moreover, this uncertainty can not be reduced by applying the relevant geometric constraints. Besides, in contrast to the previous cases of $d \leq 3$, even for the robot KUKA iiwa with specific kinematics, there were no special properties detected that allow to reduce the solution set. So, if the interaction force is applied to the lower links with $k \leq 2$, the identification of the interaction parameters is not possible for the robots equipped with the joint torque sensors only. Although in practice, this case is not important since it is not typical for human-robot collaboration in an industrial environment.

"Zero-torque" cases. In addition to the above presented singular cases, there may be other specific situations when a non-zero interaction force causes zero torque in the robot joints. So, even the fact of the human-robot interaction cannot be detected. Geometrically, such cases arise when the force action line simultaneously intersects all joint axes, starting from the first to the kth one. As a result, the interaction

Table 4.4: Examples of "zero-torque" cases with $\tau_i = 0$, $\forall i$ and $\mathbf{F} \neq \mathbf{0}$, where the interaction force amplitude is non-identifiable but the force direction can be found

Case 8: $d = 3,\ k = 4$	**Case 9:** $d = 4,\ k = 3$	**Case 10:** $d = 4,\ k = 3$
		
$rank(\mathbf{D}_k) = 3$	$rank(\mathbf{D}_k) = 2$	$rank(\mathbf{D}_k) = 2$
$\mathbf{D}_k = \begin{bmatrix} 0 & 2 & 0 & 0 & 1 & 0 \\ 0 & 0 & -2 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{4\times6}$	$\mathbf{D}_k = \begin{bmatrix} 0 & 2 & 0 & 0 & 1 & 0 \\ 0 & 0 & -2 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{3\times6}$	$\mathbf{D}_k = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & -1 & 1 & 0 & -1 \\ 0 & -1 & 0 & 0 & -1 & 0 \end{bmatrix}_{3\times6}$
$rank(\mathbf{J}_k^w) = 4$	$rank(\mathbf{J}_k^w) = 3$	$rank(\mathbf{J}_k^w) = 3$
$(\mathbf{J}_k^w)^T = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 & 1 \\ 0 & 0 & -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix}_{4\times6}$	$(\mathbf{J}_k^w)^T = \begin{bmatrix} 0 & 2 & 0 & 0 & 0 & 1 \\ 0 & 0 & -2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}_{3\times6}$	$(\mathbf{J}_k^w)^T = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 \end{bmatrix}_{3\times6}$
$null(\mathbf{D}_k) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}_{6\times3}$	$null(\mathbf{D}_k) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}_{6\times4}$	$null(\mathbf{D}_k) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}_{6\times4}$

Note: In these examples the manipulator kinematics is similar to KUKA iiwa robot. Besides, in order to improve the clarity of numerical expressions, the lengths of the robot links were set to 1, and the manipulator joint angles to either 0 or $\pi/2$, which yields the following vectors $\mathbf{e}_i$ and $\mathbf{p}_i$:

$$\mathbf{e}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{e}_4 = -\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix};\quad \mathbf{e}_5 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{e}_6 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\mathbf{p}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix};\quad \mathbf{p}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_3 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_4 = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix};\quad \mathbf{p}_5 = \begin{bmatrix} 2 \\ 0 \\ 2 \end{bmatrix};\quad \mathbf{p}_6 = \begin{bmatrix} 2 \\ 0 \\ 3 \end{bmatrix}$$

force is completely compensated by the robot mechanical structure and "not visible" for the joint torque sensors. A number of such "zero-torque" cases is presented in Table 4.4, which also shows relevant matrices $\mathbf{D}_k$ and $\mathbf{J}_k^w$. As follows from this table, for some specific robot configurations and interaction force directions, the identification technique based on the considered static equilibrium equations is not able to find the force amplitude. However, the force action line can be identified either in a unique or non-unique way. So, such "zero-torque" cases can be also treated in the frame of the identification technique proposed in this thesis.

Resume of singular cases. The above-presented analysis of singular cases related to the rank deficiency d of the linearized static equilibrium system of k equations can be summarized as follows:

- $\boldsymbol{d = 1,\ k \geq 5}$: the interaction force $\mathbf{F}$ is identified in a unique way, the solution set of force application points $\{\mathbf{p}\}$ belong to the uniquely identified force action line passing through "true" force application point $\mathbf{p}^0$;
- $\boldsymbol{d = 2,\ k \geq 4}$: the solution sets of possible interaction forces $\{\mathbf{F}\}$ and their application points $\{\mathbf{p}\}$ form a single pencil of straight lines, belonging to a uniquely identified plane passing through "true" force $\mathbf{F}^0$ and its application point $\mathbf{p}^0$;
- $\boldsymbol{d = 3,\ k \geq 3}$: generally, the sets of interaction forces $\{\mathbf{F}\}$ and their application points $\{\mathbf{p}\}$ form a three-dimensional vector field but for KUKA iiwa robot the set of points $\{\mathbf{p}\}$ is constrained by a plane, where the force $\mathbf{F}$ is non-unique;
- $\boldsymbol{d \geq 4,\ k \geq 1}$: the system of static equilibrium equations provides extremely high ambiguity for the solution sets of $\{\mathbf{F}\}$,$\{\mathbf{p}\}$, so the identification of the desired interaction parameters is not feasible using such limited measurement data.

It should be noted that the above-described solution sets are valid for robot kinematics similar to KUKA iiwa, i.e. for the robot with three intersecting joint axes at the base, which also matches a large variety of commercially available collaborative industrial manipulators. These solutions sets will be further used in the next section, where the force direction and surface constraint are applied to reduce these sets and obtain a unique solution.

4.3.2 Computing the interaction parameters solutions set

Generally, the desired solutions for the interaction parameters $\{\mathbf{F}, \mathbf{p}\}$ must satisfy both the nonlinear system of static equilibrium equations (1.11) and geometric constraints $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$, $\mathbf{p} \in \boldsymbol{\Omega}_k$. In the previous subsection, the linearized version of the static equilibrium system was analyzed in detail, without taking into account these constraints. The latter allowed us to discover the most essential properties of the solution sets, such as geometry and dimensions of the possible interaction force vectors $\{\mathbf{F}\}$ and

their application points $\{\mathbf{p}\}$. In particular, it was discovered that the static equilibrium system never provides a unique solution and there are three main cases, which should be considered separately. Even in the best case, the solution for the interaction force $\mathbf{F}$ is unique but solutions for its application point $\{\mathbf{p}\}$ form a straight line. In other cases, the ambiguity in identification of $\mathbf{F}$, $\mathbf{p}$ is essentially higher, with $\{\mathbf{F}\}$ forming some pencils and $\{\mathbf{p}\}$ forming a plane, etc. To reduce this ambiguity, the relevant geometrical constraints should be also applied, which allows to obtain physically feasible solutions only (or even a unique one). So, computing of such reduced solution set is the focus of this subsection.

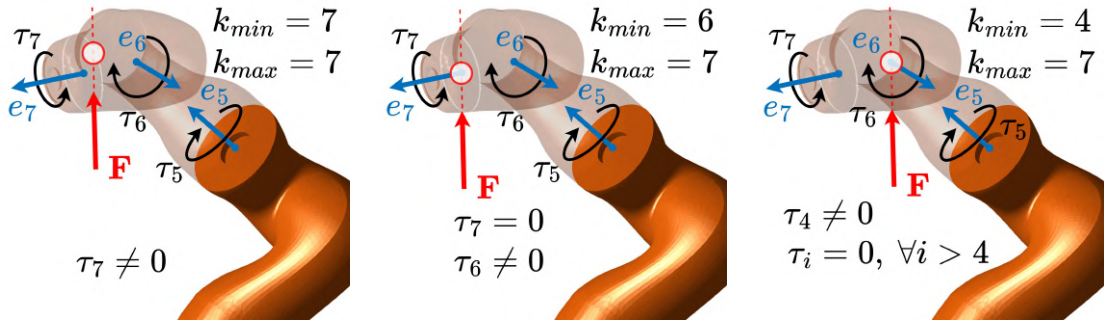
While computing the desired solution sets, it is assumed that the robot *link surfaces* are described by discretized model and are presented as conventional *triangular meshes*, which are widely used in 3D graphics and CAD modeling. In such representation, each triangular element locally describes the robot link surface to which the interaction force can be applied to, and its normal vector defines possible force directions. The latter can be directly used for applying the surface constraint $\mathbf{\Omega}_k$ and force direction constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ in the considered interaction parameters identification problem.

Below, two techniques for computing reduced set of the interaction parameters $\{\mathbf{F}, \mathbf{p}\}$ are proposed. The first of them is based on a straightforward enumeration of surface primitives and selecting ones satisfying both the static equilibrium system and the geometric constraints. The second technique utilizes some useful geometric properties of the solution sets discovered in the previous subsection. Both of these approaches are described in detail, evaluated, and compared in terms of their accuracy, time performance.

Approach 1: Straightforward enumeration of triangle mesh elements

Basic Idea. *To obtain the desired set of interaction parameters* $\{\mathbf{F}, \mathbf{p}\}$*, enumerate all triangle mesh elements describing the* k*th robot link to which the force can be applied to. For each of these elements, apply the least squares technique to the modified static equilibrium system integrating the geometric constraints and compute relevant residuals. Then, create the solution set composed of the estimated interaction parameters* $\{\mathbf{F}, \mathbf{p}\}$ *corresponding to the smallest least-squares residuals.*

In the frame of the first approach, it is assumed that the number of triangular primitives $\mathbf{\Delta}_i$ describing the robot links surfaces is large enough and their size is rather small. So, a single center point $\mathbf{p}_c^i$ of the triangle and its normal vector $\mathbf{n}^i$ can be used to represent the whole element. The latter allows us to "fix" the position of the potential force application point $\mathbf{p} = \mathbf{p}_c^i$ and find the corresponding interaction force $\mathbf{F}(\mathbf{p}_c^i)$ using the modified least squares technique applied to the system of k static equilibrium



(a) No axis intersection (b) Intersects one axis (c) Intersects three axes

Figure 4.4: Estimation of k-index range for different cases when the force action line intersects joint axis producing zero torque(s).

equations. It is clear that only the force direction constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ must be integrated into these modified equations since the surface constraint $\mathbf{p} \in \boldsymbol{\Omega}_k$ is already taken into account by considering a single mesh element. Further, by enumerating all mesh elements, it is possible to compose a solution set consisting of the force vectors $\{\mathbf{F}\}$ and their application points $\{\mathbf{p}\}$, which yield zero least squares residuals. However, because of the applied surface discretization, pure zero residuals can never be obtained in practice, so some threshold ε_w must be established to select the mesh elements belonging to the final solution set.

To estimate the number of equations k included in the considered static equilibrium system, the decision rule similar to one used for the planar 2D case (3.36) can be also applied. For the 3D case, the lower bound of the k-index is estimated using the basic expressions (1.12). Then, the upper bound of k is obtained by successively increasing the number of considered static equilibrium equations and verifying the extended system (4.2) consistency. This consistency test is based on the residual analysis of the corresponding least square solution. It is clear that some tolerances should be also defined to take into account the measurement noise. In a more formal way, this technique for computing the k-index range is presented as

$$\begin{aligned} &\textbf{\textit{Lower bound:}}\ k_{\min} = \arg\max_i \left(|\tau_i| \geq \delta_\tau\right) \\ &\textbf{\textit{Upper bound:}}\ k_{\max} = \arg\max_i \left(\left\| \left(\mathbf{J}_i^w\left(\mathbf{q}, \mathbf{p}^0\right)\right)^T \cdot \mathbf{W}_i - \boldsymbol{\tau} \right\| \leq \delta_w\right) \end{aligned} \tag{4.26}$$

where the $\mathbf{W}_i$ denotes the least squares solution of the extended system composed of the first i static equilibrium equations and δ_τ and δ_w are the tolerances for the joint torque measurements and least squares residuals respectively.

A graphical illustration of the above decision rule is presented in Fig. 4.4 where three typical cases are shown. In the case, Fig. 4.4a the force is applied to the manipulator

Algorithm 10: Estimation of interaction parameters using Approach 1

```
1  Function Approach1(q, τ, Robot):
       Input  : q is joints positions, τ is torque measurements, Robot is a
                structure with robot geometric and kinematic parameters.
       Output: interaction parameters F, p
2      τ_b = |τ| > δ_τ;
3      k = find(τ_b ≠ 0, last);
4      while k ≤ n do
5          J = J_k^w(q, p^0);
6          W = (J^#)^T · τ;
7          τ_d = J^T · W − τ;
8          if ||τ_d|| ≤ δ_w then
9              for i = Ω_k do
10                 p_c = (sv_1^{k,i} + sv_2^{k,i} + sv_3^{k,i})/3;
11                 α, ε = min_{α≥0} ||τ − J_k(q, p_c, F^α)^T α||;
12                 if ε ≤ ε_w then
13                     F_i = Σ_{j=1}^{4} α_j · F_j^α;
14                     p_i = p_c;
15             k = k + 1;
16         else
17             break;
```

7th link and does not intersect any of the joint axes. So, the k-index range consists of one index only. The case Fig. 4.4b is similar but the force action line is intersecting the 7th joint axis, which leads to zero torque $\tau_7 = 0$. Here, the basic decision rule (1.12) produces the undervalued interacting link index $k = 6$, which is used as a lower bound k_{min}. To take into account above mentioned particularity, the k-index upper bound estimate is based on residual analysis and yields $k_{max} = 7$, which coincides with the actual k-value. A more complex case is shown in Fig. 4.4c, where the force action line intersects the last three joint axes at the same time. So, the last non-zero torque is obtained at the 4th joint and the k-index range includes indexes of the four last links. The latter case is rare but possible, since the spherical wrist kinematic configuration, where the last three joint axes intersect at the same point, is quite popular in robotics.

In a more formal way, Approach 1 is described in Algorithm 10. Here, at lines 2, 3 the k-index range is estimated and the *while loop* enumerates through the $k_{min}, ..., k_{max}$. It is clear that generally, it is necessary to try to find a solution for each of the kth links from this range, but for some of them, there may be no feasible solution at all. The main *for loop* (see line 9) enumerates through all mesh elements of the kth link and estimates the interaction force $\mathbf{F}$ by solving the modified static equilibrium

Table 4.5: Simulation results of Approach 1 without measurement noise

Param.	Metrics	k-index					avg
		7	**6**	**5**	**4**	**3**	
$\|\|\Delta\mathbf{F}\|\|$ $[N]$	mean	0.24	0.54	0.74	1.79	2.06	1.09
	std	0.18	0.59	0.91	1.77	2.29	1.56
	max	1.14	5.23	9.92	11.4	17.7	17.7
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	1.27	2.11	2.53	2.68	2.92	2.32
	std	0.52	0.92	1.13	1.18	1.37	1.22
	max	3.05	4.72	5.45	6.03	6.69	6.69
$\mathbf{T}_c$ $[ms]$	mean	50.6	89.4	108.4	153.4	205.6	124.0
	std	0.5	21.2	19.1	12.5	16.1	56.2
	min	49.4	49.1	97.4	146.4	194.2	49.1
	max	52.1	103.8	154.2	203.8	254.6	254.6
$\mathbf{f}_c$ $[Hz]$	mean	19.7	11.2	9.22	6.52	4.86	8.06

Note: $\Delta\mathbf{F}$ - estimation error of the interaction force $\|\mathbf{F}\| = 100\ N$;
$\Delta\mathbf{p}$ - estimation error of the force application point;
$\mathbf{T}_c$, $\mathbf{f}_c$ - time performance of the estimation algorithm.

system with the force direction constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ using least squares technique. Here, each triangular element is defined as $\boldsymbol{\Delta} = (\mathbf{sv}_1^{k,i},\ \mathbf{sv}_2^{k,i},\ \mathbf{sv}_3^{k,i},\ \mathbf{n}^{k,i})$, where $\mathbf{sv}_j^{k,i}$ is the vertex point, $\mathbf{n}^{k,i}$ is normal to the plane crossing all vertices and the center point is computed as $\mathbf{p}_c^{k,i} = (\mathbf{sv}_1^{k,i} + \mathbf{sv}_2^{k,i} + \mathbf{sv}_3^{k,i})/3$. To speed up computations the friction cone is approximated by a polyhedron, which is defined via four support vectors (see Fig. 2.6 for more details). The latter allows to solve the considered constrained least squares problem using a convex quadratic programming technique (2.8). Further, the desired set of interaction parameters is obtained by simple selection of the triangles with residuals lower than a certain threshold ε_w. It should be mentioned that Algorithm 10 operates with three thresholds that should be tuned to achieve better performance. Two of them δ_τ and δ_w affect the k-index range, while ε_w is used to limit the total number of mesh elements that are included in the solution set.

The efficiency of Approach 1 was evaluated via a simulation study, relevant results are presented in Table 4.5. In this simulation experiment, the interaction force of amplitude $\|\mathbf{F}\| = 100\ N$ was applied at some random points of the robot links #3,..7 and was directed perpendicular to the link surface. In total, 10^3 random configurations were considered. Also, it was assumed that there is no measurement noise for both joint angles $\mathbf{q}$ and torques $\boldsymbol{\tau}$. This study used the 3D CAD model of KUKA iiwa collaborative robot with the edge length of each triangular element about 10 mm. To evaluate the estimation accuracy, two metrics $||\boldsymbol{\Delta}\mathbf{F}||$ and $||\boldsymbol{\Delta}\mathbf{p}||$ were used, which

represent the euclidean distances between the "real" parameter $\mathbf{F}$ or $\mathbf{p}$ and the closest one from the obtained solution set. The simulation result showed that the estimation accuracy is quite low, because of very rough simplification, where each triangle is represented by its center point only. So, the estimated force application points almost never coincide with the real one that may be located at any point of the triangle. In particular, for the force amplitude, the mean estimation error was about 1%, with its maximum value of 17%. For the force application point, the mean estimation error was 2.3 mm, with its maximum value of 6.7 mm, which is in an agreement with the triangle mesh size of about 10 mm. The time performance of this technique is also rather low, on average the computing time was about 120 ms. The latter means that it allows to update estimates of the interaction parameters approximately 8 times per second, but only 4 times per second in the worst case. Also, it should be noted that the algorithm is faster if the interaction force is applied to the links that are closer to the end-effector, which obviously yields a smaller k-index range. It is clear that both the algorithm accuracy and its time performance highly depend on the mesh size. But these criteria are competing: smaller discretization gives better accuracy but leads to the computing time increasing.

Approach 2: Intersection of $\{\mathbf{F}, \mathbf{p}\}$-solutions set with triangle mesh

Basic Idea. *To obtain the desired set of interaction parameters* $\{\mathbf{F}, \mathbf{p}\}$*, find first an unconstrained solution for the system of k static equilibrium equations by applying the iterative non-linear least squares technique. Then, using this partial solution, obtain the unconstrained general solution based on the nullspace of the matrix* $\mathbf{D}_k$ *computed for the linearized system. Further, find the geometric intersection of the unconstrained general solution and the triangular mesh describing the robot surface constraint* $\mathbf{p} \in \boldsymbol{\Omega}_k$*. Finally, create the solution set of the interaction parameters* $\{\mathbf{F}, \mathbf{p}\}$ *composed of the obtained points and forces additionally satisfying the force direction constraint* $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$*.*

This approach is based on some useful properties of the solution sets, discovered in the previous subsection. In particular, for typical collaborative manipulators with their first three joint axes intersecting at the same point, the geometrical representation of the solution set for the force application points $\{\mathbf{p}\}$ can be either a plane or a line. The most common example of such manipulators is KUKA iiwa collaborative robot considered in this work. As it was shown above, the geometric shape of the solution set depends on the matrix $\mathbf{D}_k$ rank, which is obtained by linearization of the static equilibrium system in the neighborhood of some solution. So, in this approach, it is proposed to start with some partial solution of the original system of static equilibrium equations (1.11) and estimate the rank deficiency d of the corresponding linearized

Algorithm 11: Estimation of interaction parameters using Approach 2

Function `Approach2(`$\mathbf{q}, \boldsymbol{\tau}, \mathbf{Robot}$`):`
  Input : $\mathbf{q}$ is joints positions, $\boldsymbol{\tau}$ is torque measurements, **Robot** is a structure with robot geometric and kinematic parameters.
  Output: set of identified interaction parameters $\mathbf{F}, \mathbf{p}$.
  $\boldsymbol{\tau}_b = |\boldsymbol{\tau}| > \delta_\tau$;
  $k = \text{find}(\boldsymbol{\tau}_b \neq 0,\ last)$;
  while $k \leq n$ **do**
    $\mathbf{p} = \mathbf{p}^k$;
    $\mathbf{F} = (\mathbf{J}^{\#})^T \cdot \boldsymbol{\tau}$;
    for $i = 1 : i_{max}$ **do**
      $\mathbf{J} = \mathbf{J}_k(\mathbf{q}, \mathbf{p})$;
      $\boldsymbol{\tau}_r = \mathbf{J}^T \cdot \mathbf{F}$;
      $\boldsymbol{\tau}_d = \boldsymbol{\tau}_r - \boldsymbol{\tau}$;
      $\mathbf{D}_k = [\mathbf{J}_k(\mathbf{q}, \mathbf{p}) \quad \mathbf{H}_k(\mathbf{q}, \mathbf{F})]$;
      $[\delta\mathbf{F},\ \delta\mathbf{p}]^T = (\mathbf{D}_k^{\#})^T \cdot \boldsymbol{\tau}_d$;
      $\mathbf{F} = \mathbf{F} + \delta\mathbf{F}$;
      $\mathbf{p} = \mathbf{p} + \delta\mathbf{p}$;
      $d = 6 - \text{rank}(\mathbf{D}_k)$;
      if $||\boldsymbol{\tau}_{prev} - \boldsymbol{\tau}_r|| \leq \varepsilon_\tau$ **then**
        break;
      $\boldsymbol{\tau}_{prev} = \boldsymbol{\tau}_r$;
    if $||\boldsymbol{\tau}_d|| \leq \delta_w$ **then**
      Solution$(k) = [\mathbf{F},\ \mathbf{p},\ d]$;
      $k = k + 1$;
    else
      break;
  $\mathbf{F},\ \mathbf{p} = \text{ApplyConstraints}(\mathbf{Solution},\ \mathbf{Robot})$;

system matrix $\mathbf{D}_k$. To find the partial solution of the original nonlinear system, any common technique can be applied. Further, the desired general solution is generated by using the nullspace of matrix $\mathbf{D}_k$. In particular, if the rank deficiency of this matrix is $d = 1$, the general solution consists of a unique interaction force vector $\mathbf{F}$ and a straight line for potential force application points $\{\mathbf{p}\}$. In contrast, for the rank deficiencies $d = 2$ and $d = 3$, the potential force application points $\{\mathbf{p}\}$ belong to a plane, where the force vector $\mathbf{F}(\mathbf{p})$ is either unique for $d = 2$ or non-unique for $d = 3$. It is worth mentioning that such a general solution set, obtained from the static equilibrium system only, does not take into account the geometric constraints $\mathbf{p} \in \boldsymbol{\Omega}_k$ and $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. The latter means that some solutions of this set may be practically unfeasible and must be eliminated.

Algorithm 12: Applying of the geometrical constraint for solutions set

Function `ApplyConstraints(`**Solution**, **Robot**`)`:
  Input : **Solution** is general solution with set of $\{\mathbf{F}, \mathbf{p}\}$, **Robot** is a structure with robot geometric and kinematic parameters.
  Output: set of interaction parameters $\mathbf{F}, \mathbf{p}$.
  for $k = 1 : n$ **do**
    switch **Solution**$(k).d$ **do**
      case $d = 1$ **do**
        $\mathbf{p}_{int} = \text{IntersectLineMesh}\,(\textbf{Solution}(k),\ \textbf{Robot})$;
        for $i = 1 : \text{size}(\mathbf{p}_{int})$ **do**
          if $\left|\mathbf{n}^{k,i} \times \textbf{Solution}(k).\mathbf{F}\right| \leq \mu \left(\mathbf{n}^{k,i} \cdot \textbf{Solution}(k).\mathbf{F}\right)$ **then**
            $\mathbf{F} = \textbf{Solution}(k).\mathbf{F}$;
            $\mathbf{p} = \mathbf{p}^{i}_{int}$;
      case $d = 2$ **do**
        $\mathbf{p}_{int} = \text{IntersectPlaneMesh}\,(\textbf{Solution}(k),\ \textbf{Robot})$;
        for $i = 1 : \text{size}(\mathbf{p}_{int})$ **do**
          $\mathbf{p}_c = (\mathbf{p}^{i}_{int} + \mathbf{p}^{i+1}_{int})/2$;
          $\mathbf{F}_c = \text{pencil}(\mathbf{p}_c,\ \textbf{Solution}(k))$;
          if $\left|\mathbf{n}^{k,i} \times \mathbf{F}_c\right| \leq \mu \left(\mathbf{n}^{k,i} \cdot \mathbf{F}_c\right)$ **then**
            $\mathbf{F} = \mathbf{F}_c$;
            $\mathbf{p} = \left[\mathbf{p}^{i}_{int},\ \mathbf{p}^{i+1}_{int}\right]$;
      case $d = 3$ **do**
        $\mathbf{p}_{int} = \text{IntersectPlaneMesh}\,(\textbf{Solution}(k),\ \textbf{Robot})$;
        for $i = 1 : \text{size}(\mathbf{p}_{int})$ **do**
          $\mathbf{p}_c = (\mathbf{p}^{i}_{int} + \mathbf{p}^{i+1}_{int})/2$;
          $\mathbf{F}_c = \text{set\_pencils}(\mathbf{p}_c,\ \textbf{Solution}(k))$;
          if $\left|\mathbf{n}^{k,i} \times \mathbf{F}_c\right| \leq \mu \left(\mathbf{n}^{k,i} \cdot \mathbf{F}_c\right)$ **then**
            $\mathbf{F} = \mathbf{F}_c$;
            $\mathbf{p} = \left[\mathbf{p}^{i}_{int},\ \mathbf{p}^{i+1}_{int}\right]$;

To apply the geometric constraints $\mathbf{p} \in \boldsymbol{\Omega}_k$ and $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ to the unconstrained general solution set, it is reasonable to start with possible locations of the force application point $\mathbf{p}$, which obviously must belong to the robot link surface. Since for the considered type of robots, the unconstrained solution set for $\{\mathbf{p}\}$ can be either a line or a plane in 3D space, imposing the surface constraint $\mathbf{p} \in \boldsymbol{\Omega}_k$ is computationally simple and is equivalent to finding a line or a plane intersection with the triangular mesh elements. In fact, in computation geometry there exists a number of efficient techniques to solve this problem, such as Möller-Trumbore algorithm [72]. Further, the force direction constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ is applied in a straight-forward way by examining the force vectors $\{\mathbf{F}\}$ feasibility at the points $\{\mathbf{p}\}$ obtained at the previous step, where

the intersections of the line/plane with the triangular mesh elements are computed. The latter yields the final reduced solution set composed of feasible force vectors $\{\mathbf{F}\}$ and their application points $\{\mathbf{p}\}$ satisfying both the static equilibrium equations and the considered geometric constraints.

In a more formal way, Approach 2 is described in Algorithm 11. Similarly to the previous approach, lines 2, 3 deal with the estimation of the k-index range and the *while loop* enumerate through the $k_{min}, ..., k_{max}$, but here the residual $\boldsymbol{\tau}_d$ is computed for the original system of static equilibrium equation (1.11). In the main *while loop*, the partial solution is obtained for each of the kth robot links using the Gauss-Newton method. As the initial guess, it is used the kth joint center point $\mathbf{p}^k$ for the force application point, and the corresponding force vector $\mathbf{F}^k$ computed from the original static equilibrium system. This method is quite simple, it requires only the first-order derivatives that are combined in the matrix $\mathbf{D}_k$. It is applied iteratively and yields some partial unconstrained solution of the original system of the static equilibrium equations. The maximum number of iterations is bounded by parameter i_{max} at the line 6. However, there is an additional break statement at line 16 to stop the loop if the residual variation is too small compared to the previous iteration. In our simulation study, the average number of iterations was about 6, which allowed to achieve the desired accuracy in an acceptable time. It should be noted that, in order to avoid high computational errors, the pseudoinverse in this algorithm is implemented using the singular value decomposition. In contrast to the classical Moore-Penrose inverse, it can provide a solution even for the rank deficient matrix $\mathbf{D}_k$.

In the above-described algorithm, the geometrical constraints are applied at line 24. The relevant technique is described in detail in Algorithm 12. Depending on the rank deficiency d of the matrix $\mathbf{D}_k$ there are three possible cases implemented as *switch statement*, which is applied to all robot links (see line 2). In the case of $d = 1$ the intersection of the triangular mesh describing the kth link surface and the obtained force action line is computed at line 5. It is assumed here that multiple intersections are possible, so all of them are verified to satisfy the force direction constraint (i.e. friction cone) at line 7. The cases $d = 2$ and $d = 3$ are handled in a similar way, except for the intersection computing, where the line-mesh is replaced by the plane-mesh crossing. Since the intersection for such cases $d = 2$ and $d = 3$ produce the set of line segments on the robot surface, their center points $\mathbf{p}_c$ are used to calculate corresponding force vectors $\mathbf{F}_c$, because the triangular mesh elements are assumed to be small enough. Further, the desired set of interaction parameters is obtained by including in it all intersection points $\{\mathbf{p}\}$ or line segments (defined by their endpoints) and corresponding forces $\{\mathbf{F}\}$, which satisfy the friction cone constraints. It should

Table 4.6: Simulation results of Approach 2 without measurement noise

Param.	Metrics	k-index					avg
		7	**6**	**5**	**4**	**3**	
$\|\|\Delta\mathbf{F}\|\|$ $[N]$	mean	10^{-14}	10^{-11}	10^{-10}	10^{-10}	10^{-14}	10^{-10}
	std	10^{-13}	10^{-10}	10^{-9}	10^{-9}	10^{-14}	10^{-10}
	max	10^{-12}	10^{-9}	10^{-9}	10^{-9}	10^{-13}	10^{-9}
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	10^{-8}	10^{-7}	10^{-8}	10^{-8}	10^{-7}	10^{-7}
	std	10^{-8}	10^{-7}	10^{-7}	10^{-7}	10^{-6}	10^{-7}
	max	10^{-7}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
$\mathbf{T}_c$ $[ms]$	mean	0.33	0.59	0.73	1.99	2.84	1.36
	std	0.03	0.06	0.27	0.57	0.59	1.05
	min	0.26	0.45	0.49	1.29	2.04	0.26
	max	3.05	4.72	5.45	6.03	6.55	6.55
$\mathbf{f}_c$ $[Hz]$	mean	2989	1680	1362	500	353	738

Note: $\Delta\mathbf{F}$ - estimation error of the interaction force $\|\mathbf{F}\| = 100\ N$;
$\Delta\mathbf{p}$ - estimation error of the force application point;
$\mathbf{T}_c$, $\mathbf{f}_c$ - time performance of the estimation algorithm.

be mentioned that Algorithm 11 operates with four parameters that should be tuned to achieve better performance. Two of them δ_τ and δ_w affect the k-index range, while i_{max} and ε_τ are used to limit the total number of iterations.

The efficiency of Approach 2 was evaluated via a simulation study, relevant results are presented in Table 4.6. In this simulation experiment, the interaction force of amplitude $\|\mathbf{F}\| = 100\ N$ was applied at some random points of the robot links #3,..7 and was directed perpendicular to the link surface. In total, 10^3 random configurations were considered. Also, it was assumed that there is no measurement noise for both joint angles $\mathbf{q}$ and torques $\boldsymbol{\tau}$. This study used the same as the above 3D CAD model of KUKA iiwa collaborative robot. To evaluate the estimation accuracy, two metrics $||\mathbf{\Delta F}||$ and $||\mathbf{\Delta p}||$ were used, which represent the euclidean distances between the "real" parameter $\mathbf{F}$ or $\mathbf{p}$ and the closest one from the obtained solution set. The simulation result showed that the estimation accuracy is high and is limited by round-off errors. So, the estimated force application points $\{\mathbf{p}\}$ coincide with the real ones that may be generally located at any point of the mesh triangle. The time performance of this technique is also rather high, on average the computing time was about 1.36 ms. The latter means that Approach 2 allows to update estimates of the interaction parameters approximately 740 times per second on average, but only 155 times per second in the worst case. Also, it should be noted that the computation is much faster if the interaction force is applied to the last links when the k-index range is smaller.

Table 4.7: Comparison of Approach 1 and Approach 2

	mean $\|\|\Delta\mathbf{F}\|\|$	**mean** $\|\|\Delta\mathbf{p}\|\|$	**max** $\mathbf{T}_c$	**mean** $\mathbf{T}_c$
Approach 1	1.09	2.32	254.6	124.0
Approach 2	10^{-10}	10^{-7}	6.55	1.36
Improvement	–	–	~ 40	~ 90

Comparison study: Advantages/Disadvantages of Approaches 1 and 2

To define application areas of each of the above-presented approaches, let us compare them in detail paying the main attention to the *accuracy*, *real-time performance* as well as *capability of singular cases handling*. Also, it is worth comparing the achieved performances with the required ones summarized at the end of Chapter 2.

The principal performance indices of these two approaches are summarized in Table 4.7, which aggregates the most important data from previous Tables 4.5 and 4.6. These data are obtained for the same benchmark problems using the same computing facilities. As follows from them, Approach 2 completely outperforms Approach 1 in terms of estimation *accuracy* for both the force vector $\mathbf{F}$ and its application point $\mathbf{p}$. Also, the time performance of the second approach is essentially better compared to the first one. In particular, its average *cycle time* of 1.36 ms (corresponding to 738 Hz) is more suitable for real-time applications allowing integration in a typical robot control loop, which for collaborative robots usually is running with the frequency of 500 Hz. It should be noted that in order to make this comparison fair, both techniques were implemented as conventional Matlab functions without any parallelization or precompiled code. Their performances were evaluated using a PC with Intel Core i5-4210H 3GHz CPU, 8Gb RAM. Obviously, implementation in a C++ environment, which utilizes multiple CPU cores, allows for significantly reducing the runtime. Nevertheless, Approach 1 has some hidden advantages, which will be clarified below.

It is worth mentioning that both techniques were also evaluated for *capability of singularity handling*. In the frame of the considered problem, the singularities arise either in some specific manipulator configurations (manipulator kinematic singularity) or in the case of some specific force directions and its application point locations (interaction force singularity). To simulate the latter case, the interaction force was applied to the lower robot links, where the joint torque measurements do not allow to obtain a unique solution. In addition, there were considered cases where the force action line intersects the last joint sensor axes yielding zero torques, which makes the estimation of the k-index non-trivial. From relevant simulation studies, it was confirmed that

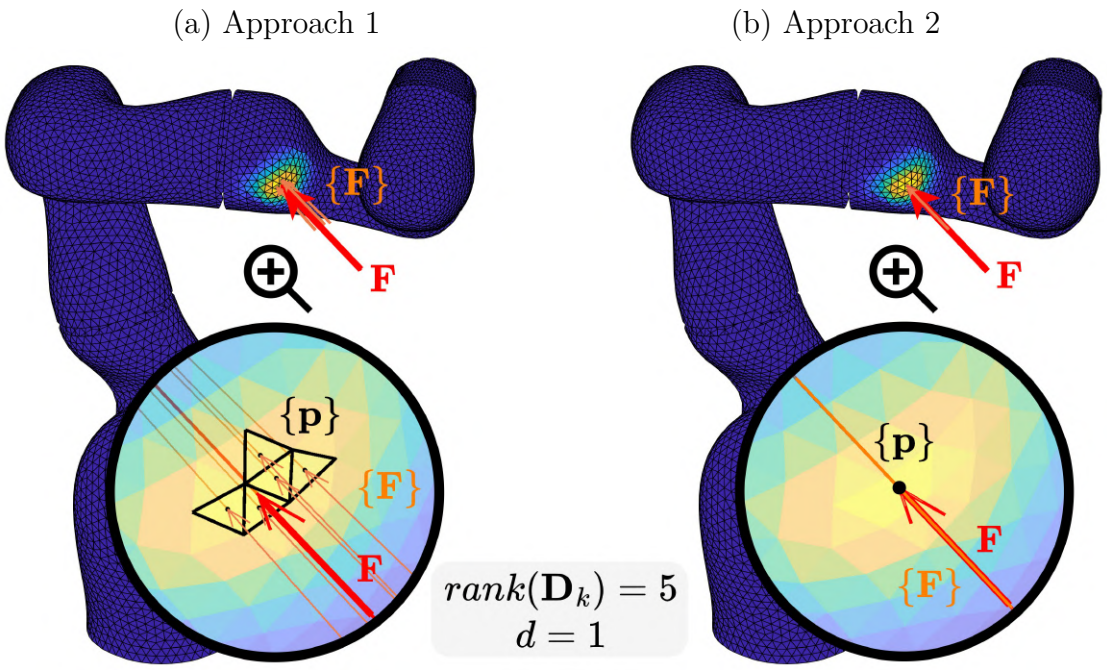


Figure 4.5: Ambiguity of the interaction parameters identification for case $d = 1$. Approach 1 generates several combinations $\{\mathbf{F}\}$, $\{\mathbf{p}\}$ with the force application points located at the centers of triangular mesh elements. Approach 2 provides a unique solution equal to the "real" one. Here, the real interaction force $\mathbf{F}$ is shown with a red arrow, the estimated forces $\{\mathbf{F}\}$ and their application points $\{\mathbf{p}\}$ are shown with orange and black colors respectively.

both approaches are capable to handle the singular configurations producing multiple potential solutions for the interaction parameters $\{\mathbf{F}\}$, $\{\mathbf{p}\}$. This is the essential advantage compared to the existing techniques reviewed in Chapter 2, which can provide a single solution only. On the other hand, the non-uniqueness of the obtained parameters $\mathbf{F}$, $\mathbf{p}$ creates some ambiguity at the following stages dealing with the human-robot interaction handling.

