Human-Robot Collaboration for Kinesthetic Teaching

Salt Ducaju, Julian

2023

Document Version:
Publisher's PDF, also known as Version of record

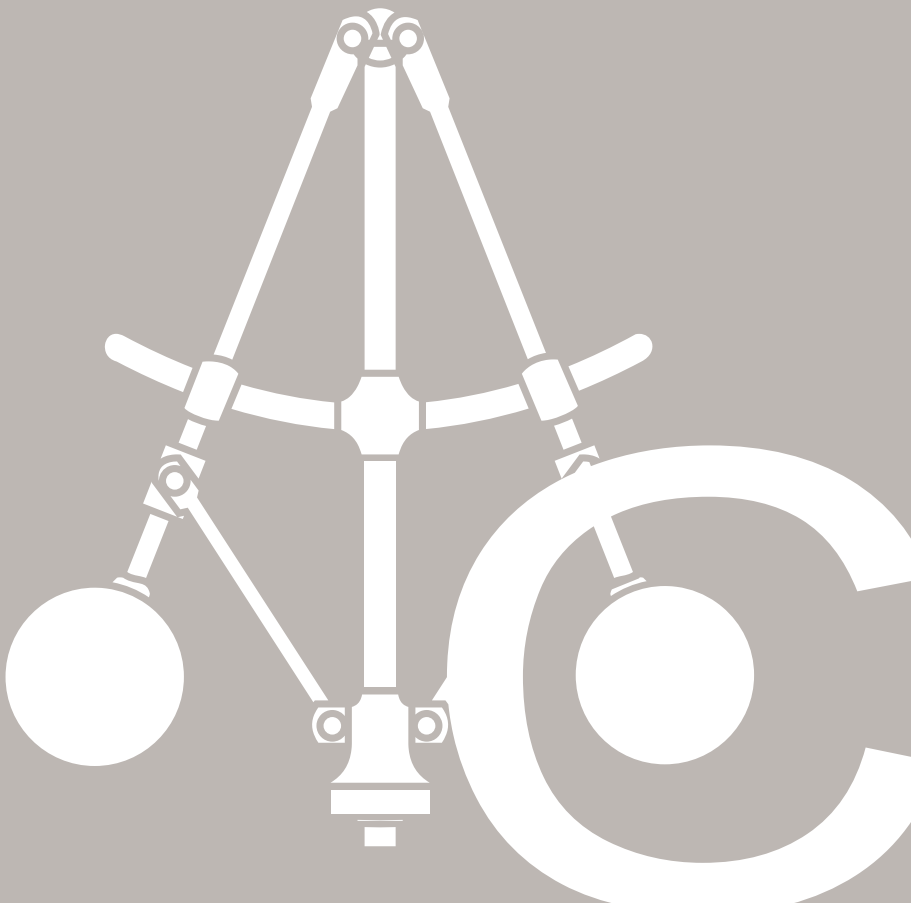
Human–Robot Collaboration for Kinesthetic Teaching

JULIAN M. SALT DUCAJU
DEPARTMENT OF AUTOMATIC CONTROL | LUND UNIVERSITY

Printed by Media-Tryck, Lund 2023 NORDIC SWAN ECOLABEL 3041 0903

Department of Automatic Control
P.O. Box 118, 221 00 Lund, Sweden
www.control.lth.se

ISRN LUTFD2/TFRT--3278--SE
ISSN 0280–5316



Human–Robot Collaboration for Kinesthetic Teaching

Julian M. Salt Ducaju

LUND
UNIVERSITY

Department of Automatic Control

Lic. Tech. Thesis TFRT-3278
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Media-Tryck.
Lund 2022

Abstract

Recent industrial interest in producing smaller volumes of products in shorter time frames, in contrast to mass production in previous decades, motivated the introduction of human–robot collaboration (HRC) in industrial settings, as an attempt to increase flexibility in manufacturing applications by incorporating human intelligence and dexterity to these processes. This thesis presents methods for improving the involvement of human operators in industrial settings where robots are present, with a particular focus on kinesthetic teaching, *i.e.*, manually guiding the robot to define or correct its motion, since it can facilitate non-expert robot programming.

To increase flexibility in the manufacturing industry implies a loss of a fixed structure of the industrial environment, which increases the uncertainties in the shared workspace between humans and robots. Two methods have been proposed in this thesis to mitigate such uncertainty. First, null-space motion was used to increase the accuracy of kinesthetic teaching by reducing the joint static friction, or stiction, without altering the execution of the robotic task. This was possible since robots used in HRC, *i.e.*, collaborative robots, are often designed with additional degrees of freedom (DOFs) for a greater dexterity. Second, to perform effective corrections of the motion of the robot through kinesthetic teaching in partially-unknown industrial environments, a fast identification of the source of robot–environment contact is necessary. Fast contact detection and classification methods in literature were evaluated, extended, and modified to use them in kinesthetic teaching applications for an assembly task. For this, collaborative robots that are made compliant with respect to their external forces/torques (as an active safety mechanism) were used, and only embedded sensors of the robot were considered.

Moreover, safety is a major concern when robotic motion occurs in an inherently uncertain scenario, especially if humans are present. Therefore, an online variation of the compliant behavior of the robot during its manual guidance by a human operator was proposed to avoid undesired parts of the workspace of the robot. The proposed method used safety control barrier functions (SCBFs) that considered the rigid-body dynamics of the robot, and the method's stability was guaranteed using a passivity-based energy-storage formulation that includes a strict Lyapunov function.

All presented methods were tested experimentally on a real collaborative robot.

Acknowledgments

First, I would like to thank my supervisor Prof. Anders Robertsson, and my co-supervisors Prof. Rolf Johansson and Dr. Björn Olofsson, for sharing your knowledge and experience with me. I am very fortunate to have received your advice and guidance throughout these years.

I am also grateful to the administrative staff: Dr. Eva Westin, Mika Nishimura, Cecilia Edelborg Christensen, and Monika Rasmusson. Thanks for ensuring that our department runs smoothly, and especially, thanks for making my life so much easier. On this behalf, I cannot forget about all the people that have helped me on technical problems: Dr. Anders Nilsson, Anders Blomdell, Pontus Andersson, Alexander Pisarevskiy, Dr. Marcus Klang and Anton Tetov Johansson. Thanks for your great patience.

Thanks to my colleagues at the Department of Automatic Control for creating such a wonderful environment to work in, and to all the guests that have visited the department during my time here. Thanks for the interesting seminars and events, for our conversations during lunch and fika, and for being such great travel companions and friends.

Moreover, I would like to acknowledge my colleagues from the RobotLab: Prof. Jacek Malec, Dr. Elin Anna Topp, Prof. Volker Krueger, Dr. Maj Stenmark, Hampus Åström, Johan Oxenstierna, Momina Rizwan, Faseeh Ahmad, and especially to Alexander Dürr and Matthias Mayr. Thanks for the interesting discussions.

And last but not least, I am thankful to my parents, to Angelica, and to my friends, for bringing comfort and joy to my life, and for your endless support.

Contents

1. Introduction **9**
1.1 Problem Formulation 10
1.2 Background and Motivation 10
1.3 Publications 13
1.4 Thesis Outline 14
1.5 Contributions 14
1.6 Discussion and Future Research 14
1.7 Conclusion 15
Bibliography **16**
Paper I. Joint Stiction Avoidance with Null-Space Motion in Real-Time Model Predictive Control for Redundant Collaborative Robots **21**
1 Introduction 22
2 Methods 24
3 Experiments and Results 30
4 Conclusion 37
References 37
Paper II. Fast Contact Detection and Classification for Kinesthetic Teaching in Robots using only Embedded Sensors **41**
1 Introduction 42
2 Method 44
3 Experiments 48
4 Results 51
5 Discussion 53
6 Conclusion 55
References 56
Paper III. Robot Cartesian Compliance Variation for Safe Kinesthetic Teaching using Safety Control Barrier Functions **59**
1 Introduction 60
2 Mathematical Background 61

3 Method 62
4 Experiments 70
5 Results 71
6 Discussion 73
7 Conclusion 75
References 76
A Proof of Lemma 3.2 77
A. Notation Overview **81**

1

Introduction

Robots were largely adopted in industry during the second half of the twentieth century, as they were deemed as the machinery needed to automatize mass production. These industrial robots could reproduce repetitive tasks with high speed and accuracy, thus increasing the productivity of an industry whose goal at the time was to manufacture large amounts of identical products in their assembly lines [Kurfess, 2005, Ch. 1]. Human workers were then displaced out of such industrial environments, characterized by being rigid and structured to facilitate mass production.

However, recent trends in the manufacturing industry indicate a shift from mass production to mass customization [Schou et al., 2013], where smaller volumes of products are manufactured during shorter time frames. The intelligence and dexterity of human operators can be useful to increase the manufacturing flexibility required in this novel scenario [Cencen et al., 2018]. Thereby, the robots that were once used to replace human labor are currently starting to be seen as direct collaborators to humans in an industrial working structure that can best exploit the different and complementary aptitudes of humans and robots [IFR, 2018b]. Moreover, the robot manufacturing industry has addressed this trend by developing collaborative robots, also known as cobots, in recent years. The lightweight and compliant design of collaborative robots makes them better suited to share their workspace with human operators than traditional industrial robots [IFR, 2020]. There is an increasing interest in the use of collaborative robots, especially by Small and Medium-sized Enterprises (SMEs), since these robots are economically more viable than traditional industrial robots [Cencen et al., 2018; Suomalainen et al., 2022], thus providing an economically-viable entry-point to robotic automation [IFR, 2020], which historically has been restricted to large companies, since they could finance its high capital cost [IFR, 2018a].

An interesting way that humans and robots can collaborate in a manufacturing task is through physical Human–Robot Interaction (pHRI), *i.e.*, by manually guiding the robot [Hirzinger, 1986]. This method is known as kinesthetic teaching [Argall et al., 2009; Akgun et al., 2012; Wrede et al., 2013; Karayiannidis et al., 2014] and consists in leading-through the robot to program a robot trajectory, or to reprogram a segment of a pre-existing trajectory, allowing a human operator

to easily modify the robot's motion to adapt it to a manufacturing process whose requirements have changed online. Kinesthetic teaching is the most common approach for introducing human demonstrations in manufacturing applications, since it facilitates non-expert robot programming [Ravichandar et al., 2020] by assuming that a human can perform the manufacturing task efficiently [Suomalainen et al., 2022]. However, it is still a resource-intensive design activity to enable humans and robots to effectively and efficiently share a workspace [Cencen et al., 2018], which hinders the advantages that the adoption of collaborative robots by industry can provide to meet the current demand for flexibility of the industry.

1.1 Problem Formulation

The previously described circumstances have motivated the research presented in this thesis, whose objective is to improve the involvement of human operators in industrial environments where robots are present through kinesthetic teaching applications (an application example is shown in Fig. 1.1), to increase the flexibility of the manufacturing industry. Two main research questions have arisen from this research problem:

1. How to reduce the uncertainty in pHRI occurring in such dynamic collaborative industrial environments to achieve an effective human manual guidance of the robot?

2. How to increase the necessary safety during human manual guidance of the robot?

The first research question has been addressed in this thesis by

- Decreasing the uncertainty of the necessary force that a human should apply to the robot, to improve the accuracy of kinesthetic teaching.

- Quickly detecting robot–environment contacts and distinguishing their source.

and the second research question has been addressed by

- Adapting the behavior of the robot with respect to the external forces/torques applied from its environment to avoid undesired collisions.

1.2 Background and Motivation

For an accurate application of kinesthetic teaching, human operators should be comfortable with the necessary amount of force/torque needed to be applied to the robot. A source of uncertainty, or dispersion, in the relationship between the force/torque

applied to a robot and its displacement is joint static friction, or stiction [Haug et al., 1986], which can be observed in joints with zero velocity and which is caused by interactions between the asperities of the surfaces in contact in a robot joint [Bittencourt and Gunnarsson, 2012; Bagge Carlson et al., 2015]. Joint stiction is a prevalent issue in collaborative robots, since these robots are often designed with additional degrees of freedom (DOFs) to increase their dexterity [Crowe, 2022]. This leads to the possibility that, in the absence of singularities, a robot trajectory generated for a robot with 7 or more DOFs can allow its end-effector or attached tool to reach any pose (position and orientation) in its task frame by only moving 6 of its joints, thus leaving the additional robot joints stationary. Previously, several methods that used joint torque feedforward to compensate friction in robot joints were proposed, *e.g.*, using iterative learning control [Norrlöf and Gunnarsson, 2020]. Another method for friction reduction, called dithering [Ipri and Asada, 1995], consisted in using high-frequency zero-mean signal as feedforward torque, and was proposed to reduce the uncertainties caused by joint stiction for force estimation [Linderoth et al., 2013]. However, dithering may cause vibration of the robot and wear of its motors and joints if the amplitude of the dithering signal is too high. To avoid these issues, an alternative method to dithering for reducing joint friction-torque dispersion to facilitate the teaching process for redundant collaborative robots was proposed in Paper I [Salt Ducaju et al., 2021]. The proposed method exploited joint redundancy and consisted in adding joint motion in the null-space of the task frame, *i.e.*, a linear combination of joint angular velocities that causes no change in the velocity of the end-effector of the robot [Sadeghian et al., 2013], to ensure that no joint remains still during the trajectory execution, thus suppressing stiction.

Moreover, one of the main challenges that comes from the desire in the manufacturing industry to increase its flexibility is having to deal with partially unknown industrial environments, where humans and robots can effectively cooperate [Jaberzadeh Ansari and Karayiannidis, 2017], but where unexpected collisions with the environment may also occur. Then, for an effective kinesthetic teaching in such partially unknown environments, it is necessary to detect in a quick and accurate manner if a contact has occurred between the robot and its environment, and also to distinguish if this contact has been caused by voluntary human cooperation or by an accidental collision with an obstacle to identify the source of robot–environment contact. External force/torque measurements or estimates [Haddadin et al., 2017] are often used for this purpose, and there are two main sets of methods, namely analyzing the frequency response of these signals or using machine learning (ML) [Cioffi et al., 2020], where the faster detection and classification of frequency response analysis can be advantageous compared to ML-proposals. In Paper II [Salt Ducaju et al., 2022b], a method to quickly detect contacts and distinguish if they have been caused by voluntary human cooperation or by accidental collisions was proposed for an assembly task using collaborative robots. The method proposed in Paper II [Salt Ducaju et al., 2022b] included necessary modifications and extensions to previous proposals to detect and classify contacts in any direction for a

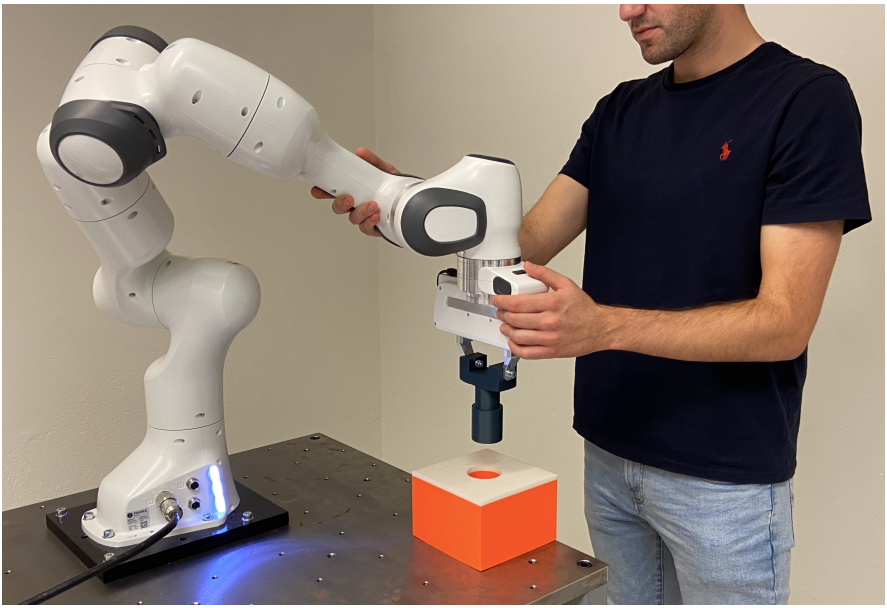


Figure 1.1 Example of the use of kinesthetic teaching for a collaborative assembly task. The image shows a human operator manually guiding a Franka Emika Panda robot [Franka Emika, 2019] mounted on a table to perform a cylinder insertion (peg-in-hole).

collaborative assembly task relying only on embedded sensors of a collaborative robot. A Cartesian impedance controller [Hogan, 1985], which establishes a mass-spring-damper relationship between the Cartesian pose variation from its reference and the Cartesian force in terms of the end-effector of the robot [Albu-Schäffer and Hirzinger, 2002], was used to achieve a compliant behavior of the robot.

Compliant controllers for robots, such as the one used in Paper II [Salt Ducaju et al., 2022b], have gained increasing popularity in parallel to the appearance and proliferation of collaborative robots, even though initially proposed some decades earlier [Hogan, 1985; Kazerooni et al., 1986]. These types of controllers improve safety in contact-rich environments, which is a key concern when operating in these partially unknown environments, and they also allow physical human cooperation with the robot. However, the workspace shared by a human operator guiding a robot during a kinesthetic teaching task may not be entirely available, *e.g.*, other robots, operators, or sensitive equipment may also be sharing this workspace. Artificial potential field (APF) methods have been used in the past for robot obstacle avoidance [Khatib, 1985], but safety control barrier functions (SCBFs) have in recent years become a popular alternative to APF-based methods, since they only modify a robot controller when needed, in a minimally-invasive manner, and provide formal guar-

antees for obstacle avoidance by enforcing the forward invariance of a safe set of states of the robot [Ames et al., 2019]. In Paper III [Salt Ducaju et al., 2022a], a method to further improve safety in kinesthetic teaching was proposed, and it consisted in varying the robot compliant behavior using SCBFs to avoid that a human operator could guide the robot to an unsafe position. To guarantee adherence to the safety constraints, the rigid-body dynamics of the robot was explicitly considered in the method to formulate the SCBF.

1.3 Publications

This thesis is based on the following publications:

Paper I

Salt Ducaju, J. M., B. Olofsson, A. Robertsson, and R. Johansson (2021). "Joint stiction avoidance with null-space motion in real-time model predictive control for redundant collaborative robots". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. Aug. 8–12. Vancouver, Canada (Virtual), pp. 307–314.

Paper II

Salt Ducaju, J. M., B. Olofsson, A. Robertsson, and R. Johansson (2022). "Fast contact detection and classification for kinesthetic teaching in robots using only embedded sensors". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. Aug. 29–Sep. 2. Naples, Italy, pp. 1138–1145.

Paper III

Salt Ducaju, J. M., B. Olofsson, A. Robertsson, and R. Johansson (2022). "Robot Cartesian compliance variation for safe kinesthetic teaching using safety control barrier functions". In: *IEEE International Conference on Automation Science and Engineering (CASE)*. Aug. 20–24. Mexico City, Mexico, pp. 2259–2266.

An overview of the notation used in each of these publications is presented in Appendix A. Moreover, the following publications, where the author has made contributions, were decided not to be part of the present thesis:

Salt Ducaju, J. M., J. J. Salt Llobregat, Á. Cuenca, and M. Tomizuka (2021). "Autonomous ground vehicle lane-keeping LPV model-based control: dual-rate state estimation and comparison of different real-time control strategies". *Sensors* **21**:4, p. 1531.

Salt Ducaju, J. M., C. Tang, M. Tomizuka, and C.-Y. Chan (2020). "Application specific system identification for model-based control in self-driving cars". In: *IEEE Intelligent Vehicles Symposium (IV)*. Oct. 19–Nov. 13, pp. 384–390.

1.4 Thesis Outline

This thesis consists of three papers. In Paper I, null-space motion is used to reduce the friction-torque dispersion of the joint of a redundant collaborative robot to aid a human collaborator in predicting the force/torque necessary to move the robot, thus facilitating kinesthetic teaching. An experimental evaluation of the proposed method for a trajectory generated online using model predictive control [Ghazaei Ardakani et al., 2019] was included in this paper.

In Paper II, fast contact detection and classification methods based on the frequency-response analysis of the estimated external force [Kouris et al., 2016; Kouris et al., 2018] were experimentally evaluated, and necessary modifications and extensions were proposed for kinesthetic teaching applications in an assembly task when only using sensors conventionally embedded in commercial collaborative robots and using robot compliant control.

In Paper III, safety control barrier functions [Ames et al., 2019] have been used to online modify the Cartesian compliant behavior of a robot to avoid that an operator could guide the end-effector of the robot to an unsafe position, in the context of safe kinesthetic teaching. A passivity-based energy-storage formulation [Ferraguti et al., 2013] has been modified to include a strict Lyapunov function and was used to ensure the stability of the proposed method.

1.5 Contributions

The main contributions of this thesis are:

- The proposal and experimental evaluation of a method to facilitate kinesthetic teaching in redundant robots by the use of null-space motion.
- The experimental evaluation of fast contact detection and classification methods for compliant robots using only data from embedded sensors and the proposal of necessary modifications and extensions to use these methods for kinesthetic teaching applications in an assembly task.
- The proposal and experimental evaluation of a method to modify the compliant behavior of a robot (while ensuring its strict stability) to avoid unsafe situations during kinesthetic teaching using safety control barrier functions.

1.6 Discussion and Future Research

Motivated by recent trends in the manufacturing industry, the relevance of human–robot collaboration (HRC) in industrial environments has increased in the last decades. This thesis focused on kinesthetic teaching, which is a topic of HRC that considers using human manual guidance of robots to exploit the intelligence and

dexterity that human operators can provide to the manufacturing industry processes. Several contributions have been made for research problems in this area regarding kinesthetic teaching: to facilitate physical interaction with robots for human operators (Paper I), to increase the application scope of kinesthetic teaching (Paper II), and to ensure safety in a shared workspace (Paper III).

Nevertheless, the involvement of human operators through kinesthetic teaching applications in industry can be further improved in several research directions utilizing both well-established and novel concepts in the robotics and automatic control fields. First, the tendency of collaborative robot manufacturers to purposefully design robots with redundant DOFs with respect to a task in Cartesian space of their end-effector to increase their dexterity can be exploited further than our proposal (Paper I) of using joint redundancy to reduce joint-torque dispersion. It could be promising to use these additional DOFs of the robot to increase the safety of human operators using SCBFs, while simultaneously facilitating human collaboration by increasing the manipulability of the robot. Additionally, well-grounded control strategies such as Model Predictive Control (MPC) [Mayne et al., 2000] that allow a longer-horizon prediction of the behavior of the robot and the human operator could be considered to improve safety in their shared workspace and to further facilitate kinesthetic teaching from the perspective of the operator.

1.7 Conclusion

The research presented in this thesis aimed to improve the involvement of human operators in industrial environments using kinesthetic teaching with robots. Different methods, focused on reducing the uncertainty in pHRI and on increasing safety during human guidance, were proposed to achieve this goal. First, the addition of null-space motion to robot trajectories showed to reduce uncertainty in the force needed for human guidance caused by joint static friction, thus facilitating kinesthetic teaching. The structural vibrations and possible wear of the robot components caused using other state-of-the-art methods, such as dithering, was avoided in the proposed method, although the application scope of the proposed method was limited to redundant manipulators. Second, necessary modifications and extensions were proposed in this thesis to state-of-the-art methods to achieve fast contact detection and classification in any contact direction for kinesthetic teaching applications. The proposed method provided an accurate distinction between voluntary human cooperation and accidental collisions with stiff and static obstacles in a collaborative assembly task with collaborative robots. Third, using safety control barrier functions to online modify the Cartesian compliant behavior of a robot provided a stable and effective method for improving safety in kinesthetic teaching applications. This proposed method was able to avoid that a human operator could guide a robotic manipulator to an undesired part of the workspace of the robot.

Bibliography

Akgun, B., M. Cakmak, J. W. Yoo, and A. L. Thomaz (2012). "Trajectories and keyframes for kinesthetic teaching: a human-robot interaction perspective". In: *ACM/IEEE International Conference on Human-Robot Interaction*. Mar. 5–8. Boston, MA, USA, pp. 391–398.

Albu-Schäffer, A. and G. Hirzinger (2002). "Cartesian impedance control techniques for torque controlled light-weight robots". In: *IEEE International Conference on Robotics and Automation (ICRA)*. Vol. 1. May 11–15. Washington DC, USA, pp. 657–663.

Ames, A. D., S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada (2019). "Control barrier functions: theory and applications". In: *European Control Conference (ECC)*. Jun. 25–28. Naples, Italy, pp. 3420–3431.

Argall, B. D., S. Chernova, M. Veloso, and B. Browning (2009). "A survey of robot learning from demonstration". *Robotics and Autonomous Systems* **57**:5, pp. 469–483.

Bagge Carlson, F., A. Robertsson, and R. Johansson (2015). "Modeling and identification of position and temperature dependent friction phenomena without temperature sensing". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Sep. 28–Oct. 2. Hamburg, Germany, pp. 3045–3051.

Bittencourt, A. C. and S. Gunnarsson (2012). "Static friction in a robot joint—Modeling and identification of load and temperature effects". *Journal of Dynamic Systems, Measurement, and Control* **134**:5, pp. 1–10.

Cencen, A., J. C. Verlinden, and J. Geraedts (2018). "Design methodology to improve human-robot coproduction in small-and-medium-sized enterprises". *IEEE/ASME Transactions on Mechatronics* **23**:3, pp. 1092–1102.

Cioffi, G., S. Klose, and A. Wahrburg (2020). "Data-efficient online classification of human-robot contact situations". In: *European Control Conference (ECC)*. May 12–15. Saint Petersburg, Russia, pp. 608–614.

Crowe, S. (2022). *Collaborative robots comparison tool*. URL: `https://www.cobottrends.com/cobot-comparison-tool/` (visited on 2022-10-25).

Ferraguti, F., C. Secchi, and C. Fantuzzi (2013). "A tank-based approach to impedance control with variable stiffness". In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 6–10. Karlsruhe, Germany, pp. 4948–4953.

Franka Emika (2019). *Franka Emika Panda – Data Sheet*. `https://www.generationrobots.com/media/panda-franka-emika-datasheet.pdf`. (Visited on 2022-09-13).

Ghazaei Ardakani, M., B. Olofsson, A. Robertsson, and R. Johansson (2019). "Model predictive control for real-time point-to-point trajectory generation". *IEEE Transactions on Automation Science and Engineering* **16**:2, pp. 972–983.

Haddadin, S., A. De Luca, and A. Albu-Schäffer (2017). "Robot collisions: a survey on detection, isolation, and identification". *IEEE Transactions on Robotics* **33**:6, pp. 1292–1312.

Haug, E. J., S. C. Wu, and S. M. Yang (1986). "Dynamics of mechanical systems with Coulomb friction, stiction, impact and constraint addition-deletion—I Theory". *Mechanism and Machine Theory* **21**:5, pp. 401–406.

Hirzinger, G. (1986). "Robot systems completely based on sensory feedback". *IEEE Transactions on Industrial Electronics* **IE-33**:2, pp. 105–109.

Hogan, N. (1985). "Impedance control: an approach to manipulation: Parts I–III". *Journal of Dynamic Systems, Measurement, and Control* **107**:1, pp. 1–24.

IFR (2018a). *Robots and the workplace of the future*. `https://ifr.org/downloads/hidden/IFR_Robots_and_the_Workplace_of_the_Future_Positioning_Paper_V01.pdf`. A positioning paper by the International Federation of Robotics (Visited on 2022-11-10).

IFR (2018b). *The impact of robots on productivity, employment and jobs*. `https://ifr.org/downloads/hidden/IFR_The_Impact_of_Robots_on_Employment_Positioning_Paper_updated_V02.pdf`. A positioning paper by the International Federation of Robotics (Visited on 2022-11-10).

IFR (2020). *Demystifying collaborative industrial robots*. `https://ifr.org/downloads/hidden/IFR_Demystifying_Collaborative_Robots_Update_v03.pdf`. A positioning paper by the International Federation of Robotics (Visited on 2022-11-10).

Ipri, S. L. and H. Asada (1995). "Tuned dither for friction suppression during force-guided robotic assembly". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*. Vol. 1. Aug. 5–9. Pittsburgh, PA, USA, pp. 310–315.

Jaberzadeh Ansari, R. and Y. Karayiannidis (2017). "Reducing the human effort for human-robot cooperative object manipulation via control design". *IFAC-PapersOnLine* **50**:1, pp. 14922–14927.

Karayiannidis, Y., C. Smith, and D. Kragic (2014). "Mapping human intentions to robot motions via physical interaction through a jointly-held object". In: *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. Aug. 25–29. Edinburgh, Scotland, UK, pp. 391–397.

Kazerooni, H., T. Sheridan, and P. Houpt (1986). "Robust compliant motion for manipulators, Part I: The fundamental concepts of compliant motion". *IEEE Journal on Robotics and Automation* **2**:2, pp. 83–92.

Khatib, O. (1985). "Real-time obstacle avoidance for manipulators and mobile robots". In: *IEEE International Conference on Robotics and Automation (ICRA)*. Vol. 2. Mar. 25–28. St. Louis, USA, pp. 500–505.

Kouris, A., F. Dimeas, and N. Aspragathos (2016). "Contact distinction in human-robot cooperation with admittance control". In: *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. Oct. 9–12. Budapest, Hungary, pp. 1951–1956.

Kouris, A., F. Dimeas, and N. Aspragathos (2018). "A frequency domain approach for contact type distinction in human–robot collaboration". *IEEE Robotics and Automation Letters* **3**:2, pp. 720–727.

Kurfess, T. R. (2005). *Robotics and Automation Handbook*. Vol. 414. Boca Raton, FL, USA.

Linderoth, M., A. Stolt, A. Robertsson, and R. Johansson (2013). "Robotic force estimation using motor torques and modeling of low velocity friction disturbances". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Nov. 3–7. Tokyo, Japan, pp. 3550–3556.

Mayne, D., J. Rawlings, C. Rao, and P. Scokaert (2000). "Constrained model predictive control: Stability and optimality". *Automatica* **36**:6, pp. 789–814.

Norrlöf, M. and S. Gunnarsson (2020). "An ILC approach to feed-forward friction compensation". *IFAC-PapersOnLine* **53**:2, pp. 1409–1414.

Ravichandar, H., A. S. Polydoros, S. Chernova, and A. Billard (2020). "Recent advances in robot learning from demonstration". *Annual Review of Control, Robotics, and Autonomous Systems* **3**, pp. 297–330.

Sadeghian, H., L. Villani, M. Keshmiri, and B. Siciliano (2013). "Task-space control of robot manipulators with null-space compliance". *IEEE Transactions on Robotics* **30**:2, pp. 493–506.

Salt Ducaju, J. M., B. Olofsson, A. Robertsson, and R. Johansson (2021). "Joint stiction avoidance with null-space motion in real-time model predictive control for redundant collaborative robots". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. Aug. 8–12. Vancouver, Canada (Virtual), pp. 307–314.

Salt Ducaju, J. M., B. Olofsson, A. Robertsson, and R. Johansson (2022a). "Fast contact detection and classification for kinesthetic teaching in robots using only embedded sensors". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. Aug. 29–Sep. 2. Naples, Italy, pp. 1138–1145.

Salt Ducaju, J. M., B. Olofsson, A. Robertsson, and R. Johansson (2022b). "Robot Cartesian compliance variation for safe kinesthetic teaching using safety control barrier functions". In: *IEEE International Conference on Automation Science and Engineering (CASE)*. Aug. 20–24. Mexico City, Mexico, pp. 2259–2266.

Schou, C., J. Damgaard, S. Bøgh, and O. Madsen (2013). "Human-robot interface for instructing industrial tasks using kinesthetic teaching". In: *44th IEEE International Symposium on Robotics (ISR)*. Oct. 19–27. Seoul, Korea, pp. 1–6.

Suomalainen, M., Y. Karayiannidis, and V. Kyrki (2022). "A survey of robot manipulation in contact". *Robotics and Autonomous Systems* **156**, p. 104224.

Wrede, S., C. Emmerich, R. Grünberg, A. Nordmann, A. Swadzba, and J. Steil (2013). "A user study on kinesthetic teaching of redundant robots in task and configuration space". *Journal of Human-Robot Interaction* **2**:1, pp. 56–81.

Paper I

Joint Stiction Avoidance with Null-Space Motion in Real-Time Model Predictive Control for Redundant Collaborative Robots

Julian M. Salt Ducaju **Björn Olofsson**
Anders Robertsson **Rolf Johansson**

Abstract

Model Predictive Control (MPC) is an efficient point-to-point trajectory-generation method for robots that can be used in situations that occur under time constraints. The motion plan can be recalculated online to increase the accuracy of the trajectory when getting close to the goal position. safety control barrier functions have implemented this strategy in a Franka Emika Panda robot, a redundant collaborative robot, by extending previous research that was performed on a 6-DOFs robot. We have also used null-space motion to ensure a continuous movement of all joints during the entire trajectory execution as an approach to avoid joint stiction and allow accurate kinesthetic teaching. As is conventional for collaborative and industrial robots, the Panda robot is equipped with an internal controller, which allows to send position and velocity references directly to the robot. Therefore, null-space motion can be added directly to the MPC-generated velocity references. The observed trajectory deviation caused by discretization approximations of the Jacobian matrix when implementing null-space motion has been corrected experimentally using sensor feedback for the real-time velocity-reference recalculation and by performing a fast sampling of the null-space vector. Null-space motion has been experimentally seen to contribute to reducing the friction torque dispersion present in static joints.

1. Introduction

Trajectory generation is a well-studied problem in the robotics field. It consists of defining the path and the course of motion as a function of time. An overview of the many ways for doing this task is provided in [Kröger, 2010]. In an industrial setting, it is common to aim for performing a task in the shortest time possible to increase productivity. To this purpose, the robot should perform the given task under time constraints, making it convenient to formulate the problem as an optimal control problem, which provides a performance metric by means of an objective function [LaValle, 2006].

Model Predictive Control (MPC) [Mayne et al., 2000; Maciejowski, 2002] is a well-grounded option for trajectory generation in robotic applications, since its formulation can include a final-state constraint to be satisfied at the end of its prediction horizon while respecting states' and inputs' limits during the motion. MPC uses a model of the robot to predict the future states and outputs based on the solution's choice of the input sequence. In the presence of an internal controller with a short time constant considering the robot dynamics, position or velocity references can be used directly making a complex dynamic model not necessary in the MPC. Therefore, a purely kinematic model can be used.

Moreover, online MPC trajectory recalculation can be performed to increase the resolution of the computed trajectory by setting a fixed final time while keeping the number of discretization points of the MPC prediction horizon constant. Then, the continuous-time prediction horizon of the MPC problem will shrink, successively causing a reduction of the sampling period every time the trajectory is recalculated online [Ghazaei Ardakani et al., 2019].

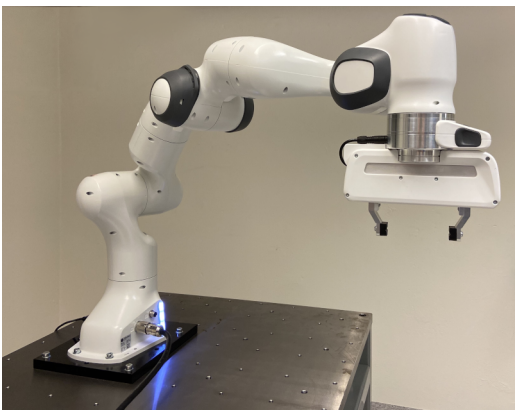


Figure 1. Franka Emika Panda robot used in the experiments.

In the context of robot trajectory reprogramming, it is convenient that a human

operator guides the robot through direct interaction [Capurso et al., 2017], which is known as kinesthetic teaching [Wrede et al., 2013] or lead-through programming (LTP), throughout the entire trajectory or parts of it. For an operator to be able to teach the robot, it is necessary to apply force on the robot end-effector or links. In this situation, the human operator should be comfortable with the physical interaction with the robot. Thus, it is important to be familiar with the force/torque required for leading the robot. Furthermore, the online MPC trajectory recalculation scheme is useful in the presence of human-robot interaction (HRI) since, after the human intervention is over, it can still be possible to reach the robot's goal pose without violating the problem's fixed-time constraint.

Therefore, the necessary force should not vary greatly between different human interventions. In order to ensure that the force that the operator needs to apply is always similar, joint stiction should be avoided. Joint static friction, or stiction, occurs when a joint has zero velocity and it becomes locked and constrained against relative motion [Haug et al., 1986]. This phenomenon is caused by the interactions between the asperities of the surface in contact in a robot joint, such as gears, bearings, and shafts [Bittencourt and Gunnarsson, 2012].

Dithering has been proven as a successful method to reduce these uncertainties [Linderoth et al., 2013]. However, it may cause vibration of the robot if the torque feedforward signal's amplitude is too high. Another option to avoid stiction, only available for robots that have more than 6 degrees of freedom (DOF), is to use null-space motion [Sadeghian et al., 2013]. Null-space motion is a linear combination of joint angular velocities in an over-actuated robot that causes no change in the end-effector's pose ($Jb = 0$ with $b \neq 0$, being J the Jacobian and b the null-space vector) [Siciliano and Khatib, 2016]. It has previously been used in kinesthetic teaching to modify the robot's configuration without altering the end-effector's pose [Wahrburg et al., 2016]. However, it can also be added to the trajectory reference to ensure that no joint remains still during the trajectory execution.

A trajectory generated for a 7 (or more) DOF robot may not necessarily involve varying the angular position of all of its joints, since it is a redundant system and it might be able to reach any end-effector's goal pose by only moving 6 of its joints. However, there could be an unexpected robot response if the operator tries to move a stationary joint since the force/torque required will be difficult to predict because of joint stiction [Haug et al., 1986]. The method that we propose to avoid joint stiction consists of adding null-space motion to an MPC-generated trajectory reference.

The purpose of this paper is to experimentally analyze the effects of adding null-space motion to an MPC-generated point-to-point trajectory reference to evaluate the possible advantages and drawbacks of this method. Moreover, HRI, facilitated by the addition of null-space motion, would allow the operator to locally modify the robot's path, which could be relevant in this context since the trajectory reference is generated considering only an initial and a final point. The implementation has been performed on the Panda robot by Franka Emika [Franka Emika, 2019], a collaborative robot [Colgate et al., 1996], which can be seen in Fig. 1.