To evaluate the *ambiguity* of the solutions provided by Approach 1 and Approach 2, let us compare them from the geometrical point of view. For the case $d = 1$, relevant illustrations are presented in Fig. 4.5. Here, the static equilibrium equations yield a unique force action line that must be integrated with the surface constraint $\mathbf{p} \in \mathbf{\Omega}_k$. For Approach 1, the surface constraint is presented as $\mathbf{p} \in \{\mathbf{p}_c^1,\ \mathbf{p}_c^2, ...\}$, i.e. the set $\mathbf{\Omega}_k$ is composed of the triangular mesh centers. So, the identification algorithm provides several discrete points $\mathbf{p}_c^i$ with the residuals, which are lower than some threshold ε_w. It is clear that generally, the "real" force application point may be located at any point of the triangle but because of such discretization, the desired interaction parameter $\mathbf{p}$ can not be identified exactly using Approach 1. In contrast, Approach 2 is not limited by such discretization and operates with all points of the triangular mesh elements

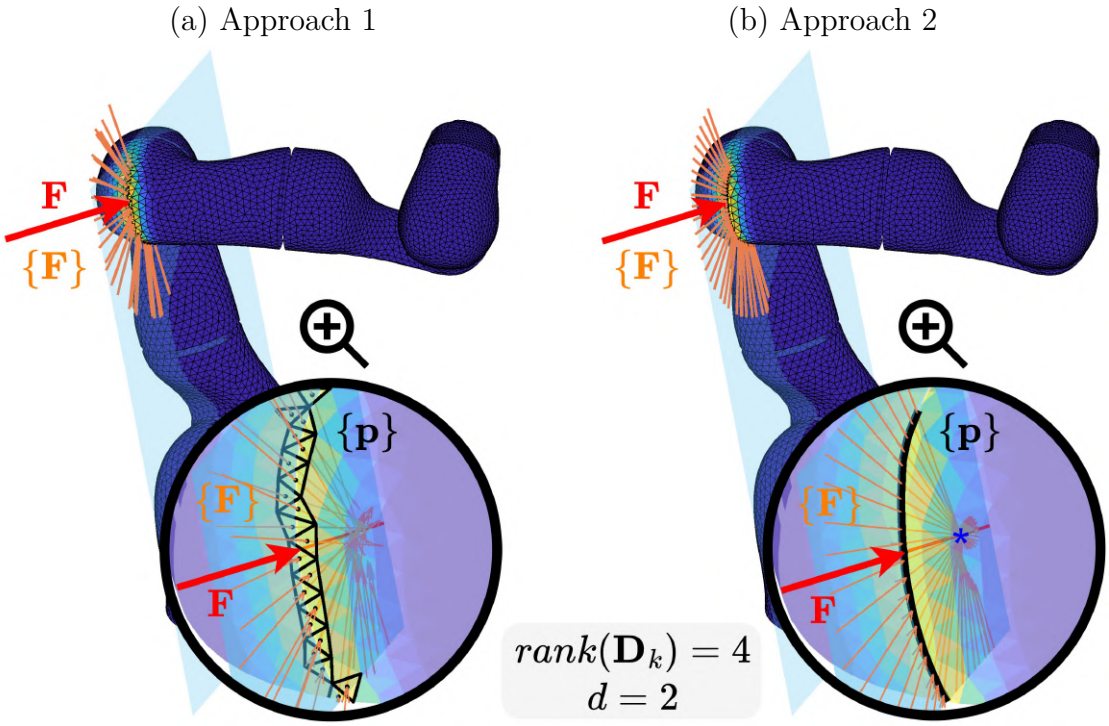


Figure 4.6: Ambiguity of the interaction parameters identification for case $d = 2$. Approach 1 generates a number of combinations $\{\mathbf{F}\}$, $\{\mathbf{p}\}$ with the force application points located at centers of triangular mesh elements, which approximate the intersection of the robot link surface and the plane of potential force action lines. Approach 2 provides a set of $\{\mathbf{F}\}$, $\{\mathbf{p}\}$ that correspond to their exact intersection including the "real" solution.

presenting the surface constraint in the form $\mathbf{p} \in \{\boldsymbol{\Delta}^1, \boldsymbol{\Delta}^2, ...\}$. So, it is able to obtain the set $\{\mathbf{p}\}$, composed of exact intersections between the triangular elements and the force action line. Hence, in the case $d = 1$ Approach 2 provides an exact solution for both the force $\{\mathbf{F}\}$ and its application point $\{\mathbf{p}\}$, while Approach 1 yields some ambiguity, generating the sets $\{\mathbf{F}\}$ and $\{\mathbf{p}\}$ in the neighborhood of the "real" interaction parameters.

For a more hard case with $d = 2$ presented in Fig. 4.6 the ambiguity is essentially higher. As it was proved above, here, the potential solutions for $\{\mathbf{p}\}$ are located on the intersection between a plane and the link surface mesh, while the potential force vectors $\{\mathbf{F}\}$ form by the pencil of straight lines intersecting at the same point. Consequently, Approach 1 generates a number of combinations $\{\mathbf{F}\}$, $\{\mathbf{p}\}$ with the force application points located at centers of triangular mesh elements, which approximate the intersection of the robot link surface and the plane of potential force action lines. It is worth mentioning that these center points are not exactly located at the intersection line, producing slightly inaccurate force vectors whose action lines do not intersect at the same point. In contrast, Approach 2 provides the exact intersection yielding a line of

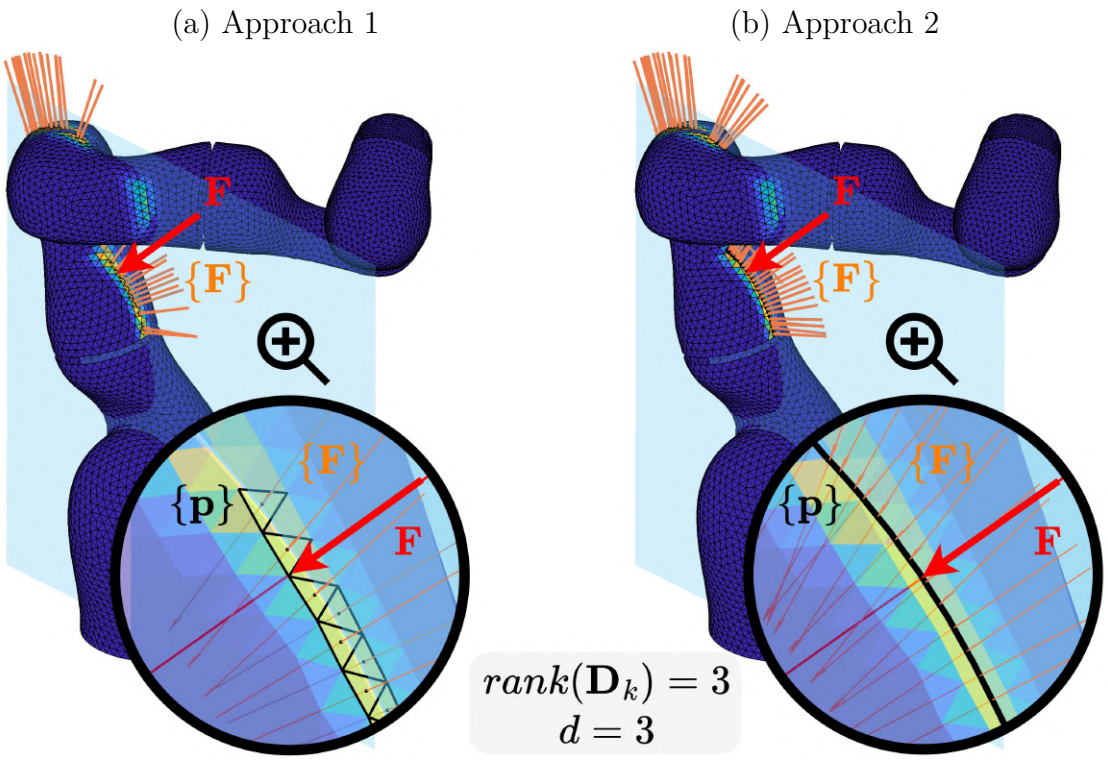


Figure 4.7: Ambiguity of the interaction parameters identification for case $d = 3$. Approach 1 generates a large number of combinations $\{\mathbf{F}\}$, $\{\mathbf{p}\}$ with the force application points located at centers of triangular mesh elements, which approximate the intersection of the robot link surface and the plane of potential force action lines. Approach 2 provides a set of $\{\mathbf{F}\}$, $\{\mathbf{p}\}$ that correspond to their exact intersection including the "real" solution. The similarity with the case $d = 2$ is caused by the special kinematic of robot KUKA iiwa but force action lines do not form a pencil.

possible force application points $\{\mathbf{p}\}$. So, here the force vectors $\{\mathbf{F}\}$ and corresponding action lines computed for any $\mathbf{p} \in \{\mathbf{p}\}$ are intersecting at the pencil vertex shown with blue $*$ symbol in the figure. Hence, in the case of $d = 2$ both approaches yield certain ambiguity that is obviously higher compared to the case $d = 1$.

Finally, for the extremely hard case with $d = 3$ presented in Fig. 4.7 the ambiguity is significantly higher than for $d = 2$. At first sight, this case looks similar to the previous one because the geometry of the potential force application points $\{\mathbf{p}\}$ is the same for both approaches. However, here the potential force vectors $\{\mathbf{F}\}$ possess higher ambiguity and can not be described by the pencil of straight lines. It is worth mentioning that such geometry is valid for robots with special kinematics only, similar to KUKA iiwa. Hence, in the case of $d = 3$ both approaches yield very high ambiguity that must be resolved later at the stage of the human-robot interaction handling.

As follows from the above-presented simulation study, the second approach overcomes the first one for both, accuracy and real-time performance. Nevertheless, there

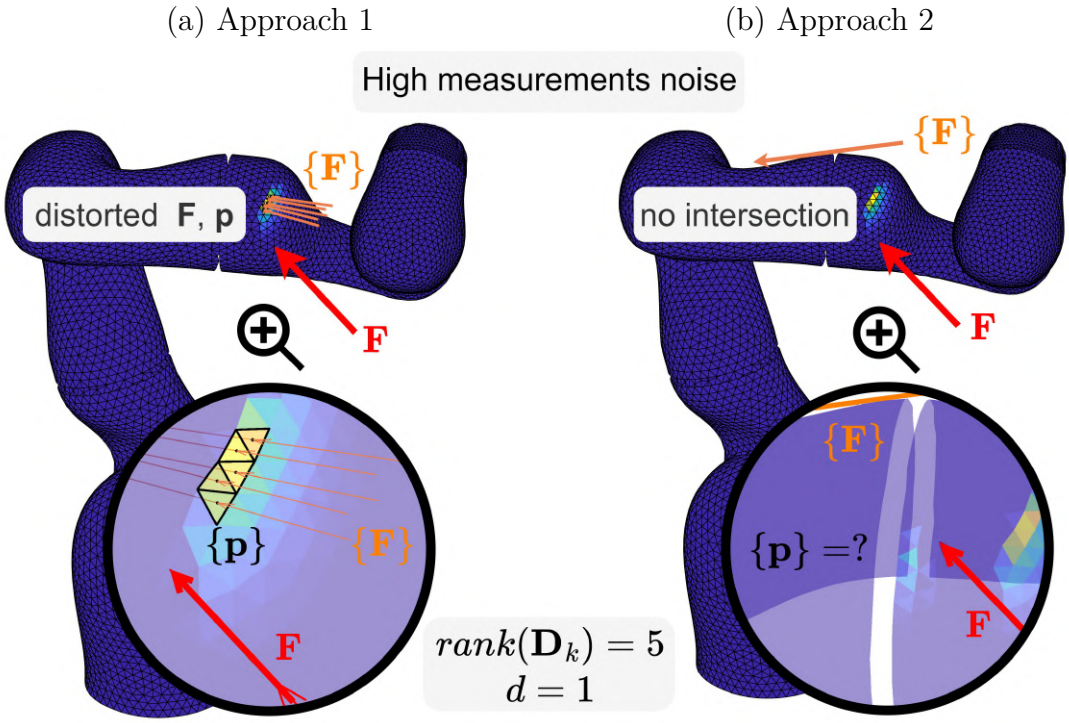


Figure 4.8: Measurements noise influence on the identification process. Approach 1 generates distorted solutions $\{\mathbf{F}\}$, $\{\mathbf{p}\}$. Approach 2 does not provide any solution, because there is no intersection of the force action line and the robot triangular mesh.

is another important feature that makes the first approach quite useful in real-life applications. In particular, the presented simulation study was conducted without taking into account the measurements noise, which obviously exists in practice. It is clear that for Approach 2, the torque measurement errors may disturb the force action line in such a way that the intersections with the triangular mesh elements $\boldsymbol{\Delta}$ disappear completely, so the solutions set may be even empty. Similar behavior of the identification algorithm was also observed for the 2D case presented at the end of Chapter 3. In some related works [89], this difficulty was also observed and overcome by redefining the force application point $\mathbf{p}$ as the closest point on the robot link surface. Nevertheless, such a method may violate the force direction constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. In contrast, Approach 1 always guarantees the satisfaction of the geometric constraints, providing a non-exact but practically acceptable solution, where the force $\mathbf{F}$ direction is inside of the friction cone and the force application point $\mathbf{p}$ is on the robot surface. A graphical illustration of the measurement noise influence on the identification process is shown in Fig. 4.8. In this example, Approach 1 generates some distorted solutions with deviated force vectors $\{\mathbf{F}\}$ and their application points $\{\mathbf{p}\}$. The deviation degree depends obviously on the measurement noise level. In contrast, Approach 2 does not provide any solution at all, because here there is no intersection of the force action line and the

robot triangular mesh. Hence, from the above-presented simulation study, which was conducted without any measurements noise, the robustness of these two approaches remains unclear.

Summarizing Subsection 4.3, dealing with the interaction parameters identification for singular cases, it is worth mentioning that both developed techniques (referred above as Approach 1 and 2) have their application areas. In contrast to the non-singular cases, they provide sets of solutions for potential interaction force vectors $\{\mathbf{F}\}$ and their application points $\{\mathbf{p}\}$. It was also shown that Approach 2 overcomes Approach 1 in terms of both accuracy and runtime if the measurements noise is negligible. However, the robustness of Approach 2 with respect to the torque measurement noise is questionable and should be analyzed in more detail, which is in the focus of the next section.

4.4 Robustness of interaction parameters identification technique

As was shown in the previous subsection, the joint torque measurement noise may essentially influence the identification results. In particular, the noise-corrupted torque measurements can lead to wrong estimation of the force action line, which can even not intersect the robot surfaces. The latter occurs for both non-singular and singular cases described above since in both cases the interaction parameters identification algorithms start with the estimation of the force action line. For the non-singular cases, this problem was reported in a number of works [29], [68], [70], [76], [89] but none of them provides the resolution technique for this problem. Graphically, the noise influence issue is presented in Fig. 4.9, where a set of force action lines corresponding to different noise samples is shown in addition to the "true" one. It is clear that such disturbances in torque measurements may cause essential estimation errors, which should be properly analyzed and evaluated.

As follows from the relevant literature study, there are still a number of open questions related to identification accuracy. In particular, to our knowledge, the problem of measurement noise influence on the identification accuracy was not directly addressed in existing works yet. For these reasons, this section focuses on the accuracy analysis of the conventional identification approach for the non-singular case, as well as its robustness with respect to the measurement noise. It also proposes further enhancements allowing to increase the identification accuracy and reduce the noise sensitivity for both, non-singular and singular cases.

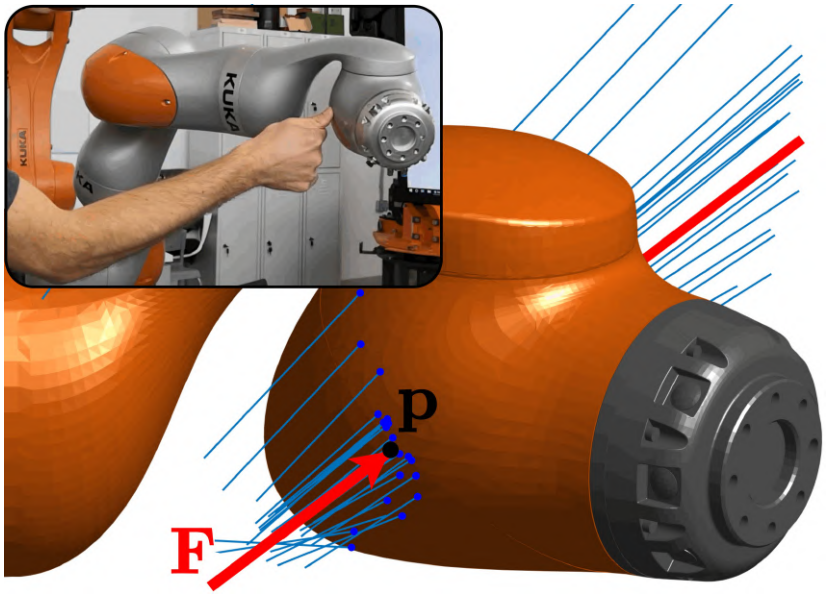


Figure 4.9: Influence of the torque measurement noise on the identification accuracy of the interaction force **F** and its application point **p**. The true interaction force is shown in red, while the estimated force action lines disturbed by the noise are shown in blue.

4.4.1 Measurement noise influence on identification accuracy

Let us first consider *non-singular case*, ensuring that the rank of the corresponding extended system is full, i.e. $rank(\mathbf{J}_k^w) = 6$. The conventional algorithm for non-singular cases presented in Section 4.2 is simple and quite fast but does not directly take into account the statistical properties of the torque measurement errors, which may differ from joint to joint. In fact, all its basic expressions are obtained under the zero-noise assumption. On the other hand, because of the pseudoinverse in (4.4) and (4.9), this algorithm should produce a statistically optimal result in the case when the measurement errors satisfy the usual i.i.d. assumption (independent and identically distributed). Let us evaluate the robustness of this algorithm with respect to the measurement noise and propose some enhancements. In this study, we implicitly assume that the standard deviations of the torque measurement errors are not identical.

The above-mentioned noise property can be explained by the fact that robot manufacturers usually use different motors and sensors for different joints in order to satisfy robot torque, weight, and speed requirements. To prove that the torque measurement errors are not identical in practice, a series of experiments were conducted. In these experiments, the time-series of the torque measurements were analyzed, which included data from 20 different static robot configurations with and without interaction. The estimated standard deviations for different robot joints are presented in Table 4.8 that confirms the non-identical assumption.

Table 4.8: Torque sensor measurement errors: experimental results for standard deviation [Nm]

σ_τ^1	σ_τ^2	σ_τ^3	σ_τ^4	σ_τ^5	σ_τ^6	σ_τ^7
0.8	0.8	0.5	0.5	0.1	0.1	0.1

Let us obtain first the covariance matrix for the wrench estimates $\hat{\mathbf{W}} = [\hat{\mathbf{F}}, \hat{\mathbf{M}}]^T$ from expression (4.4). To do so, apply the linear regression analysis, which is a statistical approach that allows to obtain the linear relation between model input and output variables. For the considered problem (4.4), the inputs are the torque measurements $\boldsymbol{\tau}$ and output is the interaction wrench $\mathbf{W}$

$$\boldsymbol{\tau} = \mathbf{J}_k^w(\mathbf{q}, \mathbf{p})^T \cdot \mathbf{W} \tag{4.27}$$

It is clear that in practice, the measurements often include errors because of the inaccuracy of measurement devices. For this reason, the original model should be rewritten as

$$\tilde{\boldsymbol{\tau}} = \mathbf{J}_k^w(\mathbf{q}, \mathbf{p})^T \cdot \mathbf{W} + \boldsymbol{\varepsilon} \tag{4.28}$$

where ε_i is the measurement error that is usually assumed to be the unbiased i.i.d. (independent and identically distributed) a random variable with zero expectation and the standard deviation σ. Because of the measurement errors influence, the system (4.28) (which should be overdetermined to reduce the noise impact) cannot be solved exactly. For this reason, the least square approach is applied, which minimizes the sum of squared residuals. Using this method, the parameter estimation can be expressed as

$$\hat{\mathbf{W}} = \left(\mathbf{J}_k^w \cdot (\mathbf{J}_k^w)^T\right)^{-1} \cdot (\mathbf{J}_k^w \cdot \tilde{\boldsymbol{\tau}}) \tag{4.29}$$

It can be proved that this expression provides us with an unbiased estimate of the desired parameters, i.e. $E\left(\hat{\mathbf{W}}\right) = \mathbf{W}$ but for any particular measurements set the estimated parameters can differ from the "true" value $\mathbf{W}$. In practice, the degree of this variation is evaluated by the covariance matrix, which in the case of the above-adapted assumptions can be expressed as

$$\operatorname{cov}\left(\hat{\mathbf{W}}\right) = \sigma^2\left(\mathbf{J}_k^w \cdot (\mathbf{J}_k^w)^T\right)^{-1} \tag{4.30}$$

However, in practice, the measurement noise varies for different joint torque sensors as was shown in Table 4.8, so the previous expression should be revised. It can be

proved that in the general case the covariance matrix is expressed as

$$\operatorname{cov}\left(\hat{\mathbf{W}}\right) = \left(\mathbf{J}_k^w \cdot (\mathbf{J}_k^w)^T\right)^{-1} \left(\mathbf{J}_k^w \cdot \boldsymbol{\Sigma}_\tau^2 \cdot (\mathbf{J}_k^w)^T\right) \left(\mathbf{J}_k^w \cdot (\mathbf{J}_k^w)^T\right)^{-1} \tag{4.31}$$

The same expression for the desired covariance can be derived in a compact form

$$\operatorname{cov}(\hat{\mathbf{W}}) = \left[\mathbf{J}_k^w\left(\mathbf{q}^{(k)}, \mathbf{p}\right)^T\right]^{\#} \cdot \boldsymbol{\Sigma}_\tau^2 \cdot \left[\mathbf{J}_k^w\left(\mathbf{q}^{(k)}, \mathbf{p}\right)^T\right]^{\#T} \tag{4.32}$$

which includes the diagonal matrix $\boldsymbol{\Sigma}_\tau = diag\left[\sigma_\tau^1, \sigma_\tau^2 ... \sigma_\tau^k\right]$ composed of the standard deviations of the torque measurement errors σ_τ^i. It is worth mentioning that in the case of identical measurement errors $\boldsymbol{\Sigma}_\tau = \sigma\mathbf{I}$, this expression is simplified down to (4.30), which provides (4.4) with a property of statistically optimal estimator.

It is clear that the obtained matrix

$$\operatorname{cov}(\hat{\mathbf{W}}) = \begin{bmatrix} \operatorname{cov}(\hat{\mathbf{F}}) & * \\ * & \operatorname{cov}(\hat{\mathbf{M}}) \end{bmatrix} \tag{4.33}$$

allows us also to find the desired covariance matrices $\operatorname{cov}(\hat{\mathbf{F}})$, $\operatorname{cov}(\hat{\mathbf{M}})$ for the force and torque vectors separately.

Further, to find the covariance matrix $\operatorname{cov}(\hat{\mathbf{p}})$ for the force application point, a similar technique can be applied to the system (4.12). Relevant derivation yields the following expression

$$\operatorname{cov}(\hat{\mathbf{p}}) = \begin{bmatrix} \mathbf{S}(\hat{\mathbf{F}}) \\ \mathbf{n}^{k,j} \end{bmatrix}^{T\,\#} \cdot \boldsymbol{\Sigma}_p^2 \cdot \begin{bmatrix} \mathbf{S}(\hat{\mathbf{F}}) \\ \mathbf{n}^{k,j} \end{bmatrix}^{\#T} \tag{4.34}$$

where the quasi-diagonal matrix

$$\boldsymbol{\Sigma}_p^2 = \begin{bmatrix} \operatorname{cov}(\hat{\mathbf{M}}) & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}_{4\times 4} \tag{4.35}$$

incorporates the measurement noise statistical properties via the 3×3 matrix $\operatorname{cov}(\hat{\mathbf{M}})$ and also includes some zero elements because the geometric constraint in (4.12) does not depend on measurement errors.

To verify the above expressions for $\operatorname{cov}(\hat{\mathbf{F}})$ and $\operatorname{cov}(\hat{\mathbf{p}})$, a relevant simulation study was carried out using the robot KUKA LBR iiwa 14 model. In this study, three typical manipulator configurations were considered for the force amplitude 100 N and 10^4 random samples were generated using the experimental data from Table 4.8. The obtained results presented in Table 4.9 confirm the validity of the covariance matrix expressions

Table 4.9: Estimation accuracy of conventional technique for non-singular cases

	Analytical		**Simulated**	
	$\boldsymbol{\sigma}_F$ $[N]$	$\boldsymbol{\sigma}_p$ $[mm]$	$\boldsymbol{\sigma}_F$ $[N]$	$\boldsymbol{\sigma}_p$ $[mm]$
Conf. 1	3.92	11.93	3.93	11.88
Conf. 2	9.32	25.11	9.21	25.45
Conf. 3	8.96	24.11	8.99	24.15

* For each configuration 10 000 random samples were evaluated.

and also show that for real-life noise parameters, the expected estimation errors are not negligible for both interaction force and its application point. In particular, for the force vector, the identification errors evaluated using the $3{\cdot}\sigma_F$ rule was up to 30% of its amplitude, while for the force application point, the $3{\cdot}\sigma_p$ error was up to 75 mm. For comparison, the length of the 6th link where the interaction force is typically applied is equal to 120 mm. The latter motivates us for the accuracy enhancement of the conventional identification technique, which is the focus of the following subsection.

4.4.2 Enhancement of identification technique accuracy and robustness

To improve the algorithm accuracy and its robustness for *non-singular cases* with respect to the measurement noise, let us introduce an additional step based on the simplified version of the original static equilibrium system (1.11), which is linearized in the neighborhood of the solution $\hat{\mathbf{F}}, \hat{\mathbf{p}}$ provided by the conventional method. It can be easily proved that the desired linearized system can be presented as following

$$\begin{bmatrix} \mathbf{J}_k(\mathbf{q}, \hat{\mathbf{p}}) & \mathbf{H}_k(\mathbf{q}, \hat{\mathbf{F}}) \end{bmatrix} \cdot \begin{bmatrix} \delta\mathbf{F} \\ \delta\mathbf{p} \end{bmatrix} = \delta\boldsymbol{\tau} \tag{4.36}$$

where $\delta\mathbf{F}$, $\delta\mathbf{p}$ are correction vectors for the interaction force and its application point respectively, $\delta\boldsymbol{\tau}$ is the vector of torque residuals, $\mathbf{J}_k$ is the manipulator Jacobian matrix defined above by (1.13), and the matrix $\mathbf{H}_k$ is expressed as

$$\mathbf{H}_k(\mathbf{q}, \mathbf{F}) = \begin{bmatrix} \mathbf{F} \times \mathbf{e}_1 & \dots & \mathbf{F} \times \mathbf{e}_k \end{bmatrix} \tag{4.37}$$

It is worth mentioning that the above presented linear system (4.36) is similar to the one used for the analysis of singular cases (4.16). However, there are two essential differences. The first one is the solution around which the system of static equilibrium

Table 4.10: Estimation accuracy of enhanced technique for non-singular cases

	Analytical		**Simulated**	
	$\boldsymbol{\sigma}_F$ $[N]$	$\boldsymbol{\sigma}_p$ $[mm]$	$\boldsymbol{\sigma}_F$ $[N]$	$\boldsymbol{\sigma}_p$ $[mm]$
Conf. 1	2.86	3.72	2.87	3.74
Conf. 2	5.51	4.49	5.48	4.50
Conf. 3	8.07	6.45	8.09	6.51

* For each configuration 10^4 random samples were evaluated.

equations is linearized. In singular case analysis from the previous section, it was implicitly assumed that the "true" solution is known and the system was linearized around it. Here, the estimation of the interaction parameters $\hat{\mathbf{F}}, \hat{\mathbf{p}}$ is used, which was obtained by Algorithm 9. The second one is the right-hand side of the linearized system, it is clear that the torque residuals are not equal to zero in the presence of measurements noise.

Further, to take into account that the statistical properties of the torque measurement errors are non-identical and differ from joint to joint (see Table 4.8), let us apply the weighted least squares technique. As it is known, for uncorrelated measurement noise the best estimate is achieved if the weights are equal to the reciprocal of the measurement variances $1/(\sigma_\tau^i)^2, \quad i = 1, ...k$. This leads to the classical quadratic optimization problem with linear equality constraint

$$\left[\mathbf{n}^{k,j}\right]^T \cdot \delta\mathbf{p} = 0 \tag{4.38}$$

that originates from the second line of (4.12) representing the geometric constraint (1.14). The solution can be found using the Lagrangian

$$L(\delta\mathbf{F}, \delta\mathbf{p}, \lambda) = \left\| \delta\boldsymbol{\tau} - \begin{bmatrix}\mathbf{J}_k & \mathbf{H}_k\end{bmatrix} \cdot \begin{bmatrix}\delta\mathbf{F} \\ \delta\mathbf{p}\end{bmatrix} \right\|^2 + 2\lambda \cdot \begin{bmatrix}\mathbf{0} \\ \mathbf{n}^{k,j}\end{bmatrix}^T \cdot \begin{bmatrix}\delta\mathbf{F} \\ \delta\mathbf{p}\end{bmatrix} \tag{4.39}$$

By including the measurements noise, the desired corrections $\delta\mathbf{F}$, $\delta\mathbf{p}$ can be obtained from the KKT optimality conditions, which in our case are presented as

$$\begin{bmatrix}\mathbf{A}_{\boldsymbol{\Sigma}} & \mathbf{n}^{k,j} \\ \mathbf{n}^{k,j^T} & 0\end{bmatrix} \cdot \begin{bmatrix}\begin{bmatrix}\delta\mathbf{F} \\ \delta\mathbf{p}\end{bmatrix} \\ \lambda\end{bmatrix} = \begin{bmatrix}\mathbf{B}_{\boldsymbol{\Sigma}} \\ 0\end{bmatrix} \cdot \delta\boldsymbol{\tau} \tag{4.40}$$

Algorithm 13: Enhanced identification method for non-singular cases

Function `EnhancedMethod(q, τ̃, k, Robot)`:

Input : $\mathbf{q}$ is joints positions, $\tilde{\boldsymbol{\tau}}$ is torque measurements, k is link index, **Robot** is a structure with robot geometric and kinematic parameters.

Output: interaction parameters $\hat{\mathbf{F}}, \hat{\mathbf{p}}$.

$[\hat{\mathbf{p}}, \hat{\mathbf{F}}] = \text{BasicMethod}(\mathbf{q}, \tilde{\boldsymbol{\tau}}, k)$;

while $||\mathbf{p}_{prev} - \hat{\mathbf{p}}|| > \varepsilon_p$ **do**

$\delta\boldsymbol{\tau} = \tilde{\boldsymbol{\tau}} - \mathbf{J}_k(\mathbf{q}, \hat{\mathbf{p}})^T \cdot \hat{\mathbf{F}}$;

$\mathbf{A} = \begin{bmatrix} \mathbf{J}_k(\mathbf{q}, \hat{\mathbf{p}})^T & \mathbf{H}_k(\mathbf{q}, \hat{\mathbf{F}})^T \end{bmatrix}$;

$\mathbf{C} = \begin{bmatrix} \mathbf{0} & \mathbf{n}^{k,j} \end{bmatrix}$;

$d = \mathbf{n}^{k,j} \cdot (\mathbf{s}\mathbf{v}^{k,j} - \hat{\mathbf{p}})$;

$$\begin{bmatrix} \delta\mathbf{F} \\ \delta\mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{A}^T \cdot \boldsymbol{\Sigma}_\tau^{-2} \cdot \mathbf{A} & \mathbf{C}^T \\ \mathbf{C} & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{A}^T \cdot \boldsymbol{\Sigma}_\tau^{-2} \cdot \delta\boldsymbol{\tau} \\ d \end{bmatrix};$$

$\hat{\mathbf{F}} = \hat{\mathbf{F}} + \delta\mathbf{F}$;

$\hat{\mathbf{p}} = \hat{\mathbf{p}} + \delta\mathbf{p}$;

where λ is scalar Lagrangian multiplier and

$$\begin{aligned} \mathbf{A}_{\boldsymbol{\Sigma}} &= \mathbf{J}_k \boldsymbol{\Sigma}_\tau^{-2} \mathbf{J}_k^T + \mathbf{H}_k \boldsymbol{\Sigma}_\tau^{-2} \mathbf{H}_k^T \\ \mathbf{B}_{\boldsymbol{\Sigma}} &= \mathbf{J}_k \boldsymbol{\Sigma}_\tau^{-2} + \mathbf{H}_k \boldsymbol{\Sigma}_\tau^{-2} \end{aligned} \tag{4.41}$$

with $\boldsymbol{\Sigma}_\tau^{-2} = diag[1/(\sigma_\tau^1)^2, ...1/(\sigma_\tau^k)^2]$. It is clear that such linearization can be applied several times, yielding the desired solution with minimal estimation error.

In more detail, the identification technique is presented in Algorithm 13. It can be also proved that the corresponding covariance matrix can be expressed as

$$\text{cov}(\hat{\mathbf{F}}, \hat{\mathbf{p}}, \lambda) = \mathbf{A}_{\Sigma\Omega}^{-1} \cdot \begin{bmatrix} \boldsymbol{\Sigma}_\tau^2 & 0 \\ 0 & 0 \end{bmatrix} \cdot \mathbf{A}_{\Sigma\Omega}^{-T} \tag{4.42}$$

where

$$\mathbf{A}_{\Sigma\Omega} = \begin{bmatrix} \mathbf{A}_\Sigma & \mathbf{n}^{k,j} \\ \mathbf{n}^{k,j^T} & 0 \end{bmatrix} \tag{4.43}$$

Similar to the previous section, the above expression for the covariance matrix was verified via the stochastic simulation using the KUKA LBR iiwa 14 robot model. The

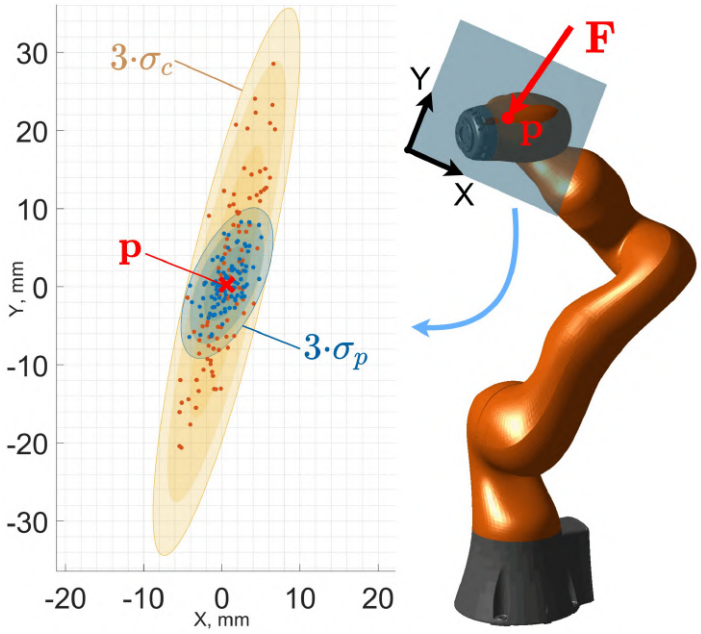


Figure 4.10: Accuracy of conventional and enhanced methods for non-singular case: estimation errors of the force application point $\mathbf{p}$ presented via 3σ-dispersion ellipses.

Table 4.11: Comparison of conventional and enhanced techniques for non-singular cases

	Conventional		**Proposed**		**Improvement**	
	$\boldsymbol{\sigma}_F$ $[N]$	$\boldsymbol{\sigma}_p$ $[mm]$	$\boldsymbol{\sigma}_F$ $[N]$	$\boldsymbol{\sigma}_p$ $[mm]$	k_F	k_p
Conf. 1	3.92	11.88	2.86	3.74	1.37	3.17
Conf. 2	9.20	25.45	5.48	4.50	1.68	5.65
Conf. 3	8.99	24.16	8.11	6.51	1.11	3.71

*For each configuration 10^4 random samples were evaluated.

obtained results are presented in Table 4.10 that demonstrates good agreement of the analytical and simulated variances, which proves the correctness of the linearized model. The obtained values are also essentially lower than the previous ones presented in Table 4.9. In more detail, advantages of the proposed technique are shown in Fig. 4.10 and Table 4.11, where the accuracy improvement up to 1.7 times for the force $\hat{\mathbf{F}}$ and up to 5.7 times for its application point $\hat{\mathbf{p}}$ was achieved. It should be mentioned here that algorithm converges in less than 5 iterations, which introduces a very small additional computational cost to the conventional approach.

Let us now consider the *singular cases*, arising when the number of the independent equations in the static equilibrium system is not sufficient. The accuracy estimation using the above-mentioned technique for such cases is complicated since the rank of the corresponding matrices in (4.42) is not full. The latter is connected to the multi-

Table 4.12: Simulation results of Approach 1 with measurement noise

Param.	Metrics	k-index					avg
		7	**6**	**5**	**4**	**3**	
$\|\|\Delta\mathbf{F}\|\|$ $[N]$	mean	2.49	3.22	5.21	19.4	18.9	10.0
	std	3.44	3.67	8.69	72.6	21.2	35.5
	max	41.5	32.2	98.4	969.8	140.1	969.8
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	2.21	2.65	5.86	15.0	19.5	9.17
	std	5.12	2.69	15.1	38.6	18.5	21.8
	max	59.8	20.7	180.9	397.0	138.5	397.0
$\mathbf{T}_c$ $[ms]$	mean	49.9	99.5	143.8	185.1	232.5	144.2
	std	0.66	1.20	15.4	22.6	23.8	64.6
	min	48.7	96.9	96.8	144.9	193.6	48.7
	max	52.2	106.2	153.6	206.2	253.1	253.1
$\mathbf{f}_c$ $[Hz]$	mean	20.1	10.0	6.95	5.40	4.30	6.93
fail %		0	0	0	0	0	0

Table 4.13: Simulation results of Approach 2 with measurement noise

Param.	Metrics	k-index					avg
		7	**6**	**5**	**4**	**3**	
$\|\|\Delta\mathbf{F}\|\|$ $[N]$	mean	3.64	8.87	74.1	11.7	2.61	19.1
	std	4.62	28.6	442.9	14.9	3.68	190.3
	max	42.3	362.6	5233	80.2	38.1	5233
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	4.51	8.64	33.2	12.0	6.59	12.8
	std	4.35	9.21	54.5	37.8	9.59	31.5
	max	29.4	73.9	297.2	406.6	102.5	406.6
$\mathbf{T}_c$ $[ms]$	mean	0.30	0.57	0.91	2.06	3.13	1.51
	std	0.03	0.08	0.37	0.43	0.46	1.12
	min	0.24	0.44	0.46	1.11	2.11	0.24
	max	0.45	1.25	3.46	5.29	5.30	5.30
$\mathbf{f}_c$ $[Hz]$	mean	3284	1748	1088	484	318	664
fail %		24	8.33	19.4	2.41	0	10.4

Note: $\Delta\mathbf{F}$ - estimation error of the interaction force $\|\mathbf{F}\| = 100\ N$;
$\Delta\mathbf{p}$ - estimation error of the force application point;
$\mathbf{T}_c$, $\mathbf{f}_c$ - time performance of the estimation algorithm;
fail - no intersection cases, resulting in empty solution set.

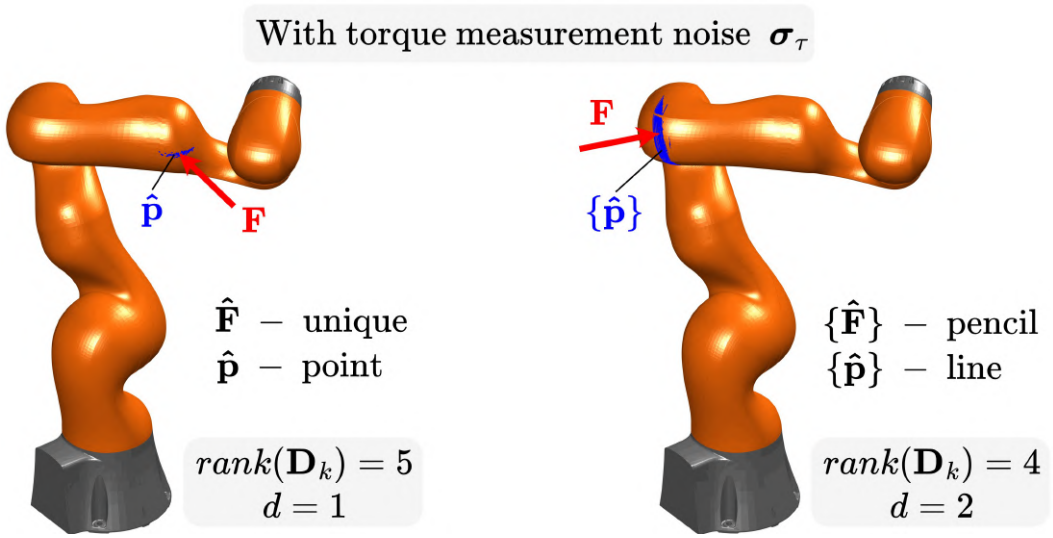


Figure 4.11: Accuracy of the enhanced identification technique for singular cases. For each configuration, 100 random samples were evaluated, the standard deviation of torque measurements noise is set to $\boldsymbol{\sigma}_\tau$ obtained from the real robot. Here, the applied interaction force $\mathbf{F}$ is shown with a red arrow, the estimated force application points $\{\mathbf{p}\}$ are shown with blue color.

ple possible solutions of the original system of static equilibrium equations described in the previous sections of this chapter. So, the direct estimation of the covariance matrix $\text{cov}(\hat{\mathbf{F}}, \hat{\mathbf{p}})$ is possible only for the singular cases when the rank deficiency of the linearized system is $d = 1$ and the interaction force applied to fifth or following robot links. The graphical example of two singular cases with the rank deficiency $d = 1$ and $d = 2$ is presented in Fig. 4.11. Here, the interaction force application point $\hat{\mathbf{p}}$ is estimated several times using torque measurements $\tilde{\boldsymbol{\tau}}$ with noise. The case with $d = 1$ provides a unique solution and is equivalent to the non-singular case, so (4.42) can be used to estimate the covariance matrix. In contrast, the case with $d = 2$ provides a non-unique solution, which corresponds to the line for possible force application points. It is clear that (4.42) cannot be used here, so some stochastic simulations should be utilized in order to evaluate the accuracy of the algorithms.