Furthermore, in previous research [Ghazaei Ardakani et al., 2019], an open-loop strategy was implemented for the online trajectory recalculation where the initial state of the robot used to solve the MPC problem was estimated using the previous MPC solution. However, the addition of null-space motion to the MPC-generated reference may increase the error in the initial state estimation at every trajectory recalculation period, or metaperiod, and this error will accumulate at every online recalculation. For this reason, it is also a goal of this paper to evaluate the influence of joint angular position sensor feedback in the estimation of the initial state of the robot at the beginning of every metaperiod when adding null-space motion to an MPC-generated trajectory reference. The use of sensor feedback is referred to as the closed-loop strategy, as opposed to the previously used open-loop strategy [Ghazaei Ardakani et al., 2019].

This paper is outlined as follows: Sec. 2 presents the method for solving the problem that is being considered. Section 3 explains the experimental setup and the experiments performed, and presents the results obtained. Finally, conclusions are drawn in Sec. 4.

2. Methods

In this section, we introduce the MPC formulation used for trajectory generation and explain a strategy to add null-space motion to it. The goal of the trajectory-generation formulation used is that the robot reaches a final configuration under a time constraint. Additionally, we outline a hybrid dual-mode controller that would allow to switch between an MPC-based trajectory-following controller with null-space motion, and an admittance controller for human interaction.

2.1 Trajectory Generation Using MPC

The motion plan generated by MPC consists of a sequence of joint angular velocity references, since the robot's internal controller takes care of applying the necessary torques to each of the joints. Therefore, the optimization problem can be formulated in the joint space of the robot, using the robot's joint configuration $q \in \mathbb{R}^7$ since the robot has 7 joints, instead of formulating the problem in terms of the robot end-effector pose, $\xi \in SE(3)$, which is composed by the end-effector's position and orientation.

The initial and final joint configurations, q_0 and q_F, of the problem are obtained from the initial and desired end-effector poses, ξ_0 and ξ_F, respectively, by means of inverse kinematics [Corke, 2013]:

$$q = \mathscr{K}^{-1}(\xi) \tag{1}$$

Since this problem considers a 7 DOF robot, there will be an infinite number of solutions. Therefore, redundancy can be conveniently exploited to meet additional

constraints on the kinematic control problem in order to obtain greater manipulability in terms of manipulator configurations, interaction with the environment, and null-space motion.

Moreover, since the robot is equipped with an internal controller that allows a velocity-reference control mode and we assume good tracking performance without exceeding the torque limits [Bäuml et al., 2010], the MPC does not need to use a complex nonlinear robot dynamic model where the torque is the input, and a simpler kinematic linear model is considered where the motion is defined in terms of position, velocity, and other higher-order time derivatives of position [Ghazaei, 2016]. Also, the internal controller reduces the effect of dynamic coupling between joints by means of torque feedforward.

Then, as in previous research [Ghazaei Ardakani et al., 2019], the continuous-time model chosen can be constructed by multiple decoupled chains of integrators. Thus, the continuous-time state vector, $x_c \in \mathbb{R}^{21}$, is composed by the angular position, q_i, velocity, $\dot{q}_i$, and acceleration, $\ddot{q}_i$, of each of the robot joints $i = 1,\ldots,7$:

$$x_c = \begin{bmatrix} q_1 & \dot{q}_1 & \ddot{q}_1 & \cdots & q_7 & \dot{q}_7 & \ddot{q}_7 \end{bmatrix}^{\mathrm{T}} \tag{2}$$

The continuous-time linear model can thus be written as:

$$\dot{x}_c(t) = A_c x_c(t) + B_c u_c(t) \tag{3}$$
$$y_c(t) = C_c x_c(t) \tag{4}$$

with

$$A_c = \mathrm{blkdiag}([\tilde{A}_c,\ldots,\tilde{A}_c]), \quad B_c = \mathrm{blkdiag}([\tilde{B}_c,\ldots,\tilde{B}_c])$$

and $C_c = I_{21}$, where I_{21} is the identity matrix in $\mathbb{R}^{21\mathrm{x}21}$, blkdiag$(\cdot)$ forms a block diagonal matrix from the given list of matrices, $A_c \in \mathbb{R}^{21\mathrm{x}21}$, $B_c \in \mathbb{R}^{21\mathrm{x}7}$, and

$$\tilde{A}_c = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad \tilde{B}_c = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^{\mathrm{T}}$$

The continuous-time input is the angular jerk of the joints, $u_c = \dddot{q} \in \mathbb{R}^7$.

For the choice of sampling period, h, to discretize the continuous-time linear system, a sampling period different from the one of the controlled system was chosen for the discretization of the kinematics in the optimization. Then, a linear interpolation of the calculated input sequence is used to provide references at the sampling rate of the robot [Ghazaei Ardakani et al., 2019]. This justifies the use of a predictive first-order-hold (FOH) sampling method [Åström and Wittenmark, 2013]:

$$x_{k+1} = \Phi x_k + \frac{1}{h}\Gamma_1 u_{k+1} + \left(\Gamma - \frac{1}{h}\Gamma_1\right) u_k \tag{5}$$
$$y_k = C x_k \tag{6}$$

with

$$\begin{aligned}\Phi &= \text{blkdiag}([\tilde{\Phi},\ldots,\tilde{\Phi}]), \quad \Gamma_1 = \text{blkdiag}([\tilde{\Gamma}_1,\ldots,\tilde{\Gamma}_1]),\\ \Gamma &= \text{blkdiag}([\tilde{\Gamma},\ldots,\tilde{\Gamma}])\end{aligned}$$

where $\Phi \in \mathbb{R}^{21\text{x}21}$, $\Gamma_1, \Gamma \in \mathbb{R}^{21\text{x}7}$, and:

$$\begin{aligned}\tilde{\Phi} &= \begin{bmatrix}1 & h & h^2/2\\ 0 & 1 & h\\ 0 & 0 & 1\end{bmatrix}\\ \tilde{\Gamma} &= \begin{bmatrix}h^3/6 & h^2/2 & h\end{bmatrix}^{\text{T}}\\ \tilde{\Gamma}_1 &= \begin{bmatrix}h^4/24 & h^3/6 & h^2/2\end{bmatrix}^{\text{T}}\\ C &= C_c\end{aligned}$$

As developed in previous research [Ghazaei Ardakani et al., 2019], the discrete-time model obtained from the FOH sampling method (5), (6) can be rewritten in the standard form by using a new discrete state variable, $\zeta \in \mathbb{R}^{21}$:

$$\zeta_{k+1} = A\zeta_k + Bu_k \tag{7}$$
$$y_k = C\zeta_k + Du_k \tag{8}$$

where

$$A = \Phi, \quad B = \Gamma + \frac{1}{h}(\Phi - I_{21})\Gamma_1, \quad D = \frac{\Gamma_1}{h}$$

Since $y_k = x_k$ because of (6) and $C = I_{21}$, we can from (8) obtain the relation:

$$x_k = C\zeta_k + Du_k \tag{9}$$

It should be mentioned that the input u is the discretized counterpart of u_c, and the discrete controlled variable x is the discretized counterpart of x_c. On the contrary, the discrete-time state ζ is not a discretized version of any variable found in the continuous-time state-space system formulation (3), (4).

Moreover, the quadratic cost function chosen for solving this problem at time step k is:

$$V_k(U_k) = \sum_{j=k+1}^{k+H} x_j^{\text{T}} Q x_j + \sum_{j=k}^{k+H-1} u_j^{\text{T}} R u_j \tag{10}$$

where $U_k = [u_k,\ldots,u_{k+H-1}] \in \mathbb{R}^{7\text{x}H}$ is the input signal sequence over the control horizon of H steps that minimizes the cost function over the MPC prediction horizon of H steps at every metaperiod, and $Q \in \mathbb{R}^{21\text{x}21}$ and $R \in \mathbb{R}^{7\text{x}7}$ are positive semi-definite weight matrices that penalize the controlled variables and inputs, respectively.

This optimization problem is subject to the discrete-time model of the system (7), (9). Additionally, a hard constraint on the value of the discrete-time final controlled variables is used to ensure that the robot reaches the desired configuration at the end of the trajectory:

$$x_{k+H} = x_{\text{goal}} \tag{11}$$

In addition, a set of linear constraints must be included to bound the admissible range of the inputs and controlled variables:

$$F[u_k^{\mathrm{T}}, \ldots, u_{k+H-1}^{\mathrm{T}}]^{\mathrm{T}} \leq f \tag{12}$$

$$G[x_{k+1}^{\mathrm{T}}, \ldots, x_{k+H}^{\mathrm{T}}]^{\mathrm{T}} \leq g \tag{13}$$

The choice of the cost function as convex, as well as a linear model and convex constraint sets, makes the whole problem convex, which is beneficial for the computation of the problem since if a solution exists, it is the globally optimal [Boyd et al., 2004].

Finally, this convex problem is solved at every trajectory recalculation period, or metaperiod. The sampling period, h, used in the discretization is equal to:

$$h = \frac{T_F - t_k}{H} \tag{14}$$

where H is the number of discrete steps in the prediction horizon, T_F is the final time where the goal state must be reached, and t_k is the time when the robot starts using the newly recalculated trajectory reference. As mentioned earlier, the continuous-time prediction horizon of the problem will shrink since, as time goes by, t_k will increase while the final time T_F and H are constant, thus increasing the resolution of the computed trajectory as the goal state, x_{goal}, is approached.

2.2 Null-Space Motion Addition to the Reference Trajectory

The manipulator's Jacobian matrix, $J(q) \in \mathbb{R}^{6\text{x}7}$, maps the joint angular velocities, $\dot{q}$, to the end-effector's twist, $\gamma = [\omega^{\mathrm{T}}, v^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^6$, with v and ω denoting the linear and angular velocity of the end-effector, respectively:

$$\gamma = J(q)\dot{q} \tag{15}$$

Therefore, null-space motion is constructed by using the null-space vector of this Jacobian matrix:

$$\dot{q} = J^{\dagger}(q)\gamma + N(q)\dot{q}_{\text{a}} \tag{16}$$

where the matrix $N(q) = I_7 - J^{\dagger}(q)J(q) \in \mathbb{R}^{7\text{x}7}$ projects the additional arbitrary joint angular velocity, $\dot{q}_{\text{a}}$, into the null space so that it is independent of the end-effector Cartesian motion [Corke, 2013].

The first term of (16) is the relationship between the joint velocity $\dot{q}$ and the end-effector's twist γ by means of the manipulator Jacobian (15), and superscript †

denotes the Moore-Penrose pseudoinverse matrix given by $J^{\dagger} = (J^{\mathrm{T}}J)^{-1}J^{\mathrm{T}}$ [Ben-Israel and Greville, 2003]. This term is shared for both 6 and 7 DOF robots, although in the case of 6 DOFs, the Jacobian is a square matrix. However, the second term of (16) is the null-space motion, which only appears in redundant manipulators. The null-space motion unitary vector is calculated as:

$$\dot{q}_{\mathrm{nsu}} = \frac{N(q)\dot{q}_{\mathrm{a}}}{\|N(q)\dot{q}_{\mathrm{a}}\|} \tag{17}$$

Also, since the Jacobian matrix is particular for each robot configuration, this vector should be sampled in real-time.

The null-space unitary vector given in (17) has to be scaled before being included with the MPC-generated angular velocity references. A sinusoidal signal has been chosen to smoothly transition between positive and negative scaling values to avoid reaching any joint limit. Its frequency depends on the length of the trajectory execution, to make sure that the first and last velocity references sent are equal to 0. Additionally, $\alpha \in \mathbb{R}$ is a constant used to scale its amplitude:

$$\dot{q}_{\mathrm{NS}} = \dot{q}_{\mathrm{nsu}}\alpha \sin\left(\frac{2\pi t}{T_F}\right) \tag{18}$$

Then, null-space motion is calculated at each robot sampling instant and added to the velocity references calculated by the optimization to avoid joint stiction:

$$\dot{q}_{\mathrm{ref}} = \dot{q}_{\mathrm{MPC}} + \dot{q}_{\mathrm{NS}} \tag{19}$$

where $\dot{q}_{\mathrm{ref}}$ is the velocity references sent to the robot, $\dot{q}_{\mathrm{MPC}}$ is the linearly interpolated velocity reference sequence calculated by the MPC, and $\dot{q}_{\mathrm{NS}}$ is the null-space motion component obtained from (18).

Moreover, the controlled-variable constraint (13) should consider the superposition of the null-space motion on the MPC solution to avoid any possible constraint violation. Therefore, when solving the MPC optimization the joint-velocity range should be reduced for every joint in a proportional way to the maximum possible joint-velocity component corresponding to the added null-space motion. With this approach, it is guaranteed that the joint-velocity limits are fulfilled. Also, the joint-acceleration range should be conservative to never exceed the joints' torque limits [Bäuml et al., 2010].

Finally, joint angular position sensor feedback can be used to reduce the mismatch between the estimation of the initial state used for the online optimization-problem calculation at every metaperiod and its real value caused by the addition of null-space motion to the MPC-generated trajectory. Therefore, a closed-loop form of the problem is proposed to obtain a more accurate estimation of the initial state to be used in the MPC. However, data samples from the robot's sensors cannot be directly used as the initial state, since there is a planned computational delay that

accounts for the time required to solve the optimization problem in the MPC. Therefore, in order to provide a precise initial state estimation it is necessary to use the system's model described by (7) and (9) to estimate the state evolution between the sampling time of the sensor feedback and the time where the new trajectory velocity references are deployed to the system.

2.3 Human-Robot Interaction (HRI)

Even though the main focus of this research is to analyze the effects of a method that facilitates HRI by reducing joint stiction, we also provide an illustrative example of one possible way that a human operator can interact with the robot. Then, we outline a hybrid dual-mode controller where the robot receives commands from the MPC-generated trajectory that includes null-space motion (19), or from human-robot interaction, but never from both sources simultaneously, as summarized in Algorithm 1.

Since human input is, in this scenario, a path correction to the previously generated MPC trajectory reference, admittance control [Wahrburg et al., 2016] is a suitable strategy for the human-interaction control mode. Another common human-robot interaction control strategy such as compliance control [Hogan, 1985] is less appropriate for this application since its virtual spring component would try to bring the robot closer to the MPC reference rather than allowing the human to freely operate the robot.

If a joint-torque interface is available, a simple way to implement admittance control is to supply the robot with joint torque commands. For this, the rigid-body dynamics of the robot is used [Siciliano and Khatib, 2016]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) + \tau_{\text{fric}} = \tau_{\text{mot}} \tag{20}$$

where $M(q) \in \mathbb{R}^{7\text{x}7}$ is the generalized inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{7\text{x}7}$ describes the Coriolis and centripetal forces effects, $g(q) \in \mathbb{R}^{7\text{x}1}$ captures the gravity-induced torques, and $\tau_{\text{fric}} \in \mathbb{R}^{7\text{x}1}$ and $\tau_{\text{mot}} \in \mathbb{R}^{7\text{x}1}$ represent the friction and motor torques, respectively.

Then, if the admittance controller is active (Algorithm 1, Line 2), it will send commanding torques to joint motors that are equal to the sensed external joint torques, $\tau_{\text{ext}} \in \mathbb{R}^{7\text{x}1}$:

$$\tau_{\text{mot}} = \tau_{\text{ext}} \tag{21}$$

where part of the commanded joint torque is used for robot motion, but a fraction of the commanded joint torque is used to overcome joint friction, as seen in Eq. (20).

Friction is present in any element that involves relative motion in robot mechanisms. All friction models have in common a significant change of friction magnitude in the zero-velocity vicinity, as shown in Fig. 2, which is the major concern of friction compensation [Cai and Song, 1993; Karnopp, 1985; Freidovich et al., 2009; Shiriaev et al., 2003]. For this reason, avoiding joint stiction is helpful for

the human operator that interacts with the robot to predict beforehand the necessary force that he/she should apply to the robot to achieve the desired displacement.

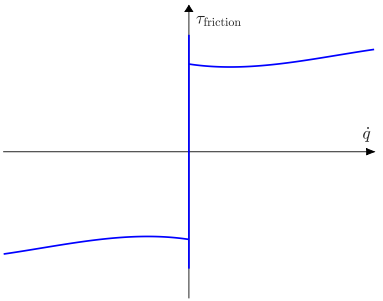


Figure 2. Joint friction as a function of joint angular velocity.

Finally, a switching mechanism between both control modes (Algorithm 1, Line 1), trajectory following with null-space motion addition and admittance control, based on external torque sensor feedback, can be either automatic, following a collision detection and classification method (a summary of different strategies can be found in [Cioffi et al., 2020]), or manually determined by the human operator.

Algorithm 1 Hybrid Dual-Mode Controller

1: **if** human is interacting with robot **then**
2: *HRI mode*: Send τ_{ext} (21) as command to the robot's torque-reference interface.
3: **else**
4: *Trajectory-following mode*: Send $\dot{q}_{\text{ref}}$ (19) as command to the robot's velocity-reference interface.
5: **end if**

3. Experiments and Results

The experiments presented in this section evaluated the performance of the addition of null-space motion onto an MPC-generated trajectory.

3.1 Implementation and Experimental Setup

The robot used in the experiments is the Franka Emika Panda [Franka Emika, 2019], a 7-DOF robotic arm. In addition to the joint velocity interface, the robot's internal controller also allows the operator to send joint position and torque commands. The

Panda robot has a sampling rate of 1 kHz, and therefore, references should be sent to it every 1 ms. A photo of the robot used is shown in Fig. 1.

This collaborative robot, or *cobot* [Colgate et al., 1996], is designed to share its workspace with humans in a safe manner, and it allows the human operator to set different maximum external-torque thresholds for each of the control modes so that if an accidental collision between the robot and the operator happened, the robot would perform a security shutdown.

As for the design choices for trajectory generation, the MPC prediction horizon was chosen to be equal to $H = 25$ as a trade-off between trajectory resolution and real-time computational cost, and the recalculation metaperiod was equal to 0.1 s. Also, the weighting matrix Q penalized the joint velocity and acceleration, but not the joint angular position, since there was no specific desired position between the initial and the final states [Ghazaei Ardakani et al., 2019], and since a hard constraint (11) was imposed on the final joint position, $Q = \text{blkdiag}([\tilde{Q}, \ldots, \tilde{Q}])$ where $\tilde{Q} = \text{diag}(\begin{bmatrix}0 & 1 & 1\end{bmatrix})$. Additionally, the input was less penalized than the states, $R = 0.001\, I_7$.

The first experiment presented analyzed the detrimental effects of slowing the sampling rate of the null-space vector of the Jacobian matrix. Then, the second experiment showed the suitability of using sensor feedback when adding null-space motion to MPC-generated trajectory references. Moreover, the third experiment focused on the results obtained for a closed-loop, fast null-space sampling approach where one of the joints would have remained static if null-space motion had not been included. Finally, the last experiment evaluated the dispersion of friction torque as a function of the joint angular velocity.