In order to evaluate the accuracy of the interaction identification techniques for the singular cases, stochastic simulations were carried out. Two methods (Approach 1 and Approach 2) from the previous section were used as the identification techniques. In this study, 10^3 random manipulator configurations were considered, the interaction forces were applied to the random feasible point on the robot link surface with an amplitude of 100 N, the generated torque measurements were corrupted according to the experimental data from Table 4.8. The results are presented in Tables 4.12 and 4.13 for Approaches 1 and 2 correspondingly. It should be mentioned that this simulation

study is similar to the one provided in Tables 4.5 and 4.6, except for the additional measurement noise.

As follows from the presented simulation study, the accuracy of both approaches decreased with the addition of the measurement noise. Compared to the same study without noise (Tables 4.5 and 4.6), Approach 1 now shows better accuracy than Approach 2. Moreover, as was discussed at the end of the previous section, Approach 2 has essential problems with its robustness. In particular, the table last row ("fail") shows the percentage of cases, where the method failed to provide any solutions at all. The latter is caused when the force action line fails to intersect the robot triangular mesh. Consequently, it is not possible to provide a solution in every tenth sample on average, which makes Approach 2 hard to apply in practice.

To address the accuracy issues of the original Approaches 1 and 2, let us introduce some enhancements based on the presented analysis. The main problem of Approach 1, besides decreased accuracy in the presence of noise, is its real-time capabilities. Both of these issues can be solved simultaneously by replacing the optimization problem (see line 11) with weighted least squares, where the weights are chosen as reciprocal of measurement noise standard deviation $\boldsymbol{\sigma}_\tau$. It should be mentioned that line 11 of Algorithm 10 is the most time-consuming step since it requires an iterative technique for each of the triangular faces representing the robot link surface. The replacement of this constrained optimization with the regular weighted pseudoinverse allows to improve both the accuracy and real-time performance of the algorithm. In addition, the execution time can be further reduced by vectorizing the main loop (see line 9 in Algorithm 10) and its execution in parallel, which allows to calculate the interaction force for all triangles at the same time. It should be noted that the interaction force $\mathbf{F}$ is unique for given force application point $\mathbf{p}$ for the $k > 3$ only, so in case of $k = 3$ the force direction is assumed to be directed as triangular face normal vector. Although it obviously reduces the accuracy of Approach 1 for the third link, it is not so important as the improvement in the algorithm execution time that is achievable because of the pseudoinverse.

The main problem of Approach 2 is the presence of empty solutions in some hard cases. As was previously mentioned such cases arise when the noise in torque measurements pushes the estimated force action line out of the robot surface and results in no intersection case (see "fail" in Table 4.13). Most of them correspond to the $7th$ and $5th$ robot links, where the physical size of $7th$ link is relatively small and it is easy to miss. Concerning Algorithm 11 itself, it can be noted that it is similar to Algorithm 13, successfully used for accuracy enhancement of non-singular cases, both of them use Gauss-Newton algorithm based on the linearized static equilibrium equations. Thus,

similar enhancements can be applied to Algorithm 11 by taking into account the measurement noise properties as weights $\mathbf{\Sigma}_\tau$, which allows to reduce the estimation error as well as the number of no-intersection cases. This modification allows to improve the algorithm accuracy in general but it does not significantly reduce "fail" cases for the $5th$ link. After the relevant analysis, it becomes clear that the latter is connected to the wrong estimation of the matrix $\mathbf{D}_k$ rank deficiency d. Generally, the rank deficiency is equal to $d = 1$, when the interaction force $\mathbf{F}$ is applied to the fifth link $k = 5$, as was shown in Fig. 4.1. However, in some particular cases and robot configurations, the matrix $\mathbf{D}_k$ possess a high condition number making this problem more singular. This issue can be overcome by choosing a different type of solution when the condition number is greater than some tolerance value. The latter allows to consider the interaction in the $5th$ link with $d = 2$, which closer describes the interaction for these hard singular cases.

To test the enhanced Approaches 1 and 2, the simulation study was conducted, and the relevant results are presented in Tables 4.14 and 4.15. Here, the same experimental setup and the same input data were used as in the previous study Tables 4.12 and 4.13. The developed enhancement in the case of Approach 1 allowed to significantly reduce the computation time by using the weighted pseudoinverse and parallelization of computations. The minor disadvantage of the enhanced technique is the reduced accuracy of estimation in $3rd$ link, where the interaction force is assumed to be directed along the surface normal vector. For Approach 2, the enhancements allowed to essentially reduce the number of no-intersection cases, especially for the $7th$ and $5th$ links. The latter becomes possible by taking into account the torque measurement noise properties. The comparison of the main parameters of these two techniques is presented in Table 4.16. So, it is clear that the proposed modifications reduced the average computation time $\mathbf{T}_c$ for Approach 1 up to 60 times making it comparable to the time performance of Algorithm 2. The percentage of "fail" cases (characterized by no-intersection with robot mesh) for Approach 2 was reduced up to 17 times. In addition, the overall accuracy of both enhanced approaches was also improved, making both of the approaches attractive for real-life applications.

It is worth mentioning that there are a few ways of how the "fail" cases can be handled in order to provide an approximate solution. The typical example of a no-intersection case is presented in Fig. 4.12a, where the estimated interaction force $\hat{\mathbf{F}}$ action line is outside of the robot model. In some related works [89], this problem was also observed and a very simple solution was proposed to solve it. The solution is to select the closest point on the robot link surface as the force application point $\hat{\mathbf{p}}$ to which the estimated force $\hat{\mathbf{F}}$ is applied as shown in Fig. 4.12b. It is clear that

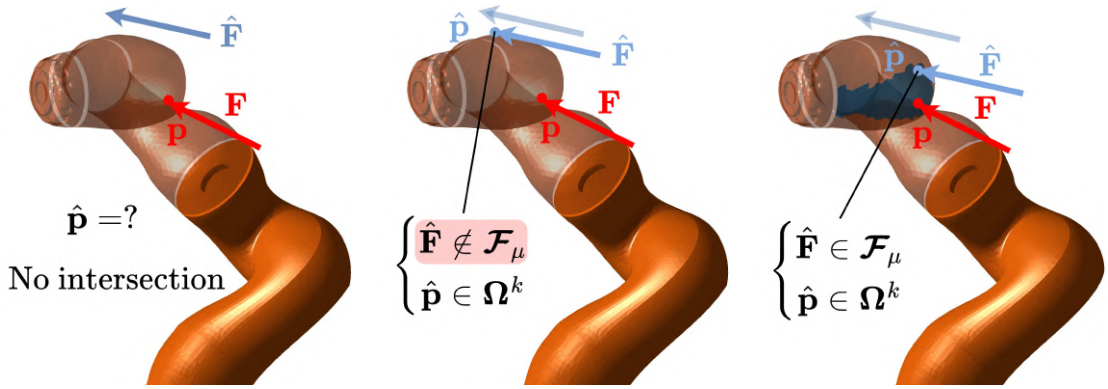
Table 4.14: Simulation results on Enhanced Approach 1 with measurement noise

Param.	Metrics	k-index					avg
		7	**6**	**5**	**4**	**3**	
$\|\|\Delta\mathbf{F}\|\|$ $[N]$	mean	2.56	3.55	4.52	12.2	43.9	13.5
	std	3.58	5.56	5.07	15.6	22.0	20.2
	max	41.6	64.7	39.7	87.9	99.1	99.1
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	2.31	2.94	5.21	10.6	10.6	6.42
	std	5.14	3.61	9.25	15.2	21.2	13.2
	max	57.6	34.24	82.0	149.7	183.2	183.2
$\mathbf{T}_c$ $[ms]$	mean	0.87	1.68	2.40	3.06	3.47	2.33
	std	0.04	0.08	0.25	0.36	0.39	0.95
	min	0.81	1.57	1.64	2.39	2.81	0.81
	max	1.03	1.94	2.81	3.56	4.51	4.51
$\mathbf{f}_c$ $[Hz]$	mean	1139	593	415	326	288	428
fail %		0	0	0	0	0	0

Table 4.15: Simulation results on Enhanced Approach 2 with measurement noise

Param.	Metrics	k-index					avg
		7	**6**	**5**	**4**	**3**	
$\|\|\Delta\mathbf{F}\|\|$ $[N]$	mean	2.03	2.51	3.34	5.13	2.12	3.05
	std	1.32	1.92	3.68	6.03	2.63	3.74
	max	9.29	14.7	41.0	45.9	27.7	45.9
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	1.89	2.57	5.20	7.89	6.10	4.79
	std	2.03	3.94	14.1	14.3	6.81	10.0
	max	20.5	40.4	136.7	129.1	43.1	136.7
$\mathbf{T}_c$ $[ms]$	mean	0.34	0.77	1.23	2.58	3.87	1.76
	std	0.03	0.38	0.80	0.87	0.74	1.43
	min	0.26	0.54	0.44	1.11	2.82	0.26
	max	0.45	4.03	5.53	7.08	7.52	7.52
$\mathbf{f}_c$ $[Hz]$	mean	2941	1294	809	387	258	567
fail %		0.57	1.38	0	0.96	0	0.6

Note: $\Delta\mathbf{F}$ - estimation error of the interaction force $\|\mathbf{F}\| = 100\ N$;
$\Delta\mathbf{p}$ - estimation error of the force application point;
$\mathbf{T}_c$, $\mathbf{f}_c$ - time performance of the estimation algorithm;
fail - no intersection cases, resulting in empty solution set.



(a) No intersection (b) Closest solution (c) Closest feasible solution

Figure 4.12: Example of "fail" case and its approximate solution when the force action line do not intersect the robot mesh. Here, the applied interaction force $\mathbf{F}$ is shown with a red arrow and the estimated force $\{\mathbf{p}\}$ is shown with blue color.

Table 4.16: Comparison of Original and Enhanced Approaches 1 and 2

	mean $\|\|\Delta\mathbf{F}\|\|$	**mean** $\|\|\Delta\mathbf{p}\|\|$	**max** $\mathbf{T}_c$	**mean** $\mathbf{T}_c$	**fail** %
Original Approach 1	10.0	9.17	253.1	144.2	0
Original Approach 2	19.1	12.8	5.30	1.50	10.4
Enhanced Approach 1	13.5	6.42	4.51	2.33	0
Enhanced Approach 2	3.05	4.79	7.52	1.76	0.6

this method allows to satisfy the surface constraint $\mathbf{p} \in \Omega_k$ but may violate the force direction constraint $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$. Another solution is to select such force application point $\hat{\mathbf{p}}$ that satisfies both constraints and is closest to the initially estimated interaction force $\hat{\mathbf{F}}$ action line. The latter allows to obtain interaction parameters, which is physically possible and most likely to be closer to the "true" interaction force application point (see Fig. 4.12c). Thus, the no-intersection cases can be treated as singular ones that required additional processing, rather than just providing an empty solution set.

Summarizing this section, the performance of techniques for the interaction parameters identification in both singular and non-singular cases was analyzed in the presence of measurement noise. Based on obtained results, some relevant enhancements were developed in order to make these approaches usable in real-life applications. Nevertheless, the solution provided by any of these techniques is ambiguous, so the relevant resolution algorithm is required and will be the focus of the next section.

4.5 Selection of unique interaction parameters from solutions set

In this section, a practice-oriented heuristic approach is proposed for singularity resolution similar to one provided for the planar case in Section 3.4. The core idea of the developed technique is to use several previous estimates $(\hat{\mathbf{F}},\ \hat{\mathbf{p}})$ and to interpolate from them in the case of singularity. It is clear that this technique can be used to select only one solution from the set $(\{\hat{\mathbf{F}}\},\ \{\hat{\mathbf{p}}\})$ of possible ones.

4.5.1 Implementation of developed singularity resolution technique

As follows from previous sections, in some cases there is an infinite number of possible solutions for the interaction force $\hat{\mathbf{F}}$ and its application point $\hat{\mathbf{p}}$. The latter is usually caused by the rank deficiency $(d > 1)$ of the relevant system $\mathbf{D}_k$, linearized in the neighborhood of the solution. In practice, the reasons for that can be divided into three main categories: the kinematic singularity of robot configuration; specific force direction when it intersects the last joint axes; specific location of the force application point leading to the insufficient number of the static equilibrium equations. It is clear that for practical application, a single solution $(\mathbf{F},\ \mathbf{p})$ corresponds to the actual one. The latter motivates us to extend the developed singularity resolution technique for the more general 3D spatial case, allowing us to overcome the above-mentioned ambiguity problem.

To resolve the ambiguity of the considered identification problem several hypotheses from Section 3.4 are used here, which were modified for the 3D case. These hypotheses are presented below and are related to the ambiguity in computing the k-index, the force direction $\mathbf{F}$, and the force application point $\mathbf{p}$ as well as the utilization of the estimation history to resolve the singular cases.

Hypothesis 5. *If the human-robot interaction model does not allow to identify uniquely the link to which the interacting force is applied, then the closest to the end-effector link is selected from possible options, corresponding to the largest k-index from the obtained range $[k_{min},\ k_{max}]$.*

Justification. The priority to the furthest potential link with the index k_{max} can be supported by the fact that in real life a human operator usually interacts with the robot end-effector or its tool, which correspond to the largest values of k. In addition,

the workspace of the collaborative work cell is usually designed in such a way that the robot base is out of reach for a human, so the interaction probability is higher for the links that are located closer to the end-effector. □

Hypothesis 6. *If the human-robot interaction model does not allow to identify uniquely the force direction, then the force* $\mathbf{F}$ *that is directed closer to the surface normal is selected from possible options.*

Justification. The surface normal direction corresponds to the center of the friction cone constraint at the considered force application point. Since in practice, the torque measurements are usually corrupted by the noise, it allows us to make some estimations $\hat{\mathbf{F}}$ of the "true" interaction force $\mathbf{F}$. It is clear that by selecting the estimated force closer to the center of the friction cone it is less likely to violate this constraint and produce a physically possible solution. □

Hypothesis 7. *If the human-robot interaction model does not allow to identify uniquely the force application point* $\mathbf{p}$*, then the center point of the primitive element describing the solution on the robot surface is selected from possible options.*

Justification. In this work, it is assumed that the robot surface model is extracted from a high-resolution CAD presentation, which provides us with a large number of surface primitives of a rather small size. Depending on the identification approach, the solution can be represented as a set of triangles or a set of points/lines. For this reason, it is rational to reduce the computational expenses by analyzing their center point only. It is clear that such simplification does not influence significantly the identification accuracy. □

Hypothesis 8. *If previous estimations of the interaction parameters* $\{\mathbf{F}_t,\ \mathbf{p}_t \mid t < t_c\}$ *are available and the current robot configuration does not allow to identify uniquely the force and its application point, then the desired parameters* $(\mathbf{F}_{t_c},\ \mathbf{p}_{t_c})$ *are computed by interpolating* $\mathbf{F}_t,\ \mathbf{p}_t$ *for* $t = t_c$ *taking into account all geometric constraints.*

Justification. In practice, during the physical human-robot interaction the human usually grasps the robot, so the force application point is almost fixed. Also, some sliding contacts are possible, when the force application point is slowly moving on the robot surface. Since the joint torque measurements are obtained with high frequency, the current location of the force application point should be close to the previous ones. The latter allows us to compute the desired interaction parameter $\mathbf{p}$ using an enhanced interpolation technique, which takes into account relevant geometric constraints. □

Algorithm 14: Parameters identification using Enhanced Approach 1

Function `EnhancedApproach1(`$\mathbf{q}, \boldsymbol{\tau}$, **Robot**`):`
  Input : $\mathbf{q}$ is joints positions, $\boldsymbol{\tau}$ is torque measurements, **Robot** is a structure with robot geometric and kinematic parameters.
  Output: unique interaction parameters $\mathbf{F}, \mathbf{p}$
  $\boldsymbol{\Sigma}_\tau = 1/\boldsymbol{\sigma}_{\boldsymbol{\tau}}^2$;
  $\boldsymbol{\tau}_b = |\boldsymbol{\tau}| > \delta_\tau$;
  $k = \text{find}(\boldsymbol{\tau}_b \neq 0,\ last)$;
  while $k \leq n$ **do**
    $\mathbf{J} = \mathbf{J}_k^w(\mathbf{q}, \mathbf{p}^0)$;
    $\mathbf{W} = (\mathbf{J}^{\#})^T \cdot \boldsymbol{\tau}$;
    $\boldsymbol{\tau}_d = \mathbf{J}^T \cdot \mathbf{W} - \boldsymbol{\tau}$;
    if $||\boldsymbol{\tau}_d|| \leq \delta_w$ **then**
      parallel calculation block;
      $\mathbf{p}_c = (\mathbf{sv}_1^k + \mathbf{sv}_2^k + \mathbf{sv}_3^k)/3$;
      $\mathbf{J}_{\Omega_k} = \mathbf{J}_k(\mathbf{q}, \mathbf{p}_c)^T$;
      $\mathbf{F}_k = (\mathbf{J}_{\Omega_k}^T \cdot \boldsymbol{\Sigma}_\tau \cdot \mathbf{J}_{\Omega_k})^{-1} \cdot \mathbf{J}_{\Omega_k}^T \cdot \boldsymbol{\Sigma}_\tau \cdot \boldsymbol{\tau}$;
      $\boldsymbol{\varepsilon} = \left\| \boldsymbol{\tau} - \mathbf{J}_{\Omega_k}^T \cdot \mathbf{F}_k \right\|$;
      for $\left| \mathbf{n}^k \times \mathbf{F}_k \right| > \mu \left(\mathbf{n}^k \cdot \mathbf{F}_k \right)$ **do**
        $\boldsymbol{\varepsilon} = Inf$;
      $k = k + 1$;
    else
      break;
  if isEmpty(**Robot.History**) **then**
    $i = \arg\min_i(\boldsymbol{\varepsilon})$;
    $\mathbf{p} = \mathbf{p}_c(i)$;
    $\mathbf{F} = \mathbf{F}_k(i)$;
  else
    $\mathbf{F},\ \mathbf{p} = \text{interpolate}(\mathbf{Robot.History}, \mathbf{p}_c, \mathbf{F}_k, \boldsymbol{\varepsilon})$;
  Robot.History.Add($\mathbf{F},\ \mathbf{p}$);

Using the above-presented Hypotheses 5 to 8, it was developed a dedicated singularity resolution technique, which is formalized in the form of Algorithms 14 and 15 for Approaches 1 and 2 correspondingly. These algorithms, in addition to the singularity resolution, include the robustness enhancements from the previous section. Their core idea is to find the whole set of potential solutions $\{\hat{\mathbf{F}}\}$, $\{\hat{\mathbf{p}}\}$ and then select only one $\hat{\mathbf{F}}$, $\hat{\mathbf{p}}$ based on Hypotheses 5 to 8. In more detail, these techniques are described below.

The Enhanced Approach 1 is presented in Algorithm 14 consists of two steps. The first step (see lines 1-19) is similar to previously described Algorithm 10 and includes some enhancements based on the known torque measurement properties as well as parallel calculation of pseudoinverse. The second step (see lines 20-26) uses mentioned hypothesis in order to obtain a unique solution. Depending on the availability of the so-called estimation history, which accumulates the unique interaction parameters estimated on the previous timestamps, two cases are possible. If the estimation history **Robot**.**History** is not available then the parameters corresponding to the lowest residual ε are the solution. Otherwise, interpolate a new solution based on the previous ones according to Hypothesis 8. It is worth mentioning that only the centers of triangular faces describing the robot surface are evaluated here, satisfying Hypothesis 7, and as result, obtained approximate solutions indirectly follow Hypotheses 5 and 6.

The Enhanced Approach 2 is shown in Algorithm 15 and based on the previously described Algorithm 11. The singularity resolution part is placed at the end of the algorithm (see lines 28-39 of Algorithm 15). In more detail, if there is no estimation history, it relies on Hypotheses 5 to 7 to extract a single solution from the set of possible ones. According to them, the interaction force is selected as the closest one to the surface normal in case of multiple potential forces. Here, the force application point $\mathbf{p}$ is computed as the center of the corresponding solution line segment presented with two endpoints. If the interaction parameters $\mathbf{F}$ and $\mathbf{p}$ are obtained uniquely, the force application point $\mathbf{p}$ is selected as the closest one to the end-effector. If the estimation history is available, it relies on Hypothesis 8. According to it, in the case of interaction parameters ambiguity, the desired single solution is obtained by interpolating the previous estimates. It should be noted that in contrast to the standard interpolation technique, here the geometric constraints are also taken into account. It is clear that the estimation history does not exist at the initial time before the interaction and is deleted after the force is no longer applied to the robot.

The performance of the proposed singularity resolution technique was evaluated during the simulation study. Different scenarios were considered and one of them is shown in Fig. 4.13 as an example. In this figure, the collaborative robot is presented in six different timestamps, where t_c is a current time and Δt is a sampling interval. It should be noted that for the singularity resolution, the presented example is rather complex since all robot configurations are singular because the interaction force is applied to the $5th$ link. In addition, the fourth and fifth configurations corresponding to $t = t_c - \Delta t$ and $t = t_c$ are kinematically singular since their Jacobian $\mathbf{J}_k$ is rank deficient. The interaction parameters for the first configuration were estimated by applying Hypothesis 5, which prioritized the force application point at the $5th$ link.

Algorithm 15: Parameters identification using Enhanced Approach 2

Function `EnhancedApproach2(`$\mathbf{q}, \boldsymbol{\tau}$, **Robot**`):`
- **Input :** $\mathbf{q}$ is joints positions, $\boldsymbol{\tau}$ is torque measurements, **Robot** is a structure with robot geometric and kinematic parameters.
- **Output:** unique identified interaction parameters $\mathbf{F}, \mathbf{p}$.

$\boldsymbol{\Sigma}_\tau = 1/\boldsymbol{\sigma}_{\boldsymbol{\tau}}^2$;
$\boldsymbol{\tau}_b = |\boldsymbol{\tau}| > \delta_\tau$;
$k = \text{find}(\boldsymbol{\tau}_b \neq 0,\ last)$;
while $k \leq n$ **do**
  $\mathbf{p} = \mathbf{p}^k$;
  $\mathbf{F} = (\mathbf{J}^{\#})^T \cdot \boldsymbol{\tau}$;
  for $i = 1 : i_{max}$ **do**
    $\mathbf{J} = \mathbf{J}_k(\mathbf{q}, \mathbf{p})$;
    $\boldsymbol{\tau}_r = \mathbf{J}^T \cdot \mathbf{F}$;
    $\boldsymbol{\tau}_d = \boldsymbol{\tau}_r - \boldsymbol{\tau}$;
    $\mathbf{D}_k = [\mathbf{J}_k(\mathbf{q}, \mathbf{p}) \quad \mathbf{H}_k(\mathbf{q}, \mathbf{F})]^T$;
    $[\delta\mathbf{F},\ \delta\mathbf{p}]^T = (\mathbf{D}_k^T \cdot \boldsymbol{\Sigma}_\tau \cdot \mathbf{D}_k)^{-1} \cdot \mathbf{D}_k^T \cdot \boldsymbol{\Sigma}_\tau \cdot \boldsymbol{\tau}_d$;
    $\mathbf{F} = \mathbf{F} + \delta\mathbf{F}$;
    $\mathbf{p} = \mathbf{p} + \delta\mathbf{p}$;
    $\mathbf{S} = \text{svd}((\mathbf{D}_k)$;
    $r = \mathbf{S} > tol$;
    $d = 6 - r$;
    if $||\boldsymbol{\tau}_{prev} - \boldsymbol{\tau}_r|| \leq \varepsilon_\tau$ **then**
      break;
    $\boldsymbol{\tau}_{prev} = \boldsymbol{\tau}_r$;
  if $||\boldsymbol{\tau}_d|| \leq \delta_w$ **then**
    Solution$(k) = [\mathbf{F},\ \mathbf{p},\ d]$;
    $k = k + 1$;
  else
    break;
$\{\mathbf{F}\},\ \{\mathbf{p}\} = \text{ApplyConstraints}(\mathbf{Solution},\ \mathbf{Robot})$;
if isEmpty(**Robot**.**History**) **then**
  $\mathbf{p}_k = \{\mathbf{p}(k)\}$;
  $\mathbf{F}_k = \{\mathbf{F}(k)\}$;
  if isPoint$(\mathbf{p}_k)$ **then**
    $\mathbf{p} = \mathbf{p}_k$;
    $\mathbf{F} = \mathbf{F}_k$;
  else
    $\mathbf{p} = (\mathbf{p}_k^i + \mathbf{p}_k^{i+1})/2$;
    $\mathbf{F} = \arg\min_i(||(\{\mathbf{F}_k^i\},\ \mathbf{n}_k^i)||)$;
else
  $\mathbf{F},\ \mathbf{p} = \text{interpolate}(\mathbf{Robot}.\mathbf{History}, \{\mathbf{F}\},\ \{\mathbf{p}\})$;
Robot.**History**.Add$(\mathbf{F},\ \mathbf{p})$;

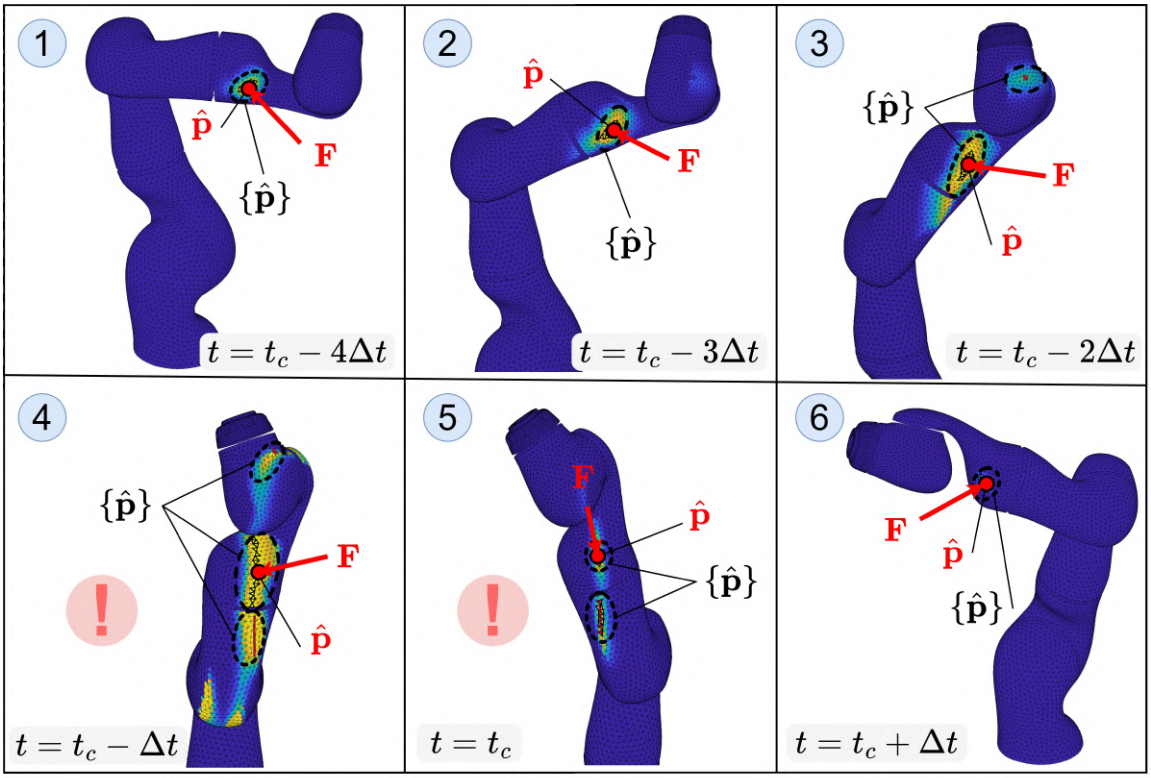


Figure 4.13: Example of the singularity resolution using the estimation history in 3D case. The red arrow corresponds to the real interaction force, the black circles show all potential force application points $\{\mathbf{p}\}$ provided by the estimation algorithm, while the red dot shows $\mathbf{p}$ chosen by the developed singularity resolution technique.

For the third timestamp with $t = t_c - 2\Delta t$, Hypothesis 8 is applied since the previous estimate of the interaction parameters for $t = t_c - 3\Delta t$ is already available. Here, the set of all potential solutions for force application point $\{\mathbf{p}\}$ consists of two points, corresponding to $k = 5$ and $k = 6$. In this case, the point $\mathbf{p}$ located on the last link with $k = 5$ is selected since it is closer to the previous estimate. It should be noted that for both timestamps $t = t_c - 2\Delta t$ and $t = t_c - 3\Delta t$ the corresponding static equilibrium systems have full rank, so the solution for interaction force $\mathbf{F}$ is unique.

For the fourth timestamp with $t = t_c - \Delta t$, Hypothesis 8 is also applied. In contrast to the previous cases, the robot configuration here is kinematically singular and the Jacobian $\mathbf{J}_k$ is rank-deficient. So, the set of all potential solutions for force application point $\{\mathbf{p}\}$ is represented by the multiple line intervals, which yields pencils of possible forces $\{\mathbf{F}\}$. Thus, the ambiguity for both interaction parameters ($\mathbf{F}$, $\mathbf{p}$) is high, but it is resolved by applying the developed interpolation technique of two previous estimates that provides a single solution. It should be noted that in this interpolation, the set of all potential solutions is used as constraints, ensuring that estimated the point $\mathbf{p}$ is located on the robot surface and the force $\mathbf{F}$ is inside of the friction cone.

Thus, as follows from this simulation study, the proposed technique is capable of resolving the singularity problem in the identification of the human-robot interaction parameters. It is clear that in industrial applications, it is challenging to reduce the ambiguity of multiple solutions produced by the identification algorithm presented in the previous sections. Besides, generally, the industry-oriented singularity resolution technique should be simple, reliable, and rather fast. So, the developed technique, which is based on several hypotheses obtained from practical experience, allows us to resolve the ambiguity problem in the interaction parameters identification.

4.5.2 Resolution of multiple simultaneous physical interactions

In some cases, several physical interactions can be applied to the collaborative robot. For example, the first interaction may be caused by some industrial process and the second is the interaction with a human. However in practice, for collaborative robots with one-axis torque sensors, it is hard to detect more than one collision. The latter requires to estimate 6 interaction parameters $(\mathbf{F},\ \mathbf{p})$ for each interaction, while the typical collaborative manipulator usually has 7 sensors at most. It is clear that a single physical interaction with joint torque sensors-equipped robot, as was shown in this chapter, is identifiable and has a unique solution for most of the practically usable cases. In the case of two simultaneous interactions where it is necessary to identify 12 interaction parameters $(\mathbf{F_1},\ \mathbf{p_1})$ and $(\mathbf{F_2},\ \mathbf{p_2})$, there is a unique solution for some particular cases only, where at least a few of the interaction parameters are known. With three interactions, there are an infinite number of solutions, which follows from the underdetermined static equilibrium system.

Another problem is distinguishing of single and multiple interactions. It is clear that for the multiple collision case, the system of static equilibrium equations is underdetermined, which means that it has no unique solution and potentially could be described as a single interaction. So, some single-interaction parameters $(\mathbf{F},\ \mathbf{p})$ can be found that

$$\begin{array}{lll} \mathbf{J}_k(\mathbf{q},\mathbf{p})^T\mathbf{F} = & \mathbf{J}_k(\mathbf{q},\mathbf{p}_1)^T\mathbf{F}_1 + & \mathbf{J}_k(\mathbf{q},\mathbf{p}_2)^T\mathbf{F}_2 \\ \mathbf{p} \in \mathbf{\Omega}_k & \mathbf{p}_1 \in \mathbf{\Omega}_{k1} & \mathbf{p}_2 \in \mathbf{\Omega}_{k2} \\ \mathbf{F} \in \mathcal{F}_\mu & \mathbf{F}_1 \in \mathcal{F}_{\mu 1} & \mathbf{F}_2 \in \mathcal{F}_{\mu 2} \end{array} \tag{4.44}$$

where $(\mathbf{F_1},\ \mathbf{p_1})$ and $(\mathbf{F_2},\ \mathbf{p_2})$ are the parameters of "true" double-interaction applied to the robot.

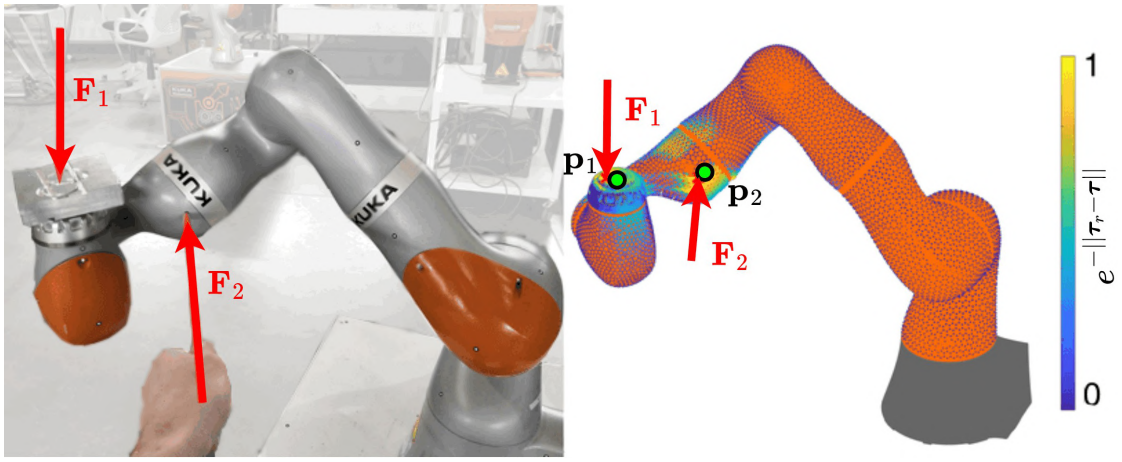


Figure 4.14: Example of multiple interactions identification. The first interaction is applied to the end-effector and the second to the fifth link. The red arrow shows the applied force and the green circle is the estimated interaction location.

In order to reliably identify both force and its application point, in case of multiple interactions, it is required to utilize additional sensors. For example, a vision sensor with or without depth channel can be used to extract approximate of interaction force application point. More expensive 6-dof force/torque sensors installed at the robot base will add six equations in the considered static equilibrium system for any application point and potentially make it overdetermined, so a simple pseudoinverse is enough to provide the solution. It is also possible to use tactile skin sensors in the robot parts where interaction is expected, here the interaction parameters can be measured directly. If it is not possible to use the additional sensors, some assumptions and apriori information can be used to separate the torque caused by the different interactions. For example, in pick and place operation, the payload in the robot gripper will generate some external torques in joints. However, if the weight and inertial parameters of the payload are known, they can be included in the robot dynamic model so the observer will provide zero external torque values. In this case, any additional interaction can be considered as the only interaction with the robot, ignoring the payload attached to the end effector.

The identification problem can also be partly avoided by assuming that interactions happen sequentially and some of the interaction parameters are known e.g. one of the interactions is applied at the end-effector of the robot, which is common for most industrial applications. If the robot performs a contact operation like pick and place, drilling, and so on, there is one physical interaction associated with a task at the end-effector and a second interaction in the intermediate point of the robot in case of a collision with the environment/human. It is clear that the amplitude of both interaction forces should be comparable, otherwise, one of the interactions could be

Algorithm 16: Parameters identification in case of multiple interactions

Function $findMulti(\mathbf{q}, \boldsymbol{\tau}, \mathbf{Robot})$**:**
  Input : $\mathbf{q}$ is joints positions, $\boldsymbol{\tau}$ is torque measurements, **Robot** is a structure with robot geometric and kinematic parameters.
  Output: set of identified interaction parameters $\mathbf{F}_i, \mathbf{p}_i$.
  if *collision* **then**
    $\mathbf{F_1},\ \mathbf{p_1} = findParams(\boldsymbol{\tau}, \mathbf{Robot})$;
    if $\left\|\boldsymbol{\tau} - \mathbf{J}_1^T\mathbf{F}_1\right\|^2 > \varepsilon_m$ **then**
      $\boldsymbol{\tau}_{new} = \boldsymbol{\tau} - \mathbf{J}_1^T\mathbf{F}_1$;
      $\mathbf{F_2},\ \mathbf{p_2} = findParams(\boldsymbol{\tau}_{new}, \mathbf{Robot})$;
      ...

identified as a noise for a significantly larger second one.

The multi-interaction case is shown in Fig. 4.14, where the first interaction force is caused by the constant gravity force of the load applied to the end-effector, and the second one by the force applied by a human.

The conventional approach could be extended for multi-interactions cases by making the following assumptions: (i) interactions appear sequentially, one after another; (ii) the first interaction does not change its location. These assumptions also allow to identify interactions in two random points on the robot surface, without the limitation that one of them should be in the end-effector. Algorithm 16 shows this procedure. The $findParams(\boldsymbol{\tau}, \mathbf{Robot})$ function finds the interaction parameters caused by the first interaction, depending on the torques $\boldsymbol{\tau}$ as was described before. When torques could not be precisely described with only one interaction and the residual is larger than ε_m, the second interaction should be considered. In this case, the first interaction is assumed to be the same as in the previous time interval before the second interaction appeared. Since each interaction is considered separately, the accuracy of the multi-interaction identification will be close to the values mentioned before. However, it is important to note that in practice the measurement noise and the presence of singular configurations make the multi-interaction identification usable in only a small number of cases.

4.6 Summary of the chapter

This chapter presents some new results on robust estimation of the interaction parameters for human-robot collaboration for the 3D spatial case. The developed techniques use the measurement data from the internal joint torque sensors only. In contrast to

other works, the torque measurements are assumed to be corrupted by unbiased, independent but *not identically distributed noise*, which contradicts to usual i.i.d. assumption widely used in related literature. Under such assumptions, two approaches for the identification of interaction force and its application point were developed. Approach 1 is based on the straightforward enumeration of triangle mesh elements describing the robot surface. Approach 2 relies on obtaining of the geometrical intersection of the unconstrained solution of the static equilibrium system with the robot triangular mesh, taking into account corresponding friction cones. In order to ensure the robustness of the developed techniques with respect to the measurement noise, some enhancements were proposed allowing to reduce essentially the estimation errors.

Besides, compared to other known techniques, the developed ones are capable to estimate the desired interaction parameters even in some *difficult singular cases*. The latter arises when the interaction force is applied to lower manipulator links, resulting in a reduced rank of the static equilibrium system, or when the force action line intersects some joint axes, leading to zero torque measurements from the corresponding sensors. In both cases, it is not possible to obtain a unique solution for the interaction parameters. To overcome this difficulty and select a single solution from the set of possible ones, the special singularity resolution technique is proposed. This technique is based on practice-inspired heuristics and interpolation of the interaction parameters obtained at the previous timestamps. The validity of the developed methodology was confirmed by a simulation study.

The **main advantages** of the interaction parameters identification techniques developed in this section can be summarized as follows:

1) The **accuracy** of the proposed techniques is rather high. It obviously allows to obtain zero estimation error in the absence of the measurement noise for Approach 2 and sufficiently low error in the case of Approach 1. For the realistic torque measurement noise observed in real collaborative robots, Approaches 1 and 2 provided the position error of about 6.4 mm and 4.8 mm, while the force estimation error was 13.5 % and 3.1 % correspondingly. It is clear that such accuracy is quite suitable for the most of practical applications.
2) The **real-time performance** of the developed algorithms is also better than the existing ones. For the considered case study, it allowed to find the desired interaction parameters in 2.3 ms and 1.8 ms on average for Approaches 1 and 2 correspondingly. Such high performance of Approach 1 is achieved due to parallel code execution and because of the analytical nature of the technique implemented in Approach 2. So, it well suits to real-time requirements of the human-robot interaction handling, where the response time is critical because of safety reasons.

3) The ability of **singularity resolution**: the developed techniques allow to obtain the desired interaction parameters even in the case of essential ambiguity of relevant mathematical equations. By using the identification history, the set of possible solutions is reduced to a single one, which is closer to the real interaction parameters. As follows from the relevant case studies, only two previous history timestamps are sufficient to cross the singularity without losing the real interaction parameters. Thus, such a simple singularity resolution method providing a unique solution for the interaction parameters essentially contributes to the safe and efficient human-robot interaction handling.

It is worth mentioning that both Approaches 1 and 2 are suitable for industrial applications. However, Approach 1 is preferable if the statistical properties of the measurements noise are unknown since the first technique is essentially more robust but slower and less accurate. Otherwise, if the noise properties are known, then Approach 2 provides some benefits. It allows to achieve 4x and 1.3x improvement in accuracy for interaction force and its application point estimation correspondingly and also 1.3x speedup. The next chapter will present some application examples and related experimental studies confirming the efficiency of the developed techniques in a real-life environment.

Chapter 5

Development of adaptive interaction handling strategy and experimental validation

Contents

This chapter focuses on the development of the adaptive technique for handling of physical interactions between a human and a robot, as well as experimental validation of the developed identification algorithms. The main goals of this technique are to ensure the safety of a human operator and improve the performance of human-robot collaboration by implementing various scenarios. In the scope of this technique, the interaction parameters identification methods proposed in the previous chapter are used for the selection of an appropriate reaction strategy. These parameters define the interaction force and its application point on the robot surface that are used for interaction classification within the set of predefined categories. Based on these categories and the current robot state, the algorithm chooses an appropriate robot reaction. At the end of this chapter, the experimental results are presented that are obtained for the real collaborative robot KUKA LBR iiwa and validate the developed techniques.

5.1 Introduction

In previous chapters, the main attention was paid to the mathematical issues related to the interaction parameters identification and development of relevant estimation algorithms and numerical techniques. Now let us concentrate on integrating these algorithms into adaptive control strategies allowing to ensure safe robot collaboration with a human operator. It is worth mentioning that here the notion of adaptivity is used for high-level robot control, in contrast to its usual relevance to the low-level control of robot actuators.

The problem of the human-robot interaction handling was the focus of the robotics community for several years. [90]. Some early works in this field [91] discussed requirements for the robot mechanics and its control systems in order to provide safe interaction with a human. It highlighted the importance of robot actuators backdrivability and the limitation of robot motor torques while contacting with a human and switching the robot controller to the so-called gravity compensation mode. More recent works proposed more sophisticated approaches for interaction handling, which operate with several possible robot reactions depending on the robot state. A formal way of defining the collaborative robot safety strategies can be found in [92]. This approach uses a graph representation of the safety mode concept, which associates a specific set of rules for each functional behavior of the monitored robot system.

To implement desired safety strategies, there are several approaches to control program architectures, which allow a robot and a human to execute tasks in the shared workspace. Several research groups applied finite state machines for interaction handling, where the states are the robot reactions. In particular, [26] proposed a practice-oriented approach, which operated with four main robot states: autonomous in the case of a human absence; high compliance mode in the case of a human presence; collaborative mode with a human in the loop; reflex reaction mode in the case of a fault. More advanced techniques were proposed in [36] and the following work [27] of these authors, who additionally used external vision sensors for the robot collision avoidance and safe human-robot coexistence. In both works, the collision handling strategies operated with the above-mentioned reactions such as switching to the emergency stop, reflex mode, kinematic redundancy mode, etc. Besides, the physical interaction parameters were also taken into account for choosing an appropriate robot reaction. Here an external RGB-D sensor was used to estimate the interaction force application point. It is clear that the practical application of such sensors is limited because of possible occlusions and low frames-per-second rates.

In this chapter, the developed interaction parameters identification techniques are integrated into high-level robot control software assuming that only internal robot sensors are available. Relevant experimental results are also presented that confirm the validity and utility of the proposed methods for safe and efficient human-robot physical interaction and collaboration.

5.2 Adaptive interaction handling strategy for safe collaboration

In order to provide safe and efficient interaction handling in the human-robot collaboration, it is important to consider the following aspects. The principal one is the prioritization of the desired characteristics of the entire workcell consisting of the robot, some auxiliary equipment and the operator. For classical industrial robots, the main performance indicator is their manufacturing efficiency characterized by accuracy, speed and load capacity. In contrast, collaborative robots share the working environment with humans, so safety is the most important issue. Thus, for the collaborative robots the prioritization of the desired performances should be the following: (i) human safety, (ii) robotic cell safety, (iii) efficient execution of the manufacturing task. It means that for the handling of each physical interaction, the collaborative robot should choose a proper reaction, which is the most riskless for the operator as well as prevents damaging the robot itself while executing the desired task in a more accurate and fast way. In practice, these can be achieved by applying advanced joint-level control, which ensures the required joint angles, velocities and, in addition, desired joint compliance. Relevant low-level controller settings are generated by the high-level controller that ensures the safe interaction between the robot and the operator.

The simplest way of interaction handling in collaborative robotics is a straightforward stop reaction, where the robot suspends all motions if the interaction is detected. Such a reaction is obviously safe but can be hardly efficient in manufacturing workcells. In particular, there are a number of so-called ”contact operations”, which involve direct physical interaction between a robotic tool and a workpiece. Hence in practice, it is required to employ more complex robot reactions depending on the interaction type and parameters. Besides, it is necessary to take into account that for the contact operations the interaction force can be caused by different sources, such as a technological process, an obstacle, or a human. It is clear that the robot reactions for each case should be different and usually pursue distinct goals.

Thus, it looks promising to use the adaptive compliance robot behavior in order to improve the robot task execution efficiency and human safety during their collaboration. To implement such behavior, it is necessary to develop several control algorithm modules ensuring switching between different reactions, as well as a set of possible robot reactions, which are described in detail below.

5.2.1 Robot behavior modes in human-robot interaction

Let us concentrate further on possible robot behavior modes that obviously depend on the existence of physical interaction and its parameters. If the robot does not interact with the operator or any object in its environment, the robot behavior mode will be further referred to as the *autonomous task execution mode*. Here, the robot simply follows the desired reference trajectory, without controlling the joint compliance. In contrast, if the human-robot or environment-robot interaction is detected, then the robot is switched to the so-called *interaction handling mode*. This mode is activated by interaction detection and remains active until it disappears. It should be noted that there is also a special case of the *autonomous task execution mode*, which corresponds to the contact operations and assumes the physical interaction between the robot and a workpiece. In addition, for safe operation, the *emergency fault mode* should be provided that is enabled when some modules of the system are not working as expected. It is clear that generally, the robot reaction and its working mode depend on the interaction type and parameters.

Let us consider first the *autonomous task execution mode*, which is characterized by the absence of any robot reactions since there is no unexpected physical interaction. As it has been mentioned above, here two sub-cases that are possible:

Non-contact mode. It corresponds to the robot following the desired trajectory without controlling the joint compliance. In this case, no obstacles or a human in the robot environment were detected. The principal goal here is the efficient execution of the task while ignoring the safety issues both for the robot and the operator.

Contact mode. It corresponds to the robot following the desired trajectory while maintaining the desired interaction force amplitude and direction. Typically such interaction force is applied to the workpiece by the robot tool at the desired time. However, any additional robot interaction with a human or environment is not expected, so the safety issues both for the robot and the operator are ignored.

Further, in the *interaction handling mode* four sub-cases are possible that differ in robot reaction to the detected interaction:

Pause mode. In such mode the robot stops its motions along the desired trajectory for some time. The motion recovery process can be initialized either manually, initialized by the operator, or automatically after a period of time. The first option is convenient for the operator and can be implemented as a reaction to the second touch: the first touch stops the robot, and the second one resumes its motion. The automatic option is useful in case of brief interactions, after which the robot can continue its motion if the interaction force is no longer applied.

Compliant mode. In this mode the robot motions are determined by the joint compliance controller, which ignores the desired trajectory. In the literature, this mode can be also referred to as the reflex mode. Here, the robot position and configuration can be changed manually, by the operator who is applying proper force to the end-effector. Such a technique can be used if the robot can not avoid an obstacle or when it is reaching the joint angle limits.

Redundant mode. In this mode the robot motions are handled by the hybrid position-compliance controllers for all joints, which try to follow the desired trajectory while ensuring certain manipulator stiffness. Here, the robot kinematic redundancy is usually used to simultaneously move the end-effector along the desired path and maintain the interaction force at a safe level. In contrast to the two previous modes, in this case, the task execution is not interrupted. This mode is available for robots with more than six degree-of-freedoms only.

Obstacle avoidance mode. In this mode the robot tries to avoid an obstacle on its way. Since it is assumed that only the internal sensors of the robot are available, the obstacle is detected after interacting with it. Further, the identified interaction parameters are used to compute the obstacle position. Depending on the robot state and the interaction type, the dedicated obstacle avoidance algorithm generates a new collision-free path in the neighborhood of the detected obstacle.

Finally, if some of the robot safety features are violated or the robot encountered some hard-to-solve problems requiring the operator intervention, the *emergency fault mode* is activated:

Emergency stop mode. In such mode the robot stops its motions along the desired trajectory and waits. Activating this mode is similar to pushing the ”red button” on the robot controller. Such a mode is enabled if the identified interaction force exceeds some safe limits. In contrast to the pause mode described above, this mode necessarily requires the operator intervention to proceed.

Hence, to ensure safe human-robot collaboration, three basic behavior modes with seven sub-cases describing the robot reactions are proposed. It is clear that these behavior modes are very basic and may be modified and adopted for any particular practical application. To ensure switching between these modes in real-time, a dedicated high-level controller should be developed, which will be the focus of the following subsection.

5.2.2 Robot reaction control based on interaction parameters

In order to achieve the desired safe human-robot collaboration, it is necessary to ensure proper switchings between the above-described robot modes. In this works, it is implemented by a high-level controller that is usually used for trajectory planning and auxiliary equipment control. This additional feature of the human-robot interaction handling may be realized by using a finite state machine, which is widely used in computer science. For the considered application, the machine states are the robot behavior modes while the transitions are the switchings between these modes. In this work, the transitions are executed using some specific conditions, which depend on the identified interaction parameters. In more detail, such finite state machine and relevant transition conditions are presented below.

The proposed *finite state machine* for the human-robot interaction handling is represented in Fig. 5.1. This machine includes seven states corresponding to different robot behavior modes described in the previous subsection. The transition between these states is executed when the interaction is detected and highly depends on the identified interaction parameters. In particular, some transitions are invoked if the interaction appears while other ones are activated when the interaction disappears. In more detail, the functioning of this finite state machine is described below.

First, let us consider the robot behavior in the case of the **non-contact operation mode**, which is typically used for the robot tool positioning. The simplest and safest reaction to any accidental interaction (**Accd**) is to stop the robot. The latter corresponds to the transition to the **pause mode**. Here, two options of this mode are used, which depend on the interaction duration. In the case of the brief interaction with the operator (**Soft**), the pause mode is activated and deactivated by touch: the first touch stops the robot motions and the second one resumes the execution of the non-contact mode. But for the long interaction, for instance, if the interaction force is caused by the collision with the robot environment (**Hard**), the pause mode will remain enabled until the force disappears. In the case of the intentional interaction (**Intl**), the robot behavior depends on the interaction parameters. If the interaction force is applied at the robot intermediate link (**Link**), the **redundant mode** is activated that does not

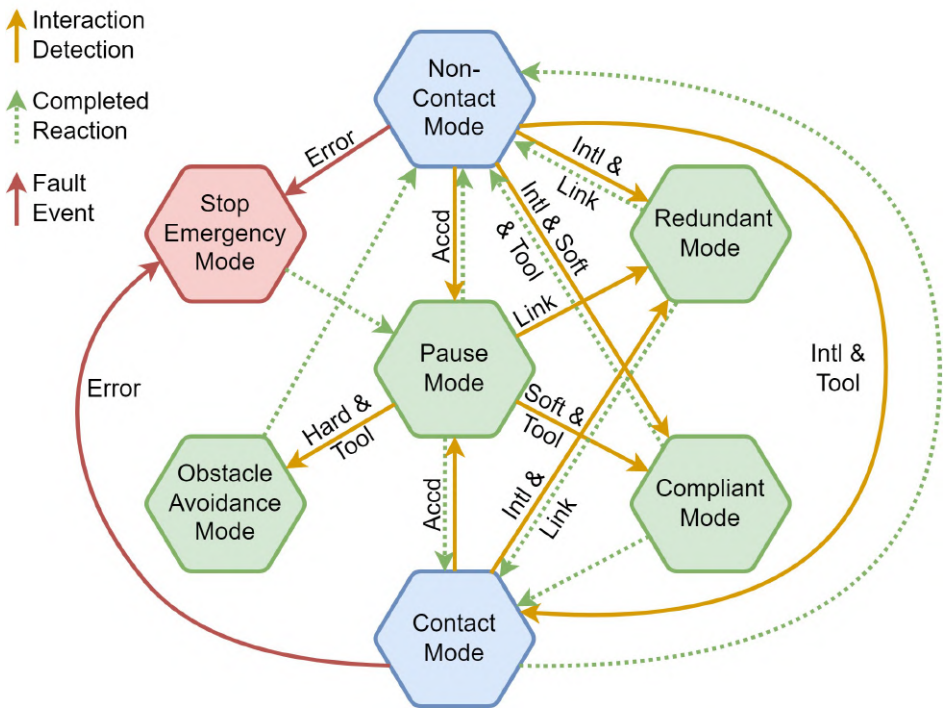


Figure 5.1: Proposed finite state machine with its states and transitions. The *states* are the robot behavior modes and *transitions* are switchings between the states depending on the interaction parameters and characteristics: applied at the tool or intermediate link (Tool/Link); having soft or hard nature (Soft/Hard); interaction intention (Intl/Accd). Only the most important states and transitions are shown.

stop the robot motion, continues executing the desired end-effector trajectory, but does not allow increasing of interaction force amplitude. In contrast, if the operator interaction with the robot end-effector was detected (**Tool & Soft**), the robot is switched to the **compliant mode**, where it stops the motions and increases the compliance in order to avoid the operator injury. Otherwise, if such interaction force is caused by the robot environment (**Tool & Hard**), then the robot is switched to the **obstacle avoidance mode**. It is worth mentioning that the considered controller also allows to switch the robot from the non-contact mode to the contact mode, which corresponds to the intentional interaction between the robot tool and the workpiece (**Intl & Tool**).

Further, consider the **contact operation mode**, where the robot interacts with a workpiece by means of its tool. In this case, the number of potential reactions is limited since the robot should maintain the desired interaction force depending on the technological process. Here, there are two options for the robot reactions, switching either to the **redundant mode** or **pause mode**. The first one is used when the force is applied to the intermediate robot link (**Intl & Link**) and it is similar to the non-contact case described above. The second reaction is applied in the case of any accidental interaction (**Accd**), it stops all robot motions since it is the safest robot

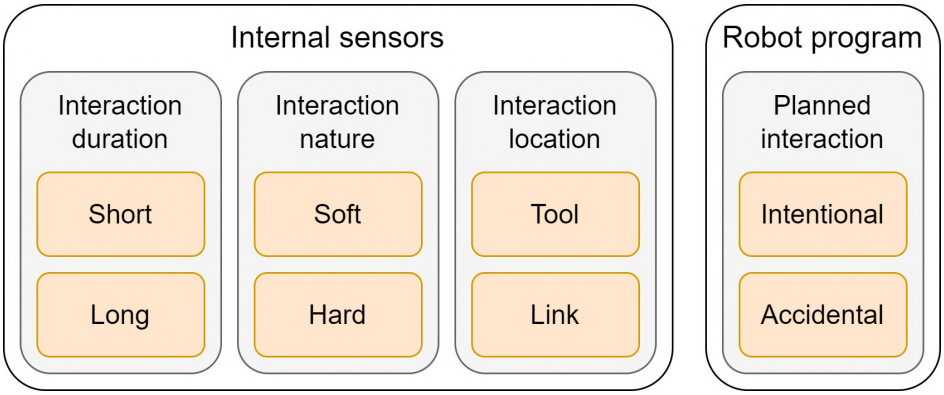


Figure 5.2: Diagram of interaction classes. By using the robot internal sensors only it is possible to determine interaction duration, nature, and location. The possibility of intentional or accidental interaction could be predefined by the robot program.

behavior here. Any other reactions are not provided for the contact mode, because of the requirements issued from the technological process specifications e.g. the desired interaction force amplitude.

It should be mentioned that for the considered finite state machine, some transition conditions require additional characteristics of the interaction, in addition to the previously described force amplitude, direction, and its application point. In particular, it is necessary to distinguish robot-operator and robot-environment interactions (soft or hard) in order to choose the proper reaction. Besides, the interaction intention (accidental or intentional) is also important since it must influence the robot behavior. The latter motivates the development of dedicated interaction classification techniques, which are presented in the following subsection.

5.2.3 Classification of human-robot interactions

To ensure safe human-robot collaboration, the robot controller must take into account not only the interaction parameters $\mathbf{F}$, $\mathbf{p}$ but also distinguish some interaction properties, which may have different nature [25], [93]–[95]. In particular, the interaction may be caused by accidental contact with the operator or by collision with the auxiliary equipment, as well as by intentional force application to the workpiece. Let us consider this issue in detail, starting from the definition of the possible interaction classes and then proposing techniques allowing to attribute these classes to the interactions.

Interaction classes. The proposed set of the interaction classes is shown in Fig. 5.2. First of all, it is necessary to distinguish the interaction duration, which is described here by **Short** and **Long** classes. Relevant classification can be done by a simple comparison of the interaction time with a predefined threshold value. The latter allows to separate easily a short accidental operator touch and dangerous colli-

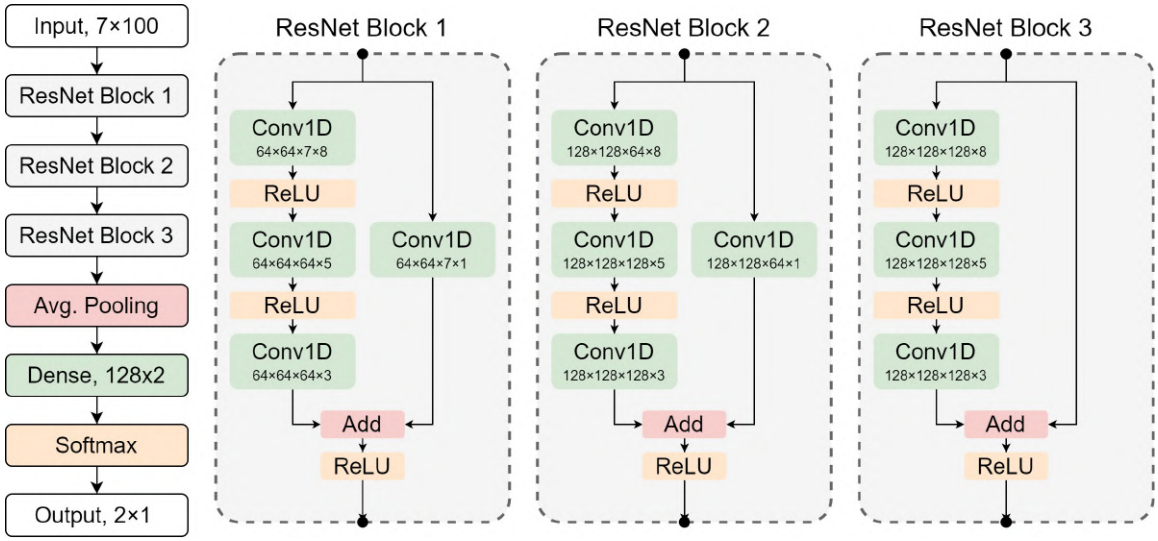
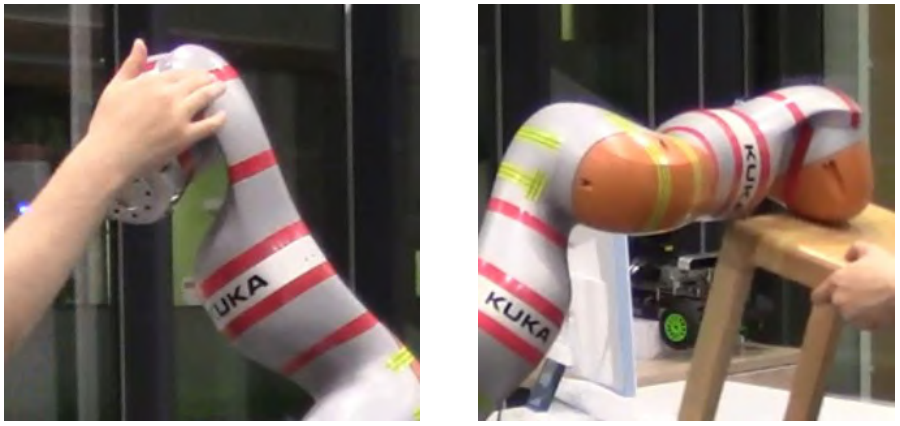


Figure 5.3: Structure of deep neural network for Soft/Hard interaction classification. As input, the fixed time window of the robot joint torque measurements is used, the output is the interaction class.

sion with an object in the robot environment. Further, the interactions differ in their nature, assumed to be either **Soft** or **Hard**. In practice, the soft interactions are usually associated with a human, while the hard ones normally occur when the robot collides with a workpiece, walls, etc. However, by using the robot internal sensors only, it is rather hard to separate the human-robot interaction and the robot interaction with some soft objects. Thus, it is explicitly assumed here that any soft interaction is related to the operator. Relevant classification can be done by analyzing the time series describing the robot joint torques evolution over time. Also, the interaction may be applied to either the robot end-effector (**Tool**) or its intermediate **Link**. Relevant classification can be easily done by using the interaction parameter $\mathbf{p}$ provided by the identification algorithm described in the previous chapters. Finally, any interaction can be either intentional (**Intl**) or accidental (**Accd**). For the non-contact operations, in most cases the interaction is accidental but for the human-robot collaborative work, the interaction is usually intentional. In practice, a relevant decision is made at the robot programming stage and does not depend on robot sensor readings.

Classification techniques. To specify the interaction class, different techniques can be applied. Let us describe those of them that were used for our experimental study. For the Short/Long, Tool/Link, and Intentional/Accidental classes, the classification techniques are rather straightforward in implementation, as was described above. In contrast, distinguishing between Soft and Hard interactions is more complicated and requires some advanced algorithms to analyze the robot joint torques evolution over time. In this study, neural network techniques were used to achieve this classification.



(a) Soft interaction (b) Hard interaction

Figure 5.4: Dataset acquisition examples for the interaction classification. The robot physically interacts either with a human or some rigid objects from its environment.

To classify time-series data obtained from the robot joint torque sensors, a deep neural network was used since this technique proved to be very efficient for similar problems [96]. In this network, the fixed time window of the torque measurements provided the input and the desired interaction class (Soft/Hard) was obtained at the output. The generalized architecture of such neural network is presented in Fig. 5.3, which implements the residual-based technique also known as ResNet and widely used in image analysis [97], robotics applications [98] and speech recognition [99].

The developed network consists of three sequential Residual blocks. Similar to the original ResNet, each of these blocks has two paths for the input data, where the first path includes three subsequent convolutions with ReLU activation functions. These convolution layers have a kernel size of 8×8, 5×5, and 3×3 correspondingly. The second path has only one convolution layer, except for the last Residual block, which has a direct connection to the end of the first path. In each of the Residual blocks, the outputs of the first and second paths are combined by the addition operation and following ReLU activation. It should be noted that the size of tensors is 64, 128, and 128 for the first, second, and third Residual blocks correspondingly. The output of the last Residual block is followed by the global average pooling and the dense layer with a softmax activation function. So, after the proposed modifications of the classical ResNet, the developed network is able to attribute the physical interaction to one of the predefined classes, Soft or Hard.

To train the developed network for the interaction type classification, the proper dataset was captured from the KUKA iiwa robot used in our experiments. To obtain the desired data, the robot followed some random trajectories and repeatedly collided with the operator and environment. In total, about 430000 data frames with 100 Hz

rate and 500 collision cases were recorded, which corresponded to both hard and soft interactions. The latter of them occurred not only in a collaborative way, where the operator guided the robot, but also in an accidental way, where the robot unintentionally collided with the operator arm, back, or body. An example of data acquisition for soft or hard interaction cases is presented in Fig. 5.4. The developed neural network was trained using the backpropagation algorithm with Adam optimizer. Since there are only two classes, Soft and Hard, the binary cross-entropy was used as the loss function.

As follows from a relevant experimental study, the developed neural network provided an accuracy of 98% on the test data. Besides, additional experiments proved the network ability to separate reliably the interactions with the operator hand, or body from the interactions with hard objects, such as metal workpieces and walls.

It should be also mentioned that the simplest one-layer feedforward neural network was also evaluated for its ability to solve the same task. Here, the Fourier transform coefficients were used as the input and the network was composed of 100 neurons. However, this simplified approach provided essentially lower accuracy of 89%, which is not enough for reliable interaction classification.

Thus, to provide safe robot interaction with the operator and the environment, the relevant controller was developed, based on the finite state machine technique. This controller allows to change the robot behavior depending on the detected interaction type and parameters. In particular, the controller distinguishes the Short and Long, Soft and Hard, Tool and Link, Intentional and Accidental interactions, which cause switching between several predefined robot behavior modes. The efficiency of this controller is demonstrated in the following sections that present some experiments with the real collaborative robot KUKA iiwa.

5.3 Experimental validation of developed interaction identification technique

This section presents experimental results, which validate the developed identification techniques for the human-robot interaction parameters from Chapters 3 and 4. Particular attention is paid to the identification accuracy, time performance, and ability to handle singular cases. At the beginning of this section, the experimental setup and relevant measurement procedure are described. Further, the obtained identification results for both non-singular and singular cases are presented and analyzed. In contrast to the other works, the provided experimental study includes a large number of singular cases, for which the identification procedure provides an ambiguous solution.

5.3.1 Experimental setup and measurements procedure

In this validation study, a number of real-life experiments were conducted, which involved direct physical interaction between the operator and the collaborative robot KUKA LBR iiwa 14. Its key feature is the redundant kinematics with seven degrees of freedom and built-in one-axis torque sensors in each joint. This robot is fully compatible with modern industrial standards for human-robot collaboration and quite popular in manufacturing. For evaluation purposes, the external measurement system was used, allowing to obtain the actual interaction parameters, which were compared to the ones provided by the developed identification techniques. This external system included the 3D scanning device GOM ATOS 5, providing an accuracy of ± 0.03 mm. The accuracy of the joint torque measurements provided by the robot built-in sensors was estimated in a separate study whose results are presented in Table 4.8.

To estimate actual values of the interaction force application points $\mathbf{p}$, special *markers* were attached to the robot link surfaces, whose Cartesian coordinates were estimated by the ATOS 5 measurement system. These markers consisted of white and black circles of diameter 3 and 7 mm respectively, they were placed at 50 different locations on the robot surface in order to provide an accurate scanning process independent of the current robot posture. Relevant marker distribution on the robot is presented in Fig. 5.5, where the left and right-hand sides correspond to the real robot and to the robot model used in the identification process.

In addition, the ATOS 5 system was also used for obtaining the 3D *model of the robot surfaces*, required by the developed identification techniques. To obtain such a model, the KUKA iiwa robot was scanned by ATOS 5 system, providing a precise 3D presentation of the robot surface. In order to make this model suitable for the developed identification algorithms, some compressing was applied since the initial model is excessively large for the considered problem (it consists of millions of small triangles). In this study, the simplified robot surface model was composed of 14000 triangles of about 1 cm in size. It should be also noted that the obtained simplified model was used to apply both the surface constraint ($\mathbf{p} \in \mathbf{\Omega}_k$) and the force direction constraint ($\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$) in the related identification problem. For the second of them, the dry friction coefficient $\mu = 0.6$ was used since it is a typical one for the leather-metal contacts and the human-robot physical interaction.

During the *experiments*, the operator repeatedly applied the interaction force to 22 marked points located at the robot links #4,...,7. It is clear that such methodology allowed easily estimate "real" value of one of the unknown interaction parameters, namely the force application point $\mathbf{p}$. The second interaction parameter, the force $\mathbf{F}$, was generated by using a metal rod with a rubber tip. The latter allowed to achieve

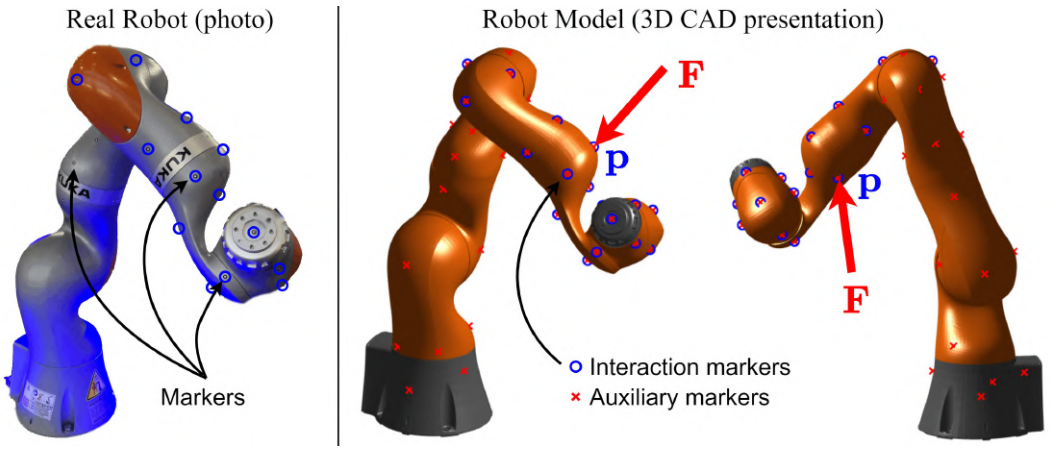


Figure 5.5: The collaboration robot KUKA iiwa with markers, which were used for both the robot surface modeling and the interaction force application. During the human-robot collaboration experiments, the force was repeatedly applied to the markers located at the four last links.

almost the "point" contact corresponding to a pure force without momentum. However, because of the human operator involvement, the exact direction and amplitude of the applied force were not measured explicitly and were not included in the experimental tables. The latter is not critical because the interaction force application point is essential for the safe interaction handling in collaborative robotics.

The experiments on human-robot collaboration were conducted for two groups of configurations. The first group considered the interaction parameters identification in non-singular cases, where the force was applied to the $6th$ robot link in three different manipulator postures. The second group included experiments for both singular and non-singular cases. Here the interaction force was applied by the operator to each of the 22 marked points located at the robot links #4,...,7 in 10 randomly chosen robot configurations, resulting in 220 measured instances.

The experimental procedure includes the following steps.

Step 1: Attach markers to the robot link surfaces ensuring that they are visible in different robot configurations.

Step 2: Using the ATOS 5 system, scan the robot link surfaces and export the obtained CAD model. Extract the markers positions from these data.

Step 3: Compress the robot surface model by reducing the number of mesh elements and by applying uniform remeshing.

Step 4: Begin the interaction experiments by moving the robot to the first configuration from the set of predefined ones.

Step 5: Using a special experimental tool, apply the interaction force to the first marker from the set of selected ones.

Step 6: Using the developed identification algorithms, estimate the interaction parameters ($\mathbf{F}$, $\mathbf{p}$) and compute the estimation errors.

Step 7: Repeat the interaction experiments for all markers from the considered set by iterating from **Step** 5.

Step 8: Repeat the interaction experiments for all robot configurations from the predefined set by iterating from **Step** 4.

It should be mentioned that the above experimental procedure includes a pre-measurements section (Steps 1-3) as well as two loops, which are repeated for all predefined robot configurations (Steps 4-8) and all selected interaction markers (Steps 5-7). Using this procedure, the experimental results were obtained that are presented and analyzed in the following subsection.

5.3.2 Analysis of the interaction identification experiments

To demonstrate the efficiency of the developed identification techniques, a number of real-life experiments were carried out that implement the experimental procedure described above. These experiments are divided into two groups, where the first one focuses on the comparison of the developed technique with the conventional method of De Luca [9], while the second group concentrates on the comparison of proposed enhanced versions of Approaches 1 and 2 from Chapter 4.

For the *first group* of experiments dealing with the comparison of the conventional and proposed algorithms, the non-singular cases were considered only. Here, because of the special selection of the manipulator posture as well as the interaction force direction and its application point located at the surface of the $6th$ link, the desired parameters are identified in a unique way. This allows to define the estimation error as the euclidean distance between the "real" force application point and the identified one. Relevant experimental results are presented in Table 5.1, which shows that the proposed algorithm allows to improve the accuracy from 1.8 up to 3.9 times compared to the conventional ones, depending on the manipulator configuration.

Table 5.1: Experimental results for estimation errors for non-singular cases

	Conventional method			**Proposed method**			
	δp_x $[mm]$	δp_y $[mm]$	δp_z $[mm]$	δp_x $[mm]$	δp_y $[mm]$	δp_z $[mm]$	$\frac{\|\|\delta\mathbf{p}^c\|\|}{\|\|\delta\mathbf{p}^p\|\|}$
Conf. 1	2.9	8.7	11.6	0.3	4.2	7.3	1.8
Conf. 2	3.6	3.3	7.1	2.2	0.2	0.1	3.9
Conf. 3	4.0	0.0	12.1	2.0	0.8	6.4	1.9

For the *second group* of experiments dealing with the comparison of the enhanced versions of Approaches 1 and 2, both the non-singular and singular cases were considered. In total, the experiments were conducted for 220 configurations, where the interaction force was applied at the robot links #4,...,7. Since the force was generated by the human operator, its exact amplitude and direction were unknown but the operator attempted to apply the force perpendicular to the link surface. It should be also mentioned here that the torque measurement data from the KUKA iiwa robot were quite noisy and do not fully satisfy the assumptions from Chapter 4. Besides, the joint torque measurements included some biases due to imperfections of the dynamic model in the relevant state observer. The identification accuracy was estimated using the same metrics as above, i.e the euclidean distances between the "real" parameter $\mathbf{p}$ and its identified value. Here, the "real" values were obtained using the marker coordinates to which the forces were applied to. For the identified values that in the singular cases may be non-unique, the closest one from the obtained solution set $\{\mathbf{p}\}$ was used. The algorithms time performance was evaluated by means of T_c, f_c metrics using a PC with Intel Core i5-4210H 3GHz CPU, 8Gb RAM. Besides, the $fail$ metric was also used to evaluate the percentage of configurations for which no solution was found.

The experimental results concerning the evaluation of *Enhanced Approach 1* are presented in Table 5.2. They show that here the estimation accuracy was about 9 mm on average, which is in good agreement with the considered mesh size of 10 mm. The corresponding σ-parameter of the obtained error distribution was 15.5 mm, which is relatively high but not critical for safe human-robot collaboration. The time performance of this technique is quite good, the average computing time was about 2.3 ms. The latter means that it is possible to update the interaction parameters estimates approximately 435 times per second. In the worst case, the update rate is 200 times per second only, which may be hardly acceptable for some collaborative robots. Also, it should be noted that the algorithm is slightly faster if the force is applied to the links that are closer to the end-effector, which obviously yields a smaller k-index range.

It is also worth mentioning that the first approach always provides zero *fail rate*, i.e. it yields some solutions even for critically corrupted torque measurements. The latter is achieved because this technique is based on the exhaustive enumeration of all mesh elements describing the robot surface. Besides, in one of the experiments, a rather large maximum estimation error of 206 mm was observed because the algorithm wrongly identified the link to which the interaction force was applied to. In addition, several computational experiments were carried out to estimate the mesh size influence on the identification accuracy. It was discovered that reducing the surface mesh size does not significantly improve the accuracy but it greatly affects the algorithm runtime.

Table 5.2: Experimental results of Enhanced Approach 1 with real robot

Param.	**Metrics**	**k-index**				**avg**
		7	**6**	**5**	**4**	
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	10.2	5.57	6.89	18.6	9.01
	std	9.05	6.72	17.9	18.1	15.5
	max	40.4	52.1	206.2	89.1	206.2
$\mathbf{T}_c$ $[ms]$	mean	0.98	1.89	2.29	3.28	2.29
	std	0.05	0.19	0.39	0.50	0.68
	min	0.92	0.99	1.86	2.73	0.92
	max	1.10	2.33	3.35	4.95	4.95
$\mathbf{f}_c$ $[Hz]$	mean	1012	527	434	304	435
fail %		0	0	0	0	0

Table 5.3: Experimental results of Enhanced Approach 2 with real robot

Param.	**Metrics**	**k-index**				**avg**
		7	**6**	**5**	**4**	
$\|\|\Delta\mathbf{p}\|\|$ $[mm]$	mean	4.54	7.20	5.23	9.71	7.09
	std	4.48	8.08	4.46	11.5	8.31
	max	13.4	57.3	27.3	69.7	69.7
$\mathbf{T}_c$ $[ms]$	mean	0.64	0.93	1.25	2.84	1.60
	std	0.21	0.42	0.82	0.96	1.12
	min	0.48	0.69	0.54	1.21	0.48
	max	1.52	3.82	7.86	7.02	7.86
$\mathbf{f}_c$ $[Hz]$	mean	1543	1070	797	351	621
fail %		0	2.81	0	0	0.88

Table 5.4: Comparison of Enhanced Approaches 1 and 2 with real robot

	mean $\|\|\Delta\mathbf{p}\|\|$	**max** $\mathbf{T}_c$	**mean** $\mathbf{T}_c$	**fail** %
Enhanced Approach 1	9.01	4.95	2.29	0
Enhanced Approach 2	7.09	7.86	1.60	0.9

Note: $\Delta\mathbf{p}$ - estimation error of the force application point;
$\mathbf{T}_c$, $\mathbf{f}_c$ - time performance of the estimation algorithm;
fail - no intersection cases, resulting in empty solution set.

The experimental results concerning the evaluation of *Enhanced Approach 2* are presented in Table 5.3. They show that here the estimation accuracy was high, about 7 mm on average. The corresponding σ-parameter of the obtained error distribution was 8.3 mm, which is relatively low. The time performance of this technique is also rather high, the average computing time was about 1.6 ms. The latter means that it is possible to update the interaction parameters estimates approximately 621 times per second on average but only 127 times per second in the worst case. Also, similar to the above, the computation is much faster if the interaction force is applied to the last links when the k-index range is smaller. In particular, the computing is almost $3\times$ faster if the force is applied to the $7th$ link.