3.2 Experiment 1: Analysis of the effects of the null-space sampling rate on the trajectory accuracy

This experiment studied the effects of null-space discretization by performing the same trajectory in different runs, the only difference being that each run was performed at a different sampling rate of the null-space vector of the Jacobian matrix (1, 2, 5, and 10 ms). Since the null-space vector depends on the robot's configuration, a slower null-space sampling increases the difference between the null-space vector that is used for the velocity reference and the actual null-space vector.

The robot's initial configuration, randomly chosen, was, in radians:

$$q_0 = \begin{bmatrix}0 & -0.79 & 0.0 & -2.36 & 0.0 & 1.57 & 0.0\end{bmatrix} \tag{22}$$

Moreover, the trajectory lasted 10 s, enough time to clearly see the detrimental effects of a slower sampling rate of the null-space vector. Also, the trajectory consisted only of null-space motion, and the unitary null-space vector was scaled by a sinusoidal wave of frequency equal to 1 Hz and an amplitude constant, α, equal to 3 in (18). Therefore, at the end of the trajectory, the end-effector should ideally have the same pose as the initial one. Furthermore, an open-loop strategy was used for

this experiment in an attempt to isolate the effects of having an insufficient sampling frequency of the null-space vector.

The temporal evolution of the robot's pose has been analyzed using Figs. 3 and 4. Figure 3 shows the temporal evolution of the end-effector's Cartesian coordinates and Fig. 4 the temporal evolution of the end-effector's orientation, by means of the Euler rotation angles (ZYX) from the robot base coordinate reference system to the end-effector's coordinate reference system. It can be observed how slowing the sampling rate caused the robot to drift from the desired constant end-effector's Cartesian pose.

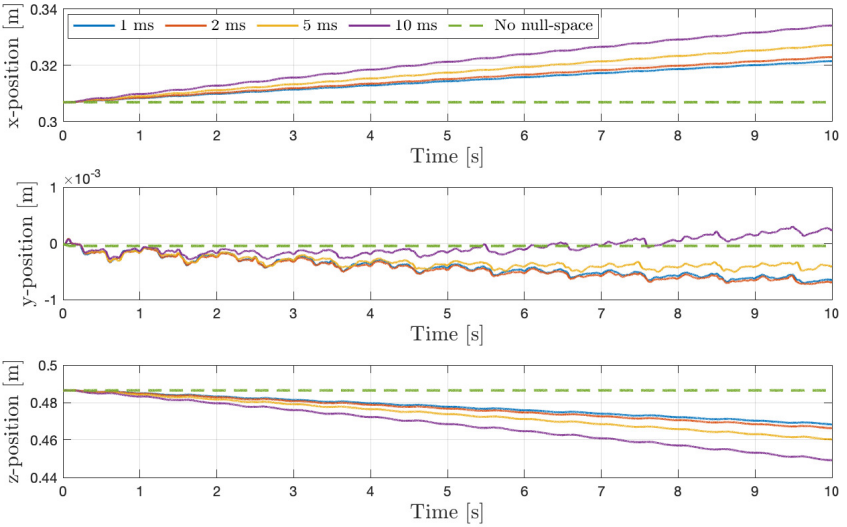


Figure 3. Experiment 1 — End-effector's position with respect to base frame.

Consequently, the null-space vector should be updated at the fastest update rate available, which in this case was the robot's sampling frequency (1 kHz). However, varying the null-space velocity references in intervals of 1 ms still introduced a slight deviation from the planned trajectory, as seen in Figs. 3 and 4. For this reason, the next experiment considered a closed-loop approach to compensate the disturbances introduced by the approximate null-space motion.

3.3 Experiment 2: Comparison of the open-loop and the closed-loop strategies

Sensor feedback from joint position sensors can be used when updating the initial state estimation for online optimization in the MPC to account for the degrading effects of low-rate null-space sampling observed in the previous experiment. To

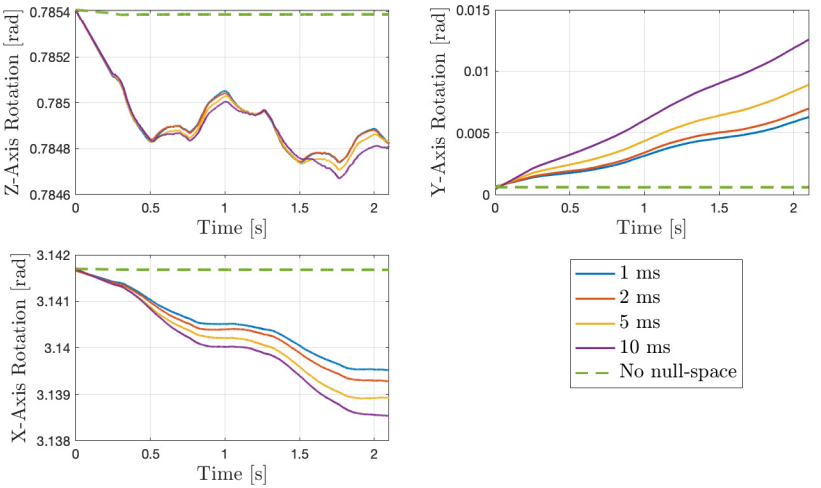


Figure 4. Experiment 1 — End-effector's orientation with respect to base frame.

show the benefits of using sensor feedback, it was necessary to compare the results of the implementation of null-space motion in the closed-loop MPC strategy versus the open-loop MPC strategy.

Several reference trajectories with different initial and goal robot configurations were used for this experiment. Additionally, each of them was executed five times. These trajectories combined null-space motion and MPC-generated trajectory references. Also, the null-space vector was sampled every ms and the sinusoidal scaling function's period was equal to the length of the trajectory.

The results of Experiment 2 are presented in Table 1, which shows the mean and standard deviation of the end-effector's Cartesian position error at the end of the trajectory. The following expression was used for calculating this error:

$$e = \sqrt{(x_G - x_F)^2 + (y_G - y_F)^2 + (z_G - z_F)^2} \tag{23}$$

where the subindex G refers to the goal position and the subindex F refers to the final position end-effector coordinate of the corresponding trial.

Table 1. End-effector's Final Cartesian Position Errors [mm]

	CL - NS	*CL*	*OL - NS*	*OL*
Mean	1.43	0.48	5.97	1.18
Std. Dev.	0.78	0.25	0.39	0.11

In Table 1, *CL* and *OL* refer to the closed-loop and the open-loop implementations, respectively, and *NS* to the runs that included null-space motion. Several assertions can be made based on Table 1. First, when no null-space motion was added, the closed-loop strategy provided a more precise final Cartesian position, since there was a better initial state estimation at each MPC trajectory recalculation. Also, in both open-loop and closed-loop scenarios, including null-space motion was detrimental to the final state precision of the trajectory. Finally, using an open-loop strategy caused a greater total final Cartesian position error and therefore, if possible, a closed-loop scheme should be used to implement null-space motion.

3.4 Experiment 3: Null-space motion integration with closed-loop MPC in a trajectory that would have left one joint static

Once the two previous sets of experiments had shown the suitability of sampling the null-space vector as fast as possible and using a closed-loop control strategy to compensate for the degrading effects of adding null-space motion to an MPC-generated trajectory reference, the results for the closed-loop controller in one of the trajectories of Experiment 2 were analyzed.

Figure 5 shows how the addition of null-space motion modified the total velocity references (19). It can be seen that Joint 3 was not used in the MPC-generated trajectory, but it was desired to have it continuously moving to avoid its stiction, thus justifying the addition of null-space motion to the trajectory.

Figures 6 and 7 show the temporal evolution of the end-effector's position in Cartesian coordinates and the temporal evolution of the end-effector's orientation parameterized in the Euler rotation angles (ZYX) between the robot's base frame and the end-effector's frame, respectively. Even though the velocity references were different, null-space motion was properly implemented in the MPC trajectory generation since the temporal evolution of the end-effector pose was very similar in both trials, and it only showed slight deviations in the y-position in Fig. 6 and in the x-axis rotation in Fig. 7, which were compensated before the motion was finished. Therefore, joint stiction in Joint 3 was addressed by adding null-space motion, while still being able to perform an accurate trajectory under the task time constraints.

3.5 Experiment 4: Friction torque dispersion

The final experiment evaluated the dispersion of the friction torque in a joint as a function of its angular velocity. For this purpose, the torque-based admittance controller in Sec. 2.3 was implemented, so that the commanded torque to each of the joints was equal to their sensed external torque signals. Also, the friction torque for all joints was estimated by rewriting Eq. (20) as:

$$\hat{\tau}_{\text{fric}} = \hat{\tau}_{\text{ext}} - (M(\hat{q})\hat{\ddot{q}} + C(\hat{q},\hat{\dot{q}})\hat{\dot{q}} + g(\hat{q})) \tag{24}$$

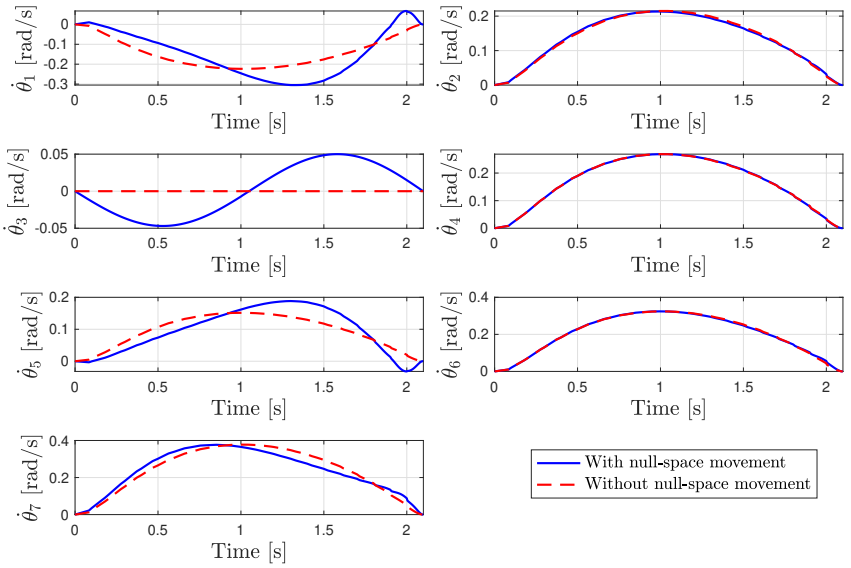


Figure 5. Experiment 3 — Joint angular commanded velocities' evolution.

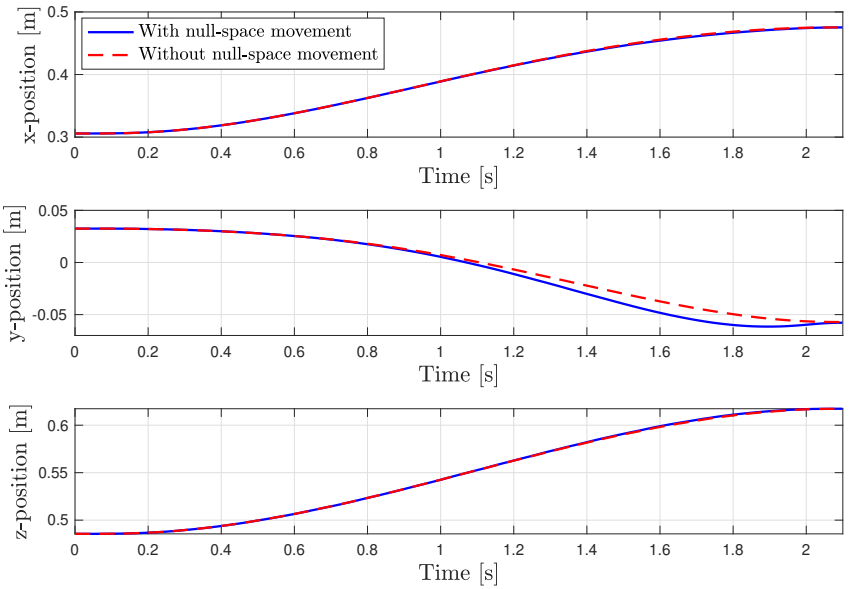


Figure 6. Experiment 3 — End-effector's position with respect to base frame.

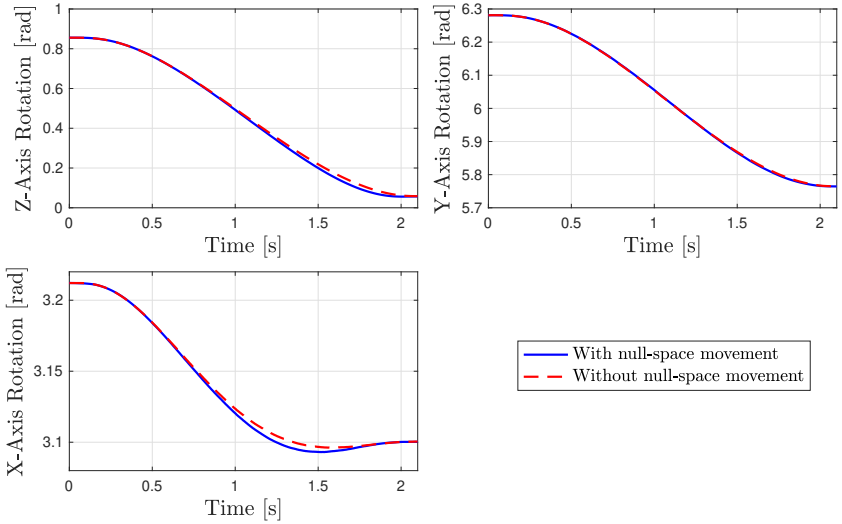


Figure 7. Experiment 3 — End-effector's orientation with respect to base frame.

where the superscript $\hat{}$ denotes a variable that has been estimated using joint position or torque sensor data.

Then, the experiment consisted of a human operator leading-through the robot for 15 s using this torque-based admittance controller. Figure 8 shows the results in terms of the standard deviation, σ, of the estimated friction torque of a joint in Eq. (24) with respect to its angular velocity.

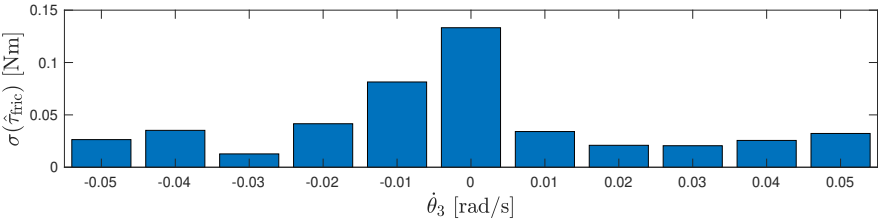


Figure 8. Experiment 4 — Standard deviation of the friction torque as a function of angular velocity of Joint 3.

The choice of joint and its angular-velocity range, shown in Fig. 8, was related to the experiment presented in Sec. 3.4, since the motion of Joint 3 was generated only by null-space motion and had the same angular velocity range, as seen in Fig. 5.

It can be seen that the standard deviation of the friction torque was greater when the angular velocity of the joint was close to zero: $\sigma(\hat{\tau}_{\text{fric}}) = 0.13$ Nm in the vicinity

of zero velocity compared to an average value of $\sigma(\hat{\tau}_{\text{fric}}) = 0.03$ Nm in the rest of the angular velocity range analyzed. Therefore, adding null-space motion to a static joint can contribute to reducing the friction torque dispersion.

4. Conclusion

We have proposed the addition of null-space motion to an MPC fixed-time point-to-point online trajectory generation method in order to facilitate kinesthetic teaching in a redundant robot. This approach allows a continuous motion of all joints throughout the trajectory execution, even if the MPC-generated trajectory does not include them in its planning, so that joint stiction is suppressed and a human operator can predict the force/torque necessary to move the robot. A reduction of the friction-torque dispersion has been experimentally observed as a consequence of adding null-space motion in a static joint.

The discrete-time control of null-space motion has been observed to be sensitive to discretization approximations of the Jacobian matrix. The experiments performed have justified the extension of a previously studied open-loop scheme [Ghazaei Ardakani et al., 2019] to a closed-loop scheme and a fast Jacobian matrix sampling to correct these slight degrading effects on the trajectory execution performance, thus allowing the addition of null-space motion to the trajectory without causing a significant loss of final-state accuracy.

An additional benefit of the presented closed-loop strategy is that, if human intervention takes place during the trajectory execution, the trajectory can be recalculated online once human intervention is concluded using an accurate estimation of the initial state in the MPC problem.

References

Åström, K. J. and B. Wittenmark (2013). *Computer-Controlled Systems: Theory and Design*. Courier Corporation, Massachusetts, USA.

Bäuml, B., T. Wimböck, and G. Hirzinger (2010). "Kinematically optimal catching a flying ball with a hand-arm-system". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Oct. 18–22. Taipei, Taiwan, pp. 2592–2599.

Ben-Israel, A. and T. N. Greville (2003). *Generalized Inverses: Theory and Applications*. Springer-Verlag, New York, USA.

Bittencourt, A. C. and S. Gunnarsson (2012). "Static friction in a robot joint—Modeling and identification of load and temperature effects". *Journal of Dynamic Systems, Measurement, and Control* **134**:5, pp. 1–10.

Boyd, S., S. P. Boyd, and L. Vandenberghe (2004). *Convex Optimization*. Cambridge Univ. Press, Cambridge, UK.

Cai, L. and G. Song (1993). "A smooth robust nonlinear controller for robot manipulators with joint stick-slip friction". In: *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 449–454.

Capurso, M., M. M. G. Ardakani, R. Johansson, A. Robertsson, and P. Rocco (2017). "Sensorless kinesthetic teaching of robotic manipulators assisted by observer-based force control". In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 29–Jun. 2. Singapore, pp. 945–950.

Cioffi, G., S. Klose, and A. Wahrburg (2020). "Data-efficient online classification of human-robot contact situations". In: *European Control Conference (ECC)*. May 12–15. Saint Petersburg, Russia, pp. 608–614.

Colgate, J. E., J. Edward, M. A. Peshkin, and W. Wannasuphoprasit (1996). "Cobots: robots for collaboration with human operators". In: *Proceedings of the ASME Dynamic Systems and Control Division: ASME International Mechanical Engineering Congress and Exposition*. Nov. 17–22. Atlanta, GA, USA.

Corke, P. (2013). *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer, Berlin, Germany.

Franka Emika (2019). *Franka Emika Panda – Data Sheet*. `https://www.generationrobots.com/media/panda-franka-emika-datasheet.pdf`. (Visited on 2022-09-13).

Freidovich, L., A. Robertsson, A. Shiriaev, and R. Johansson (2009). "LuGre-model-based friction compensation". *IEEE Transactions on Control Systems Technology* **18**:1, pp. 194–200.

Ghazaei, M. (2016). *On Trajectory Generation for Robots*. English. TFRT-1116. PhD thesis. Dept. Automat. Control, Lund Univ., Lund, Sweden. ISBN: 978-91-7753-048-0.

Ghazaei Ardakani, M., B. Olofsson, A. Robertsson, and R. Johansson (2019). "Model predictive control for real-time point-to-point trajectory generation". *IEEE Transactions on Automation Science and Engineering* **16**:2, pp. 972–983.

Haug, E. J., S. C. Wu, and S. M. Yang (1986). "Dynamics of mechanical systems with Coulomb friction, stiction, impact and constraint addition-deletion—I Theory". *Mechanism and Machine Theory* **21**:5, pp. 401–406.