It is also worth mentioning that in some hard cases, the second approach yields an empty solution set, which leads to the non-zero values of the *fail rate* shown in Table 5.3. The latter is caused by certain particularities of this approach, for which the torque measurement noise may affect the identified force action line in such a way that it can not satisfy the geometric constraints associated with the robot surface and admissible force direction. However, in practice, this difficulty can be overcome by applying the dedicated techniques proposed at the end of Chapter 4.

The *principal performances* of Approaches 1 and 2 are summarized in Table 5.4, which focuses on the accuracy, computing time as well as capability of singular cases handling. As follows from this comparison study, the second approach completely overcomes the first one in terms of the estimation *accuracy* of the interaction parameter $\mathbf{p}$, providing 21% more accurate identification on average. The corresponding σ-parameter of the obtained error distribution is also lower by up to 46 %. In addition, the average *computing time* of the second approach is 30 % better compared to the first one. In particular, its average cycle time of 1.6 ms (corresponding to 621 Hz) compared to 2.3 ms (435 Hz) for the first approach is more suitable for the real-time applications allowing integration in a typical robot control loop, which for collaborative robots usually is running with the frequency of 500 Hz. However, the maximum cycle time, which characterizes the worst-case scenario, is lower by 37 % for the first approach compared to the second one. It is worth mentioning that the *fail rate* of the second approach is less than 1 %, while the first one always provides the desired solution.

Both approaches were also evaluated for *capability of the singularity handling*. For the considered problem, the singularities arise either in some specific manipulator configurations (manipulator kinematic singularity) or in the case of some specific force directions and its application point locations (interaction force singularity). As follows from the experimental study, both approaches are capable of handling the singular cases producing multiple solutions for the desired interaction parameters $\{\mathbf{F}\}$, $\{\mathbf{p}\}$.

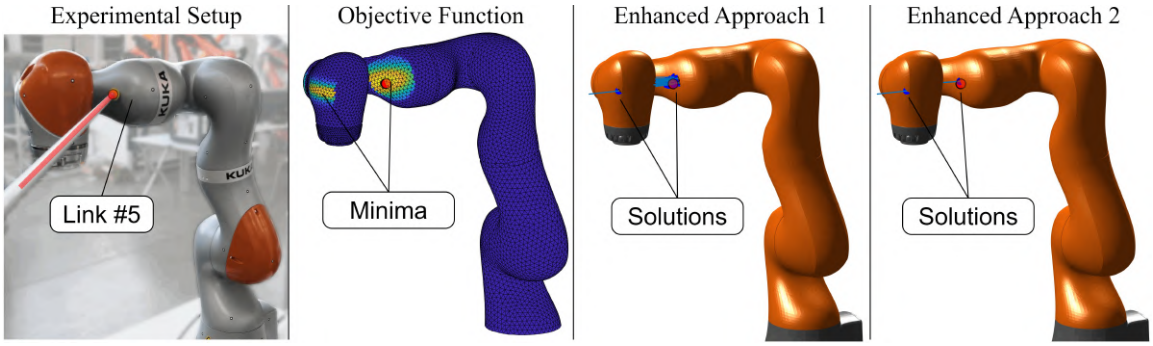


Figure 5.6: Experiment #1: identification of interaction applied for the $5th$ link.

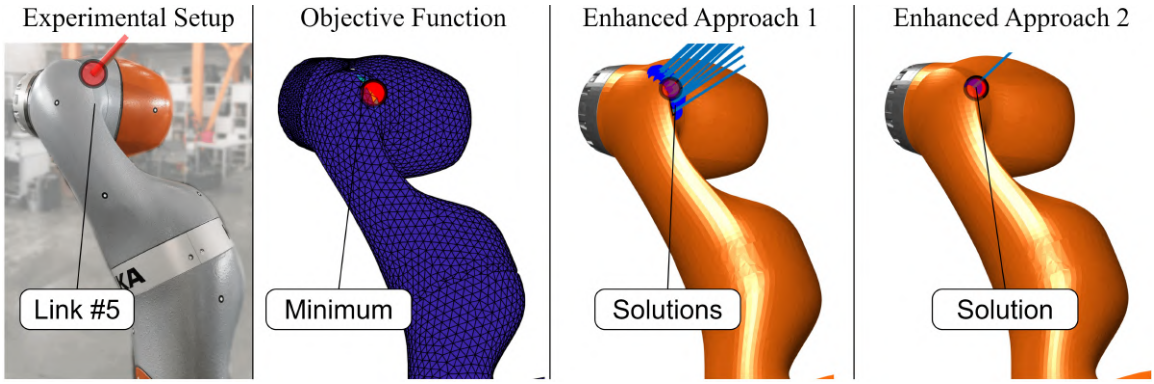


Figure 5.7: Experiment #2: identification of interaction applied for the $5th$ link.

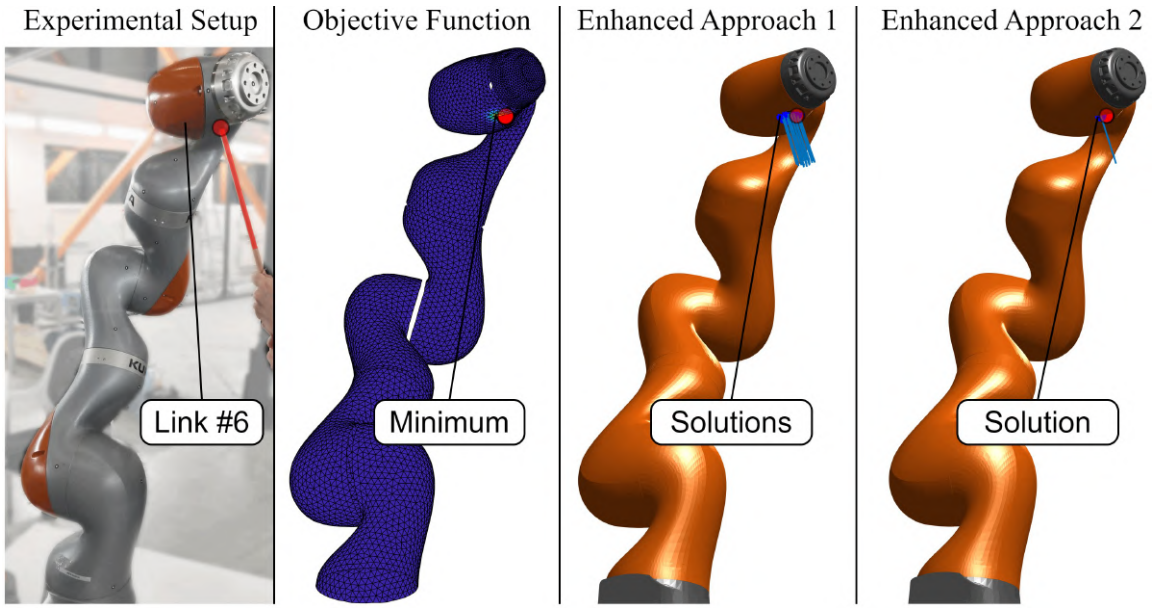


Figure 5.8: Experiment #3: identification of interaction applied for the $6th$ link.

This is the essential advantage of the proposed techniques compared to the existing ones analyzed in Chapter 2, which are able to provide a single solution only.

The **graphical illustrations** showing some details of the above experimental study are presented in Figs. 5.6 to 5.11. They cover both singular and non-singular cases and visualize the obtained interaction force and its application point as well as the colormaps of the relevant objective functions (1.18). In the experiment #1 presented in Fig. 5.6, the interaction force was applied to the $5th$ robot link in such a way that there are two minima of the objective function located at the $5th$ and $6th$ links. In this case, both proposed identification techniques successfully found these two minima, one of which corresponded to the "real" force application point. In the experiment #2 presented in Fig. 5.7, the force was also applied to the $5th$ link but here the objective function has a single minimum that yields a unique solution for the interaction parameters. In the experiment #3 shown in Fig. 5.8, the typical non-singular case was considered where the force was applied at the $6th$ link. In this case, the second technique successfully provided a unique solution, while the first one yielded several solutions in close proximity to the "real" one. In the experiment #4 presented in Fig. 5.9, the singular case was considered where the force was applied to the $5th$ link. In such a configuration, the solution should be unique but because of the ill-conditioned Jacobian, the first identification algorithm provided multiple "line"-type solutions, while the second algorithm yielded a unique solution. In the experiment #5 shown in Fig. 5.10, the hard singular case was considered where the interaction force was applied to the $4th$ link. Here, there are several minima of corresponding objective function that are located at the surfaces of the $4th$, $5th$ and $6th$ links. In this case, the first approach provided multiple solutions located at all three links, with the majority of them concentrated at the $6th$ link. In contrast, the second approach yielded a unique solution at the $6th$ link and a "line"-type multiple solutions at the $4th$ link, while the solutions at the $5th$ link were omitted since they violate the force direction constraint. Nevertheless, both approaches provided solutions located in close proximity to the "real" one. Finally, in the experiment #6 presented in Fig. 5.11, another hard singular was considered where the force was applied to the $4th$ link. Here, there are also several minima of corresponding objective function that are located at the surfaces of the different links. In this case, both approaches provided similar multiple solutions for the interaction parameters, which include the "real" one. An interesting feature of these multiple solutions is the geometrical structure of the obtained $\mathbf{F}$, $\mathbf{p}$ parameters, which looks like a "waving fan" that corresponds to the pencil of possible force action lines intersecting at the same point as was described in Chapters 3 and 4.

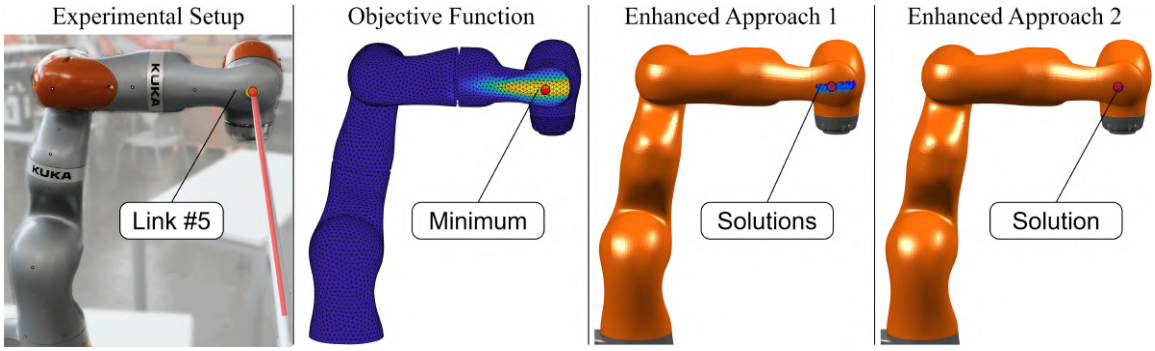


Figure 5.9: Experiment #4: identification of interaction applied for the $5th$ link.

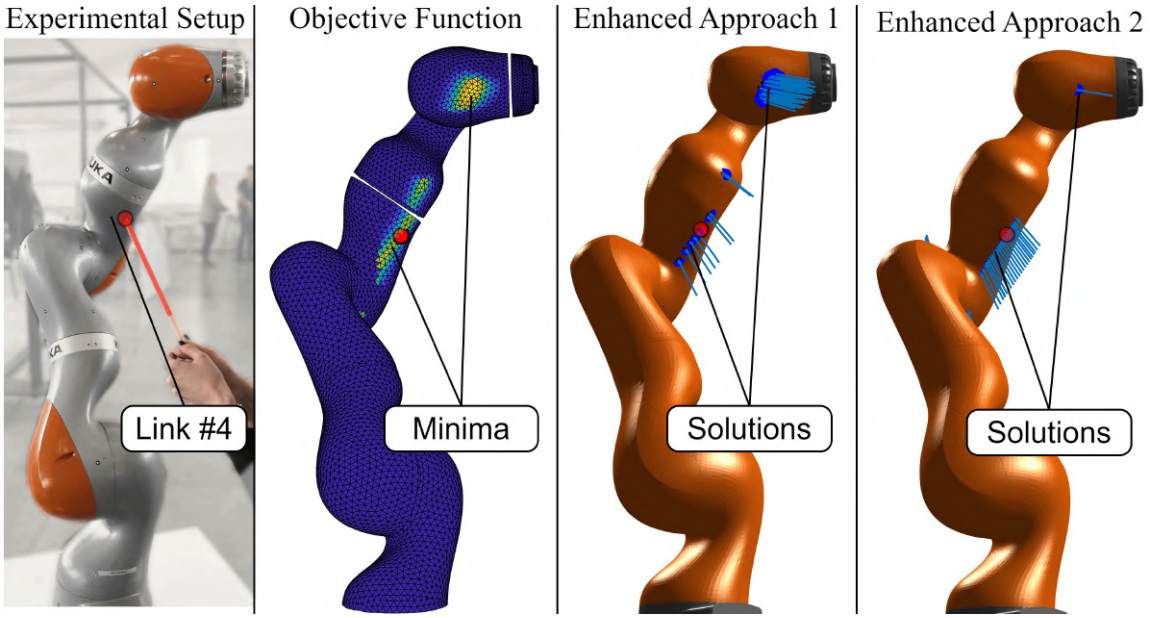


Figure 5.10: Experiment #5: identification of interaction applied for the $4th$ link.

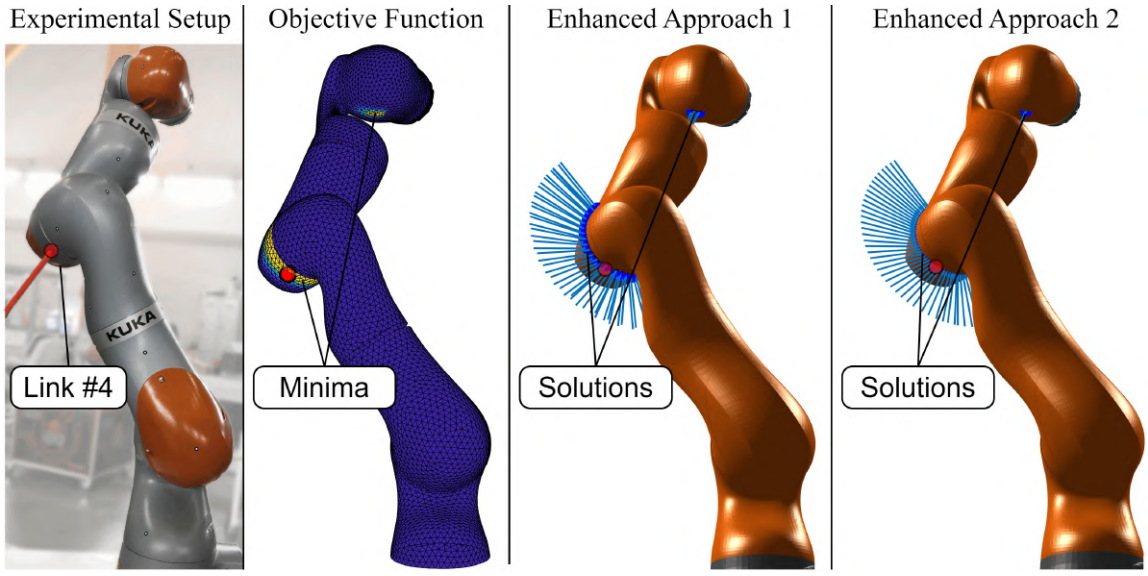


Figure 5.11: Experiment #6: identification of interaction applied for the $4th$ link.

Summary of this experimental study is presented in Table 5.5, where the proposed methods are compared with the known ones. Besides, this table evaluates the achieved performances with the required ones given at the end of Chapter 2 and shown in Table 2.3. This table allows to make the following conclusions validating the obtained results:

1) The **accuracy** of the proposed techniques is rather high. For the torque measurements data obtained from the real robot, Approaches 1 and 2 provided the position error of about 9 mm and 7 mm correspondingly. Such accuracy is acceptable for most practical applications in the field of human-robot collaboration.

2) The **real-time performance** of the developed algorithms is also better compared to the existing ones. In the considered experimental study, Approaches 1 and 2 allowed to find the desired interaction parameters in 2.3 ms and 1.6 ms on average. This also satisfies the real-time requirements of the human-robot interaction handling, where the response time is critical because of safety reasons.

3) The **singularity handling** capability of the developed techniques is their essential advantage. They allow to identify the desired interaction parameters even in some hard and ambiguous singular cases. This useful feature definitely contributes to the safe and efficient human-robot collaboration.

From a practical point of view, it is worth mentioning that both Approaches 1 and 2 are suitable for industrial applications. However, Approach 1 is preferable if the statistical properties of the measurements noise are unknown since this technique is essentially more robust while slightly slower and less accurate. Otherwise, if the noise properties are known, then Approach 2 provides some benefits. It allows to achieve a 1.2x improvement in accuracy for interaction parameters estimation and also a 1.4x speedup in computation time.

Table 5.5: Experimental results confirming efficiency of proposed approaches

Approach	**Robot**	**Δp**	**T_c**	**SH**
Proposed approaches				
Enhanced Approach 1 [Chapter 4]	KUKA iiwa	**0.9**	**1/435**	+
Enhanced Approach 2 [Chapter 4]	KUKA iiwa	**0.7**	**1/621**	+
Surface approximation approaches				
Cylinder approx. [Section 2.2]	KUKA iiwa	5.4	1/19	–
Sphere mapping [Section 2.4]	KUKA iiwa	4.2	1/100	–
DIRECT [78]	KUKA iiwa	6.0	1/30	–
DIRECT-link [57]	Kinova Jaco2	12.0	1/20	–
Monte Carlo based approaches				
PF on graph [Section 2.5]	KUKA iiwa	4.0	1/60	–
Contact particle filter [87]	KUKA iiwa	2.4	1/10	–
PF with FC binning [88]	Kinova Jaco2	11.0	1/63	–
PF with binning [88]	Kinova Jaco2	12.0	1/71	–
PF with nearest neighbor [88]	Kinova Jaco2	11.0	1/125	–
PF with weighted means [88]	Kinova Jaco2	11.0	1/159	–
Machine learning based approaches				
Feed-forward NN [Section 2.3]	KUKA iiwa	6.4	1/180	–
Random forest [57]	Kinova Jaco2	8.0	1/200	–
Multilayer perceptrons [57]	Kinova Jaco2	4.0	1/200	–
Clustered surface approach				
Clustered surface [Section 2.6]	KUKA iiwa	2.3	1/600	$\pm$

Note: **Δp** - estimation error of the force application point $[cm]$;
T_c - cycle time of the estimation algorithm $[1/Hz]$;
SH - singularity handling capabilities of the algorithm.

5.4 Practical application of adaptive interaction handling

This section presents the practical application of the developed adaptive interaction handling strategy, which integrates both the proposed identification techniques and classification methods. Particular attention is paid to the human-robot collaboration safety, the robot behaviors modes, and switchings between them depending on the interaction type. The general scheme of high-level controller ensuring safe human-robot interaction is described. The presented practical example deals with a non-contact collaborative process jointly executed by the robot and the human operator, where some physical interactions are possible. In this example, all robot behavior modes and all interaction classes from Section 5.2 are considered and analyzed.

5.4.1 Implementation of the adaptive interaction handling controller

The developed high-level robot controller ensuring safe human-robot collaboration is based on a modular structure. It consists of two independent parts, where the first one was implemented inside the original robot controller of KUKA iiwa (internal part) and the second one was realized on the external computer. The communication between the robot controller and external computer was executed by means of ROS (Robot Operating System), which allows to receive the sensor measurements from the robot and sends to it the motion commands. Here, we used the advantages of KUKA iiwa controller that supports high-level Java programming language, in contrast to the conventional robotic controllers for which highly-specialized languages (KRL, Karel, etc.) are available only. This allowed to connect the robot controller Java Core with the external ROS core by means of the open source ROS package *iiwa_stack* [80]. It should be also mentioned that some modifications were done to the original *iiwa_stack* in order to obtain required data from the sensors or the state observers as well as provide online adjustment of the joint compliances.

The general structure of the controller is presented in Fig. 5.12. Its external part consists of five main modules: Communication, Identification, Classification, Finite State Machine, and Reaction Library. The *Communication* module creates the ROS core and transforms the robot controller data into a more convenient form. In addition, this module sends the commands to the robot, which define the desired robot joint angles and compliances. The *Identification* module uses the robot joint torques and angles measurements to detect the physical interaction and to identify its parameters

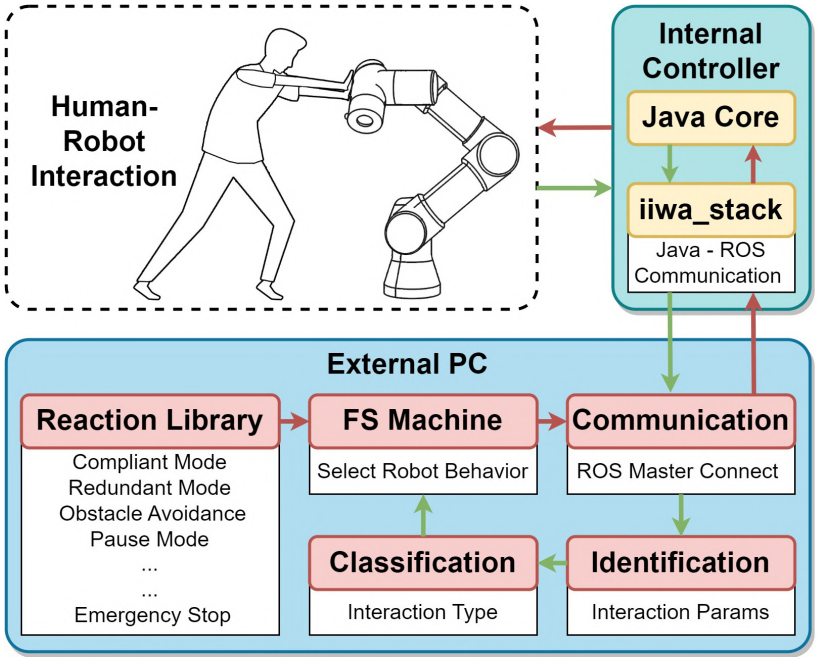


Figure 5.12: General structure of developed adaptive interaction handling controller. It consists of two independent parts, where the first one was implemented inside of the original robot controller and the second one was realized on the external computer.

($\mathbf{F}$, $\mathbf{p}$) using the developed algorithms presented in Chapter 4. The *Classification* module attributes the detected interaction to one of the predefined classes listed in Fig. 5.2 and described in detail in Section 5.2; in case of Soft/Hard classification, the deep neural network is used. Further, the interaction parameters and characteristics are transmitted to the *Finite State Machine* presented in Fig. 5.1 and described in Section 5.2. This machine defines the safe robot behaviors modes, using its states and transition between them as functions of the interaction parameters and properties obtained from the Identification and Classification modules. The set of all possible robot reactions is stored in the *Reaction Library* module, which predefines the safe robot behaviors. It should be mentioned that each part of the high-level controller is implemented as a separate module, so it is quite flexible and can be easily used to enlarge a variety of desired robot behaviors. In particular, by redefining the set of robot behavior modes/reactions and their transitions, the operator can adapt the controller to any particular industrial process that assumes intensive human-robot collaboration.

The external part of the above controller was implemented mainly in *C++* language, while the identification module was compiled as a library from the *MATLAB* functions using MATLAB Coder, and the Soft/Hard classification module was implemented in *Python* language using Keras/TensorFlow libraries. The internal part of the controller was written in *Java* language and included a number of modules from the open source software *iiwa_stack* with relevant modifications.

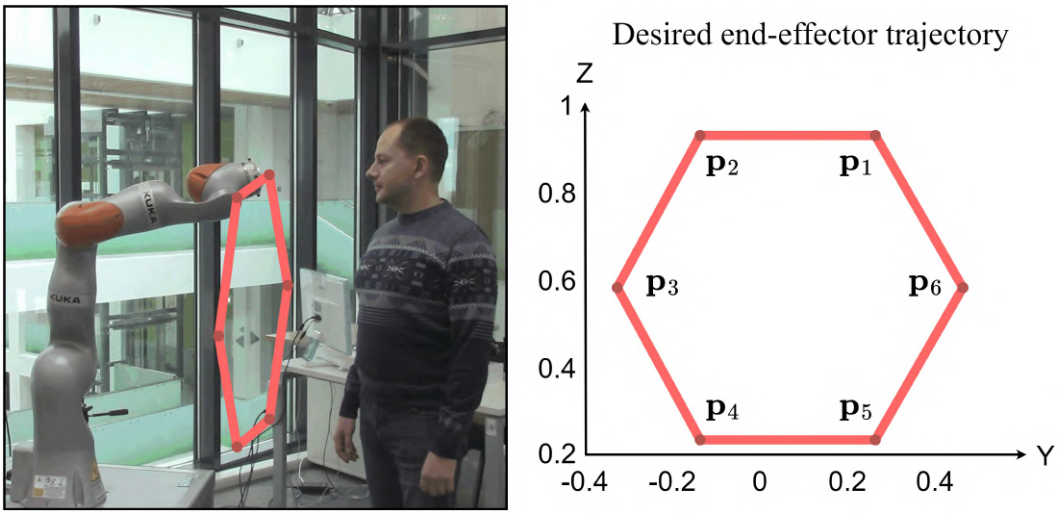


Figure 5.13: Experiment setup and desired end-effector trajectory.

The developed interaction handling controller proved to be reliable, ensured safe human-robot collaboration in our experimental study, and was successfully used in laboratory experiments validating the proposed techniques. Relevant results are presented in the following subsection.

5.4.2 Validation of proposed interaction handling technique

Let us present an application example, which confirms the efficiency of the developed interaction handling technique for real-life human-robot collaboration. In this experimental study, the robot was programmed to follow the desired hexagon trajectory shown in Fig. 5.13. While programming the robot for this task, it was assumed that normally any physical interactions may not occur, which corresponds to the Non-Contact Operation Mode of the developed Finite State Machine. Nevertheless, to ensure safe human-robot collaboration, some additional properties were assigned for each trajectory segment, which admits some unexpected interaction with the operator, either Accidental (Accd) or Intentional (Intl) ones. In particular, the complete trajectory was composed of four closed loops $\mathbf{p}_1 \rightarrow \mathbf{p}_2 \rightarrow \cdots \rightarrow \mathbf{p}_6 \rightarrow \mathbf{p}_1$, where the first two loops treated all detected interactions as Accidental (Accd) ones and the remaining loops treated them as Intentional (Intl) interactions. It should be noted that in this experimental study, the general robot behavior was controlled by the Finite State Machine described in Section 5.2.

During the experiment, while the robot followed its desired trajectory, some interactions were created with their force applied either at the robot end-effector (Tool) or its intermediate links (Link). In addition, to verify the developed controller ability to distinguish Soft and Hard interactions, some forces were applied by the human oper-

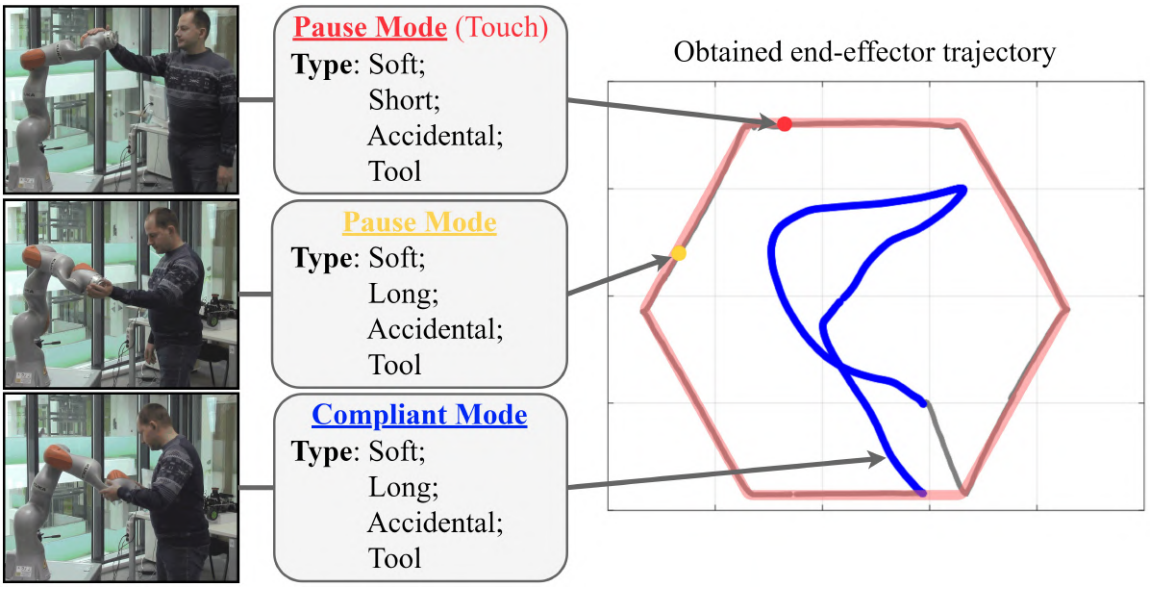


Figure 5.14: Stage #1 of the experimental study corresponded to the first loop of the robot end-effector trajectory. Here, all interactions were considered as accidental and the robot interacted with the human operator.

ator (Soft) while other ones were caused by the collision with a rigid object (Hard). Besides, the interaction duration was also different (Short/Long), in order to evaluate the reasonableness of the controller reactions.

The experimental setup included the collaborative robot KUKA LBR iiwa 14, which was connected to the external PC with Intel Core i5-4210H 3GHz CPU, 8Gb RAM. The high-level interaction handling controller was composed of two parts. The external part was based on the PC with a Linux operation system, ensuring fast communication with the internal part. In this controller, the Identification and Classification module implements the main theoretical contributions of this thesis. In general, the developed control system is capable of real-time execution of the desired robot motion program, while ensuring the maximum exchange rate between the internal and external parts of 800 times per second. However, in this application example, the exchange rate was reduced down to 100 times per second. The latter was caused by the limitations of the communication bandwidth and the necessity to obtain some additional service information from the robot. It is also worth mentioning that this experimental study involved the direct physical interaction between the human operator and the robot, which must be definitely safe even in the case of unexpected robot behavior during experimentation. So, to eliminate the potential harm, the maximum stiffness of the robot joints were deliberately set to some low values that caused the increased trajectory tracking errors. It is clear that for real-life industrial applications such stiffness modification is not required and the tracking errors will be essentially low.

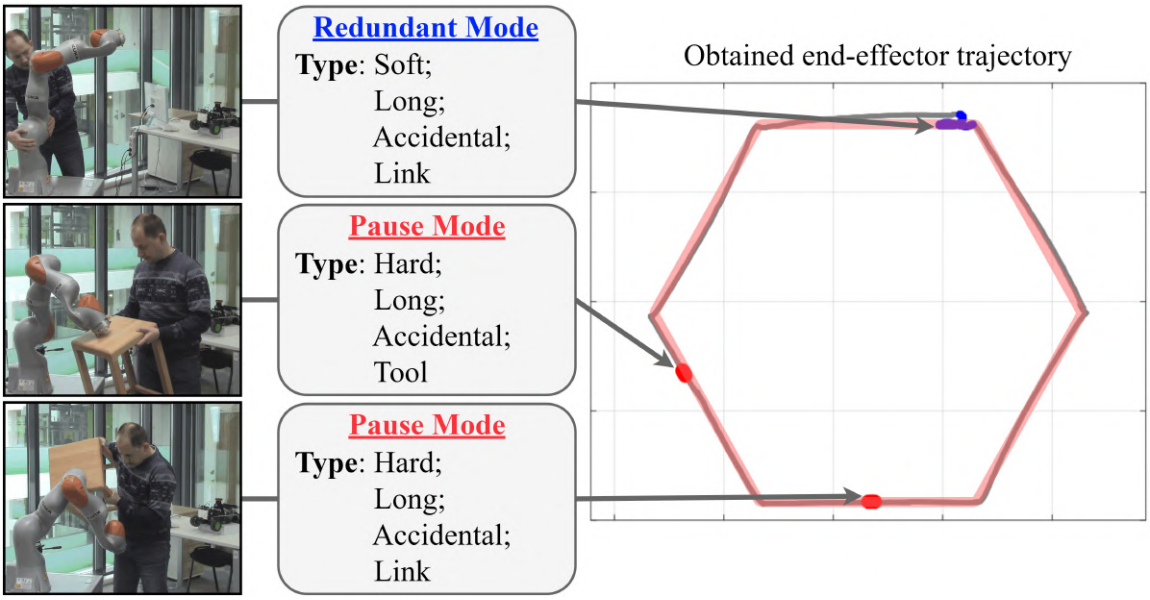


Figure 5.15: Stage #2 of the experimental study corresponded to the second loop of the robot end-effector trajectory. Here, all interactions were considered as accidental and the robot interacted with the human or environment.

Another important issue related to this experimental study is the natural operator behavior, who was allowed to apply the interaction force to any robot link or even to interact with the robot using both his hands. The latter is not in good agreement with the principal assumptions implemented in the Identification and Classification modules, where a single interaction with the links #3...7 was admitted. Nevertheless, the robot behavior observed in the experiments was safe and the interactions were handled in the expected way according to the relevant state machine. The video of the experiment is available at (https://youtu.be/kYldqaoEcZM).

The experimental study was composed of four stages corresponding to four closed loops $\mathbf{p}_1 \rightarrow \mathbf{p}_2 \rightarrow \cdots \rightarrow \mathbf{p}_1$ of the robot end-effector trajectory presented in Fig. 5.13. At each stage, assuming that Non-Contact Mode was active, the collaborative robot was subject to physical interactions of different types (Soft/Hard, Short/Long, Accd/Intl, Tool/Link). In more detail, these stages covering all possible combinations of the interaction types are described below.

The *Stage #1* is presented in Fig. 5.14, it corresponds to the first loop of the desired robot end-effector trajectory. Here, three interactions were detected, and all of them were considered as accidental (Accd). The first one was generated by the operator who briefly applied the force at the robot end-effector. It was recognized as Soft/Short/Accd/Tool, so the controller reaction was to stop the robot (Pause Mode). The related end-effector position is shown in the figure with a red dot. The subsequent brief human touch on the end-effector resumed the robot motion. The second

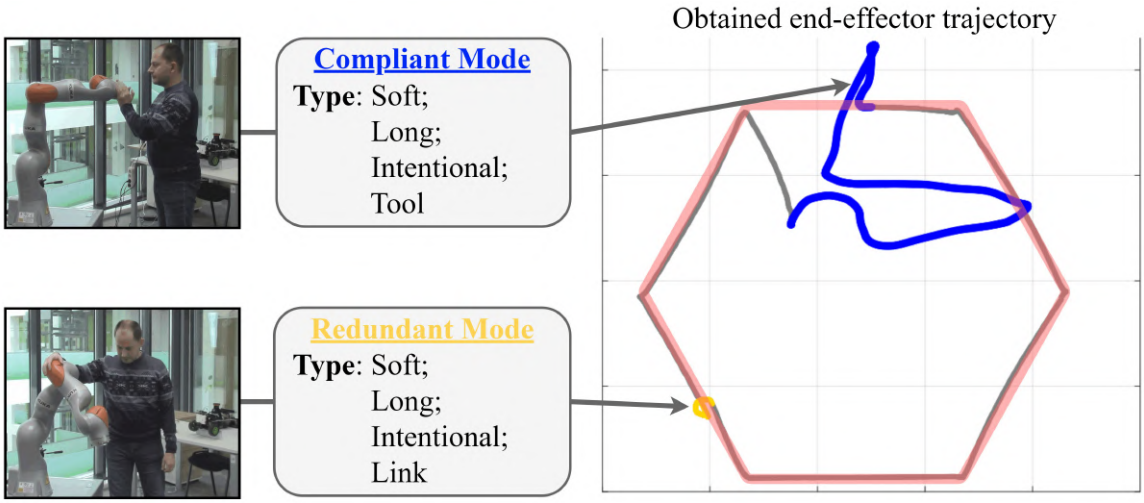


Figure 5.16: Stage #3 of the experimental study corresponded to the second loop of the robot end-effector trajectory. Here, all interactions were considered as intentional and the robot interacted with the human operator.

interaction was also generated by the operator who grasped the robot end-effector for a certain time. It was recognized as Soft/Long/Accd/Tool, so the controller reaction was to stop the robot as well (Pause Mode). The related end-effector position is shown in the figure with a yellow dot. In this case, the robot motions were suspended until this interaction disappeared. The third interaction was generated by the operator, who not simply grasped the robot end-effector but tried to move it away. It was recognized as Soft/Long/Accd/Tool, so the controller reaction was to stop the trajectory tracking and essentially increase the joint compliances (Compliant Mode). Related end-effector trajectory is shown in the figure with a blue color and was clearly determined by the human operator. In contrast to the previous interaction, here the force was applied for a certain time, its amplitude continuously increased and the robot was switched to the compliant mode when the force exceeded some predefined threshold.

The *Stage #2* is presented in Fig. 5.15, it corresponds to the second loop of the desired trajectory. Here, three interactions were detected, all of them were considered as accidental (Accd) but in contrast to the previous stage, some of them were caused by a collision with a rigid object. The first interaction was generated by the operator, who grasped the robot lower link and tried to move it away. It was recognized as Soft/Long/Accd/Link, so the controller reaction was to use the robot kinematic redundancy to continue the desired trajectory (Redundant Mode). Related end-effector trajectory is shown in the figure with a blue line, which is close to the desired path. Here, the obtained trajectory slightly differed from the desired one since the joint compliances were set to rather high values. The second interaction was caused by the collision between the robot end-effector and a rigid obstacle on its path. It was rec-

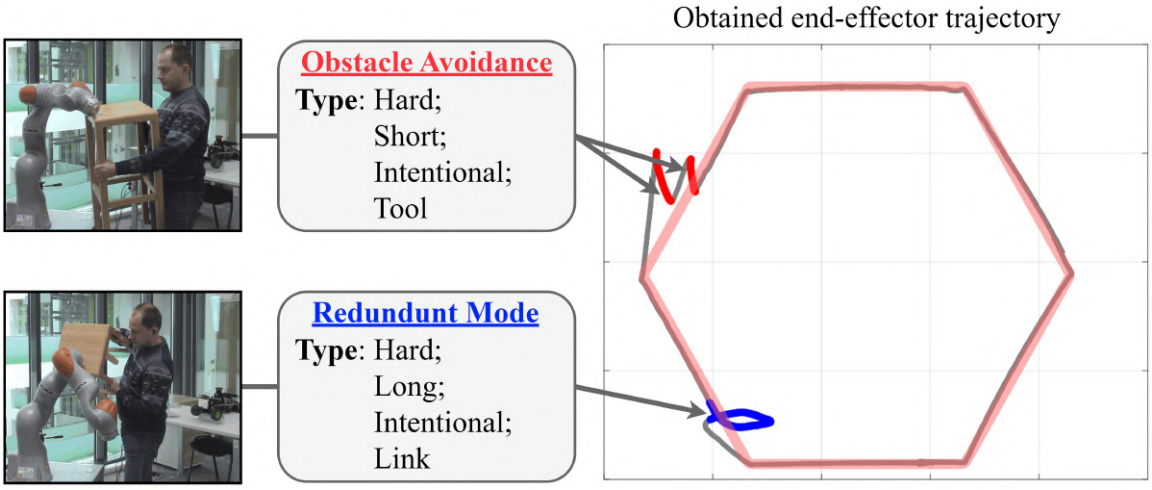


Figure 5.17: Stage #4 of the experimental study corresponded to the second loop of the robot end-effector trajectory. Here, all interactions were considered as intentional and the robot interacted with its environment.

ognized as Hard/Long/Accd/Tool, so the controller reaction was to stop the motions (Pause Mode). Related the end-effector position is shown in the figure with a red dot. In this case, the robot motions were suspended until the interaction force disappeared. The third interaction was also caused by the collision between the robot $3rd$ link and the rigid obstacle on its path. It was recognized as Hard/Long/Accd/Link, so the controller reaction was to stop the robot (Pause Mode) until the force disappeared.

The *Stage #3* is presented in Fig. 5.16 and corresponds to the third loop of the desired trajectory. Here, two interactions were detected only, all of them were considered as intentional (Intl). The first interaction was generated by the operator who grasped the robot end-effector and tried to move it away. It was recognized as Soft/Long/Intl/Tool, so the controller reaction was to stop the trajectory tracking and essentially increase the joint compliances (Compliant Mode) in order to follow the human operator hand guiding. Related end-effector trajectory is shown in the figure with a blue color. Here, the force was applied for a certain time, its amplitude continuously increased and the robot was switched to the compliant mode when the force exceeded some predefined threshold. The second interaction was also generated by the operator, who implemented a similar scenario but the interaction force was applied to the robot $4th$ link. It was recognized as Soft/Long/Intl/Link, so the controller reaction was to use the kinematic redundancy to continue the desired trajectory (Redundant Mode). The related end-effector trajectory is shown in the figure with a yellow line.

The *Stage #4* is presented in Fig. 5.17 and corresponds to the fourth loop of the desired trajectory. Here, three interactions with the rigid objects were detected, and all of them were considered as intentional (Intl). The first and second interactions

were caused by the collision between the robot end-effector and rigid obstacles on its path. They both were recognized as Hard/Short/Intl/Tool, so the controller reaction was to avoid the obstacle by changing the robot path (Obstacle Avoidance Mode). Relevant end-effector trajectories are presented in the figure with red lines, showing how the robot avoids the obstacle. The third interaction was caused by the collision between the robot $3rd$ link and a rigid obstacle on its path. It was recognized as Hard/Long/Intl/Link, so the controller reaction was to use the robot kinematic redundancy to continue the desired trajectory (Redundant Mode). A relevant trajectory is shown in the figure with a blue line, which is close to the desired path.

It should be noted that the above-described experiments cover all essential combinations of the interaction types, which may be generated from possible options Soft/Hard, Short/Long, Accd/Intl, Tool/Link. In fact, in spite of that, the number of all possible combinations is $2^4 = 16$, and only 8 of them are different from the practical point of view. For this reason, the experimental study considered a lower number of sub-cases related to the interaction types.

Thus, the presented experimental study confirms *validity* of the developed interaction handling techniques. The relevant high-level controller implements the proposed identification algorithms as well as the classification methods, which allows adapting the robot behavior to the detected interaction. In particular, depending on the interaction parameters and type, the robot may be switched to one of the predetermined modes ensuring safe human-robot collaboration while executing the manufacturing task.

5.5 Summary of the chapter

This chapter is devoted to the development of an adaptive technique for handling physical interactions between the human and the robot, as well as experimental validation of the developed identification algorithms. Particular attention has been paid to the safety issues of the human operator and performance improvement of the human-robot collaboration. In the scope of the interaction handling development, the identification methods proposed in the previous chapter were used for the selection of the appropriate reaction robot strategy. These parameters define the interaction force amplitude/direction and its application point on the robot surface that are used for the interaction classification within the set of predefined categories. Based on these categories and the current robot state, the algorithm chooses the relevant robot behavior mode. At the end of this chapter, the experimental results are presented that were obtained for the real collaborative robot KUKA LBR iiwa and validate the developed techniques.

In more detail, the result of this chapter can be summarized as follows:

1) The developed *adaptive interaction handling controller*, based on the finite state machine, provides safe robot interaction with the operator and the environment. This controller allows to change the robot behavior depending on the detected interaction type and parameters. In particular, the controller distinguishes the Short and Long, Soft and Hard, Tool and Link, Intentional and Accidental interactions, which cause switching between several predefined robot behavior modes.

2) The *experimental validation* of the interaction parameters identification techniques is presented. The relevant experimental setup consisted of the real collaborative robot KUKA LBR iiwa 14 and the measurement device GOM ATOS 5. Particular attention was paid to the identification accuracy, time performance, and ability to handle singular cases. The achieved identification results for both non-singular and singular cases show that the proposed approaches overcome the existing ones by all of the considered performance measures.

3) The *practical application* of the adaptive interaction handling controller proved the validity of the developed techniques. The relevant high-level controller integrates the proposed interaction identification algorithms as well as the classification methods. In this controller, several robot behavior modes are predefined and switchings between them are executed depending on the parameters and characteristics of the detected interaction. The latter allows to ensure the safety of the human operator during the collaboration process with the robot while executing the manufacturing task.

It should be mentioned that the adaptive interaction handling technique presented in this chapter focuses on the laboratory prototype, which is not targeted to a particular industrial or manufacturing task. However, by redefining the set of robot behavior modes/reactions and their transitions, the user can easily adapt the controller to any desired industrial process that assumes intensive human-robot collaboration.

CONCLUSION

Contributions of the thesis

The thesis focuses on the development of new techniques for modeling the physical interaction between a human and a robot with adaptive compliance in order to improve human safety and human-robot workcell performance. Special attention was paid to the accuracy issues and the interaction parameters identification in singular cases, which can arise during the physical interaction because of limited measurement information provided by robot torque sensors.

The most essential results and contributions can be summarized as follows:

1) *The new approaches for the interaction parameters identification in human-robot collaboration, allowing to find the interacting link and estimate the interaction force and its application point using the joint torque measurements only for both singular and non-singular cases.* These approaches are based on the analytical solution of the corresponding system of static equilibrium equations with relevant geometrical constraints. The latter allowed to reduce significantly the computing time, compared to the conventional approaches based on straightforward numerical optimization. The developed methods suit well for real-life industrial applications where the response time is critical because of safety reasons. In addition, the methods are capable of producing multiple potential solutions, which may arise when the interaction force is applied to the lower links or the robot in a kinematically singular configuration.
2) *Identification accuracy improvement technique allowing to reduce the deviation of the obtained parameters from the real ones caused by the measurements noise.* The performance of the developed techniques for the interaction parameters identification in both singular and non-singular cases is analyzed in the presence of measurement noise. The deviation of the parameter estimates is computed using the statistical properties of the sensors' noise. Based on obtained results, some enhancements are proposed in order to make these approaches usable in real-life applications. Besides, essential reduction of the identification errors is achieved by means of the weighted least squares and introducing the simplified version of the original static equilibrium equations, which is linearized in the neighborhood of the approximated solution.

3) *Adaptive interaction handling method covering all phases of the interaction processing and allowing changing the robot behavior in order to improve the safety and efficiency of the human-robot collaboration.* The developed interaction handling controller is based on the finite state machine and provides safe robot physical interaction with the operator and the environment. The obtained high-level controller successfully integrates the interaction identification algorithms as well as the interaction classification methods. The latter allows to define several robot behavior modes and switches between them based on the interaction parameters and characteristics. In this case, the robot behavior is separated into the autonomous and interaction handling modes, where the first one is concentrated on efficient task execution, and the second mode was focused on the human safety.

The obtained theoretical results have been validated via the experimental studies that deal with the human interacting with the collaborative robot KUKA LBR iiwa 14. In the experimental validation of the interaction parameters identification techniques, particular attention was paid to the identification accuracy, time performance, and ability to handle singular cases. The achieved identification results for both non-singular and singular cases show that the proposed approaches overcome the existing ones by all of the considered performance measures. The practical application of the adaptive interaction handling controller proved the validity of the developed techniques. The obtained high-level controller successfully integrates the interaction identification algorithms as well as the interaction classification methods. In this controller, several robot behavior modes are defined and switchings between them are done based on the interaction parameters and characteristics. The latter allows to ensure the safety of the human operator during the collaboration process with the robot.

In practice, the contributions of this thesis can be used to improve human safety in the presence of a collaborative robot. The developed interaction identification techniques allows to continuously monitor the interaction force vector amplitude for avoiding violation of safe interaction values. If it is violated, the adaptive interaction handling algorithm allows to change the robot behavior to prevent injury by utilizing different scenarios and dynamical switching between them. Moreover, the real-time capabilities of the developed techniques allow to almost instantly react to any potentially unsafe situations.

The developed methods could not only improve the safety of the human in the robot workspace but also increase the performance of the workcell by activating different collaboration scenarios. In particular, the identified interaction parameters can be used for communicating between the human and the robot by enabling special collaboration scenarios, depending on how and where the human touches the robot. Besides, the

same parameters can be used for the compensation of compliance errors, which can increase the accuracy of the robot in the presence of the interaction force, improving the overall performance of the robotic work cell.

Limitations of obtained results

In spite of several essential advantages, there are still some limitations in the obtained results that are presented below:

1) It was assumed that the internal robot joint torque sensors are available only. In this work, to identify the interaction parameters the measurements from the robot built-in sensors were used that currently are one of the most popular sensors in collaborative robotics. However, conventional industrial robots, which are widely used in manufacturing, usually are not equipped with such sensors, limiting the application area of the developed techniques.
2) It was assumed that the robot has a serial kinematics structure. In the case of parallel kinematics, the relevant system of static equilibrium equations is more complicated because of passive joints, which further increases the rank deficiency of the considered system. However, the application of parallel robots for the human-robot collaboration nowadays is quite limited.
3) It was assumed that there is only a single interaction between the human and the robot. In general, for multiple simultaneous interactions, it is not possible to identify reliably the desired interaction parameters. This issue is connected to insufficient sensory information of a typical collaborative robot equipped with the joint torque sensors.
4) It was assumed that there are no external vision systems or proximity sensors available. Here, the developed interaction handling controller can change the robot behavior after the interaction only. In such circumstances, it is impossible to use the robot obstacle avoidance techniques before the physical interaction since the obstacle is not visible to the robot. This issue can affect the safety of the human in the robot workspace.

Nevertheless, in the frame of the considered application area, the limitations related to the model assumptions are not critical because the desired accuracy has been achieved in the experimental study. Moreover, it is possible to extend/modify the developed approach further in order to apply other techniques, which only improve the overall accuracy of the identification process.

Perspectives and future work

To generalize the obtained results and increase its application area, it is reasonable to continue research in several directions and to concentrate on the following issues:

1) Enhancement of the developed interaction identification technique for the case of additional internal or external sensors. Here, the external sensors could include vision-based systems or tactile skin applied to the robot surface. The internal ones could be based on additional multi-axis force/torque sensors installed inside of the robot links. By adding such sensors, the identification problem is no longer singular and usually provides a unique solution for the desired parameters.
2) Developing the interaction handling controller for a specific manufacturing process. In this work, the controller was focused on the laboratory prototype but adapting it for the real industrial processes would allow to improve the safety and efficiency of the related workcell. For example, assembly task is quite popular for the collaborative robots and can be used as a target for developing a new interaction handling controller for this particular operation.
3) Extending the application area of the interaction parameters identification technique for different types of robots. In particular, it is reasonable to extend this technique to humanoid robots, that are typically used in the contact-rich environment. Thus, implementing a relevant interaction identification technique can be useful to improve their performance.

Publications of main results

Journal papers:

1. **D. Popov**, A. Klimchik, A. Pashkevich, "Robustness of interaction parameters identification technique for collaborative robots", *IEEE Robotics and Automation Letters (RAL)*, 2022, Vol. 7, no. 4, Pp. 8582-8589. Also presented at *IEEE International Conference on Intelligent Robots and Systems (IROS)*.
2. **D. Popov**, A. Klimchik, A. Pashkevich, "Real-time estimation of multiple potential contact locations and forces", *IEEE Robotics and Automation Letters (RAL)*, 2021, Vol. 6, no. 4, Pp. 7025-7032. Also presented at *IEEE International Conference on Automation Science and Engineering (CASE)*.
3. **D. Popov**, A. Klimchik, "The use of joints force sensors to determine the collision location and type for an industrial robot", *Mechatronics, Automation, Control (MAU)*, 2019, Vol. 20, no. 3, Pp. 171-179.

4. **D. Popov**, A. Pashkevich, A. Klimchik, "Identification of human-robot interaction in a collaborative workcell: singularities of closed form solution for a planar case", *Robotics and Computer-Integrated Manufacturing (RCIM)*, under review, submitted in August 2022.
5. **D. Popov**, A. Pashkevich, A. Klimchik, "Adaptive technique for physical human-robot interaction handling using proprioceptive sensors", *Engineering Applications of Artificial Intelligence (EAAI)*, under review, submitted in May 2023.

Conference papers:

6. **D. Popov**, A. Klimchik, A. Pashkevich, "Analytical estimation of interaction force and its application point for collaborative robots", *IFAC Conference on Manufacturing Modeling, Management and Control (MIM)*, 2022, Vol. 55, no. 10, Pp. 2451-2456.
7. **D. Popov**, S. Mikhel, R. Yagfarov, A. Klimchik, A. Pashkevich, "Multi-scenario contacts handling for collaborative robots applications", *IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2021, Pp. 2985-2992.
8. **D. Popov**, A. Klimchik, N. Mavridis, "Collision detection, localization & classification for industrial robots with joint torque sensors", *IEEE Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2017, Pp. 838-843.
9. **D. Popov**, A. Klimchik, "Transfer learning for collision localization in collaborative robotics", *Proceedings of the International Conference on Applications of Intelligent Systems (APPIS)*, 2020, Pp. 1-7.
10. **D. Popov**, A. Klimchik, "Real-time external contact force estimation and localization for collaborative robot", *IEEE International Conference on Mechatronics (ICM)*, 2019, Pp. 646-651.
11. **D. Popov**, A. Klimchik, "Multi-collision detection for collaborative robot", *IEEE Dynamics of Complex Networks and their Application in Intellectual Robotics (DCNAIR)*, 2019, Pp. 145-148.
12. **D. Popov**, S. Mikhel, A. Klimchik, "Scenarios for physical robot-environment interaction", *IEEE Dynamics of Complex Networks and their Application in Intellectual Robotics (DCNAIR)*, 2018, Pp. 109-111.
13. S. Mikhel, **D. Popov**, S. Mamedov, A. Klimchik, "Development of typical collision reactions in combination with algorithms for external impacts identification", *IFAC Conference on Manufacturing Modeling, Management and Control (MIM)*, 2019, Vol. 52, no. 13, Pp. 253-258.
14. S. Mikhel, **D. Popov**, A. Klimchik, "Collision driven multi scenario approach for human collaboration with industrial robot", *International Conference on Mechatronics and Robotics Engineering (ICMRE)*, 2018, Pp. 78-84.

15. S. Mikhel, **D. Popov**, S. Mamedov, A. Klimchik, "Advancement of robots with double encoders for industrial and collaborative applications", *IEEE Conference of Open Innovations Association (FRUCT)*, 2018, Pp. 246-252.

RESUME EN FRANÇAIS

MODÉLISATION DES INTERACTIONS PHYSIQUES ENTRE UN HUMAIN ET UN ROBOT AVEC ÉLASTICITÉ ADAPTATIVE

Introduction générale

Motivation

L'interaction homme-robot est l'un des sujets de recherche essentiels en robotique avec des méthodes allant du traitement visuel et audio à l'identification directe des contacts physiques. Bien que la plupart de ces méthodes aient été proposées il y a des décennies, elles gagnent en popularité ces dernières années en raison de la large utilisation de robots collaboratifs, qui sont généralement équipés de capteurs de force/couple qui leur permettent de détecter les interactions physiques avec un opérateur ou un environnement [1], [2]. Ce type de robot est spécialement conçu pour être humain et peut être utilisé pour la création de cellules de travail partagées homme-robot, où la sécurité humaine est une priorité.

Cependant, assurer la sécurité humaine reste un défi, car la plupart des robots industriels pourraient atteindre des vitesses et des forces assez élevées, ce qui pourrait être dangereux pour un humain [3], [4]. Bien qu'il existe actuellement un certain nombre de techniques d'évitement de collision, il arrive qu'une collision entre un humain et un robot soit inévitable ou même nécessaire pour le processus de fabrication. Pour assurer la sécurité dans la pratique, toutes les interactions physiques doivent être correctement identifiées et gérées par une réaction appropriée du robot qui peut varier d'un simple arrêt d'urgence à des scénarios plus compliqués, comme s'éloigner de l'obstacle ou réduire la rigidité de l'articulation du robot. Il est clair que pour mettre en uvre l'un de ces scénarios, la force d'interaction homme-robot doit être bien estimée, motivant le développement de techniques spéciales [5], [6].

Un autre aspect important, qui survient lors de l'interaction physique, est la performance d'une cellule de travail homme-robot. Étant donné que les robots collaboratifs sont conçus pour une interaction homme-robot sûre, la rigidité de leurs articulations

et/ou liens est essentiellement faible. Pour ces robots, même de petites forces d'interaction appliquées peuvent décaler la trajectoire de l'outil du robot de la trajectoire souhaitée. Ainsi, pour améliorer la précision du robot et la sécurité humaine lors de l'interaction physique, le comportement de conformité adaptative peut être utilisé, ce qui modifie la rigidité du robot en fonction de la tâche donnée et des interactions identifiées. Il est clair que pour mettre en uvre des scénarios d'interaction homme-robot ou un comportement de conformité adaptatif, les paramètres de l'interaction homme-robot doivent être bien estimés, ce qui motive le développement de techniques spéciales présentées dans cette thèse.

Pour identifier la force d'interaction souhaitée, différentes techniques peuvent être appliquées, allant des mesures directes aux algorithmes numériques sophistiqués. En pratique, différents capteurs internes et externes pourraient être utilisés pour extraire des paramètres d'interactions et de forces de contact [7], [8]. Dans ce travail, nous proposons d'utiliser uniquement les informations des capteurs internes du robot, c'est-à-dire de limiter le sous-système de mesure par des capteurs de couple articulaires à un axe et des codeurs angulaires. En traitant les données des capteurs, il est possible d'extraire les informations tactiles souhaitées concernant l'interaction, qui incluent la position de contact sur la surface du robot et l'amplitude et la direction de la force d'interaction.

La plupart des approches existantes pour extraire la position de contact et les paramètres d'interaction des capteurs internes sont basées sur la minimisation des carrés des résidus des équations d'équilibre statique, sous réserve de certaines contraintes résultant à la fois de la géométrie des liens du robot et d'une propriété de surface de contact [9]. Par exemple, le point de contact doit être clairement situé sur la surface de liaison du robot et la direction de la force doit être proche de la normale à la surface. Les techniques numériques correspondantes pour l'identification des paramètres d'interaction reposent sur une optimisation contrainte non linéaire simple, qui garantit généralement une bonne précision, mais offre une vitesse de calcul plutôt faible. Ce dernier est causé par la nature itérative des algorithmes d'optimisation pertinents nécessitant de nombreuses évaluations de la fonction objectif pour la recherche des paramètres d'interaction. Ainsi, en pratique, il est nécessaire de trouver un compromis raisonnable entre la vitesse et la précision de l'algorithme. D'autres difficultés surgissant ici sont associées à une incapacité à gérer les interactions dans des configurations singulières ou proches du singulier, où plusieurs minima des fonctions objectives existent, et l'interaction ne peut pas être identifiée de manière unique. Ainsi, les techniques existantes ne répondent pas parfaitement aux exigences imposées par la mise en uvre en ligne.

Objectif principal et problèmes de recherche

Le *objectif principal* de la thèse est le développement de nouvelles techniques de modélisation de l'interaction physique entre un humain et un robot, avec une compliance adaptative, afin d'améliorer la sécurité des humains et les performances des cellules de travail homme-robot. Une attention particulière est accordée aux problèmes de précision et à l'identification des paramètres d'interaction dans des cas singuliers, qui peuvent survenir lors de l'interaction physique en raison des informations de mesure limitées fournies par les capteurs de couple du robot.

Pour atteindre cet objectif, les problèmes suivants doivent être résolus :

Problème 1: Etude comparative des approches existantes d'identification des paramètres d'interaction, qui sont utilisées pour la modélisation de l'interaction physique entre un robot et un humain, afin de détecter leurs faiblesses et de proposer des améliorations appropriées.

Problème 2: Développement d'une nouvelle méthode efficace pour l'identification des paramètres d'interaction dans la collaboration physique homme-robot, capable d'estimer la force d'interaction et son point d'application en temps réel et pour les cas singuliers et non singuliers.

Problème 3: Développement de la stratégie de gestion adaptative des interactions, qui couvre toutes les phases de l'interaction depuis sa détection initiale, son identification jusqu'à sa classification et sa réaction, permettant de modifier le comportement du robot afin d'améliorer la sécurité et l'efficacité de la cellule de travail collaboratif.

Problème 4: Validation expérimentale de la méthode d'identification d'interaction développée à l'aide d'un robot collaboratif réel et application pratique de la stratégie de gestion d'interaction adaptative proposée dans un environnement en temps réel.

Principaux apports théoriques

Les résultats théoriques présentés dans ce travail se situent dans le domaine de la modélisation des interactions physiques homme-robot. Parmi eux, trois contributions peuvent être considérées comme les plus essentielles.

1) La méthode *d'identification des paramètres d'interaction*, qui permet de trouver la force d'interaction et son point d'application à la fois dans les cas *singulier* et *non singulier cas* en utilisant uniquement les mesures de couple articulaire.
2) La méthode *d'amélioration de la précision* de l'identification des paramètres d'interaction et de la résolution des ambiguïtés, qui permet de réduire la déviation des paramètres causée par le bruit des mesures.

3) La méthode de *gestion adaptative des interactions*, qui couvre tous les types d'interactions possibles et permet d'adapter le comportement du robot afin d'améliorer la sécurité et l'efficacité de la cellule de travail collaborative.

Nouveauté des résultats de la recherche

La thèse contribue à l'amélioration des méthodes numériques et analytiques pour la modélisation de l'interaction physique homme-robot, y compris l'amélioration de la précision de l'identification des paramètres d'interaction dans des cas singuliers. Contrairement aux travaux antérieurs dans ce domaine, les méthodes proposées :

- Peut être utilisé pour identifier les paramètres d'interaction dans des cas singuliers. Les travaux précédents ne considéraient que des cas non singuliers, de sorte que les techniques connues ne peuvent pas être appliquées à des configurations de robot arbitraires, des directions de force d'interaction et des points d'application de force.
- Peut produire toutes les solutions possibles pour la force d'interaction et son point d'application en cas d'ambiguïté des équations pertinentes. Dans les travaux précédents, une seule solution pour la force d'interaction et son point d'application a été fournie, qui n'est pas nécessairement correcte.
- Permet d'estimer la robustesse et la précision d'identification vis-à-vis du bruit de mesure. Dans les travaux précédents, cette question était hors de l'attention des auteurs. En outre, contrairement à la pratique courante, il a été supposé que le bruit est présenté comme des variables aléatoires non biaisées et indépendantes mais non distribuées de manière identique, ce qui est plus proche des applications réelles.
- Fournir une vitesse de calcul élevée pour l'identification des paramètres d'interaction en raison de la nature non itérative des algorithmes numériques pertinents, ce qui permet une mise en uvre efficace en temps réel.

Signification théorique et pratique

Cette thèse présente des apports théoriques dans le domaine de la modélisation de l'interaction physique homme-robot et l'identification de ses paramètres. En particulier, il permet d'estimer les paramètres d'interaction (la force d'interaction et son point d'application) dans les cas singuliers, qui mathématiquement sont liés au nombre insuffisant d'équations d'équilibre statique. Par conséquent, les paramètres recherchés doivent être trouvés à partir d'un système indéterminé avec un nombre infini de solutions. Pour lever cette ambiguïté, il est proposé d'appliquer des contraintes géométriques pertinentes décrivant uniquement les forces d'interaction physiquement possibles, ce qui permet de réduire essentiellement l'ensemble des solutions possibles

de manière analytique. De plus, pour les cas singuliers et non singuliers, les solutions obtenues sont présentées sous forme fermée, alors que les techniques précédentes sont basées sur l'optimisation numérique, qui ne fournit pas la vitesse de calcul souhaitée.

Tout d'abord, les apports pratiques de la thèse incluent l'amélioration de la sécurité humaine en présence d'un robot collaboratif. Les techniques d'identification d'interaction développées permettent de surveiller en continu l'amplitude du vecteur de force d'interaction pour violer les valeurs d'interaction sûres. S'il est violé, l'algorithme de gestion adaptative des interactions peut modifier le comportement du robot pour éviter les blessures, en utilisant différents scénarios et une commutation dynamique entre eux. En plus de l'amplitude du vecteur de force, son point d'application peut être utilisé pour améliorer encore la sécurité humaine en limitant les mouvements du robot afin d'éviter le " coincement " humain entre les liaisons du robot. De plus, les capacités en temps réel des techniques développées permettent de réagir presque instantanément à toute situation potentiellement dangereuse.

Deuxièmement, des contributions pratiques peuvent être utilisées pour améliorer l'efficacité de la cellule de travail homme-robot. D'une part, les paramètres d'interaction identifiés peuvent être utilisés pour communiquer entre l'humain et le robot en permettant des scénarios de collaboration spéciaux, selon comment et où l'humain touche le robot. D'autre part, les mêmes paramètres peuvent être utilisés pour la compensation des erreurs de conformité, ce qui peut augmenter la précision du robot en présence de la force d'interaction, améliorant ainsi les performances globales de la cellule de travail. De cette manière, les méthodes développées pourraient non seulement améliorer la sécurité d'un humain dans un espace de travail robotisé, mais également augmenter les performances de la cellule de travail en activant différents scénarios de collaboration.

Enfin, des contributions pratiques peuvent être utilisées dans la conception collaborative de robots. L'analyse de précision d'identification pertinente peut être utilisée comme méthodologie pour la conception de robots, en fournissant la précision d'identification attendue en fonction des paramètres de capteur utilisés.

Méthodes de recherche

Dans cette thèse, diverses méthodes ont été utilisées dont la théorie de la modélisation cinématique et dynamique des systèmes robotiques, l'optimisation numérique, ainsi que la modélisation des interactions physiques. Par ailleurs, des réseaux de neurones artificiels ont été appliqués à la classification des interactions ainsi qu'à l'identification de ses paramètres. Pour le module de réaction d'interaction, la machine à états finis a été utilisée. Tous les modèles et algorithmes ont été implémentés dans des environnements MATLAB et Python et testés sur du matériel réel.

Validité des résultats obtenus

La validité des principaux résultats présentés dans la thèse est confirmée par des résultats de simulation et des études expérimentales dédiées, qui ont été menées pour les robots collaboratifs industriels KUKA LBR iiwa 14 et Universal Robotics UR10e ; ainsi que par l'utilisation correcte d'appareils mathématiques connexes et par des publications dans des revues à comité de lecture et des actes de conférence.

Diffusion des résultats de la recherche

Les résultats de la thèse ont été présentés à la communauté scientifique et approuvés lors de 12 conférences, dont : huit conférences IEEE telles que International Conference on Intelligent Robots and Systems(IROS 2021, 2022), International Conference on Automation Science and Engineering (CASE 2021), International Conference on Robot and Human Interactive Communication (RO-MAN 2017), International Conference on Mechatronics (ICM 2019). Ils ont également été discutés lors de deux conférences IFAC sur Manufacturing Modelling, Management and Control (MIM 2022, 2019) et autres.

Publications

Les principaux résultats obtenus dans cette thèse ont été publiés dans 14 articles et ont été présentés lors de 12 conférences. Parmi eux, il y a 3 articles de revues (IEEE Robotics and Automation Letters; Mechatronics, Automation, Control), 2 articles de revues en cours de révision (Robotics and Computer-Integrated Manufacturing; Engineering Applications of Artificial Intelligence) et 10 articles de conférence indexés par Web of Science / Scopus : IEEE International Conference on Intelligent Robots and Systems; IEEE International Conference on Automation Science and Engineering; IEEE International Symposium on Robot and Human Interactive Communication; IEEE International Conference on Mechatronics; IFAC Conference on Manufacturing Modelling, et autres.

Contribution personnelle de l'auteur

Les principaux résultats présentés dans cette thèse sont obtenus par l'auteur. En particulier, l'auteur a été directement impliqué dans toutes les étapes de la recherche, y compris l'analyse comparative, le développement des nouvelles méthodes, des algorithmes d'identification des interactions, ainsi que leurs validations expérimentales et les applications pratiques des résultats finaux.

Structure de la thèse

Afin de présenter les principaux résultats et contributions de cette thèse, celle-ci est divisée en cinq chapitres et est organisée comme suit.

Chapitre 1. Ce chapitre est consacré à l'état de l'art et à la revue de la littérature sur la robotique collaborative, la collaboration homme-robot et la modélisation de l'in-

teraction physique. Une attention particulière est portée aux règles élémentaires de sécurité humaine lors du processus de collaboration avec un robot et à la formalisation mathématique de l'interaction physique homme-robot. L'objectif principal de ce chapitre est de passer en revue les méthodes existantes dans la modélisation de l'interaction physique et de mettre en évidence leurs avantages, inconvénients et difficultés, qui n'ont pas encore été abordés et permettent de définir l'objectif principal.

Chapitre 2. Ce chapitre présente une étude comparative des approches existantes pour l'identification des paramètres d'interaction, qui sont utilisées pour la modélisation de l'interaction physique entre un robot et un humain, afin de détecter leurs faiblesses et de proposer des améliorations appropriées. En particulier, il évalue numériquement et expérimentalement les approches existantes pour l'identification des paramètres d'interaction qui ont été mises en uvre par l'auteur de cette thèse. Dans le cadre de cette comparaison, trois critères principaux sont utilisés : la précision, le temps d'exécution et la capacité de l'algorithme à trouver plusieurs solutions.

Chapitre 3. Ce chapitre traite de l'identification des paramètres d'interaction pour le cas plan 2D. Il propose une nouvelle technique rapide non itérative pour calculer la force d'interaction et son point d'application en utilisant uniquement les données de mesure obtenues à partir des capteurs de couple internes de l'articulation. L'algorithme proposé est basé sur une solution générale paramétrée d'une équation d'équilibre statique permettant de trouver les paramètres recherchés même en cas d'ambiguïté essentielle. Contrairement aux approches existantes, la méthode proposée est applicable aux cas singuliers associés à un nombre insuffisant d'équations indépendantes dans un système d'équilibre statique produisant des paramètres non uniques.

Chapitre 4. Ce chapitre traite de l'identification des paramètres d'interaction pour le cas spatial et présente des techniques permettant d'estimer la force et son point d'application dans l'interaction physique homme-robot. On suppose que les paramètres souhaités sont estimés à l'aide de données obtenues à partir de capteurs de couple intégrés dans les articulations du robot. En pratique, les données de mesure sont altérées par du bruit, ce qui provoque des erreurs d'identification. Sur la base de l'analyse pertinente, des améliorations ont été proposées, qui améliorent la précision et la robustesse de l'identification par rapport au bruit de mesure.

Chapitre 5. Ce chapitre porte sur le développement d'une technique adaptative de gestion des interactions physiques entre un humain et un robot, ainsi que sur la validation expérimentale des algorithmes d'identification développés. Les principaux objectifs de cette technique sont d'assurer la sécurité d'un opérateur humain et d'améliorer les performances de la collaboration homme-robot en mettant en uvre divers scénarios. Dans le cadre de cette technique, les méthodes d'identification des paramètres d'in-

teraction proposées dans le chapitre précédent sont utilisées pour la sélection d'une stratégie de réaction appropriée. Ces paramètres définissent la force d'interaction et son point d'application sur la surface du robot qui sont utilisés pour la classification des interactions dans l'ensemble des catégories prédéfinies. Sur la base de ces catégories et de l'état actuel du robot, l'algorithme choisit une réaction de robot appropriée. À la fin de ce chapitre, sont présentés les résultats expérimentaux obtenus pour le robot collaboratif réel KUKA LBR iiwa et valident les techniques développées.

Enfin, *la Conclusion* résume les principaux apports de la thèse et définit des perspectives pour les travaux de recherche futurs.

Contenu de la thèse

La partie **introduction** montre la pertinence du sujet de recherche et formule son objectif principal avec la liste des tâches correspondantes, nécessaires pour l'atteindre. La nouveauté scientifique et sa signification, ainsi que des résultats pratiques avec l'approbation pertinente, finalisent la partie introductive.

Le premier chapitre est consacré aux particularités d'une modélisation d'interaction physique homme-robot. Il traite de l'état de l'art dans ce domaine, il présente également le problème général de la gestion des événements de collision en robotique collaborative (illustré à la Fig. 1), qui est traité en plusieurs étapes, qui incluent l'interaction détection, isolement, identification, classification et réaction. Il convient de mentionner que cette thèse se concentre sur les étapes d'isolation et d'identification des interactions qui sont combinées en une seule étape d'identification avec un exemple pratique de la procédure complète de traitement des interactions. Il est à noter qu'en dépit de nombreux travaux dans le domaine considéré, il reste encore plusieurs questions importantes à étudier pour les algorithmes d'identification plus en détail. En particulier, il est nécessaire d'augmenter la vitesse de calcul des algorithmes d'identification pertinents permettant leur utilisation efficace en temps réel. Par ailleurs, une attention particulière doit être portée à la faisabilité physique des solutions obtenues ainsi qu'à l'identification des interactions en cas de données de mesure insuffisantes.

La section 1.3 formule mathématiquement le problème de l'interaction physique entre un humain et un robot pour les cas planaires et 3D correspondants. Plus en détail, le modèle d'interaction peut être décrit par des équations d'équilibre statique classiques, qui définissent la relation de base entre la force $\mathbf{F} = [F_x, F_y, F_z]^T$ et le vecteur de couples articulaires $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)^T$ et peut être écrit en utilisant le manipulateur Jacobien $\mathbf{J}(\mathbf{q}, \mathbf{p})$, où $\mathbf{p}$ est la force point d'application et $\mathbf{q}$ est le vecteur de coordonnées commun. Il convient de mentionner qu'ici, la force d'interaction n'est généralement pas appliquée

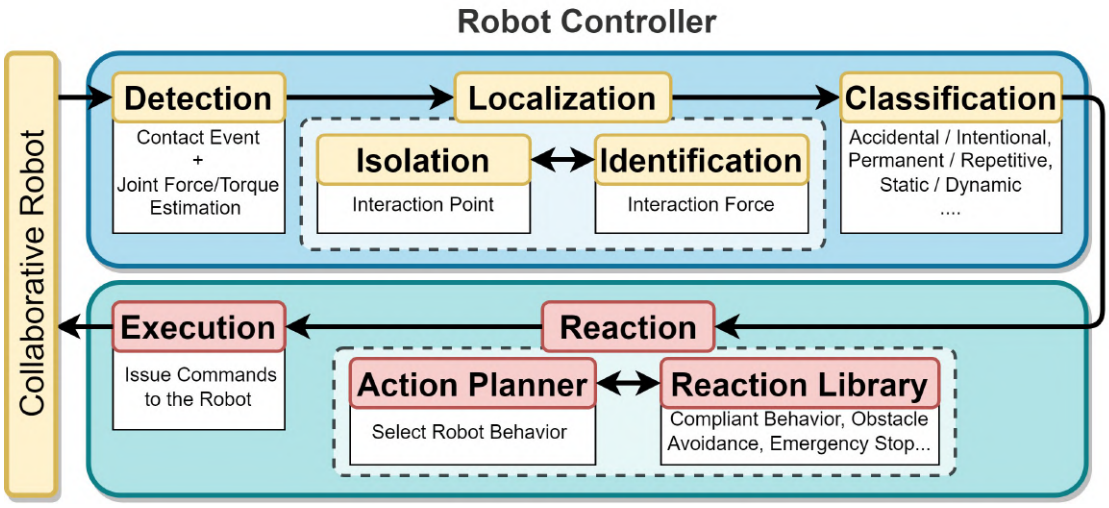


Figure 1: Étapes de base de la gestion des événements d'interaction dans la collaboration homme-robot.

à l'effecteur final mais à tout manipulateur $k$ième lien. Ainsi, le jacobien réduit $\mathbf{J}_k(\mathbf{q}, \mathbf{p})$ doit être utilisé, qui est obtenu à partir du conventionnel $\mathbf{J}(\mathbf{q}, \mathbf{p})$ par extraction de ses premières k colonnes. En utilisant ces notations, les équations d'équilibre statique souhaitées peuvent être écrites comme

$$\mathbf{J}_k\left(\mathbf{q}^{(k)}, \mathbf{p}\right)^T \cdot \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \dots \\ \tau_k \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \dots \\ \varepsilon_k \end{bmatrix} \tag{1}$$

où $\mathbf{p}$ est le point d'application de la force, $\mathbf{q}^{(k)} = (q_1, \dots, q_k)^T$ est le vecteur de coordonnées articulaires réduit et seules les k composantes significatives du vecteur de couple articulaire $\boldsymbol{\tau}$ sont utilisées. De plus, on suppose ici que toutes les mesures de couple τ_i sont perturbées par certaines composantes de bruit ε_i.

Comme il ressort de la nature physique du problème considéré, l'équation d'équilibre statique ci-dessus doit être résolue en conjonction avec certaines contraintes décrivant les surfaces de liaison du manipulateur. En pratique, il convient de décrire ces surfaces à l'aide des maillages triangulaires classiques, largement utilisés en modélisation CAO 3D. Cela permet de présenter les contraintes géométriques de base sous la forme suivante $\mathbf{p} \in \boldsymbol{\Omega}^k$, où l'exposant k indique le numéro du lien.

Une autre contrainte qu'il faut obligatoirement prendre en compte est liée aux directions réalisables de la force d'interaction $\mathbf{F}$, qui sont évidemment bornées. En tenant compte de la loi de frottement de Coulomb, les directions possibles peuvent être présentées sous la forme du "cône de frottement", qui est construit autour de la surface de liaison normale $\mathbf{n}_p$ au point de contact $\mathbf{p}$, comme le montre la Fig. 2. En

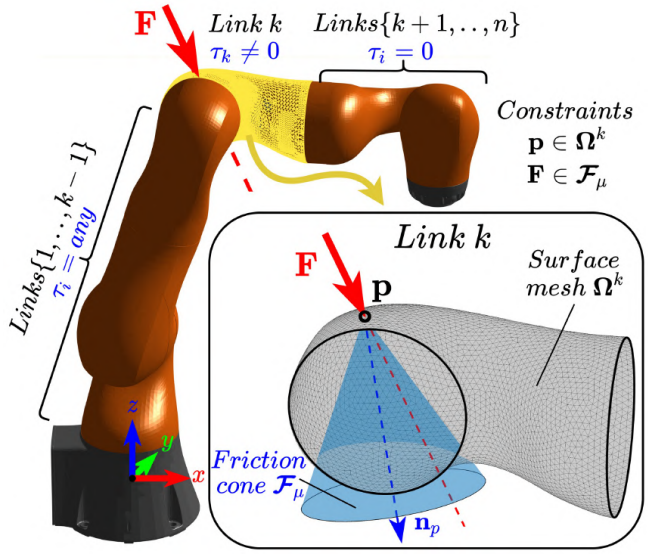


Figure 2: Représentation graphique des contraintes géométriques pour le $k$ième lien.

supposant que le coefficient de frottement est égal à μ et $\mathbf{n}_p$ est une normale pointant vers l'intérieur (c'est-à-dire pointant vers l'intérieur du lien), la contrainte associée peut être présenté de la manière suivante

$$|\angle(\mathbf{n}_p,\ \mathbf{F})| \leq atan(\mu) \tag{2}$$

Dans ce modèle, le problème principal est l'identification des paramètres d'interaction, qui consistent en un vecteur de force d'interaction $\mathbf{F}$ et son point d'application $\mathbf{p}$. L'identification de ces paramètres sera l'objet principal de la thèse.

Pour présenter l'ambiguïté possible du problème d'identification des paramètres d'interaction, plusieurs études de cas qui montrent plusieurs solutions potentielles pour les paramètres d'interaction.