Hogan, N. (1985). "Impedance control: an approach to manipulation: part I". *Journal of Dynamic Systems, Measurement, and Control* **107**:1, pp. 1–7.

Karnopp, D. (1985). "Computer simulation of stick-slip friction in mechanical dynamic systems". *Journal of Dynamic Systems, Measurement, and Control* **107**:1, pp. 100–103.

Kröger, T. (2010). *On-Line Trajectory Generation in Robotic Systems: Basic Concepts for Instantaneous Reactions to Unforeseen (Sensor) Events*. Vol. 58. Springer, Berlin, Germany.

LaValle, S. M. (2006). *Planning Algorithms*. Cambridge Univ. Press, Cambridge, UK.

Linderoth, M., A. Stolt, A. Robertsson, and R. Johansson (2013). "Robotic force estimation using motor torques and modeling of low velocity friction disturbances". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Nov. 3–7. Tokyo, Japan, pp. 3550–3556.

Maciejowski, J. M. (2002). *Predictive Control: with Constraints*. Prentice Hall, Harlow, UK.

Mayne, D., J. Rawlings, C. Rao, and P. Scokaert (2000). "Constrained model predictive control: Stability and optimality". *Automatica* **36**:6, pp. 789–814.

Sadeghian, H., L. Villani, M. Keshmiri, and B. Siciliano (2013). "Task-space control of robot manipulators with null-space compliance". *IEEE Transactions on Robotics* **30**:2, pp. 493–506.

Shiriaev, A., A. Robertsson, and R. Johansson (2003). "Friction compensation for passive systems based on the LuGre model". In: *2nd IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*. Apr. 3–5. Seville, Spain, pp. 183–188.

Siciliano, B. and O. Khatib (2016). *Springer Handbook of Robotics*. Springer, Berlin, Germany.

Wahrburg, A., J. Boes, B. Matthias, F. Dai, and H. Ding (2016). "Sensorless null-space admittance control for redundant manipulators". In: *Proceedings of ISR 2016: 47st International Symposium on Robotics*. Jun. 21–22. VDE. Munich, Germany, pp. 1–7.

Wrede, S., C. Emmerich, R. Grünberg, A. Nordmann, A. Swadzba, and J. Steil (2013). "A user study on kinesthetic teaching of redundant robots in task and configuration space". *Journal of Human-Robot Interaction* **2**:1, pp. 56–81.

Paper II

Fast Contact Detection and Classification for Kinesthetic Teaching in Robots using only Embedded Sensors

Julian M. Salt Ducaju **Björn Olofsson**
Anders Robertsson **Rolf Johansson**

Abstract

Collaborative robots have been designed to perform tasks where human cooperation may occur. Additionally, undesired collisions can happen in the robot's environment. A contact classifier may be needed if robot trajectory recalculation is to be activated depending on the source of robot–environment contact. For this reason, we have evaluated a fast contact detection and classification method and we propose necessary modifications and extensions so that it is able to detect a contact in any direction and distinguish if it has been caused by voluntary human cooperation or by accidental collision with a static obstacle for kinesthetic teaching applications. Robot compliance control is used for trajectory following as an active strategy to ensure safety of the robot and its environment. Only sensor data that are conventionally available in commercial collaborative robots, such as joint-torque sensors and joint-position encoders/resolvers, are used in our method. Moreover, fast contact detection is ensured by using the frequency content of the estimated external forces, whereas external force direction and sense relative to the robot's motion is used to classify its source. Our method has been experimentally proven to be successful in a collaborative assembly task for a number of different experimentally recorded trajectories and with the intervention of different operators.

1. Introduction

Physical Human–Robot Interaction (pHRI) has become a research topic of major interest during the later years in the robotics community [Villani et al., 2018]. The reason behind this is allowing robots to safely work in partially unknown environments where humans and robots can cooperate. One way that human operators can cooperate with the robot is through direct interaction, known as kinesthetic teaching [Wrede et al., 2013], which is useful for robot trajectory reprogramming [Karlsson et al., 2017]. Consequently, collaborative robots have increased in popularity since their lightweight, compliant design is especially convenient when robots share their workspace with humans.

As part of the desire of increasing the flexibility and versatility of robots, it is common to find applications (*e.g.*, collaborative assembly [Sadrfaridpour and Wang, 2018]) where human cooperation is not the only contact that the robots may experience with their environment, and where unexpected collisions with obstacles may also occur. For this reason, it is essential that robots are capable of quickly distinguishing if a contact has occurred, and if so, whether it has been caused by human cooperation (defined as intentional) or by an obstacle collision (defined as accidental). Therefore, contact detection and classification, while the robot behaves in a compliant way with respect to its environment, is a key concern in these applications.

1.1 Previous Research

As summarized in [Cioffi et al., 2020], there are two main sets of methods, which are primarily based on external force/torque estimation, being used to detect and classify contacts: using machine-learning approaches [Golz et al., 2015; Popov et al., 2017; Briquet-Kerestedjian et al., 2019; Cioffi et al., 2020], or analyzing their frequency content [Geravand et al., 2013; Kouris et al., 2016; Kouris et al., 2018]. In such scenarios, a fast detection and classification is essential since a successful robot trajectory reprogramming should depend on it [Karlsson et al., 2017; Wrede et al., 2013; Haddadin et al., 2008].

Machine-learning approaches have shown to provide promising results for contact detection and classification, but their fast execution may be challenging. In [Golz et al., 2015], the authors used the entire contact event to extract features that allow to discriminate between intended and unintended contacts. An extensive classification approach was presented in [Popov et al., 2017], but it cannot run in real-time. In [Briquet-Kerestedjian et al., 2019], the authors were able to classify a detected contact in a minimum of 160 ms. Finally, an online classification method using machine learning was proposed in [Cioffi et al., 2020], but it is operator dependent and needs the joint load-torque signals of a previous, uncollided, execution of the trajectory.

In contrast, frequency-response analysis methods can achieve a faster detection and classification: in [Geravand et al., 2013], the authors detected contacts in less

than 50 ms, and the authors of [Kouris et al., 2016; Kouris et al., 2018] detected and classified contacts in a single force direction in less than 10 ms. However, frequency-based methods come with their own challenges, one of the more significant being the difficulty of tuning their thresholds and cut-off frequencies. In [Geravand et al., 2013], six different thresholding parameters per joint were needed to classify the contact situation based on filtered motor-current signals, which, unfortunately, are not available in some robot controller interfaces.

Moreover, these frequency-based contact-detection and classification methods are based on the premise that human voluntary cooperation with the robot presents forces with a lower rate of change than accidental collisions, and therefore, their frequency characteristics can be differentiated: cooperation will present lower frequency components than the accidental collisions. To sustain this assumption, the authors in [Kouris et al., 2016] and [Kouris et al., 2018] presented experimental data for one force direction recorded from an external force sensor mounted between the robot's flange and a handle.

1.2 Problem Formulation

In this paper, we address the problem of fast contact-detection and classification for kinesthetic teaching applications in collaborative robots relying only on available information provided by its embedded sensors, which in most cases are the joint motor encoders/resolvers that are able to provide joint angular positions (and joint angular velocity and possibly acceleration by differentiation), and the joint-torque sensors that are used to measure the joint applied torques. These variables are then used to estimate the external forces/torques applied to the robot. We refer to [Haddadin et al., 2017] for a summary of different methods to obtain these external forces/torques, and especially for the justification of the generalized momentum observer that was used in our experiments. Moreover, robot compliant control is used to ensure safety in a contact-rich environment and to allow human cooperation.

To solve the problem addressed in this paper, while ensuring fast contact detection, we evaluated the use of frequency-response analysis of the estimated external force and the benefits of comparing the robot Cartesian motion and its sensed external force. The method should allow a fast detection and to distinguish between human cooperation and accidental collisions in any contact direction for a collaborative assembly task using data only from robot embedded sensors. To evaluate this method, several experiments were performed, using the Panda robot by Franka Emika [Franka Emika, 2019] with a peg-in-hole setup as seen in Fig. 2.

1.3 Outline

The paper is organized as follows: Sec. 2 presents the method for solving the problem described in Sec. 1. Section 3 explains the experiments performed. Then, Sec. 4 presents the results obtained. Finally, a discussion is included in Sec. 5 and conclusions are drawn in Sec. 6.

2. Method

First, we introduce the robot compliance controller used. Then, we evaluate the use of frequency-based contact detection and classification for our problem. Finally, we propose modifications and extensions to ensure contact detection in any direction and classification between human cooperation and obstacle collision for a collaborative assembly task.

2.1 Torque-Based Cartesian Impedance Control

External forces may be applied to the robot at any moment while executing a desired trajectory. Therefore, the robot must behave in a compliant way toward these forces to avoid any harm of both the robot and the colliding object. Also, a compliant robot behavior allows direct human cooperation without the need of switching to a dedicated admittance controller. The aim of a Cartesian impedance controller [Hogan, 1985] is to establish a mass-damper-spring relationship between the Cartesian pose variation from its reference, $\Delta\xi$, and the Cartesian force, F [Albu-Schäffer and Hirzinger, 2002]:

$$F = I\ddot{\xi} + B\dot{\xi} + K\Delta\xi \tag{1}$$

where I, B, and K are the virtual inertia, damping, and stiffness matrices, respectively. Further, $\Delta\xi = \begin{bmatrix}\Delta p^{\mathrm{T}} & \Delta\varepsilon^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$, where the translation variations in the Cartesian pose are calculated with $\Delta p = p_d - \hat{p}$, and the rotation variations are calculated with $\Delta\mathscr{Q} = \hat{\mathscr{Q}}^{-1}\mathscr{Q}_d$, $\Delta\varepsilon$ being the vector part of the unit-quaternion representation of the rotation variation with respect to the base frame, $\Delta\mathscr{Q}$. Here, $\hat{\xi} = \begin{bmatrix}\hat{p}^{\mathrm{T}} & \hat{\mathscr{Q}}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ is the estimated Cartesian pose of the robot end-effector computed from joint angle measurements, and $\xi_d = \begin{bmatrix}p_d^{\mathrm{T}} & \mathscr{Q}_d^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ is the reference Cartesian pose of the robot end-effector.

With F from Eq. (1), the task torque is calculated as:

$$\tau_{\text{task}} = J^{\mathrm{T}}(q)F \tag{2}$$

where $J(q)$ is the Jacobian relative to the base frame of the robot and q are the sensed joint angular positions. Finally, the contribution from Coriolis and centripetal forces, $C(q,\dot{q})$, is added to the task torque to obtain the reference torque:

$$\tau_{\text{ref}} = \tau_{\text{task}} + C(q,\dot{q})\dot{q} \tag{3}$$

where $\dot{q}$ represents the sensed joint angular velocities. The gravitational-forces term does not appear in Eq. (3) since the robot's internal controller takes care of the gravity compensation.

2.2 Frequency-Based Contact Detection and Classification

Previous research on frequency-based contact detection and classification is based on the idea that the frequency characteristics of motor currents or external force

acting on the robot in an accidental collision situation are different from the ones obtained while a human is cooperating with the robot [Geravand et al., 2013; Kouris et al., 2016; Kouris et al., 2018]. For a sliding time window, if the p–norm of the discrete Fourier transform of the force signal over a given frequency interval is greater than a user-defined force threshold, F_{th}, then it is considered that the contact should be classified an accidental obstacle collision [Kouris et al., 2016; Kouris et al., 2018]. If not, then it is classified as interference from human cooperation.

Figure 1 illustrates the L_∞-norm of the frequency content of the force signal in the frequency range between ω_{min} and ω_{max} using a sliding window of N samples at every time step, as suggested in [Kouris et al., 2016], but using the joint-torque sensors embedded in the robot to estimate the external force. This frequency range has an upper limit determined by the Nyquist frequency ($\omega_{\text{max}} \leq 1/2h$) and a lower limit determined by the measurement duration ($\omega_{\text{min}} \geq 1/Nh$), with h being the sample period. Moreover, the Cartesian impedance control parameters used are the same as in Sec. 3.1.

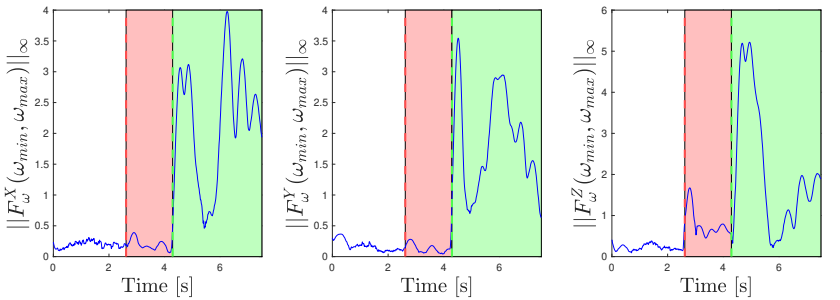


Figure 1. Temporal evolution of the L_∞-norm of the frequency content for all three force directions: F^X, F^Y, and F^Z, in a collaborative assembly task (peg-in-hole). The frequency range used is $\omega \in [10, 100]$ Hz, and the temporal sliding window is $N = 500$ samples long.

In the trajectory used for Fig. 1, the robot transitions from free, undisturbed motion (white background), to obstacle collision (red background), and then to human cooperation (green background). The obstacle collision, which occurs along the Z-direction, can be distinguishable from the free motion when analyzing the frequency content that belongs to F_Z. However, human cooperation also causes an identifiable spike in this same force direction later in the trajectory. Therefore, when the experiments are performed for a robot with compliant behavior using the joint-torque sensors embedded in the robot instead of external force sensors, the distinction of frequency content between accidental collisions and cooperation becomes uncertain, which motivates the proposal of modifications and extensions to the method.

2.3 Contact Detection

From the analysis of Fig. 1, it can be concluded that force-thresholding can be useful for contact detection, but extra variables are needed for classifying the contact if only joint-torque sensors are used. Thus, our proposal consists of a decoupled process between contact detection and classification. For contact detection, the method presented in [Kouris et al., 2016; Kouris et al., 2018] was extended to all three force directions. Therefore, the proposed detection process consists of evaluating if

$$\hat{F}_{\omega}^{i} > F_{\text{th}} \tag{4}$$

for $i \in \{X, Y, Z\}$, where F_{th} is the selected frequency-based threshold valid for any direction i for each time step, and $\hat{F}_{\omega}^{i}$ is equal to the L_{∞}-norm of the discrete Fourier transform of the external sensed force along direction i in the frequency range between ω_{min} and ω_{max} using a sliding window of N samples.

The contact-detection method, which is called at each time step, has been summarized in Algorithm 1. If the robot is moving in free motion (*STATE = FREE*), the condition (4) is evaluated in all directions (Algorithm 1, Line 2). If this condition is true, a contact is detected, and the classifier takes care of evaluating if the contact is accidental or if a human operator is collaborating with the robot (Algorithm 1, Line 5; detailed in Algorithm 2). The contact classifier uses the system's state at the exact time of the contact, which is obtained (in Algorithm 1, Line 4) by performing a backwards search in the external force signal from the contact-detection time along the contact's direction (determined in Algorithm 1, Line 3).

Algorithm 1 Contact Detection

```
1:  if STATE == FREE then
2:      if Check contact == TRUE then
3:          Get contact direction
4:          Get contact time
5:          STATE ← Contact Classifier (Algorithm 2)
6:      end if
7:  else if STATE == COLL then
8:      Get contact direction
9:      if Check new contact direction == TRUE then
10:         Get contact time
11:         STATE ← Contact Classifier (Algorithm 2)
12:     end if
13:     Update active collision directions
14: else if STATE == COOP then
15:     if Check cooperation stopped == TRUE then
16:         STATE ← FREE
17:     end if
18: end if
```

Moreover, while an accidental collision is occurring (*STATE = COLL*), an additional contact could be detected (whose source can be human cooperation) if the force threshold is violated again along any direction, with the exception of the directions that previously experienced the accidental collision (Algorithm 1, Line 9). The contact-detection algorithm will update the active collision directions if a collision along a new direction is detected or if the value of a previously collided direction has stopped violating the threshold (Algorithm 1, Line 13).

Furthermore, when a contact has been labelled as human cooperation (*STATE = COOP*), the contact-detection algorithm will only determine that the cooperation has stopped if the forces along all three directions are below the force threshold, F_{th} (Algorithm 1, Line 15).

2.4 Contact Classification

In contrast with contact detection, the frequency content of the estimated external force is not enough to classify the contact event when only using embedded sensors (as indicated in Fig. 1). Therefore, to properly categorize the contact in a kinesthetic teaching application, knowledge of the performed robot motion can be used.

Our classifier algorithm is based on two assumptions:

- **Assumption A1**: An accidental collision of the robot end-effector or attached tool with a static obstacle must occur in the direction of the movement and with the opposite sense from the one of the motion.

- **Assumption A2**: Human cooperation should have less dominant external force components in the direction of the robot's motion because of the typical spatial layout and interaction of a human operator and a robotic manipulator in kinesthetic teaching.

These two ideas are used to formulate an algorithm next, along with the explanation of the steps of the classifier algorithm.

The contact classifier will be activated once contact has been detected. The contact-classifying algorithm has been summarized in Algorithm 2. The first step is to analyze if, for any of the external forces sensed that have trespassed their threshold (where inequality (4) holds), the force is being applied in the same sense as the motion at the moment that this force signal started rising (Algorithm 2, Line 1). If this is the case, it is straightforward to affirm that human cooperation is occurring (*A1*):

$$\operatorname{sign}\left(\dot{\hat{F}}^{i}_{\text{ext}}\dot{\hat{\xi}}^{i}\right) > 0 \tag{5}$$

only when $\hat{F}^{i}_{\omega} > F_{\text{th}}$, $i \in \{X, Y, Z\}$, F_{th} being the selected frequency-based threshold, and $\dot{\hat{\xi}}$ being the time derivative of the estimated Cartesian pose of the robot end-effector. The time derivative of the estimated external force along direction i, $\dot{\hat{F}}^{i}_{\text{ext}}$,

is preferred compared to the estimated external force along direction i, $\hat{F}^i_{\text{ext}}$, since the time step where this condition is evaluated is when the force signal starts rising.

The second step is, if the inequality (5) is not true for any of the detected contact directions, to perform a new test that evaluates the direction of the external force vector relative to the Cartesian velocity vector (Algorithm 2, Line 4). The reason for this is that compared to intuitive human cooperation for kinesthetic teaching, the largest external force components in accidental collisions must come from directions where the robot's velocity is the highest (*A2*):

$$||\vec{u}_{\dot{\hat{F}}_{\text{ext}}} \oslash \vec{u}_{\dot{\xi}}||_2 < \gamma \tag{6}$$

where $\vec{u}_j$ represents a unitary vector of variable j, γ is the threshold coefficient, and $||\cdot||_2$ is the L_2-norm. Also, $\oslash$ represents the Hadamard division: $C_{jk} = A_{jk}/B_{jk}$ if $C = A \oslash B$. The smaller the threshold coefficient γ is, the closer the external force will be when compared to the Cartesian velocity. If the inequality (6) is evaluated as true, the contact is classified as accidental collision.