La dernière section de ce chapitre résume l'état actuel de l'art et formule l'objectif principal et les tâches de la recherche.

Le deuxième chapitre présente une étude comparative des approches existantes pour l'identification des paramètres d'interaction, qui sont utilisées pour la modélisation de l'interaction physique entre un robot et un humain, afin de détecter leurs faiblesses et de proposer des améliorations appropriées. En particulier, il considère les approches d'identification des paramètres d'interaction développées au cours de la thèse. Pour les évaluer, trois critères principaux sont utilisés, il s'agit de la précision, du temps d'exécution et de la possibilité de sortir plusieurs solutions potentielles si elles existent.

La section 2.2 décrit l'approche basée sur l'approximation cylindrique de la forme du robot. L'idée principale de la méthode est de représenter la forme complexe de la surface du robot sous la forme d'un cylindre. Cela permet de paramétrer l'emplacement

du point de contact avec une seule valeur. De plus, la direction de la force d'interaction est supposée être dirigée colinéairement à la normale au point de contact. Bien que cette approche soit essentiellement simple, elle a montré une faible précision en raison de l'approximation de la forme du robot et d'un temps d'exécution assez élevé.

La section 2.3 présente les approches basées sur les réseaux de neurones capables d'identifier le point d'application de la force d'interaction. Le premier d'entre eux utilise le paramétrage du cylindre de la section 2.2 et prédit le point en fonction des valeurs de couple mesurées et de la pose du robot. La seconde résout le problème de classification où chaque classe est l'un des points fixes sur la surface du robot. Pour appliquer ces deux approches au robot réel, la méthodologie de base de l'apprentissage par transfert a été utilisée. Étant donné que ces deux méthodes utilisent des réseaux de neurones à anticipation, leur coût de calcul est faible mais la précision est limitée.

La section 2.4 discute de l'approche basée sur la cartographie sphérique de la surface du robot. L'idée principale de la méthode est de transformer la forme complexe de la surface du robot en une sphère, de trouver les paramètres d'interaction dans l'espace sphère et d'utiliser la transformation inverse pour obtenir les paramètres dans l'espace initial. A cet effet, un algorithme temps réel basé sur une procédure d'optimisation en deux étapes est proposé. Dans la première étape, une position approximative du point de contact est estimée, dans la deuxième étape, une optimisation locale est effectuée afin de spécifier l'emplacement exact du contact et les forces externes agissantes. L'algorithme est capable de fonctionner avec une boucle de 100 Hz et assure 2 cm pour la localisation du point de contact dans les 3 dernières liaisons avec du bruit dans une détection de couple.

La section 2.5 décrit l'approche basée sur la théorie des graphes et le filtre particulaire. Cette approche utilise les idées de base du filtre à particules de contact proposé par Manuelli et Tedrake mais avec une durée d'exécution améliorée. La réduction des coûts de calcul est obtenue en représentant la surface du robot sous forme de graphique, ce qui remplace essentiellement l'étape de projection de particules la plus chronophage de la méthode d'origine.

La section 2.6 présente l'approche basée sur la représentation hiérarchique de la surface du robot. Cette approche peut détecter et évaluer plusieurs solutions plus de 3 fois plus rapidement sans perte de précision. Pour obtenir des performances en temps réel, une représentation hiérarchique du robot et un prétraitement du maillage de surface ont été utilisés, ce qui nous a permis d'obtenir une accélération de 50 fois en un temps d'exécution, par rapport à la vérification de tous les points de contact. Le prétraitement du maillage comprend le remaillage isotrope et le regroupement en deux étapes dans l'espace des vecteurs normaux des sommets et des positions des sommets.

Table 1: Résultats expérimentaux sur la précision et la durée d'exécution des approches existantes

Approach	**Robot**	$\Delta\mathbf{p}$	$\mathbf{T}_c$
Surface approximation approaches			
Cylinder approx. [Section 2.2]	KUKA iiwa	5.4	1/19
Sphere mapping [Section 2.4]	KUKA iiwa	4.2	1/100
DIRECT [78]	KUKA iiwa	6.0	1/30
DIRECT-link [57]	Kinova Jaco2	12.0	1/20
Monte Carlo based approaches			
PF on graph [Section 2.5]	KUKA iiwa	4.0	1/60
Contact particle filter [87]	KUKA iiwa	2.4	1/10
PF with FC binning [88]	Kinova Jaco2	11.0	1/63
PF with binning [88]	Kinova Jaco2	12.0	1/71
PF with nearest neighbor [88]	Kinova Jaco2	11.0	1/125
PF with weighted means [88]	Kinova Jaco2	11.0	1/159
Machine learning based approaches			
Feed-forward NN [Section 2.3]	KUKA iiwa	6.4	1/180
Random forest [57]	Kinova Jaco2	8.0	1/200
Multilayer perceptrons [57]	Kinova Jaco2	4.0	1/200
Clustered surface approach			
Clustered surface [Section 2.6]	KUKA iiwa	2.3	1/600

Note: $\mathbf{\Delta p}$ - estimation error of the force application point $[cm]$;
$\mathbf{T}_c$ - cycle time of the estimation algorithm $[1/Hz]$.

En exploitant cette structure, l'exécution en temps réel de notre algorithme sur des fréquences supérieures à 600 Hz a été réalisée.

Les résultats de la comparaison des approches existantes et susmentionnées sont présentés dans la section 2.7 et présentés dans le tableau 1. Malgré le fait que le temps d'exécution de l'algorithme de structure arborescente de robot lui permet d'être utilisé dans une application critique en temps réel et sans danger pour l'homme, ses performances se dégradent toujours dans les cas singuliers. Étant donné que l'approche est basée sur une optimisation simple, elle n'est pas capable de produire plusieurs solutions possibles pour le même lien, ce qui se produit dans des cas singuliers lorsque le système d'équilibre statique est indéterminé. Ceci motive le développement d'une nouvelle technique présentée dans le chapitre suivant.

Le troisième chapitre est consacré au développement d'une nouvelle méthode efficace pour l'identification des paramètres d'interaction dans la collaboration physique homme-robot pour le cas planaire, qui est capable d'estimer la force d'interaction et son point d'application en temps réel et pour les cas singuliers et non singuliers.

Comme il ressort du chapitre précédent, le problème d'optimisation pour l'identification des paramètres d'interaction peut avoir une solution unique ou non unique. Premièrement, un cas non singulier est présenté, lorsqu'il existe une solution unique pour la force $\mathbf{F}$ et son point d'application $\mathbf{p}$. Dans ce cas, toutes les équations d'équilibre statique sont complètement satisfaites et la fonction objectif considérée atteint sa valeur minimale possible égale à zéro. Pour simplifier le processus, le problème sera résolu en deux étapes, où la première ignore les contraintes géométriques et la seconde les prend en compte afin d'obtenir la solution souhaitée.

Pour le cas non singulier, supposons que le système est cohérent et que le point d'application de la force $\mathbf{p}$ est connu. Pour trouver une solution sous forme fermée du système (1), introduisons un système étendu des équations d'équilibre statique de la forme suivante

$$\left[\mathbf{J}_k^w(\mathbf{q},\mathbf{p})^T\right]_{k\times 3}\cdot\begin{bmatrix}F_x\\F_y\\M_z\end{bmatrix}=\begin{bmatrix}\tau_1\\\dots\\\tau_k\end{bmatrix} \tag{3}$$

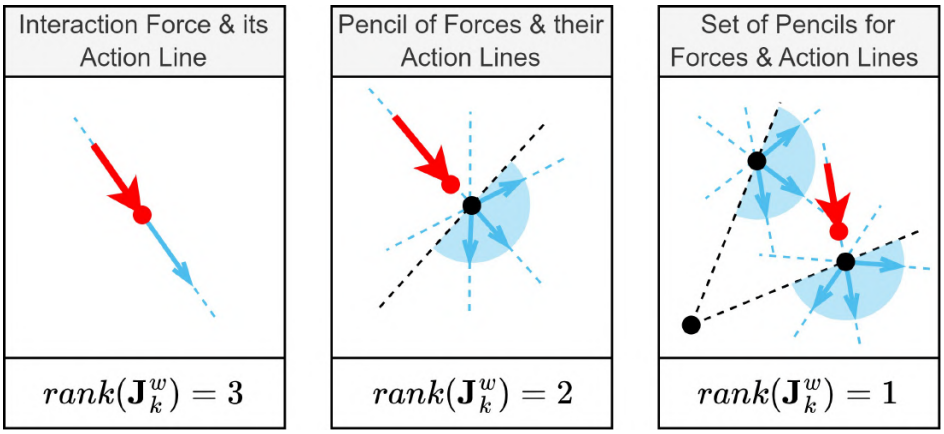
où le vecteur de force inconnu $\mathbf{F}=[F_x,F_y]^T$ est complété par le valeur de couple M_z et l'original $2\times k$ Jacobian $\mathbf{J}_k\left(\mathbf{q}^{(k)},\mathbf{p}\right)$ est remplacé par la matrice jacobienne étendue de taille $3\times k$. Ensuite, ce système peut être facilement résolu pour la clé $\mathbf{W}=[\mathbf{F},M_z]^T$ en utilisant l'expression suivante

$$\mathbf{W}=\left[\mathbf{J}_k^w\left(\mathbf{q}^{(k)},\mathbf{p}\right)^T\right]^{\#}\cdot\boldsymbol{\tau} \tag{4}$$

où $\boldsymbol{\tau}=[\tau_1,\dots,\tau_k]^T$ est le vecteur de couple et $[.]^{\#}$ désigne la pseudo-inverse de la matrice de Moore-Penrose. De plus, il est prouvé que, pour tout $\mathbf{p}$ arbitraire, la clé obtenue $\mathbf{W}$ nous fournit toujours les composantes désirées de la force d'interaction $\mathbf{F}$ du système d'origine (1). Ensemble, la force d'interaction $\mathbf{F}$ et un certain $\mathbf{p}$ arbitraire peuvent être représentés par la force d'interaction $\mathbf{F}$ et sa ligne d'action.

La solution générale des équations d'équilibre statique a été obtenue en fournissant le vecteur force $\mathbf{F}$ et sa ligne d'action en ignorant les contraintes géométriques $\mathbf{p}\in\boldsymbol{\Omega}_k,\mathbf{F}\in\boldsymbol{\mathcal{F}}_\mu$. Pour cette raison, la section suivante se concentre sur l'application de ces contraintes à la solution obtenue, ce qui nous permet finalement d'identifier les paramètres d'interaction désirés $\mathbf{F},\mathbf{p}$. Cela peut être fait en effectuant les calculs

Table 2: Solutions possibles pour la force d'interaction $\mathbf{F}$ et sa ligne d'action : trois cas possibles selon $rank(\mathbf{J}_k^w)$.



Note: Real force is shown in red, possible solutions are shown in blue.

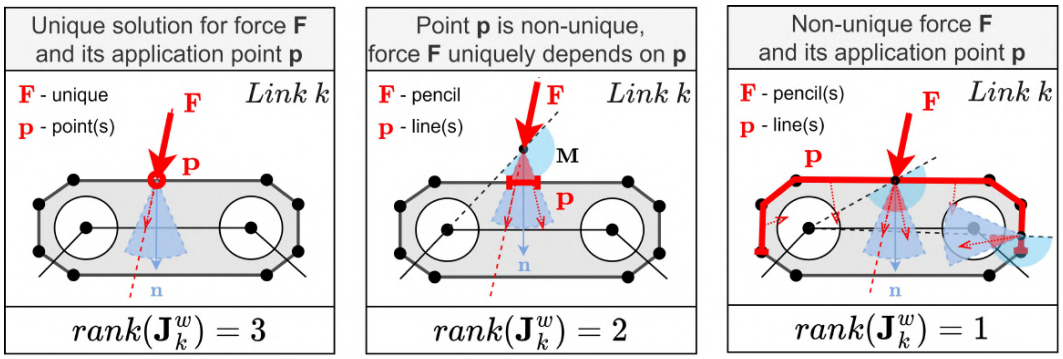
suivants : (i) Vérification de la contrainte du cône de frottement ; (ii) Vérification de l'existence de l'intersection des segments de ligne ; (iii) Calcul du point d'intersection du segment de ligne (s'il existe). Ce dernier nous permet d'exclure certaines étapes inutiles si au moins une des contraintes est violée.

La section 3.3 se concentre sur les cas singuliers où la force d'interaction $\mathbf{F}$, sa ligne d'action ou le point d'application de la force $\mathbf{p}$ ne peuvent pas être calculés de manière unique. Il est clair que ces cas sont associés au déficit de rang du jacobien étendu, qui est ici inférieur à trois. De plus, dans les cas singuliers, la fonction objectif n'atteint pas son minimum global en un seul point isolé, ce qui donne de multiples solutions au problème considéré qui doivent évidemment inclure la "vraie" correspondant à l'interaction physique réelle. Par conséquent, à la fois le vecteur de force, sa ligne d'action et le point d'application de la force peuvent être exprimés sous une forme paramétrique, qui contient certaines valeurs arbitraires.

Le résumé de tous les cas possibles (singuliers et non singuliers) pour l'identification de la force d'interaction est présenté dans le tableau 2. Comme il ressort de ce tableau, si $rank(\mathbf{J}_k^w) = 3$, il existe une solution unique pour le vecteur de force et sa ligne d'action. En revanche, si $rank(\mathbf{J}_k^w) = 1$, il existe un nombre infini de solutions $\mathbf{F}$ pour tout point considéré (x, y). Et si $rank(\mathbf{J}_k^w) = 2$, l'ensemble des solutions possibles est également infini, mais il peut être représenté comme un seul crayon de lignes droites pour les directions de force possibles.

L'identification du point d'application de la force $\mathbf{p}$ a également été effectuée pour les cas singuliers avec $rank(\mathbf{J}_k^w) = 2$ et $rank(\mathbf{J}_k^w) = 1$. Contrairement au cas non singulier, la solution est représentée non pas par un point mais par un intervalle sur la surface de liaison. Évidemment, la procédure proposée doit être appliquée à tous

Table 3: Résumé des solutions possibles pour le point d'application de la force d'interaction $\mathbf{p}$: trois cas possibles en fonction de $rank(\mathbf{J}_k^w)$.



les segments m décrivant la $k$ième forme de lien pour obtenir la solution complète. Le résumé de tous les cas possibles, y compris les cas singuliers et non singuliers, est présenté dans le tableau 3, qui, avec le tableau 2, couvre tous les types de solution pour les deux, la force d'interaction $\mathbf{F}$ et son point d'application $\mathbf{p}$ en conséquence. Pour résoudre l'ambiguïté du problème d'identification considéré, plusieurs hypothèses sont utilisées ici, qui sont basées sur l'expérience pratique. Ces hypothèses sont présentées ci-dessous et sont liées à l'ambiguïté du calcul de l'indice k, de la direction de la force $\mathbf{F}$ et du point d'application de la force $\mathbf{p}$ ainsi qu'à l'utilisation de l'historique d'estimation pour résoudre les cas singuliers.

Pour résumer ce chapitre, l'identification du point d'application de la force d'interaction a été fournie pour les trois cas possibles pour le manipulateur planaire en série. L'utilisation de contraintes géométriques a permis de réduire considérablement le nombre de solutions possibles pour le système donné d'équations d'équilibre statique même dans les cas singuliers.

Un point faible de la méthode proposée est sa robustesse plutôt faible au bruit de mesure, qui a été observée dans l'étude de simulation. Bien que la précision de l'algorithme soit acceptable pour des niveaux de bruit pratiquement observés, pour des bruits très élevés il est tout à fait possible que la technique analytique développée n'apporte aucune solution. Dans ce dernier cas, la ligne d'action de force estimée diffère essentiellement de la ligne réelle et ne croise aucune surface de liaison du robot. Pour cette raison, les problèmes de robustesse feront l'objet du chapitre suivant.

Le quatrième chapitre traite de l'identification des paramètres d'interaction pour le cas spatial et présente des techniques permettant d'estimer la force et son point d'application dans l'interaction physique homme-robot. On suppose que les paramètres souhaités sont estimés à l'aide de données obtenues à partir de capteurs de couple internes intégrés dans les articulations du robot. En pratique, les données de mesure sont altérées par du bruit, ce qui provoque des erreurs d'identification. Sur la base de

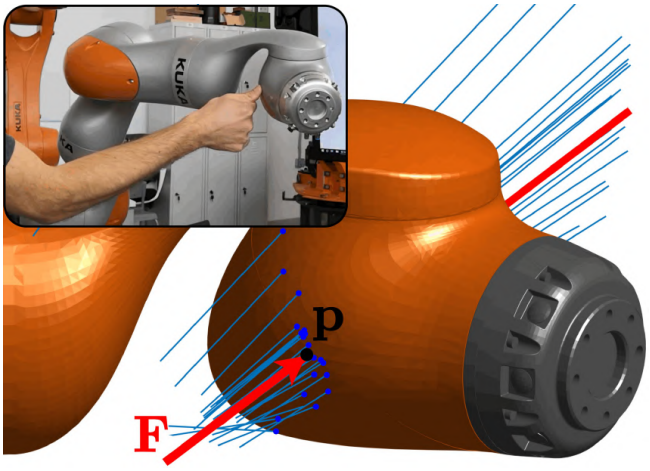


Figure 3: Influence du bruit de mesure du couple sur la précision d'identification de la force d'interaction $\mathbf{F}$ et de son point d'application $\mathbf{p}$. La véritable force d'interaction est indiquée en rouge, tandis que les lignes d'action de la force estimée perturbées par le bruit sont indiquées en bleu.

l'analyse pertinente, des améliorations ont été proposées, qui améliorent la précision et la robustesse de l'identification par rapport au bruit de mesure. Un exemple pertinent est présenté sur la Fig. 3, où un ensemble de lignes d'action de force correspondant à différents échantillons de bruit est représenté. Il est clair que de telles perturbations dans les mesures de couple peuvent provoquer des erreurs d'estimation essentielles, qui doivent être correctement analysées et évaluées.

La section 4.1 présente l'algorithme d'identification pour le cas non singulier, qui reprend les idées de base de De Luca et Haddadin [9] mais inclut également quelques améliorations permettant d'améliorer sa robustesse et sa précision. Cet algorithme est basé sur l'hypothèse habituelle que le système d'équations d'équilibre statique (1) est cohérent, ce qui est évidemment valable pour le cas sans bruit, c'est-à-dire lorsque $\varepsilon_i = 0,\ \forall je$. Ce dernier nous permet de trouver la solution souhaitée sous une forme fermée en utilisant une procédure en deux étapes, où la première étape se concentre uniquement sur les équations d'équilibre statique, en omettant les contraintes géométriques sur les variables inconnues $\mathbf{F}, \mathbf{p}$ mais en étendant ces équations en incluant des variables supplémentaires. Ici, la solution générale de ces équations est obtenue et intégrée avec les contraintes $\mathbf{p} \in \mathbf{\Omega}_k, \mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu$ at la deuxième étape donnant les paramètres d'interaction souhaités.

La section 4.2 suivante se concentre sur les cas singuliers où la force d'interaction $\mathbf{F}$, sa ligne d'action ou le point d'application de la force $\mathbf{p}$ ne peuvent pas être calculés de manière unique. Il est clair que ces cas sont associés au défaut de rang du jacobien étendu $\mathbf{J}_k^w(\mathbf{q}, \mathbf{p})$, qui dans la section précédente était supposé être de rang complet. Semblable au cas planaire considéré dans chapitre 3, ici le problème d'optimisation

de contraintes associé donne également plusieurs solutions pour les paramètres d'interaction souhaités. Cependant, contrairement au cas planaire, il est plutôt difficile d'obtenir des expressions de forme fermée ou paramétriques décrivant des ensembles de forces d'interaction possibles et leurs points d'application. Pour cette raison, l'accent principal de cette section est sur les propriétés géométriques de ces ensembles et leur calcul numérique.

Supposons que les "vrais" paramètres d'interaction $\mathbf{F}, \mathbf{p}$ sont connus et considérons une version simplifiée du système d'équilibre statique original $\mathbf{J}_k(\ mathbfq, \mathbf{p})^T \cdot \mathbf{F} = \boldsymbol{\tau}$, qui est linéarisé dans son voisinage. Il peut être facilement prouvé que le système linéarisé souhaité peut être présenté comme

$$[\mathbf{e}_i \times (\mathbf{p} - \mathbf{p}_i)]^T \cdot \delta\mathbf{F} + [\mathbf{F} \times \mathbf{e}_i]^T \cdot \delta\mathbf{p} = 0, \quad \forall i = 1, \ldots, k \tag{5}$$

où $\delta\mathbf{F}$, $\delta\mathbf{p}$ sont de petites variations des "vrais" paramètres d'interaction $\mathbf{F}, \mathbf{p}$. Comme le sait l'algèbre linéaire, le système homogène ci-dessus a toujours la solution triviale $[\delta\mathbf{F},\ \delta\mathbf{p}]^T = \mathbf{0}$ et dans certains cas, lorsque la matrice $\mathbf{D}_k$ est de rang déficient, le nombre de solutions est infini. Dans le cas général, l'ensemble des variations possibles $\delta\mathbf{F},\ \delta\mathbf{p}$ satisfaisant le système d'équilibre statique linéarisé peut être présenté comme

$$\begin{bmatrix}\delta\mathbf{F} \\ \delta\mathbf{p}\end{bmatrix}_{6\times 1} = \begin{bmatrix}\mathbf{0} \\ \mathbf{0}\end{bmatrix}_{6\times 1} + [null(\mathbf{D}_k)]_{6\times d} \cdot \begin{bmatrix}\mu_1 \\ \cdots \\ \mu_d\end{bmatrix}_{d\times 1} \tag{6}$$

où $null(\mathbf{D}_k)$ désigne l'espace nul de la matrice $\mathbf{D}_k$, $[\mu_1, ...\mu_d]^T$ est un *arbitraired*-dimensionnel, et d est le défaut de rang de $\mathbf{D}_k$ calculé comme $d = 6 - rank(\mathbf{D}_k)$.

L'analyse des cas singuliers liés au défaut de rang d du système d'équilibre statique linéarisé d'équations k peut être résumée comme suit :

- $\boldsymbol{d = 1,\ k \geq 5}$: la force d'interaction $\mathbf{F}$ est identifiée de manière unique, l'ensemble solution des points d'application de force $\{\mathbf{p}\}$ appartient à l'unique ligne d'action de force identifiée passant par le point d'application de force "vrai" $\mathbf{p}^0$;
- $\boldsymbol{d = 2,\ k \geq 4}$: les ensembles solutions des forces d'interaction possibles $\{\mathbf{F}\}$ et leurs points d'application $\{\mathbf{p}\}$ forment un seul faisceau de droites droites, appartenant à un plan identifié de manière unique passant par la force "vraie" $\mathbf{F}^0$ et son point d'application $\mathbf{p}^0$;
- $\boldsymbol{d = 3,\ k \geq 3}$: généralement, les ensembles de forces d'interaction $\{\mathbf{F}\}$ et leurs points d'application $\{\mathbf{p}\}$ forment un vecteur tridimensionnel mais pour le robot KUKA iiwa, l'ensemble de points $\{\mathbf{p}\}$ est contraint par un plan, où la force $\mathbf{F}$ n'est pas unique ;

- $\boldsymbol{d \geq 4,\ k \geq 1}$: le système d'équations d'équilibre statique fournit une ambiguïté extrêmement élevée pour les ensembles de solutions de $\{\mathbf{F}\}$,$\{\mathbf{p}\}$, de sorte que l'identification des paramètres d'interaction souhaités n'est pas réalisable en utilisant de telles données de mesure limitées.

Généralement, les solutions souhaitées pour les paramètres d'interaction $\{\mathbf{F}, \mathbf{p}\}$ doivent satisfaire à la fois le système non linéaire d'équations d'équilibre statique et les contraintes géométriques $\mathbf{F} \in \boldsymbol{\mathcal{F}}_\mu,\ \mathbf{p} \in \boldsymbol{\Omega}_k$. Dans la sous-section précédente, la version linéarisée du système d'équilibre statique a été analysée en détail, sans tenir compte de ces contraintes. Ce dernier nous a permis de découvrir les propriétés les plus essentielles des ensembles de solutions, telles que la géométrie et les dimensions des vecteurs de force d'interaction possibles $\{\mathbf{F}\}$ et leurs points d'application $\{\mathbf{p}\}$. En particulier, il a été découvert que le système d'équilibre statique ne fournit jamais une solution unique et qu'il existe trois cas principaux, qui doivent être considérés séparément. Même dans le meilleur des cas, la solution pour la force d'interaction $\mathbf{F}$ est unique mais les solutions pour son point d'application $\{\mathbf{p}\}$ forment une ligne droite. Dans d'autres cas, l'ambiguïté dans l'identification de $\mathbf{F}$, $\mathbf{p}$ est essentiellement plus élevée, avec $\{\mathbf{F}\}$ formant des crayons et $\{\mathbf{p}\}$ formant un plan, etc. Pour réduire cette ambiguïté, les contraintes géométriques pertinentes doivent également être appliquées, ce qui permet d'obtenir uniquement des solutions physiquement réalisables (voire une solution unique).

Deux techniques de calcul de l'ensemble réduit des paramètres d'interaction $\{\mathbf{F}, \mathbf{p}\}$ sont proposées. La première d'entre elles (Approche 1) est basée sur une simple énumération des primitives de surface et la sélection de celles qui satisfont à la fois au système d'équilibre statique et aux contraintes géométriques. La deuxième technique (Approche 2) utilise certaines propriétés géométriques utiles des ensembles de solutions découverts dans la sous-section précédente. Ces deux approches sont décrites en détail, évaluées et comparées en termes de précision et de performances temporelles.

Comme il ressort de l'étude de simulation présentée ci-dessus, la deuxième approche surmonte la première pour la précision et les performances en temps réel. Néanmoins, il existe une autre caractéristique importante qui rend la première approche très utile dans les applications réelles. En particulier, l'étude de simulation présentée a été menée sans tenir compte du bruit des mesures, qui existe évidemment en pratique. Il est clair que pour l'approche 2, les erreurs de mesure de couple peuvent perturber la ligne d'action de la force de telle sorte que les intersections avec les éléments de maillage triangulaires $\boldsymbol{\Delta}$ disparaissent complètement, de sorte que l'ensemble de solutions peut même être vide. Un comportement similaire de l'algorithme d'identification a également été observé pour le cas 2D présenté à la fin de chapitre 3. Dans certains

travaux connexes, cette difficulté a également été observée et surmontée en redéfinissant le point d'application de la force $\mathbf{p}$ comme le point le plus proche sur la surface de liaison du robot. Néanmoins, une telle méthode peut violer la contrainte de direction de force $\mathbf{F} \in \boldsymbol{\mathcal{F}}_{\mu}$. En revanche, l'approche 1 garantit toujours la satisfaction des contraintes géométriques, fournissant une solution non exacte mais pratiquement acceptable, où la direction de la force $\mathbf{F}$ est à l'intérieur du cône de frottement et le point d'application de la force $\mathbf{p}$ est sur la surface du robot.

En résumant la sous-section 4.3, traitant de l'identification des paramètres d'interaction pour les cas singuliers, il convient de mentionner que les deux techniques développées (appelées Approche 1 et 2 ci-dessus) ont leurs domaines d'application. Contrairement aux cas non singuliers, ils fournissent des ensembles de solutions pour les vecteurs de force d'interaction potentiels $\{\mathbf{F}\}$ et leurs points d'application $\{\mathbf{p}\}$. Il a également été montré que l'approche 2 surpasse l'approche 1 en termes de précision et de temps d'exécution si le bruit des mesures est négligeable. Cependant, la robustesse de l'approche 2 vis-à-vis du bruit de mesure du couple est discutable et doit être analysée plus en détail, ce qui fait l'objet de la section suivante.

La section 4.4 propose l'extension de la technique développée du chapitre 3 à la 3D. La principale différence est la contrainte de surface du robot et sa description. Contrairement aux segments de ligne dans le cas planaire, il est maintenant décrit avec des faces triangulaires. En pratique, cela signifie que le point d'application de la force se trouve non pas par l'intersection de deux droites mais par l'intersection de la ligne d'action de la force et d'un plan formé par une face triangulaire. De plus, la taille du jacobien étendu $\mathbf{J}_k^w\left(\mathbf{q}^{(k)}, \mathbf{p}\right)$ devient 6 $foisk$, ce qui correspond au vecteur clé à six composantes $\mathbf{W} = [\mathbf{F}, \mathbf{M}]^T$.

On the other hand, because of the pseudoinverse in (4), this algorithm should produce a statistically optimal result in the case when the measurement errors satisfy the usual i.i.d. assumption (independent and identically distributed). L'approche susmentionnée ne prend pas directement en compte les propriétés statistiques des erreurs de mesure de couple, qui peuvent différer d'une articulation à l'autre. D'autre part, à cause de la pseudo-inverse dans (4), cet algorithme devrait produire un résultat statistiquement optimal dans le cas où les erreurs de mesure satisfont les i.i.d. habituels. hypothèse (indépendante et identiquement distribuée).

La sous-section suivante évalue la robustesse de cet algorithme vis-à-vis du bruit de mesure et propose quelques pistes d'amélioration. Ici, on suppose implicitement que les écarts-types des erreurs de mesure de couple ne sont pas identiques, comme cela est confirmé par des résultats expérimentaux.

Tout d'abord, obtenez la matrice de covariance pour les estimations de clé $\hat{\mathbf{W}} =$

$[\hat{\mathbf{F}}, \hat{\mathbf{M}}]^T$ de l'expression (4). Après les dérivations habituelles bien connues de la littérature statistique, on peut obtenir l'expression suivante pour la covariance souhaitée

$$\text{cov}(\hat{\mathbf{W}}) = \left[\mathbf{J}_k^w\left(\mathbf{q}^{(k)}, \mathbf{p}\right)^T\right]^{\#} \cdot \boldsymbol{\Sigma}_\tau^2 \cdot \left[\mathbf{J}_k^w\left(\mathbf{q}^{(k)}, \mathbf{p}\right)^T\right]^{\#T} \tag{7}$$

qui inclut la matrice diagonale $\boldsymbol{\Sigma}_\tau = diag\left[\sigma_\tau^1, \sigma_\tau^2 ... \sigma_\tau^k\right]$ composé des écarts types des erreurs de mesure de couple σ_τ^i. De plus, pour trouver la matrice de covariance $\text{cov}(\hat{\mathbf{p}})$ pour le point d'application de la force, une technique similaire peut être appliquée

$$\text{cov}(\hat{\mathbf{p}}) = \begin{bmatrix} \mathbf{S}(\hat{\mathbf{F}})^T \\ \mathbf{n}^{k,j} \end{bmatrix}^{\#} \cdot \boldsymbol{\Sigma}_p^2 \cdot \begin{bmatrix} \mathbf{S}(\hat{\mathbf{F}})^T \\ \mathbf{n}^{k,j} \end{bmatrix}^{\#T}, \quad \boldsymbol{\Sigma}_p^2 = \begin{bmatrix} \text{cov}(\hat{\mathbf{M}}) & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}_{4\times 4} \tag{8}$$

où $\mathbf{S}(\cdot)$ est une matrice symétrique asymétrique et $\mathbf{n}^{k,j}$ est un vecteur normal de la surface. Cette expression intègre les propriétés statistiques du bruit de mesure via la matrice 3×3 $\text{cov}(\hat{\mathbf{M}})$ et inclut également des éléments nuls car la contrainte géométrique ne dépend pas des erreurs de mesure.

Pour améliorer la précision de l'algorithme et sa robustesse vis-à-vis du bruit de mesure, une étape supplémentaire est introduite basée sur une version simplifiée du système d'équilibre statique original (1), qui est linéarisé au voisinage du solution $\hat{\mathbf{F}}, \hat{\mathbf{p}}$ fournie par l'algorithme conventionnel. Il peut être facilement prouvé que le système linéarisé souhaité peut être présenté sous la forme suivante

$$\left[\mathbf{J}_k(\mathbf{q}^{(k)}, \hat{\mathbf{p}}) \quad \mathbf{H}_k(\mathbf{q}^{(k)}, \hat{\mathbf{F}})\right] \cdot \begin{bmatrix} \delta\mathbf{F} \\ \delta\mathbf{p} \end{bmatrix} = \delta\boldsymbol{\tau}^{(k)} \tag{9}$$

où $\delta\mathbf{F}$, $\delta\mathbf{p}$ sont respectivement des vecteurs de correction pour la force d'interaction et son point d'application, $\delta\boldsymbol{\tau}^{(k)}$ est le vecteur des résidus de couple, $\mathbf{J}_k$ est la matrice jacobienne du manipulateur et la matrice $\mathbf{H}_k$ est exprimée par

$$\mathbf{H}_k\left(\mathbf{q}^{(k)}, \mathbf{F}\right) = \left[\mathbf{F} \times \mathbf{e}_1 \quad ... \quad \mathbf{F} \times \mathbf{e}_k\right] \tag{10}$$

où $\mathbf{e}_k$ est le $k$ième axe de rotation de l'articulation.

De plus, pour tenir compte du fait que les propriétés statistiques des erreurs de mesure de couple ne sont pas identiques et diffèrent d'une articulation à l'autre, la technique des moindres carrés pondérés peut être appliquée. Comme on le sait, pour le bruit de mesure non corrélé, la meilleure estimation est obtenue si les poids sont égaux à l'inverse des variances de mesure $1/(\sigma_\tau^i)^2$, $i = 1, ...k$. Cela conduit au problème classique d'optimisation quadratique avec contrainte d'égalité linéaire. Par conséquent, les corrections souhaitées $\delta\mathbf{F}$, $\delta\mathbf{p}$ peuvent être obtenues à partir des conditions d'optimal-

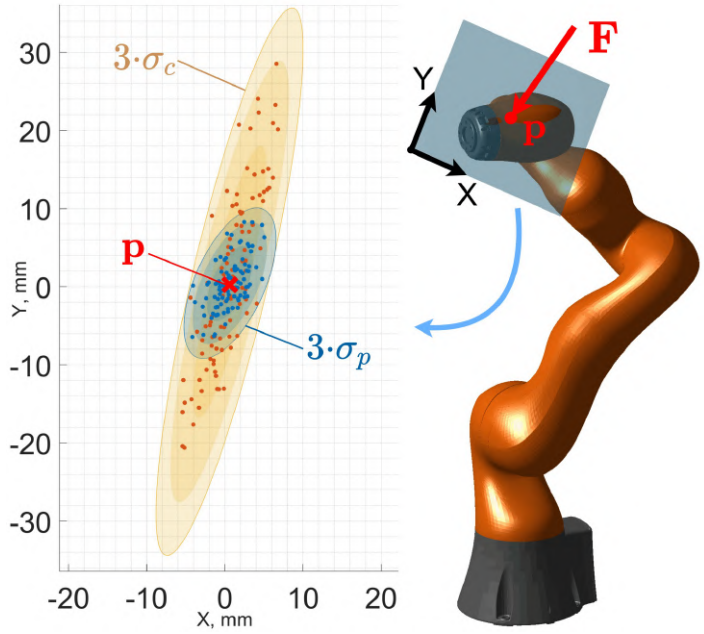


Figure 4: Précision des méthodes conventionnelles et proposées : erreurs d'estimation du point d'application de la force $\mathbf{p}$ présentées via des ellipses de dispersion 3σ.

ité KKT, qui dans notre cas sont présentées comme

$$\begin{bmatrix} \mathbf{A}_{\mathbf{\Sigma}} & \mathbf{n}^{k,j} \\ \mathbf{n}^{k,j^T} & 0 \end{bmatrix} \cdot \begin{bmatrix} \delta\mathbf{F} \\ \delta\mathbf{p} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{B}_{\mathbf{\Sigma}} \\ 0 \end{bmatrix} \cdot \delta\boldsymbol{\tau}^{(k)} \tag{11}$$

où λ est le multiplicateur lagrangien scalaire et

$$\mathbf{A}_{\mathbf{\Sigma}} = \mathbf{J}_k \mathbf{\Sigma}_\tau^{-2} \mathbf{J}_k^T + \mathbf{H}_k \mathbf{\Sigma}_\tau^{-2} \mathbf{H}_k^T, \qquad \mathbf{B}_{\mathbf{\Sigma}} = \mathbf{J}_k \mathbf{\Sigma}_\tau^{-2} + \mathbf{H}_k \mathbf{\Sigma}_\tau^{-2} \tag{12}$$

avec $\mathbf{\Sigma}_\tau^{-2} = diag[1/(\sigma_\tau^1)^2, ...1/(\sigma_\tau^k)^2]$. Il est clair qu'une telle linéarisation peut être appliquée plusieurs fois, donnant la solution souhaitée avec une erreur d'estimation minimale. On peut également prouver que la matrice de covariance correspondante peut être exprimée comme

$$\mathrm{cov}(\hat{\mathbf{F}}, \hat{\mathbf{p}}, \lambda) = \mathbf{A}_{\Sigma\Omega}^{-1} \cdot \begin{bmatrix} \mathbf{\Sigma}_\tau^2 & 0 \\ 0 & 0 \end{bmatrix} \cdot \mathbf{A}_{\Sigma\Omega}^{-T}, \qquad \mathbf{A}_{\Sigma\Omega} = \begin{bmatrix} \mathbf{A}_\Sigma & \mathbf{n}^{k,j} \\ \mathbf{n}^{k,j^T} & 0 \end{bmatrix} \tag{13}$$

La précision de l'approche conventionnelle et proposée est représentée graphiquement sur la Fig. 4.

Ce chapitre présente quelques nouveaux résultats sur l'estimation robuste des paramètres d'interaction pour la collaboration homme-robot pour le cas spatial 3D.

Table 4: Comparaison des nhanced Approaches 1 et 2 avec un robot réel

	mean $\|\|\Delta\mathbf{p}\|\|$	**max** $\mathbf{T}_c$	**mean** $\mathbf{T}_c$	**fail** %
Enhanced Approach 1	9.01	4.95	2.29	0
Enhanced Approach 2	7.09	7.86	1.60	0.9

Note: $\Delta\mathbf{p}$ - estimation error of the force application point;
$\mathbf{T}_c$, $\mathbf{f}_c$ - time performance of the estimation algorithm;
fail - no intersection cases, resulting in empty solution set.

Les techniques développées utilisent uniquement les données de mesure des capteurs de couple internes de l'articulation. Contrairement à d'autres travaux, les mesures de couple sont supposées être corrompues par un bruit non biaisé, indépendant mais textit non distribué de manière identique, ce qui contredit l'i.i.d. habituel. hypothèse largement utilisée dans la littérature connexe. Sous de telles hypothèses, deux approches pour l'identification de la force d'interaction et de son point d'application ont été développées. L'approche 1 est basée sur l'énumération directe d'éléments de maillage triangulaire décrivant la surface du robot. L'approche 2 repose sur l'obtention de l'intersection géométrique de la solution sans contrainte du système d'équilibre statique avec le maillage triangulaire du robot, en tenant compte des cônes de frottement correspondants. Afin d'assurer la robustesse des techniques développées vis-à-vis du bruit de mesure, des améliorations ont été proposées permettant de réduire essentiellement les erreurs d'estimation.

De plus, par rapport aux autres techniques connues, celles développées sont capables d'estimer les paramètres d'interaction souhaités même dans certains *cas singuliers difficiles*. Ce dernier survient lorsque la force d'interaction est appliquée aux liaisons inférieures du manipulateur, ce qui entraîne un rang réduit du système d'équilibre statique, ou lorsque la ligne d'action de la force coupe certains axes d'articulation, ce qui entraîne des mesures de couple nulles à partir des capteurs correspondants. Dans les deux cas, il n'est pas possible d'obtenir une solution unique pour les paramètres d'interaction. Pour surmonter cette difficulté et sélectionner une solution unique parmi l'ensemble des solutions possibles, la technique spéciale de résolution de singularité est proposée. Cette technique est basée sur une heuristique inspirée de la pratique et sur l'interpolation des paramètres d'interaction obtenus aux horodatages précédents. La validité de la méthodologie développée a été confirmée par une étude de simulation.