Moreover, the inequality (6) is equivalent to evaluating if the unitary external force vector is contained in the ellipsoid defined by the robot's unitary Cartesian velocity vector:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} < 1 \tag{7}$$

where $[x,y,z] = \vec{u}_{\dot{\hat{F}}_{\text{ext}}}$ and $[a,b,c] = \gamma \vec{u}_{\dot{\xi}}$.

Furthermore, unitary vectors have been chosen to avoid having a dependence on the trajectory or on the applied force magnitude, since the classification should be trajectory-independent and also human-operator independent. Therefore, the algorithm only relies on the external force-vector direction with respect to the tangential direction of the motion.

Algorithm 2 Contact Classification

```
1: if A1 == TRUE then
2:     STATE ← COOP
3: else
4:     if A2 == TRUE then
5:         STATE ← COLL
6:     else
7:         STATE ← COOP
8:     end if
9: end if
```

3. Experiments

The goal of the experiments was to obtain realistic data of a collaborative assembly task where human operators were instructed to cooperate intuitively with the robot

to evaluate the contact detection and classification method for kinesthetic teaching applications proposed in Sec. 2.

3.1 Experimental Platform

The experiments were performed using the Panda robot by Franka Emika [Franka Emika, 2019] mounted onto a table; the robot was able to record with a sample frequency of 1 kHz (sample period $h = 1$ ms), using the setup shown in Fig. 2. As mentioned earlier, we only used data from embedded sensors to estimate the variables of interest, $\hat{\xi}$ and $\hat{F}_{\text{ext}}$, which were used for selecting and evaluating the threshold parameters for contact detection, F_{th}, and contact classification, γ. The end-effector Cartesian pose was obtained by applying forward kinematics, $\mathscr{K}$, to the joint angular-position readings provided by the joint encoders [Corke, 2013]:

$$\hat{\xi} = \mathscr{K}(q) \tag{8}$$

Moreover, the estimate of the external forces was obtained from the external joint-torques, which were estimated based on the generalized momentum observer for the Panda robot that was introduced in [Haddadin et al., 2017], by using the Jacobian relative to the base frame of the robot in an inverse way compared to the one presented in Eq. (2):

$$\hat{F}_{\text{ext}} = J^{\dagger}(q)\,\hat{\tau}_{\text{ext}} \tag{9}$$

where the superscript † denotes the Moore-Penrose pseudoinverse.

Furthermore, the Cartesian impedance control parameters (K, B, and I) of Eq. (1) were chosen to be as follows:

- The stiffness K was equal to 150 $[\text{N}/\text{m}]$ for the translational degrees of freedom and equals to 10 $[\text{N}/\text{rad}]$ for the rotational degrees of freedom.
- The damping B was equal to $2\sqrt{K}$.
- The inertia I was equal to 0.

The relationship between the Cartesian position variation and the task force will, with these parameters, behave along all degrees of freedom as a first-order system with a time constant equal to $2/\sqrt{K}$ [Lawrence, 1988]. This way, we ensured stability of the system and proper following of the trajectory reference (overdamped behavior).

3.2 Experimental Procedure

A cylinder insertion, or peg-in-hole, was the collaborative assembly task chosen for the experiments, as shown in Fig. 2. The reason for the selection of this task was that it presents a high amount of interaction with the environment: the hole where the piston must be inserted was narrow in comparison to the piston, and also the piston must make a vertical descent to avoid contacts. Therefore, the probability of

an accidental collision was high if the reference trajectory was not accurate or if the controller introduced uncertainty in the motion. Furthermore, it is an application where the aid of a human operator can be valuable and it did not require a high level of skill for the operator.

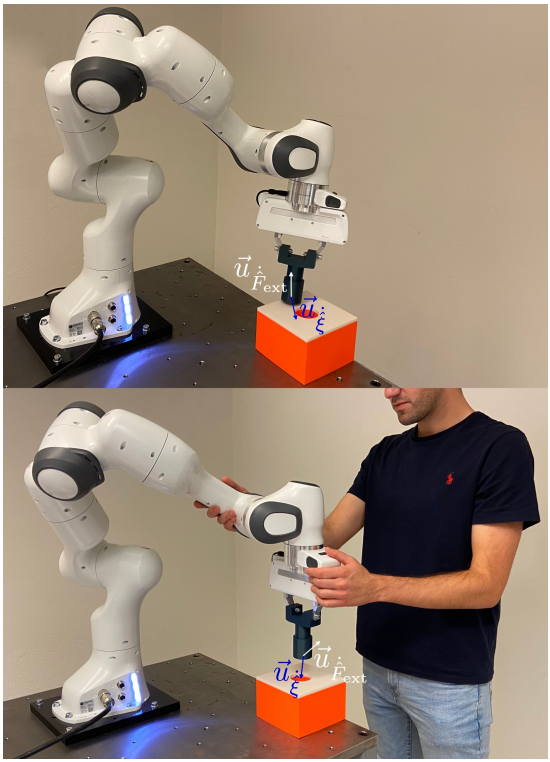


Figure 2. Setup for the collaborative assembly task (peg-in-hole) used for the experiments. Figure 2-A (top) shows an accidental collision scenario, and Fig. 2-B (bottom) shows a human cooperation event. The unitary vector of external forces (white) and the unitary vector of Cartesian velocity (blue) were used in the contact-classification algorithm proposed in Sec. 2.4. A video of the experiments can be found at [*Experiments video* 2022].

The trajectories used range from almost-ideal trajectories, where the robot could complete the insertion task and the only collisions were with the borders of the hole of the box, to failed trajectories where the robot collided with the side of the box and the robot was not able to overcome this collision and insert the piston in the hole without human intervention. Other trajectories used were flawed with manifest collisions with the top of the box, and depending on the nature of the

trajectory, the robot might be able to find its way to the hole with no human input. All trajectories were recorded several times using different initial poses to avoid a trajectory-dependent contact detection and classification.

The desired trajectory of Cartesian poses, $\xi_d(t)$, were recorded before the experiments by leading through the robot and recording the Cartesian pose of the robot end-effector. These trajectories served as the reference for the Cartesian impedance controller of Eq. (1). The reference trajectory was solely time-dependent and did not rely on the robot's current pose, since ideally, the contact-detection and classification algorithm should be implemented in a time-constrained scenario.

Additionally, regarding human cooperation, to test the validity of the assumptions proposed in Sec. 2.4 for a kinesthetic teaching application, the operators were instructed to cooperate in an intuitive manner with the robot to either help the robot with its cylinder insertion task or to push/pull the robot out of its trajectory to avoid colliding with the box. Moreover, since human cooperation in a kinesthetic teaching application may occur at different points of the trajectory in each of the runs, some of the human interventions occurred while the piston was in collision with the box and others while the robot was in free motion. Also, for the sake of data completeness, the operators were also instructed to vary the location of contact with the robot so that the human-cooperation events took place throughout all the links of the robot and not only at a location close to the robot end-effector.

Furthermore, as commented in previous research, human–robot cooperation may be very operator-dependent [Geravand et al., 2013; Briquet-Kerestedjian et al., 2019; Cioffi et al., 2020]. Therefore, a total of four different operators (including three external participants) individually manipulated the robot during the recording of the experiments to test the sensitivity and robustness of the classification. Also, the operators had different experience levels with robot manipulation to analyze the role of this variable for the contact-classification method proposed.

4. Results

The total amount of data that were recorded included 266 contact events. These collision events were divided into 148 accidental obstacle collisions and 118 voluntary human cooperation events. In total, 28 accidental collisions and 28 human cooperation events (from a single human operator) were used for the parameter tuning, and the remaining contact events were used for the method's evaluation.

4.1 Tuning and Evaluation of the Method

First, for contact detection, the force threshold parameter has been assigned a value of $F_{\text{th}} = 0.85$ N, for all Cartesian directions. This value of the force-detection threshold allowed that all contacts recorded in the evaluation experiments were detected, and that no contact-detection false positive was detected. The frequency range used for detection was $\omega \in [10, 100]$ Hz.

Accidental collisions were detected within 71 ms on average, with a standard deviation of 31 ms. On the contrary, voluntary human-cooperation events were detected within 133 ms on average, with a standard deviation of 66 ms. Thus, the capacity of this method to detect contacts fast can be confirmed.

Additionally, the contact-classification threshold parameter was chosen to have a value of $\gamma = 6.2$. This value should be chosen conservatively high, since it is preferred to misclassify human cooperation events than accidental collisions. This idea will be further developed in Sec. 5.

The results for the evaluation of the classification method are shown in Tables 1 and 2. Table 1 displays the confusion matrix for all evaluation experiments performed. It can be seen how 93.3% of the accidental collisions were correctly classified (specificity), whereas for the cooperation events, 88.9% of them were correctly classified (sensitivity).

Table 1. Confusion Matrix for Evaluation Experiments

	Classified as	
	Collision	Cooperation
Collision	112 (93.3%)	8 (6.7%)
Cooperation	10 (11.1%)	80 (88.9%)

Moreover, Table 2 breaks down the recorded cooperation events of Table 1 into the four different operators involved in the evaluation experiments. As commented before, the success rate of the contact classifier varied depending on the human operator. For the method proposed, the sensitivity ranged from 85.2% to 96.2%. Therefore, the sensitivity achieved using this method was still high for the human operator with the lowest classification rate. Furthermore, the sensitivity of the method for experienced operators (Operators 1 and 2 in Table 2) was on average 91% with a standard deviation of 5.3%, which is higher than the sensitivity of the method for inexperienced operators (Operators 3 and 4 in Table 2), which was equal to 86.4% with a standard deviation of 1.2%.

Table 2. Detail of Confusion Matrix for each Human Operator

	Cooperation classified as	
	Collision	Cooperation
Operator 1	1 (3.8%)	25 (96.2%)
Operator 2	3 (14.3%)	18 (85.7%)
Operator 3	4 (14.8%)	23 (85.2%)
Operator 4	2 (12.5%)	14 (87.5%)

4.2 Contact Detection and Classification Example

Figure 1 provides an example of the data extracted for one trajectory execution. The robot was initialized in free motion. It can be seen that at $t = 2.616$ s, a contact was detected along the Z-direction. The contact was detected 87 ms after the external force signal along the Z-direction starts rising. Once contact was detected, the contact-classifying part of the algorithm analyzed the force sense along the Z-direction and compared it to the motion component along this direction using condition (5). Since their signs were opposite, it cannot be determined that the contact was a human-cooperation event. Then, the inequality (6) was used. Since $||\vec{u}_{\hat{F}_{\text{ext}}} \oslash \vec{u}_{\hat{\xi}}||_2 = 2.97 < 6.2 = \gamma$, it can be concluded that the contact was an accidental collision.

Moreover, at $t = 4.290$ s, a new contact was detected along the Y-direction just 85 ms after this new contact occurred. Now, by evaluating condition (5) at the contact time, it was seen that both the force component along the Y-direction and the motion along this direction share the same sign and therefore it is concluded that the contact belongs to the human cooperation category. Furthermore, if the inequality (6) had been evaluated in this situation, the contact would also have been labelled as a human cooperation, since $||\vec{u}_{\hat{F}_{\text{ext}}} \oslash \vec{u}_{\hat{\xi}}||_2 = 41.84 > 6.2 = \gamma$.

Finally, no false positives occurred for contact detection for the entire trajectory shown in Fig. 1, since, for the accidental collision, no force violated the threshold along the X and Y-directions and no force violated the threshold along the Z-direction once the value was lower than this threshold. Also, for the human-cooperation segment, the force was at all times above the force threshold for some of the three Cartesian directions after $t = 4.290$ s.

5. Discussion

In the event that only embedded sensors are available and the external force signal is estimated using the generalized momentum observer [Haddadin et al., 2017], which is currently implemented in commercial collaborative robots such as the Franka Emika Panda [Franka Emika, 2019] and the KUKA LBR product family [Bischoff et al., 2010], the assumption, considered in [Geravand et al., 2013; Kouris et al., 2016; Kouris et al., 2018], that the frequency content of the estimated force is easily distinguishable between voluntary human cooperation and accidental collisions with static obstacles is not certain anymore in a collaborative assembly task. However, we have experimentally demonstrated that the frequency content of the external force signal can still be used for contact detection in this application. Nevertheless, additional sensor information, provided by the embedded joint-position sensors, regarding the robot's motion prior to the detected contact, can be used to classify the contact.

Moreover, several aspects of the implementation of the proposed method allow

freedom to the designer for their selection, and this also has several consequences. First, there is a trade-off between the contact-detection time, defined as the time between the contact occurs and when it is detected, and the force threshold parameter F_{th}: if shorter detection times are desired, more false positives in the contact detection will occur since F_{th} would be smaller. Using only embedded joint-torque sensors causes longer detection times when compared to previous research that included this same force threshold parameter but used external force/torque sensors for a single force-direction detection [Kouris et al., 2016], [Kouris et al., 2018]. Nevertheless, the force-threshold parameter value used in our experiments has been proven able to provide faster response times for all three force directions than alternative machine-learning methods [Golz et al., 2015; Popov et al., 2017; Briquet-Kerestedjian et al., 2019], while presenting no false positives in contact detection.

Second, the contact classifier's threshold parameter, γ, can be varied depending on the desired ratio between the sensitivity (percentage of human cooperation events correctly classified) and the specificity (percentage of accidental collisions correctly classified), since it is not possible to obtain a threshold parameter that allows no ambiguity in the classifier part. Here, specificity must be prioritized to avoid false positives in human cooperation. This is because the proposed method is aimed to be used in a collaborative assembly task where the presence of accidental collisions is expected, and if human cooperation is detected, the cooperation event can be used for trajectory reprogramming using kinesthetic teaching [Karlsson et al., 2017].

Third, the method proposed solely requires tuning of two thresholding parameters (F_{th} and γ) to achieve a proper contact detection and classification along all three force directions, compared to the 6 parameters per joint used for tuning the method in [Geravand et al., 2013] and to the single parameter needed in [Kouris et al., 2016; Kouris et al., 2018] for a single force direction. Also, as discussed in [Kouris et al., 2016], the choice of the virtual inertia, damping, and mass of Eq. (1) will have an effect on the sensed external force signal, and therefore, the two thresholding parameters used in our proposed method must be varied if the desired impedance behavior of the robot is different from the one described using the values defined in Sec. 3.1.

The proposed method was not tested to detect transitions between accidental collision to free motion, or from cooperation to accidental collision since we were not interested in these situations in the experimental application used for evaluation. First, the peg-in-hole application would not have the accidental-collision to free-motion situation, since when the piston impacts the cylinder, it would not stop its impact without human intervention. Second, for this application, a human intervention for kinesthetic teaching would not end up in a purposeful direct transition to an obstacle collision. Also, the proposed method can detect human-cooperation events while an accidental collision with an obstacle is occurring, whereas this transition has not been tested by machine-learning methods [Golz et al., 2015; Popov et al., 2017; Briquet-Kerestedjian et al., 2019; Cioffi et al., 2020] or by the previously-proposed frequency-based meth-

ods [Geravand et al., 2013; Kouris et al., 2016; Kouris et al., 2018]. This feature is especially relevant for applications that use kinesthetic teaching for corrective trajectory demonstration [Karlsson et al., 2017].

In addition, the proposed method's accuracy (percentage of total contacts correctly classified) outperformed other methods previously presented (91.4% for the proposed method, 86.3% for the method in [Cioffi et al., 2020], 89.5% for the method in [Popov et al., 2017], and 81.9% for the method in [Briquet-Kerestedjian et al., 2019]). The method presented in [Golz et al., 2015] provides the highest accuracy, 97.8%, but only one human operator was used for gathering experimental data. Also, the proposed method's accuracy (91.4%) was higher than the accuracy obtained when using the same relevant variables ($\vec{u}_{\hat{F}_{\text{ext}}}$ and $\vec{u}_{\hat{\xi}}$) as parameter estimates in Fisher's Linear Discriminant [Fisher, 1938] for contact classification (83.3%).

Moreover, our method is novel compared to the methods in [Cioffi et al., 2020; Golz et al., 2015; Popov et al., 2017; Briquet-Kerestedjian et al., 2019] in that it has been designed for kinesthetic teaching applications, where a human operator can lead-through the robot for trajectory reprogramming [Karlsson et al., 2017]: the robot's compliant behavior, contrary to the stiffer robot behavior in [Cioffi et al., 2020; Golz et al., 2015; Popov et al., 2017; Briquet-Kerestedjian et al., 2019], allows lead-through without controller switching (as well as providing safety for both the robot and its environment), and also the method is able to classify a human-cooperation event happening while an accidental collision is occurring.

Furthermore, the proposed method, although its evaluation involved only four participants, can be considered robust with respect to different operators since the standard deviation between operators of the sensitivity was equal to 4.4 percentage points, which was lower than in other methods (10.1 percentage points in [Cioffi et al., 2020], where four operators were involved, and 7.3 percentage points in [Briquet-Kerestedjian et al., 2019], where three operators were involved). Also, the difference in accuracy between trained and untrained operators was lower than in [Briquet-Kerestedjian et al., 2019], being 4.6 percentage points (91% and 86.4%, respectively) the difference in our method compared to 14.6 percentage points (86.4% and 71.9%, respectively) in [Briquet-Kerestedjian et al., 2019], which showed the validity of the assumptions for the intuitive human cooperation in kinesthetic teaching that were included in Sec. 2.4 for both trained and untrained operators in a collaborative assembly task. Thus, the proposed method can be used by different operators for kinesthetic teaching in these tasks without the need for retuning.

6. Conclusion

Fast contact detection and classification based on the frequency-response analysis of the estimated external force signals was evaluated, and necessary modifications and extensions to detect and classify a contact in any direction for kinesthetic teaching applications were proposed. Cartesian impedance control was used to allow safe

human cooperation. The only sensors used for obtaining the external force estimate were sensors that are conventionally embedded in commercial collaborative robots and whose values were easily attainable: joint-torque sensors and joint-position encoders/resolvers.

The proposed modified method was proven to provide accurate results for both accidental collision with stiff and static obstacles and voluntary human cooperation in a collaborative assembly task. In addition, the method is trajectory-independent, and was tested for a meaningful number of different operators, showing interesting results for both trained and untrained operators.

References

Albu-Schäffer, A. and G. Hirzinger (2002). "Cartesian impedance control techniques for torque controlled light-weight robots". In: *IEEE International Conference on Robotics and Automation (ICRA)*. Vol. 1. May 11–15. Washington DC, USA, pp. 657–663.

Bischoff, R., J. Kurth, G. Schreiber, R. Koeppe, A. Albu-Schäffer, A. Beyer, O. Eiberger, S. Haddadin, A. Stemmer, G. Grunwald, and G. Hirzinger (2010). "The KUKA-DLR lightweight robot arm—A new reference platform for robotics research and manufacturing". In: *41st International Symposium on Robotics (ISR)*. Jun. 7–9. Munich, Germany, pp. 1–8.

Briquet-Kerestedjian, N., A. Wahrburg, M. Grossard, M. Makarov, and P. Rodriguez-Ayerbe (2019). "Using neural networks for classifying human-robot contact situations". In: *18th European Control Conference (ECC)*. Jul. 25–28. Naples, Italy, pp. 3279–3285.

Cioffi, G., S. Klose, and A. Wahrburg (2020). "Data-efficient online classification of human-robot contact situations". In: *European Control Conference (ECC)*. May 12–15. Saint Petersburg, Russia, pp. 608–614.

Corke, P. (2013). *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer, Berlin, Germany.

Experiments video (2022). `https://youtu.be/5Bq4oB6nDbw`.

Fisher, R. A. (1938). "The statistical utilization of multiple measurements". *Ann. Eugen.* **8**:4, pp. 376–386.

Franka Emika (2019). *Franka Emika Panda – Data Sheet*. `https://www.generationrobots.com/media/panda-franka-emika-datasheet.pdf`. (Visited on 2022-09-13).

Geravand, M., F. Flacco, and A. De Luca (2013). "Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture". In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 6–10. Karlsruhe, Germany, pp. 4000–4007.

Golz, S., C. Osendorfer, and S. Haddadin (2015). "Using tactile sensation for learning contact knowledge: discriminate collision from physical interaction". In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 26–30. Seattle, USA, pp. 3788–3794.

Haddadin, S., A. Albu-Schäffer, A. De Luca, and G. Hirzinger (2008). "Collision detection and reaction: a contribution to safe physical human-robot interaction". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Sep. 22–26. Nice, France, pp. 3356–3363.

Haddadin, S., A. De Luca, and A. Albu-Schäffer (2017). "Robot collisions: a survey on detection, isolation, and identification". *IEEE Transactions on Robotics* **33**:6, pp. 1292–1312.

Hogan, N. (1985). "Impedance control: an approach to manipulation: part I". *Journal of Dynamic Systems, Measurement, and Control* **107**:1, pp. 1–7.

Karlsson, M., A. Robertsson, and R. Johansson (2017). "Autonomous interpretation of demonstrations for modification of dynamical movement primitives". English. In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 29–Jun. 2. Singapore, pp. 316–321.

Kouris, A., F. Dimeas, and N. Aspragathos (2016). "Contact distinction in human-robot cooperation with admittance control". In: *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. Oct. 9–12. Budapest, Hungary, pp. 1951–1956.

Kouris, A., F. Dimeas, and N. Aspragathos (2018). "A frequency domain approach for contact type distinction in human–robot collaboration". *IEEE Robotics and Automation Letters* **3**:2, pp. 720–727.

Lawrence, D. A. (1988). "Impedance control stability properties in common implementations". In: *IEEE International Conference on Robotics and Automation (ICRA)*. Apr. 24–29. Philadelphia, USA, pp. 1185–1190.

Popov, D., A. Klimchik, and N. Mavridis (2017). "Collision detection, localization & classification for industrial robots with joint torque sensors". In: *26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. Aug. 28–31. Lisbon, Portugal, pp. 838–843.

Sadrfaridpour, B. and Y. Wang (2018). "Collaborative assembly in hybrid manufacturing cells: an integrated framework for human–robot interaction". *IEEE Transactions on Automation Science and Engineering* **15**:3, pp. 1178–1192.

Villani, V., F. Pini, F. Leali, and C. Secchi (2018). "Survey on human–robot collaboration in industrial settings: safety, intuitive interfaces and applications". *Mechatronics* **55**, pp. 248–266.

Wrede, S., C. Emmerich, R. Grünberg, A. Nordmann, A. Swadzba, and J. Steil (2013). "A user study on kinesthetic teaching of redundant robots in task and configuration space". *Journal of Human-Robot Interaction* **2**:1, pp. 56–81.

Paper III

Robot Cartesian Compliance Variation for Safe Kinesthetic Teaching using Safety Control Barrier Functions

Julian M. Salt Ducaju **Björn Olofsson**
Anders Robertsson **Rolf Johansson**

Abstract

Kinesthetic teaching allows human operators to reprogram part of a robot's trajectory by manually guiding the robot. To allow kinesthetic teaching, and also to avoid any harm to both the robot and its environment, Cartesian impedance control is here used for trajectory following. In this paper, we present an online method to modify the compliant behavior of a robot toward its environment, so that undesired parts of the robot's workspace are avoided during kinesthetic teaching. The stability of the method is guaranteed by a well-known passivity-based energy-storage formulation that has been modified to include a strict Lyapunov function, *i.e.*, its time derivative is a globally negative-definite function. Safety Control Barrier Functions (SCBFs) that consider the rigid-body dynamics of the robot are formulated as inequality constraints of a quadratic optimization (QP) problem to ensure forward invariance of the robot's states in a safe set. An experimental evaluation using a Franka Emika Panda robot is provided.

1. Introduction

Physical Human–Robot Interaction (pHRI) has become a popular topic in the robotics community, since it addresses the recent trend in the manufacturing industry to replace mass production for mass customization [Schou et al., 2013]. As part of this change of paradigm, human operators have become direct collaborators in robotic tasks, and robots that are compliant toward their environment have gained relevance.

An interesting application of human collaboration in robotics is to reprogram part of the robot's trajectory [Karlsson et al., 2017] by manually guiding the robot, which is known as kinesthetic teaching [Schou et al., 2013]. However, the workspace that humans and robots share may not be entirely available, *e.g.*, if another robot arm is occupying part of the workspace, or if there is sensitive equipment in the workspace. Then, the robot's compliant behavior toward its environment should be modified so that the human operator cannot guide the robot to unsafe situations. In addition, the compliance variations must be done in such a way that the stability of the robot's controller is ensured. Passivity-based energy storage has been used previously to provide a stable variation of the impedance parameters of a robot [Ferraguti et al., 2013; Landi et al., 2018].

Moreover, Safety Control Barrier Functions (SCBFs) have gained attention in recent years [Ames et al., 2019; Wang et al., 2017; Landi et al., 2019; Ferraguti et al., 2020; Rauscher et al., 2016; Singletary et al., 2021], because they provide more formal guarantees for obstacle avoidance than the artificial potential-field methods used in the past for this purpose [Khatib, 1985]. Safety control barrier functions provide safety by enforcing forward invariance of a set, *i.e.*, SCBFs ensure that a system does not leave a safe set [Ames et al., 2019]. They can be formulated as inequality constraints of a quadratic optimization (QP) problem to modify the input to the system [Ames et al., 2019; Wang et al., 2017]. Additionally, SCBFs have been used to perform a minimally-invasive modification of the robot's behavior to avoid safety threats, such as obstacle collisions [Landi et al., 2019; Ferraguti et al., 2020; Rauscher et al., 2016; Singletary et al., 2021].

In this paper, we address the problem of safe kinesthetic teaching by modifying the Cartesian compliant behavior of a robot with respect to its environment in a strictly stable manner, such that we can ensure that the robot's end-effector avoids undesired parts of its workspace. Safety control barrier functions that consider the rigid-body dynamics of the robot are used as inequality constraints of a quadratic optimization problem to online modify the robot's compliance behavior in a minimally-invasive way, so that the human operator can still manipulate the robot while avoiding any safety threat.

The paper is organized as follows: Sec. 2 introduces relevant mathematical concepts that are used in our method. Then, Sec. 3 presents the method for solving the described problem. Section 4 explains the experiments performed, and Sec. 5 presents the results obtained. Finally, a discussion is included in Sec. 6 and conclu-

sions are drawn in Sec. 7.

2. Mathematical Background

In this section, we discuss two relevant mathematical concepts. First, SCBFs for safe set forward invariance. Second, passivity-based energy storage for stable variation of the robot compliant behavior with respect to its environment.

2.1 Safety Control Barrier Functions (SCBFs)

Consider a nonlinear control-affine system:

$$\dot{x} = f(x) + g(x)u \tag{1}$$

that has closed-loop system dynamics with a state-feedback controller k according to

$$\dot{x} = f_{cl}(x,t) = f(x) + g(x)k(x,t) \tag{2}$$

Moreover, define a safe set $\mathscr{C}$, with boundary $\partial\mathscr{C}$ and interior $\text{Int}(\mathscr{C})$, as [Ames et al., 2019]

$$\mathscr{C} = \{x \in \mathbb{R}^n \mid h(x) \geq 0\} \tag{3}$$
$$\partial\mathscr{C} = \{x \in \mathbb{R}^n \mid h(x) = 0\} \tag{4}$$
$$\text{Int}(\mathscr{C}) = \{x \in \mathbb{R}^n \mid h(x) > 0\} \tag{5}$$

For $\mathscr{C}$ to be forward invariant [Ames et al., 2019],

$$\sup_{u \in \mathscr{U}} [L_f h(x) + L_g h(x) u] \geq -\kappa(h(x)) \tag{6}$$

for all $x \in \mathscr{D}$, being h the SCBF, $h : \mathscr{D} \to \mathbb{R}$ with $\mathscr{C} \subseteq \mathscr{D} \subset \mathbb{R}^n$, κ an extended class-$\mathscr{K}_\infty$ function (strictly monotonically increasing), $L_f h(x) = (\partial h/\partial x) f(x)$, and $L_g h(x) = (\partial h/\partial x) g(x)$. Also, the authors in [Wang et al., 2017] highlight the possibility of choosing $\kappa(h) = \gamma h^Z$ ($\gamma > 0$) for any positive odd integer Z.

Furthermore, a quadratic optimization (QP) problem can be formulated to minimize the difference between the input to the system, u, and the nominal state-feedback controller in (2), k^d, while using SCBFs to formulate an inequality constraint that allows obstacle avoidance [Ames et al., 2019]:

$$\begin{aligned} k(x,t) = \arg\min_{u \in \mathbb{R}^m} & \frac{1}{2} \left|\left| u - k^d(x,t) \right|\right|_2^2 \\ \text{s.t. } & \dot{h}(x,t,u) \geq -\kappa(h(x,t)) \end{aligned} \tag{7}$$

2.2 Passivity-Based Energy Storage

Energy storage has previously been used to handle stiffness variations in robots [Ferraguti et al., 2013; Landi et al., 2018]. This formulation is based on the idea of keeping the energy introduced to the system lower that the energy dissipated by the system. The energy dissipated by the system's damping is stored in an energy reservoir with state $z(t) \in \mathbb{R}$ and dynamics

$$\dot{z} = \frac{\varphi}{z} P_D - \frac{\sigma}{z} P_K \tag{8}$$

where P_D and P_K represent the dissipated power due to damping and the power caused by the stiffness variation, respectively. Also, the parameter $\varphi \in \{0,1\}$ controls the storage of dissipated energy and disables the storage if the energy stored is higher than an upper bound $\bar{T}$, and the parameter $\sigma \in \{\varphi,1\}$ controls the injection or extraction of energy from the storage. The energy stored is

$$T(z) = \frac{1}{2} z^2 \tag{9}$$

and its time derivative is

$$\dot{T}(z) = z\dot{z} = \varphi P_D - \sigma P_K \tag{10}$$

A lower bound δ is used for the minimum amount of energy stored. In addition, to avoid singularities, $z(t=0) > 0$ with $T(z(0)) \geq \delta$. Then, the authors in [Ferraguti et al., 2013; Landi et al., 2018] showed that the system is passive with respect to the pair $(F^{\text{ext}}, \dot{\xi})$ if $T(t) \geq \delta$, where $F^{\text{ext}} \in \mathbb{R}^6$ is the external force and $\xi \in SE(3)$ is the end-effector pose of the robot.

3. Method

We aim to formulate a state-feedback controller (2) that allows safe kinesthetic teaching. Here, the nominal state-feedback controller, k^d, represents the robot's desired Cartesian compliant behavior. Then, the robot's compliant behavior is modified by a quadratic optimization problem (7) to ensure that the robot's states stay in a safe set.

3.1 Robot System

The rigid-body dynamics of the robot can be written in the joint space of the robot, $q \in \mathbb{R}^n$, as [Siciliano and Khatib, 2016]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + \tau^{\text{ext}} \tag{11}$$

where $M(q) \in \mathbb{R}^{n \times n}$ is the generalized inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{n \times n}$ describes the Coriolis and centripetal forces effects, $G(q) \in \mathbb{R}^n$ captures the gravity-induced

torques, and $\tau \in \mathbb{R}^n$ represents the input torques, n being the number of joints of the robot. Finally, $\tau^{\text{ext}} \in \mathbb{R}^n$ represents the external torques.

The rigid-body equation of the robot can be written in terms of its end-effector pose, ξ, which is composed by the end-effector's position and orientation:

$$M_\xi(q)\ddot{\xi} + C_\xi(q,\dot{q})\dot{\xi} + G_\xi(q) = F + F^{\text{ext}} \tag{12}$$

where $F \in \mathbb{R}^6$ is the input force, and, for a fully-actuated robot ($n = 6$), $M_\xi \in \mathbb{R}^{6\times6}$, $C_\xi \in \mathbb{R}^{6\times6}$, and $G_\xi \in \mathbb{R}^6$ are equal to

$$M_\xi = J^{-\mathrm{T}}(q)M(q)J^{-1}(q) \tag{13}$$

$$C_\xi = J^{-\mathrm{T}}(q)(C(q,\dot{q}) - M(q)J^{-1}(q)\dot{J}(q))J^{-1}(q) \tag{14}$$

$$G_\xi = J^{-\mathrm{T}}(q)G(q) \tag{15}$$

assuming that the Jacobian relative to the base frame of the robot, $J(q) \in \mathbb{R}^{6\times6}$, has full rank [Khatib, 1987].

By applying partial feedback linearization [Khalil, 2014, Ch. 9], we can write the input, $u \in \mathbb{R}^6$, to the system as the gravity-compensated force:

$$u = F + F^{\text{ext}} - G_\xi(q) \tag{16}$$

Then, by choosing the state vector as $x = [\xi^{\mathrm{T}}, \dot{\xi}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{12}$, the linearized system is

$$\dot{x} = A(q,\dot{q})x + B(q)u \tag{17}$$

where

$$A = \begin{bmatrix} 0_6 & I_6 \\ 0_6 & -M_\xi^{-1}(q)C_\xi(q,\dot{q}) \end{bmatrix}, \; B = \begin{bmatrix} 0_6 \\ M_\xi^{-1}(q) \end{bmatrix} \tag{18}$$

LEMMA 3.1
$M_\xi(q)$ is invertible since $J(q)$ is also invertible. □

Proof. We know that $M(q)$ is invertible because $M(q)$ is a symmetric positive-definite matrix ($M(q) \in S^n_{++}$) [Siciliano and Khatib, 2016]. Then, it can be obtained from (13) that

$$M_\xi^{-1}(q) = (J^{-\mathrm{T}}(q)M(q)J^{-1}(q))^{-1} = J(q)M^{-1}(q)J^{\mathrm{T}}(q) \tag{19}$$

which holds since we have assumed that $J(q)$ has full rank to formulate the rigid-body equation (12). □

3.2 Cost Function

The nominal state-feedback controller (2), $k^d \in \mathbb{R}^6$, should achieve the robot's desired Cartesian compliant behavior. A Cartesian impedance controller [Hogan, 1985] is used to establish a mass-spring-damper relationship between the Cartesian pose variation from its reference, $\Delta\xi = \xi_d - \xi$, ξ_d being the Cartesian reference, and the external Cartesian force, F^{ext}:

$$F^{\text{ext}} = M_\xi(q)\ddot{\xi} + (D + C_\xi(q,\dot{q}))\dot{\xi} - K\Delta\xi \tag{20}$$

where D and K are the virtual damping and stiffness matrices, respectively. The virtual inertia is chosen equal to the robot inertia, $M_\xi(q)$, to avoid inertia shaping [Ott, 2008, Ch. 3.2], so that the input force F does not require feedback from the external forces and is equal to

$$F = K\Delta\xi - D\dot{\xi} + G_\xi(q) \tag{21}$$

Therefore, the gravity-compensated nominal state-feedback controller is

$$k^d = K\Delta\xi - D\dot{\xi} + F^{\text{ext}} \tag{22}$$

and we can formulate a new cost function analogous to the cost function in (7),

$$L(\xi,\dot{\xi},u,F^{\text{ext}}) = \frac{1}{2}\left|\left|u - K\Delta\xi + D\dot{\xi} - F^{\text{ext}}\right|\right|_2^2 \tag{23}$$

Then, the cost function (23) can be expressed in terms of the states and inputs of the system, assuming that $\dot{\xi}_d = 0$, $(x - x_d) = \left[-\Delta\xi^{\text{T}}, \quad \dot{\xi}^{\text{T}}\right]^{\text{T}}$:

$$L(x,u,F^{\text{ext}}) = \frac{1}{2}\left|\left|u + \begin{bmatrix} K, & D \end{bmatrix}(x - x_d) - F^{\text{ext}}\right|\right|_2^2 \tag{24}$$

3.3 Inequality Constraint

A safety function can be formulated to ensure that the safety distance is always greater or equal than the current distance to the obstacles subtracted by the distance needed to brake the system into a full stop with constant and instantaneous acceleration [Wang et al., 2017; Ferraguti et al., 2020]. For our problem, each of these three elements can be formulated as:

- The safety distance D_s is a constant parameter that can be formulated as

$$D_s = r_{rb} + r_o \tag{25}$$

  where r_{rb} and r_o are the radii of two protective spheres around the robot end-effector and an obstacle that represents the undesired part of the workspace, respectively.

- The current distance $||\Delta\rho||$ is defined using the difference between the robot end-effector position and the obstacle's position,

$$\Delta\rho = \rho - \rho_o \tag{26}$$

where $\rho = \begin{bmatrix}\xi_x, & \xi_y, & \xi_z\end{bmatrix}^{\mathrm{T}}$ is the robot's position vector and $\rho_o = \begin{bmatrix}\xi_{o,x}, & \xi_{o,y}, & \xi_{o,z}\end{bmatrix}^{\mathrm{T}}$ is the position of the obstacle. The parameters in ρ_o are constant parameters, since we are considering a static (or semi-static) obstacle.

- The distance needed to brake the robot to full stop is slightly more elaborated. For a constant acceleration, $a_{\mathrm{br}} > 0$, the total distance between a final position ρ_{F} and an initial position ρ_0 after an elapsed time t of an object that starts at ρ_0 with relative velocity $v_{\mathrm{rel}} < 0$ is

$$||\rho_{\mathrm{F}} - \rho_0|| = -v_{\mathrm{rel}}t - \frac{1}{2}a_{\mathrm{br}}t^2 \tag{27}$$

and since the time to brake to full stop is $t = -v_{\mathrm{rel}}/a_{\mathrm{br}}$, the braking distance is equal to

$$||\rho_{\mathrm{F}} - \rho_0|| = \frac{v_{\mathrm{rel}}^2}{2a_{\mathrm{br}}} \tag{28}$$

v_{rel} being the velocity prior to braking in the direction of the obstacle,

$$v_{\mathrm{rel}} = \frac{\Delta\rho^{\mathrm{T}}}{||\Delta\rho||}v \tag{29}$$

where $v = \begin{bmatrix}\dot{\xi}_x, & \dot{\xi}_y, & \dot{\xi}_z\end{bmatrix}^{\mathrm{T}}$. Also, a_{br} is a parameter defined by the user that other authors commonly define as the maximum braking ability of the robot [Wang et al., 2017; Ferraguti et al., 2020]. However, one could decide to choose a smaller value to have even larger margins.