Les **principaux avantages** des techniques d'identification des paramètres d'interaction développées dans cette section peuvent être résumées comme suit :

1) La **précision** des techniques proposées est plutôt élevée. Cela permet évidemment d'obtenir une erreur d'estimation nulle en l'absence du bruit de mesure pour l'Approche 2 et une erreur suffisamment faible dans le cas de l'Approche 1. Pour le bruit de mesure de couple réaliste observé dans de vrais robots collaboratifs, les Approches 1 et 2 ont fourni l'erreur de position de environ $6,4\ mm$ et $4,8\ mm$, tandis que l'erreur d'estimation de la force était de 13,5 % et 3,1 % en conséquence. Il est clair qu'une telle précision convient tout à fait à la plupart des applications pratiques.
2) Les **performances en temps réel** des algorithmes développés sont également meilleures que celles existantes. Pour l'étude de cas considérée, cela a permis de trouver les paramètres d'interaction souhaités en 2.3 ms et 1.8 ms en moyenne pour les Approches 1 et 2 correspondantes. Ces performances élevées de l'approche 1 sont obtenues grâce à l'exécution de code parallèle et à la nature analytique de la technique mise en uvre dans l'approche 2. Ainsi, elle convient bien aux exigences en temps réel de la gestion de l'interaction homme-robot, où le temps de réponse est critique pour des raisons de sécurité.
3) La capacité de **résolution de singularité** : les techniques développées permettent d'obtenir les paramètres d'interaction souhaités même en cas d'ambiguïté essentielle des équations mathématiques pertinentes. En utilisant l'historique d'identification, l'ensemble des solutions possibles est réduit à une seule, plus proche des paramètres d'interaction réels. Comme il ressort des études de cas pertinentes, seuls deux horodatages de l'historique précédent sont suffisants pour traverser la singularité sans perdre les paramètres d'interaction réels. Ainsi, une méthode de résolution de singularité aussi simple fournissant une solution unique pour les paramètres d'interaction contribue essentiellement à la gestion sûre et efficace de l'interaction homme-robot.

Il convient de mentionner que les approches 1 et 2 conviennent aux applications industrielles. Cependant, l'approche 1 est préférable si les propriétés statistiques du bruit des mesures sont inconnues car la première technique est essentiellement plus robuste mais plus lente et moins précise. Sinon, si les propriétés du bruit sont connues, l'approche 2 offre certains avantages. Il permet d'obtenir une amélioration de 4x et 1,3x de la précision de la force d'interaction et de son estimation du point d'application en conséquence, ainsi qu'une accélération de 1,3x. Le chapitre suivant présentera quelques exemples d'application et des études expérimentales connexes confirmant l'efficacité des techniques développées dans un environnement réel.

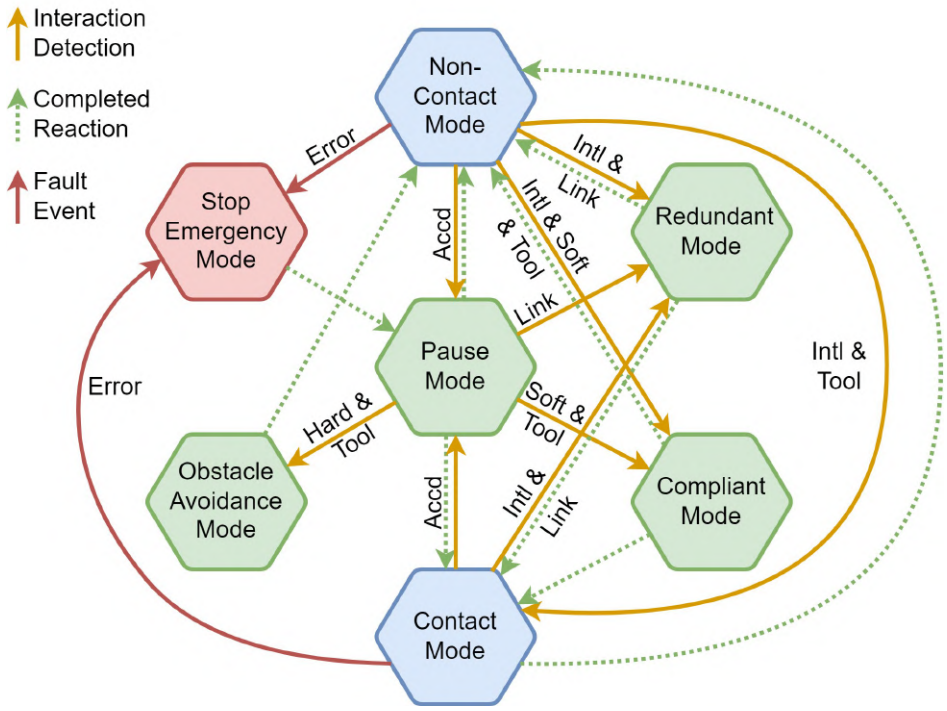


Figure 5: Machine à états finis proposée avec ses états et transitions. Les *états* sont les modes de comportement du robot et les *transitions* sont des basculements entre les états en fonction des paramètres et caractéristiques d'interaction : appliqués au niveau de l'outil ou du lien intermédiaire (Link/Tool) ; ayant une nature douce ou dure (Soft/Hard); intention d'interaction (Intl/Accd). Seuls les états et transitions les plus importants sont affichés

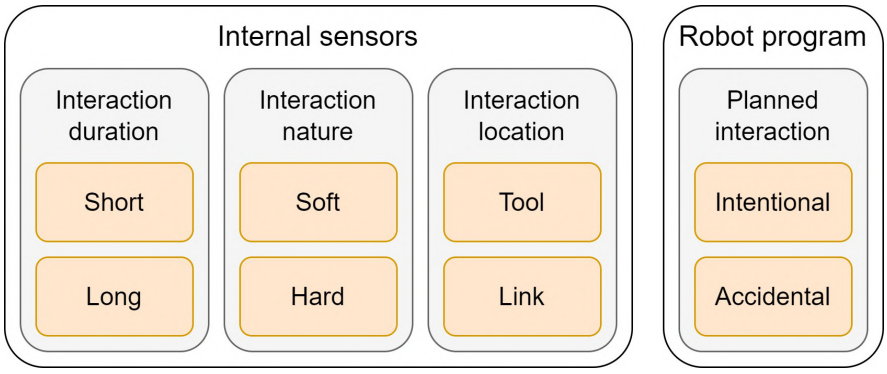


Figure 6: Diagramme des classes d'interaction. En utilisant uniquement les capteurs internes du robot, il est possible de déterminer la durée, la nature et l'emplacement de l'interaction. La possibilité d'une interaction intentionnelle ou accidentelle pourrait être prédéfinie par le programme du robot.

Le cinquième chapitre est consacré au développement de la stratégie de gestion adaptative de l'interaction, qui couvre toutes les phases de l'interaction depuis sa détection initiale, son identification jusqu'à sa classification et sa réaction, permettant de modifier le comportement du robot afin d'améliorer la sécurité et efficacité de la cellule de travail collaboratif. La validation expérimentale avec un robot réel a été réalisée afin de tester la robustesse et l'amélioration de la précision de la technique d'identification d'interaction proposée. La configuration expérimentale est illustrée à la Fig. 4 et comprend un robot KUKA LBR iiwa 14 équipé d'un capteur de couple à 1 degré de liberté dans chaque articulation.

Stratégie de gestion d'interaction adaptative axée sur la modification dynamique du comportement du robot en fonction des paramètres d'interaction. Il est nécessaire de créer un algorithme de contrôle qui peut basculer entre différentes réactions et un ensemble de politiques de réaction. Le résultat de l'identification des paramètres d'interaction sera utilisé pour estimer les réactions souhaitées du robot qui peuvent être choisies pour l'ensemble de politiques prédéfinies ou de stratégies adaptatives. Pour cet exemple pratique, cinq réactions de base du robot ont été choisies : exécution normale (faible conformité), pause, réaction du coude (conformité élevée du ddl redondant), réaction de l'effecteur final (évitement d'obstacles) et mode conforme (conformité élevée).

L'objectif principal du robot collaboratif est d'être sûr pour les humains, l'environnement et eux-mêmes, et d'exécuter simultanément la tâche le plus efficacement possible. Pour atteindre cet objectif, le robot doit avoir une stratégie pour chaque collision possible et un interrupteur qui activera et désactivera cette stratégie. Ici, une machine à états finis est proposée pour cette tâche et présentée sur la Fig. 5, avec les types de collisions suivants par emplacement : dans l'effecteur, dans le coude; par nature : soft, hard ; par tâche en cours : volontaire, accidentel. Les états de la machine à états finis sont des stratégies de réaction, les événements de contact sont des transitions et les propriétés de contact sont des conditions de transition. L'utilisateur final peut définir des états et des transitions entre eux en fonction du comportement souhaité du robot. Chaque état comprend plusieurs actions du robot, chaque transition a une condition sur les classes de collision.

Les propriétés de contact, autres que son emplacement, peuvent être trouvées en évaluant les caractéristiques de la collision. Selon le contexte d'un événement de collision, le résultat sera différent. Ici, le contexte d'une collision doit être compris en se basant uniquement sur les capteurs internes du robot. La classification commence lorsqu'une collision est détectée. Actuellement, 6 critères ont été définis pour l'événement de collision, mais il pourrait être facilement étendu. Le schéma de classification

est présenté sur la Fig. 6. La classe accidentelle/volontaire est donnée au stade de la programmation du robot et dépend d'une tâche spécifique. Au lieu de cela, les classes simples/continues, souples/dures et statiques/dynamiques sont estimées en ligne à partir des propriétés de la collision. En particulier l'identification des classes soft/hard effectuée par le réseau neuronal profond.

Le chapitre est finalisé par l'exemple pratique utilisant le module d'identification, de classification et de réaction. L'exemple comprend des expériences avec un véritable robot collaboratif KUKA iiwa LBR 14, qui effectue une opération sans contact. À des intervalles aléatoires, certaines interactions physiques peuvent se produire, y compris des contacts accidentels avec un bras humain, des contacts avec un humain pour une collaboration et des contacts accidentels avec l'environnement dynamique.

La conclusion présente les principaux apports de la thèse.

Les contributions de la thèse

La thèse porte sur le développement de nouvelles techniques de modélisation de l'interaction physique entre un humain et un robot avec une compliance adaptative afin d'améliorer la sécurité humaine et les performances de la cellule de travail homme-robot. Une attention particulière a été accordée aux problèmes de précision et à l'identification des paramètres d'interaction dans des cas singuliers, qui peuvent survenir lors de l'interaction physique en raison des informations de mesure limitées fournies par les capteurs de couple du robot.

Les résultats et contributions les plus essentiels peuvent être résumés comme suit :

1) *Les nouvelles approches pour l'identification des paramètres d'interaction dans la collaboration homme-robot, permettant de trouver le lien d'interaction et d'estimer la force d'interaction et son point d'application en utilisant les mesures de couple articulaire uniquement pour les cas singuliers et non singuliers.* Ces approches sont basées sur la solution analytique du système correspondant d'équations d'équilibre statique avec des contraintes géométriques pertinentes. Cette dernière a permis de réduire significativement le temps de calcul, par rapport aux approches classiques basées sur l'optimisation numérique simple. Les méthodes développées conviennent bien aux applications industrielles réelles où le temps de réponse est critique pour des raisons de sécurité. De plus, les procédés sont capables de produire de multiples solutions potentielles, qui peuvent survenir lorsque la force d'interaction est appliquée aux liaisons inférieures ou au robot dans une configuration cinématiquement singulière.

2) *Technique d'amélioration de la précision d'identification permettant de réduire l'écart des paramètres obtenus par rapport aux paramètres réels causé par le bruit des mesures.* Les performances des techniques développées pour l'identification des paramètres d'interaction dans les cas singuliers et non singuliers sont analysées en présence de bruit de mesure. L'écart des estimations des paramètres est calculé à l'aide des propriétés statistiques du bruit des capteurs. Sur la base des résultats obtenus, certaines améliorations sont proposées afin de rendre ces approches utilisables dans des applications réelles. En outre, la réduction essentielle des erreurs d'identification est obtenue au moyen des moindres carrés pondérés et en introduisant la version simplifiée des équations d'équilibre statique d'origine, qui est linéarisée au voisinage de la solution approchée.

3) *Méthode de gestion adaptative des interactions couvrant toutes les phases du traitement des interactions et permettant de modifier le comportement du robot afin d'améliorer la sécurité et l'efficacité de la collaboration homme-robot.* Le contrôleur de gestion des interactions développé est basé sur la machine à états finis et fournit interaction physique sûre du robot avec l'opérateur et l'environnement. Le contrôleur de haut niveau obtenu intègre avec succès les algorithmes d'identification des interactions ainsi que les méthodes de classification des interactions. Ce dernier permet de définir plusieurs modes de comportement du robot et bascule entre eux en fonction des paramètres et des caractéristiques d'interaction. Dans ce cas, le comportement du robot est séparé en modes de manipulation autonome et d'interaction, où le premier est concentré sur l'exécution efficace des tâches, et le second mode est axé sur la sécurité humaine.

Les résultats théoriques obtenus ont été validés via les études expérimentales qui traitent de l'interaction humaine avec le robot collaboratif KUKA LBR iiwa 14. Lors de la validation expérimentale des techniques d'identification des paramètres d'interaction, une attention particulière a été accordée à la précision d'identification, aux performances temporelles et capacité à traiter des cas singuliers. Les résultats d'identification obtenus pour les cas non singuliers et singuliers montrent que les approches proposées dépassent celles existantes par toutes les mesures de performance considérées. L'application pratique du contrôleur de gestion d'interaction adaptative a prouvé la validité des techniques développées. Le contrôleur de haut niveau obtenu intègre avec succès les algorithmes d'identification des interactions ainsi que les méthodes de classification des interactions. Dans ce contrôleur, plusieurs modes de comportement du robot sont définis et les commutations entre eux sont effectuées en fonction des paramètres et des caractéristiques d'interaction. Ce dernier permet d'assurer la sécurité de l'opérateur humain lors du processus de collaboration avec le robot.

En pratique, les apports de cette thèse peuvent être utilisés pour améliorer la sécurité humaine en présence d'un robot collaboratif. Les techniques d'identification d'interaction développées permettent de surveiller en continu l'amplitude du vecteur de force d'interaction pour éviter la violation des valeurs d'interaction sûres. S'il est violé, l'algorithme de gestion des interactions adaptatives permet de modifier le comportement du robot pour éviter les blessures en utilisant différents scénarios et une commutation dynamique entre eux. De plus, les capacités en temps réel des techniques développées permettent de réagir presque instantanément à toute situation potentiellement dangereuse.

Les méthodes développées pourraient non seulement améliorer la sécurité de l'humain dans l'espace de travail du robot, mais également augmenter les performances de la cellule de travail en activant différents scénarios de collaboration. En particulier, les paramètres d'interaction identifiés peuvent être utilisés pour communiquer entre l'humain et le robot en permettant des scénarios de collaboration spéciaux, selon comment et où l'humain touche le robot. En outre, les mêmes paramètres peuvent être utilisés pour la compensation des erreurs de conformité, ce qui peut augmenter la précision du robot en présence de la force d'interaction, améliorant ainsi les performances globales de la cellule de travail robotique.

Limitations des résultats obtenus

Malgré plusieurs avantages essentiels, il existe encore quelques limites dans les résultats obtenus qui sont présentés ci-dessous :

1) Il a été supposé que seuls les capteurs de couple internes des articulations du robot étaient disponibles. Dans ce travail, pour identifier les paramètres d'interaction, les mesures des capteurs intégrés au robot ont été utilisées, qui sont actuellement l'un des capteurs les plus populaires en robotique collaborative. Cependant, les robots industriels conventionnels, largement utilisés dans la fabrication, ne sont généralement pas équipés de tels capteurs, ce qui limite le domaine d'application des techniques développées.
2) On a supposé que le robot avait une structure cinématique en série. Dans le cas de la cinématique parallèle, le système pertinent d'équations d'équilibre statique est plus compliqué à cause des articulations passives, ce qui augmente encore le déficit de rang du système considéré. Cependant, l'application des robots parallèles pour la collaboration homme-robot est aujourd'hui assez limitée.
3) On a supposé qu'il n'y avait qu'une seule interaction entre l'humain et le robot. En général, pour plusieurs interactions simultanées, il n'est pas possible d'iden-

tifier de manière fiable les paramètres d'interaction souhaités. Ce problème est lié à l'insuffisance des informations sensorielles d'un robot collaboratif typique équipé des capteurs de couple articulaire.

4) On a supposé qu'il n'y avait pas de systèmes de vision externes ou de capteurs de proximité disponibles. Ici, le contrôleur de gestion d'interaction développé peut modifier le comportement du robot après l'interaction uniquement. Dans de telles circonstances, il est impossible d'utiliser les techniques d'évitement d'obstacles du robot avant l'interaction physique puisque l'obstacle n'est pas visible pour le robot. Ce problème peut affecter la sécurité de l'homme dans l'espace de travail du robot.

Néanmoins, dans le cadre du domaine d'application considéré, les limitations liées aux hypothèses du modèle ne sont pas critiques car la précision souhaitée a été atteinte dans l'étude expérimentale. De plus, il est possible d'étendre/modifier davantage l'approche développée afin d'appliquer d'autres techniques, qui ne font qu'améliorer la précision globale du processus d'identification.

Perspectives et travaux futurs

Pour généraliser les résultats obtenus et élargir son domaine d'application, il est raisonnable de poursuivre la recherche dans plusieurs directions et de se concentrer sur les problématiques suivantes :

1) Amélioration de la technique d'identification d'interaction développée pour le cas de capteurs supplémentaires internes ou externes. Ici, les capteurs externes pourraient inclure des systèmes basés sur la vision ou une peau tactile appliquée à la surface du robot. Les capteurs internes pourraient être basés sur des capteurs de force/couple multiaxes supplémentaires installés à l'intérieur des liaisons du robot. En ajoutant de tels capteurs, le problème d'identification n'est plus singulier et fournit généralement une solution unique pour les paramètres recherchés.

2) Développement du contrôleur de gestion des interactions pour un processus de fabrication spécifique. Dans ce travail, le contrôleur s'est concentré sur le prototype de laboratoire mais l'adapter aux processus industriels réels permettrait d'améliorer la sécurité et l'efficacité de la cellule de travail associée. Par exemple, la tâche d'assemblage est très populaire pour les robots collaboratifs et peut être utilisée comme cible pour développer un nouveau contrôleur de gestion d'interaction pour cette opération particulière.

3) Extension du domaine d'application de la technique d'identification des paramètres d'interaction pour différents types de robots. En particulier, il est raisonnable d'étendre cette technique aux robots humanoïdes, qui sont généralement utilisés dans l'environnement riche en contacts. Ainsi, la mise en place d'une technique d'identification des interactions pertinentes peut être utile pour améliorer leurs performances.

BIBLIOGRAPHY

[1] S. Robla-Gómez, V. M. Becerra, J. R. Llata, E. Gonzalez-Sarabia, C. Torre-Ferrero, and J. Perez-Oria, "Working together: A review on safe human-robot collaboration in industrial environments", *IEEE Access*, vol. 5, pp. 26 754–26 773, 2017.

[2] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications", *Mechatronics*, vol. 55, pp. 248–266, 2018.

[3] J. A. Falco, J. A. Marvel, R. J. Norcross, *et al.*, "Collaborative robotics: Measuring blunt force impacts on humans", *Chest*, vol. 140, no. 210, p. 45, 2012.

[4] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control", *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 882–893, 2016.

[5] S. Haddadin, A. Albu-Schaffer, A. De Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical human-robot interaction", in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 3356–3363.

[6] M. Lippi and A. Marino, "Enabling physical human-robot collaboration through contact classification and reaction", in *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2020, pp. 1196–1203.

[7] B. D. Argall and A. G. Billard, "A survey of tactile human–robot interactions", *Robotics and autonomous systems*, vol. 58, no. 10, pp. 1159–1176, 2010.

[8] U. E. Ogenyi, J. Liu, C. Yang, Z. Ju, and H. Liu, "Physical human-robot collaboration: Robotic systems, learning methods, collaborative strategies, sensors, and actuators", *IEEE transactions on cybernetics*, 2019.

[9] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification", *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[10] J. E. Colgate, J. Edward, M. A. Peshkin, and W. Wannasuphoprasit, "Cobots: Robots for collaboration with human operators", 1996.

[11] F. Vicentini, "Terminology in safety of collaborative robotics", *Robotics and Computer-Integrated Manufacturing*, vol. 63, p. 101 921, 2020.

[12] A. A. Malik and A. Bilberg, "Developing a reference model for human–robot interaction", *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 13, no. 4, pp. 1541–1547, 2019.

[13] F. Flacco and A. De Luca, "Safe physical human-robot collaboration", in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 2072–2072.

[14] E. ISO 10218–1, "10218: Robots and robotic devices–safety requirements for industrial robots–part 1: Robots", *International Organization for Standardization: Geneva, Switzerland*, 2011.

[15] E. ISO 10218–2, "Robots and robotic devices–safety requirements for industrial robots–part 2: Robot systems and integration", *International Organization for Standardization: Geneva, Switzerland*, 2011.

[16] E. ISO/TS 15066, "Robots and robotic devices-collaborative robots", *International Organization for Standardization: Geneva, Switzerland*, 2016.

[17] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human–robot interaction", *Mechanism and Machine Theory*, vol. 43, no. 3, pp. 253–270, 2008.

[18] A. De Luca, A. Albu-Schaffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the dlr-iii lightweight manipulator arm", in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 1623–1630.

[19] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion learning and adaptive impedance for robot control during physical interaction with humans", in *IEEE International Conference on Robotics and Automation*, 2011, pp. 4326–4332.

[20] A. Bolotnikova, S. Courtois, and A. Kheddar, "Multi-contact planning on humans for physical assistance by humanoid", *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 135–142, 2019.

[21] T. Harden, K. A. Schroeder, and M. W. Pryor, "On the use of joint torque sensors for collision detection in a confined environment", Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 2011.

[22] S. Takakura, T. Murakami, and K. Ohnishi, "An approach to collision detection and recovery motion in industrial robot", in *15th Annual Conference of IEEE Industrial Electronics Society*, 1989, pp. 421–426.

[23] Y. J. Heo, D. Kim, W. Lee, H. Kim, J. Park, and W. K. Chung, "Collision detection for industrial collaborative robots: A deep learning approach", *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 740–746, 2019.

[24] S. Golz, C. Osendorfer, and S. Haddadin, "Using tactile sensation for learning contact knowledge: Discriminate collision from physical interaction", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3788–3794.

[25] A. Kouris, F. Dimeas, and N. Aspragathos, "A frequency domain approach for contact type distinction in human–robot collaboration", *IEEE robotics and automation letters*, vol. 3, no. 2, pp. 720–727, 2018.

[26] S. Parusel, S. Haddadin, and A. Albu-Schäffer, "Modular state-based behavior control for safe human-robot interaction: A lightweight control architecture for a lightweight robot", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 4298–4305.

[27] E. Magrini and A. De Luca, "Human-robot coexistence and contact handling with redundant robots", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 4611–4617.

[28] N. Likar and L. lajpah, "External joint torque-based estimation of contact information", *International Journal of Advanced Robotic Systems*, vol. 11, no. 7, p. 107, 2014.

[29] L. Manuelli and R. Tedrake, "Localizing external contact using proprioceptive sensors: The contact particle filter", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 5062–5069.

[30] A. Zwiener, R. Hanten, C. Schulz, and A. Zell, "Armcl: Arm contact point localization via monte carlo localization", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 7105–7111.

[31] A. Cirillo, F. Ficuciello, C. Natale, S. Pirozzi, and L. Villani, "A conformable force/tactile skin for physical human–robot interaction", *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 41–48, 2015.

[32] F. J. A. Chavez, J. Kangro, S. Traversaro, F. Nori, and D. Pucci, "Contact force and joint torque estimation using skin", *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3900–3907, 2018.

[33] A. Jain, M. D. Killpack, A. Edsinger, and C. C. Kemp, "Reaching in clutter with whole-arm tactile sensing", *The International Journal of Robotics Research*, vol. 32, no. 4, pp. 458–482, 2013.

[34] R. S. Dahiya, P. Mittendorfer, M. Valle, G. Cheng, and V. J. Lumelsky, "Directions toward effective utilization of tactile skin: A review", *IEEE Sensors Journal*, vol. 13, no. 11, pp. 4121–4138, 2013.

[35] E. Magrini, F. Ferraguti, A. J. Ronga, F. Pini, A. De Luca, and F. Leali, "Human-robot coexistence and interaction in open industrial cells", *Robotics and Computer-Integrated Manufacturing*, vol. 61, p. 101 846, 2020.

[36] A. De Luca and F. Flacco, "Integrated control for phri: Collision avoidance, detection, reaction and collaboration", in *EEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, 2012, pp. 288–295.

[37] A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, and P. Fraisse, "Collaborative manufacturing with physical human–robot interaction", *Robotics and Computer-Integrated Manufacturing*, vol. 40, pp. 1–13, 2016.

[38] R.-J. Halme, M. Lanz, J. Kämäräinen, R. Pieters, J. Latokartano, and A. Hietanen, "Review of vision-based safety systems for human-robot collaboration", *Procedia CIRP*, vol. 72, pp. 111–116, 2018.

[39] J. G. Garca, A. Robertsson, J. G. Ortega, and R. Johansson, "Sensor fusion for compliant robot motion control", *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 430–441, 2008.

[40] W. McMahan, J. M. Romano, and K. J. Kuchenbecker, "Using accelerometers to localize tactile contact events on a robot arm", in *Workshop on Advances in Tactile Sensing and Touch-Based Human-Robot Interaction, ACM/IEEE International Conference on Human-Robot Interaction*, Citeseer, 2012, pp. 1–6.

[41] I. Iakubchik, A. Iakubchik, and Y. Nakamura, "Acoustic determination of contact on the exterior surface of the robot", in *IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, IEEE, 2021, pp. 387–389.

[42] M. Koga, K. Kosuge, K. Furuta, and K. Nosaki, "Coordinated motion control of robot arms based on the virtual internal model", *IEEE Transactions on Robotics and Automation*, vol. 8, no. 1, pp. 77–85, 1992.

[43] M. Geravand, F. Flacco, and A. De Luca, "Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture", in *IEEE International Conference on Robotics and Automation*, 2013, pp. 4000–4007.

[44] S. Lu, J. H. Chung, and S. A. Velinsky, "Human-robot collision detection and identification based on wrist and base force/torque sensors", in *IEEE international Conference on Robotics and Automation*, 2005, pp. 3796–3801.

[45] G. Buondonno and A. De Luca, "Combining real and virtual sensors for measuring interaction forces and moments acting on a robot", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 794–800.

[46] E. Matheson, R. Minto, E. G. Zampieri, M. Faccio, and G. Rosati, "Human–robot collaboration in manufacturing applications: A review", *Robotics*, vol. 8, no. 4, p. 100, 2019.

[47] A. De Luca and R. Mattone, "Actuator failure detection and isolation using generalized momenta", in *IEEE international conference on robotics and automation*, vol. 1, 2003, pp. 634–639.

[48] G. Garofalo, N. Mansfeld, J. Jankowski, and C. Ott, "Sliding mode momentum observers for estimation of external torques and joint acceleration", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6117–6123.

[49] W.-H. Chen, D. J. Ballance, P. J. Gawthrop, and J. O'Reilly, "A nonlinear disturbance observer for robotic manipulators", *IEEE Transactions on industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.

[50] V. Sotoudehnejad, A. Takhmar, M. R. Kermani, and I. G. Polushin, "Counteracting modeling errors for sensitive observer-based manipulator collision detection", in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 4315–4320.

[51] J.-B. Song *et al.*, "Collision detection algorithm robust to model uncertainty", *International Journal of Control, Automation and Systems*, vol. 11, no. 4, pp. 776–781, 2013.

[52] J. A. Moreno and M. Osorio, "A lyapunov approach to second-order sliding mode controllers and observers", in *IEEE conference on decision and control*, 2008, pp. 2856–2861.

[53] A. Mohammadi, M. Tavakoli, H. J. Marquez, and F. Hashemzadeh, "Nonlinear disturbance observer design for robotic manipulators", *Control Engineering Practice*, vol. 21, no. 3, pp. 253–267, 2013.

[54] A. Bolotnikova, S. Courtois, and A. Kheddar, "Contact observer for humanoid robot pepper based on tracking joint position discrepancies", in *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2018, pp. 29–34.

[55] A. De Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control", in *IEEE international conference on robotics and automation*, 2005, pp. 999–1004.

[56] T. Tomi, C. Ott, and S. Haddadin, "External wrench estimation, collision detection, and reflex reaction for flying robots", *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1467–1482, 2017.

[57] A. Zwiener, C. Geckeler, and A. Zell, "Contact point localization for articulated manipulators with proprioceptive sensors and machine learning", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 323–329.

[58] Z. Liu, F. Yu, L. Zhang, and T. Li, "Real-time estimation of sensorless planar robot contact information", *Journal of Robotics and Mechatronics*, vol. 29, no. 3, pp. 557–565, 2017.

[59] J. Bimbo, C. Pacchierotti, N. Tsagarakis, and D. Prattichizzo, "Collision detection and isolation on a robot using joint torque sensing", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[60] T. Pang, J. Umenberger, and R. Tedrake, "Identifying external contacts from joint torque measurements on serial robotic arms and its limitations", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6476–6482.

[61] J. Liang and O. Kroemer, "Contact localization for robot arms in motion without torque sensing", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6322–6328.

[62] T. Hellebrekers, N. Chang, K. Chin, M. J. Ford, O. Kroemer, and C. Majidi, "Soft magnetic tactile skin for continuous force and location estimation using neural networks", *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 3892–3898, 2020.

[63] A.-N. Sharkawy, P. N. Koustoumpardis, and N. Aspragathos, "Human–robot collisions detection for safe human–robot interaction using one multi-input–output neural network", *Soft Computing*, vol. 24, no. 9, pp. 6687–6719, 2020.

[64] T. Xu, J. Fan, Q. Fang, Y. Zhu, and J. Zhao, "A new robot collision detection method: A modified nonlinear disturbance observer based-on neural networks", *Journal of Intelligent & Fuzzy Systems*, vol. 38, no. 1, pp. 175–186, 2020.

[65] N. Rotella, S. Schaal, and L. Righetti, "Unsupervised contact learning for humanoid estimation and control", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 411–417.

[66] M. Kaboli, D. Feng, K. Yao, P. Lanillos, and G. Cheng, "A tactile-based framework for active object learning and discrimination using multimodal robotic skin", *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2143–2150, 2017.

[67] K. Narukawa, T. Yoshiike, K. Tanaka, and M. Kuroda, "Real-time collision detection based on one class svm for safe movement of humanoid robot", in *IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 791–796.

[68] T. Ito, K. Ayusawa, E. Yoshida, and A. Kheddar, "Experimental study for controller-friendly contact estimation for humanoid robot", in *IEEE International Conference on Advanced Robotics and its Social Impacts (ARSO)*, 2019, pp. 28–33.

[69] M. Iskandar, O. Eiberger, A. Albu-Schäffer, A. De Luca, and A. Dietrich, "Collision detection, identification, and localization on the dlr sara robot with sensing redundancy", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3111–3117.

[70] F. Cavenago, A. M. Giordano, and M. Massari, "Contact detection, isolation and estimation for orbital robots through an observer based on a centroid-joints dynamics", *Acta Astronautica*, vol. 181, pp. 40–51, 2021.

[71] F. D'Ippolito, F. Alonge, and E. Cucco, "Contact estimation in robot interaction", *International Journal of Advanced Robotic Systems*, vol. 11, no. 7, p. 96, 2014.

[72] T. Möller and B. Trumbore, "Fast, minimum storage ray-triangle intersection", *Journal of graphics tools*, vol. 2, no. 1, pp. 21–28, 1997.

[73] C. Wächter and A. Keller, "Instant ray tracing: The bounding interval hierarchy.", *Rendering Techniques*, vol. 2006, no. 139-149, p. 130, 2006.

[74] J. J. Jiménez, R. J. Segura, and F. R. Feito, "A robust segment/triangle intersection algorithm for interference tests. efficiency study", *Computational Geometry*, vol. 43, no. 5, pp. 474–492, 2010.

[75] F. Flacco, A. Paolillo, and A. Kheddar, "Residual-based contacts estimation for humanoid robots", in *IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 409–415.

[76] J. Vorndamme, M. Schappler, and S. Haddadin, "Collision detection, isolation and identification for humanoids", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4754–4761.

[77] S. Mamedov and S. Mikhel, "Practical aspects of model-based collision detection", *Frontiers in Robotics and AI*, vol. 7, p. 162, 2020.

[78] N. Likar and L. lajpah, "Estimation of contact information using nonlinear optimization", in *IEEE International Conference on Robotics in Alpe-Adria-Danube Region (RAAD)*, 2014, pp. 1–6.

[79] D. Finkel, "Direct optimization algorithm user guide", North Carolina State University. Center for Research in Scientific Computation, Tech. Rep., 2003.

[80] C. Hennersperger, B. Fuerst, S. Virga, *et al.*, "Towards mri-based autonomous robotic us acquisitions: A first feasibility study", *IEEE transactions on medical imaging*, vol. 36, no. 2, pp. 538–548, 2017.

[81] P. T. Choi, K. C. Lam, and L. M. Lui, "Flash: Fast landmark aligned spherical harmonic parameterization for genus-0 closed brain surfaces", *SIAM Journal on Imaging Sciences*, vol. 8, no. 1, pp. 67–94, 2015.

[82] M. Grant and S. Boyd, *Cvx: Matlab software for disciplined convex programming, version 2.1*, 2014.

[83] J. Mattingley and S. Boyd, "Cvxgen: A code generator for embedded convex optimization", *Optimization and Engineering*, vol. 13, no. 1, pp. 1–27, 2012.

[84] K. Crane, C. Weischedel, and M. Wardetzky, "Geodesics in heat: A new approach to computing distance based on heat flow", *ACM Transactions on Graphics (TOG)*, vol. 32, no. 5, pp. 1–11, 2013.

[85] A. Jacobson *et al.*, *gptoolbox: Geometry processing toolbox*, 2018.

[86] D.-M. Yan, B. Lévy, Y. Liu, F. Sun, and W. Wang, "Isotropic remeshing with fast and exact computation of restricted voronoi diagram", in *Computer graphics forum*, vol. 28, 2009, pp. 1445–1454.

[87] L. Manuelli, "Localizing external contact using proprioceptive sensors: The contact particle filter", Ph.D. dissertation, Massachusetts Institute of Technology, 2018.

[88] A. Zwiener, R. Hanten, C. Schulz, and A. Zell, “ARMCL - ARM Contact point Localization via Monte Carlo Localization”, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[89] J. Vorndamme and S. Haddadin, “Rm-code: Proprioceptive real-time recursive multi-contact detection, isolation and identification”, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 6307–6314.

[90] J. Fryman and B. Matthias, “Safety of industrial robots: From conventional to collaborative applications”, in *ROBOTIK*, VDE, 2012, pp. 1–5.

[91] J. Heinzmann and A. Zelinsky, “The safe control of human-friendly robots.”, in *IROS*, 1999, pp. 1020–1025.

[92] J. Guiochet, D. Powell, É. Baudin, and J.-P. Blanquart, “Online safety monitoring using safety modes”, in *Workshop on Technical Challenges for Dependable Robots in Human Environments*, 2008, pp. 1–13.

[93] G. Cioffi, S. Klose, and A. Wahrburg, “Data-efficient online classification of human-robot contact situations”, in *IEEE European Control Conference (ECC)*, 2020, pp. 608–614.

[94] N. Briquet-Kerestedjian, A. Wahrburg, M. Grossard, M. Makarov, and P. Rodriguez-Ayerbe, “Using neural networks for classifying human-robot contact situations”, in *IEE European Control Conference (ECC)*, 2019, pp. 3279–3285.

[95] C.-N. Cho, J.-H. Kim, Y.-L. Kim, J.-B. Song, and J.-H. Kyung, “Collision detection algorithm to distinguish between intended contact and unexpected collision”, *Advanced Robotics*, vol. 26, no. 16, pp. 1825–1840, 2012.

[96] P. Tchatchoua, G. Graton, M. Ouladsine, J. Muller, A. Traoré, and M. Juge, “1d resnet for fault detection and classification on sensor data in semiconductor manufacturing”, in *IEEE International Conference on Control, Automation and Diagnosis (ICCAD)*, 2022, pp. 1–6.

[97] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition”, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[98] S. Kumra and C. Kanan, “Robotic grasp detection using deep convolutional neural networks”, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 769–776.

[99] S. Hershey, S. Chaudhuri, D. P. Ellis, *et al.*, “Cnn architectures for large-scale audio classification”, in *ieee international conference on acoustics, speech and signal processing (icassp)*, 2017, pp. 131–135.

COLLEGES SCIENCES
BRETAGNE POUR L'INGENIEUR
LOIRE ET LE NUMERIQUE

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Titre : Modélisation des interactions physiques entre un humain et un robot avec élasticité adaptative

Mot clés : Interaction Physique Homme-Robot, Collaboration Homme-Robot, Identification des Paramètres d'Interaction, Résolution de Singularité, Contrôle d'Interaction Adaptatif.

Résumé : La thèse porte sur la modélisation des interactions physiques entre un humain et un robot, ce qui est extrêmement important pour la conception et le développement de cellules de travail collaboratives modernes pour de nouvelles applications industrielles. Elle se concentre sur le développement de techniques d'identification des interactions afin d'améliorer la sécurité des personnes et les performances des cellules de travail homme-robot. Une attention particulière est accordée aux problèmes de précision causés par les données de mesures bruitées et l'identification des paramètres d'interaction dans des cas singuliers, qui peuvent survenir lors de l'interaction physique en raison d'informations de mesures limitées. Elle présente de nouvelles techniques pour calculer la force d'interaction et son point d'application en utilisant uniquement les données de mesure obtenues à partir des capteurs de couple de l'articulation interne du robot. De plus, le contrôleur de gestion d'interaction adaptative est développé, celui-ci intègre l'identification d'interaction afin d'assurer la sécurité humaine en changeant le mode de comportement du robot. La validité des approches développées et leur efficacité ont été confirmées par une étude expérimentale impliquant une collaboration entre un opérateur et le robot KUKA LBR iiwa 14.

Title: Modeling of physical interactions between a human and a robot with adaptive compliance

Keywords: Physical Human-Robot Interaction, Safe Robotics, Human-Robot Collaboration, Interaction Parameters Identification, Singularity Resolution, Adaptive Interaction Handling.

Abstract: The thesis deals with the modeling of physical interactions between a human and a robot, which is extremely important for the design and development of modern collaborative work cells for new industrial applications. It focuses on the development of interaction identification techniques in order to improve human safety and human-robot work cell performance. Special attention is paid to the accuracy issues caused by the noisy measurements data and the interaction parameters identification in singular cases, which can arise during the physical interaction because of limited measurements information. This thesis also presents new techniques for computing the interaction force and its application point using measurement data obtained from the robot internal joint torque sensors only. In addition, the adaptive interaction handling controller is developed by integrating the interaction identification in order to ensure human safety by changing the robot behavior mode. The validity of the developed approaches and their efficiency was confirmed by an experimental study involving collaboration between an operator and the KUKA LBR iiwa 14 robot.