Finally, the safety function is formulated as

$$D_s \geq ||\Delta\rho|| - \frac{v_{\mathrm{rel}}^2}{2a_{\mathrm{br}}} \tag{30}$$

so the SCBF candidate $h : \mathbb{R}^n \to \mathbb{R}$ is

$$h(x) = \sqrt{2a_{\mathrm{br}}(||\Delta\rho|| - D_s)} + \frac{\Delta\rho^{\mathrm{T}}}{||\Delta\rho||}v \tag{31}$$

In addition, we know that

$$\frac{d(||\Delta\rho||)}{dt} = v_{\mathrm{rel}} = \frac{\Delta\rho^{\mathrm{T}}}{||\Delta\rho||}v \tag{32}$$

and from the system's model (17), (18),

$$\frac{d(\Delta\rho)}{dt} = v \tag{33}$$

$$\frac{dv}{dt} = -\Phi v + \Gamma \begin{bmatrix} I_3, & 0_3 \end{bmatrix} u \tag{34}$$

where

$$\Phi = \left(M_\xi^{-1}(q) C_\xi(q,\dot{q}) \right)_{[1:3,1:3]} \in \mathbb{R}^{3\times 3} \tag{35}$$

$$\Gamma = \left(M_\xi^{-1}(q) \right)_{[1:3,1:3]} \in \mathbb{R}^{3\times 3} \tag{36}$$

are submatrices composed by the first three rows and columns of their original matrices (Matlab notation). Then, considering that

$$\frac{d\left(\sqrt{2a_{\text{br}}(||\Delta\rho|| - D_s)}\right)}{dt} = \frac{a_{\text{br}}}{\sqrt{2a_{\text{br}}(||\Delta\rho|| - D_s)}} \frac{d(||\Delta\rho||)}{dt} \tag{37}$$

and

$$\frac{d\left(\frac{\Delta\rho^{\text{T}}}{||\Delta\rho||} v\right)}{dt} = \frac{d\left(\frac{\Delta\rho^{\text{T}}}{||\Delta\rho||}\right)}{dt} v + \frac{\Delta\rho^{\text{T}}}{||\Delta\rho||} \frac{dv}{dt} \tag{38}$$

with

$$\frac{d\left(\frac{\Delta\rho^{\text{T}}}{||\Delta\rho||}\right)}{dt} v = \left(\frac{v^{\text{T}}}{||\Delta\rho||} - \frac{\Delta\rho^{\text{T}} v \Delta\rho^{\text{T}}}{||\Delta\rho||^3} \right) v \tag{39}$$

the time derivative of $h(x)$ in (31) is equal to

$$\begin{aligned} \frac{dh(x)}{dt} = & \frac{a_{\text{br}} \Delta\rho^{\text{T}} v}{||\Delta\rho|| \sqrt{2a_{\text{br}}(||\Delta\rho|| - D_s)}} + \frac{\Delta\rho^{\text{T}} \Gamma \begin{bmatrix} I_3, & 0_3 \end{bmatrix} u}{||\Delta\rho||} \\ & - \frac{\Delta\rho^{\text{T}} \Phi v}{||\Delta\rho||} + \frac{||v||^2}{||\Delta\rho||} - \frac{(\Delta\rho^{\text{T}} v)^2}{||\Delta\rho||^3} \end{aligned} \tag{40}$$

Therefore, to fulfill the condition (6) that ensures that the safe set is forward invariant, we must satisfy the inequality constraint

$$\begin{aligned} & \frac{a_{\text{br}} \Delta\rho^{\text{T}} v}{||\Delta\rho|| \sqrt{2a_{\text{br}}(||\Delta\rho|| - D_s)}} + \frac{\Delta\rho^{\text{T}} \Gamma \begin{bmatrix} I_3, & 0_3 \end{bmatrix} u}{||\Delta\rho||} \\ & - \frac{\Delta\rho^{\text{T}} \Phi v}{||\Delta\rho||} + \frac{||v||^2}{||\Delta\rho||} - \frac{(\Delta\rho^{\text{T}} v)^2}{||\Delta\rho||^3} + \gamma h^Z \geq 0 \end{aligned} \tag{41}$$

which can be rewritten as

$$A_{\text{CBF}} u \leq b_{\text{CBF}} \tag{42}$$

where

$$A_{\text{CBF}} = -\Delta\rho^{\text{T}}\Gamma\begin{bmatrix} I_3, & 0_3 \end{bmatrix} \tag{43}$$

$$b_{\text{CBF}} = \frac{a_{\text{br}}\Delta\rho^{\text{T}}v}{\sqrt{2a_{\text{br}}(||\Delta\rho|| - D_s)}} + ||v||^2 - \frac{(\Delta\rho^{\text{T}}v)^2}{||\Delta\rho||^2} + ||\Delta\rho||\gamma h^Z - \Delta\rho^{\text{T}}\Phi v \tag{44}$$

3.4 Discrete-Time QP Problem Implementation

The discrete-time implementation of the nominal state-feedback controller in (22) allows to obtain the input at time-step i by using the robot state (x_i) and the estimated external force ($\hat{F}_i^{\text{ext}}$) at the same time-step. Therefore, the only free variable of the cost function in (24) is u_i,

$$L(u_i) = \frac{1}{2}\left|\left|u_i + \begin{bmatrix} K, & D \end{bmatrix}(x_i - x_{d,i}) - \hat{F}_i^{\text{ext}}\right|\right|_2^2 \tag{45}$$

The cost function in (45) can be reduced (by eliminating its constant terms) to a standard Quadratic Program (QP) problem:

$$L_r(u_i) = \frac{1}{2}u_i^{\text{T}}Qu_i + c^{\text{T}}u_i \tag{46}$$

where $Q = I_6$ and $c^{\text{T}} = \begin{bmatrix} K, & D \end{bmatrix}(x_i - x_{d,i}) - \hat{F}_i^{\text{ext}}$. It is trivial to see that the quadratic term of the cost function in (46) is positive definite, $Q \in S_{++}^n$.

Moreover, similar to the cost function (46), A_{CBF} and b_{CBF} of the SCBF-based inequality constraint (42) only depend on x_i and therefore they can be treated as constants at each time-step for this problem. Therefore, analogous to (7), the QP problem to online modify the robot's compliant behavior at each time-step i is

$$\begin{aligned} k_i = &\arg\min_{u_i \in \mathbb{R}^6} L_r(u_i) \\ &\text{s.t. } A_{\text{CBF}}u_i \leq b_{\text{CBF}} \end{aligned} \tag{47}$$

3.5 Varying the Compliant Behavior of the System

If the inequality constraint (42) of the QP problem is active, the cost function (46) will not be equal to zero ($L_r(u) > 0$). In this case, since $u \neq k^d - G_\xi(q)$, the relationship between the Cartesian pose variation from its reference and the external Cartesian force (20) is modified,

$$F^{\text{ext}} = M_\xi(q)\ddot{\xi} + (D + C_\xi(q,\dot{q}))\dot{\xi} - K\Delta\xi - \Delta u \tag{48}$$

Then, the additional force Δu can used to vary the stiffness and damping parameters,

$$K'(t) = K + \Delta K(t) \tag{49}$$

$$D'(t) = D + \Delta D(t) \tag{50}$$

where $K', D' \in S^n_{++}$, and

$$\Delta u = \Delta K \Delta\xi - \Delta D \dot{\xi} \tag{51}$$

To vary the Cartesian compliance parameters in a stable manner, we first show that, using an approach based on [Santibáñez and Kelly, 1997], the nominal state-feedback controller (22) is stable.

LEMMA 3.2
The time-varying Lyapunov function

$$V(x,t) = \frac{1}{2}\dot{\xi}^{\mathrm{T}} M_\xi(q)\dot{\xi} + \frac{1}{2}\Delta\xi^{\mathrm{T}} K \Delta\xi - \alpha \Delta\xi^{\mathrm{T}} M_\xi(q)\dot{\xi} \tag{52}$$

where $x = \begin{bmatrix}\Delta\xi^{\mathrm{T}}, & \dot{\xi}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$, shows the global asymptotic stability of the nominal state-feedback controller k^d in (22) for $\alpha > 0$ satisfying

$$\min\left(\sqrt{\frac{\lambda_{m,K}}{\lambda_{M,M_\xi}}},\ \frac{2\lambda_{m,K}}{\lambda_{M,D}},\ \frac{\lambda_{m,D}}{2(\lambda_{M,M_\xi} + k_C||\Delta\xi||)}\right) > \alpha \tag{53}$$

where $\lambda_{m,\Pi}$ and $\lambda_{M,\Pi}$ are the smallest and largest eigenvalues of a matrix Π, respectively, and k_C is a positive constant such that for all $x, y, z \in \mathbb{R}^n$ [Santibáñez and Kelly, 1997]

$$||C_\xi(x,y)z|| \leq k_C||y||||z|| \tag{54}$$

□

Proof. See Appendix A. □

Then, since $M_\xi(q), K, D \in S^n_{++}$, a passive map from the external force F^{ext} to the velocity $\dot{\xi}$ is guaranteed,

$$\dot{V} < \dot{\xi}^{\mathrm{T}} F^{\text{ext}} - \frac{1}{2}\left[\dot{\xi} - \alpha\Delta\xi\right]^{\mathrm{T}} D \left[\dot{\xi} - \alpha\Delta\xi\right] < \dot{\xi}^{\mathrm{T}} F^{\text{ext}} \tag{55}$$

where the passivity condition valid for passive environments is

$$V(t) - V(0) < \int_0^t \dot{\xi}^{\mathrm{T}}(\tau) F^{\text{ext}}(\tau) d\tau \tag{56}$$

However, the additional force Δu (51) produces extra energy, which can break the passivity of the system if the additional energy that is injected into the system causes a positive variation of the stiffness, $\dot{K}'(t) > 0$. Defining H as a Lyapunov function that is equivalent to considering (52) with time-varying $K'(t)$ and $D'(t)$, its time derivative is

$$\dot{H} < \dot{\xi}^{\mathrm{T}} F^{\text{ext}} - P_D + P_K \tag{57}$$

where

$$P_D = \frac{1}{2}\left[\dot{\xi} - \alpha\Delta\xi\right]^{\mathrm{T}} D' \left[\dot{\xi} - \alpha\Delta\xi\right] \tag{58}$$

$$P_K = \frac{1}{2}\Delta\xi^{\mathrm{T}}\dot{K}'\Delta\xi \tag{59}$$

Then, a storage function for the system can be defined as

$$W = H + T \tag{60}$$

where T is the energy stored in a reservoir (9), as in [Ferraguti et al., 2013; Landi et al., 2018]. The time derivative of W is equal to

$$\dot{W} = \dot{H} + \dot{T} < \dot{\xi}^{\mathrm{T}} F^{\mathrm{ext}} - (1-\varphi)P_D + (1-\sigma)P_K \tag{61}$$

Choosing, as in [Landi et al., 2018], that $\sigma = 1$ when $\dot{K}'(t) > 0$,

$$\dot{W} < \dot{\xi}^{\mathrm{T}} F^{\mathrm{ext}} \tag{62}$$

Therefore, analogous to (56), the passivity condition valid for passive environments is

$$W(t) - W(0) < \int_0^t \dot{\xi}^{\mathrm{T}}(\tau) F^{\mathrm{ext}}(\tau) d\tau \tag{63}$$

Moreover, enough stored energy in the reservoir is needed to ensure passivity. We can use the following metric for an arbitrary time interval $\left[t_s, \quad t_f\right]$ to ensure that the storage does not get empty [Ferraguti et al., 2013; Landi et al., 2018],

$$T(t_f) = T(t_s) + \int_{t_s}^{t_f} P_D d\tau - \int_{t_s}^{t_f} P_K d\tau \geq \delta \tag{64}$$

which gives

$$T(t_s) - \delta \geq -\int_{t_s}^{t_f} P_D d\tau + \int_{t_s}^{t_f} P_K d\tau \tag{65}$$

The energy needed to increase the stiffness is equal to

$$\int_{t_s}^{t_f} P_K d\tau = \frac{1}{2}\Delta\xi^{\mathrm{T}}\Delta K\Delta\xi \tag{66}$$

whereas the energy that we can inject into the reservoir in the time interval $[t_s, t_f]$ is

$$\int_{t_s}^{t_f} P_D d\tau = \frac{\eta}{2}\left[\dot{\xi} - \alpha\Delta\xi\right]^{\mathrm{T}} D' \left[\dot{\xi} - \alpha\Delta\xi\right] \tag{67}$$

with $\eta = t_f - t_s$ being the duration of the time interval $[t_s, t_f]$. Therefore, as long as $K'(t), D'(t) \in S^n_{++}$ and (51) is satisfied, the virtual damping coefficient, $D' = D + \Delta D(t)$, can be increased with $\Delta D(t) \geq 0$ to ensure that the energy storage does not get empty, (65), if the stiffness variation is too high.

4. Experiments

In this section, we present the experiments performed to evaluate the proposed method for a kinesthetic teaching application.

4.1 Application Scenario

The application scenario that motivated the experiments regards automatic quality-assurance processes in the food-packaging industry using images recorded from a camera mounted onto the end-effector of a robot [Kakani et al., 2020; Zhu et al., 2021]. Since the distance needed between the camera mounted on the robot and the food item for a correct food-quality analysis is unknown, and varies for different types of food, the trajectory of the robot has to be varied. Then, for each type of food, a human operator can be used to manually guide the robot arbitrarily close to the food item for robot trajectory reprogramming, while ensuring that a collision between the end-effector and the food item does not occur, so that neither of the two is damaged.

4.2 Experimental Setup

The performed experiments consisted of a robot motion in which, during the robot's trajectory execution, a human operator manually guided the robot to bring it arbitrarily close to the object of interest, here, an egg. The experiment was performed using the Panda robot by Franka Emika [Franka Emika, 2019] mounted on a table (see Fig. 1). This robot had seven rotational joints, but since the formulation for the proposed method was focused on fully-actuated non-redundant robots, we locked the last joint ($\theta_7 = -\pi/2$ rad), and then the robot used six degrees of freedom, $n = 6$.

Moreover, the initial impedance parameters used were:

- The initial virtual stiffness K was equal to 250 $[\mathrm{N/m}]$ for the translational degrees of freedom and equaled to 10 $[\mathrm{N/rad}]$ for the rotational degrees of freedom.

- The initial virtual damping D was equal to $2\sqrt{K}$ for all degrees of freedom.

Furthermore, the choice of additional parameters used for the inequality constraint of the quadratic optimization problem (47) were: $\gamma = 1$, $Z = 3$, $D_s = 0.05$ m, and $a_{\mathrm{br}} = 10\ \mathrm{m/s^2}$ (a_{br} was chosen conservatively, since its maximum value was configuration-dependent). Note that γ must be a positive number and Z must be a positive odd integer to guarantee safety [Wang et al., 2017]. Also, a new quadratic optimization problem was solved every 1 ms, since the sampling rate of the robot was 1 kHz.

Figure 1. Setup for the kinesthetic teaching task described in Sec. 4.2. A Franka Emika Panda robot is mounted on a table. The blue piece allows to attach a camera to the robot's end effector. The human operator is manually guiding the robot and displacing it away from its original trajectory and arbitrarily close to the object of interest, here, an egg.

5. Results

In this section, we evaluate the results obtained from the experiments described in Sec. 4. First, Fig. 2 shows a 3D representation of the path $\rho(t)$ traversed by the robot. It can be seen how the external force generated by the human operator displaced the robot from its unperturbed path $\rho_{\text{un}}(t)$, where no external force acted on the robot. The robot was able to avoid the undesired parts of its workspace even when the operator was manually guiding the robot, which was ensured by solving the quadratic optimization problem in (47) at each time-step.

Moreover, Fig. 3 shows the temporal evolution of the safety control barrier function h. It can be seen how the robot end-effector stayed inside the forward-invariant safe set (3), $h(x) \geq 0$, throughout the entire trajectory, thus confirming that undesired parts of the robot's workspace were avoided using the proposed method.

Furthermore, as mentioned in Sec. 3.5, the solution of the quadratic optimization problem (47) was used to online modify the impedance parameters of the Cartesian compliance controller to avoid undesired parts of the robot's workspace. Figure 4 shows the temporal variation of the external force, as well as the stable temporal variation of the virtual stiffness during the trajectory segment where the inequality constraint of the QP optimization problem (47) was active ($t = 2.658$ s

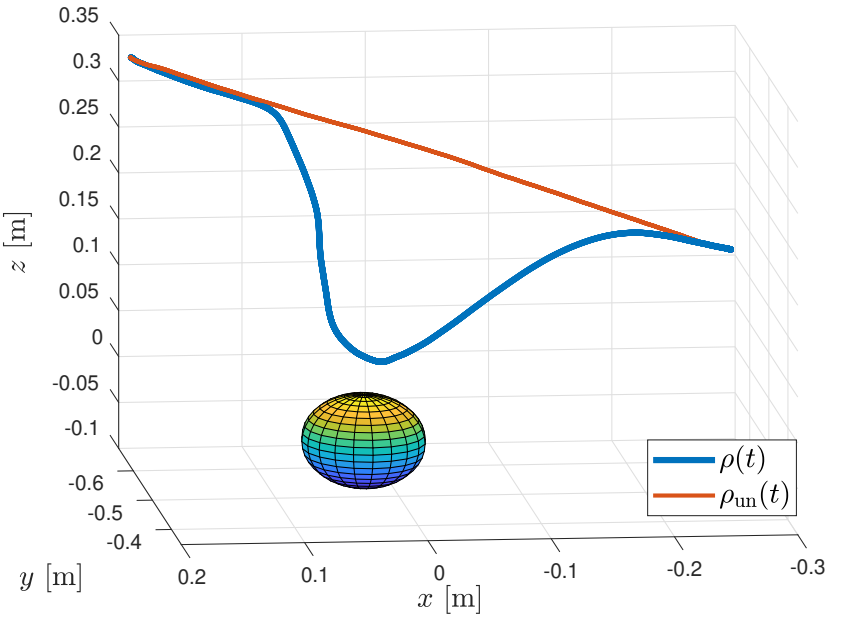


Figure 2. 3D plot of the path $\rho(t)$ traversed by the robot's end-effector. The operator displaced the robot from its unperturbed path $\rho_{\text{un}}(t)$. The plotted sphere is centered at the obstacle (egg) at ρ_o, and its radius is equal to D_s.

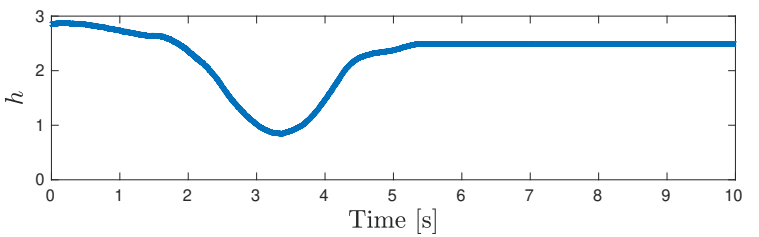


Figure 3. Temporal variation of the SCBF, $h(x)$ in (31), throughout the experiment.

to $t = 3.575$ s). Then, Fig. 5 shows the temporal variation of the joint input torques τ, which were commanded to the robot to achieve the virtual stiffness variation seen in Fig. 4. Figure 5 also shows the unmodified input torques τ_{un} that would be commanded for a constant virtual stiffness, $K' = K$ in (49). It can be seen, in both Figs. 4 and 5, how the nominal controller of the robot was only modified when needed in a minimally-invasive fashion. Therefore, when the SCBF-based inequality constraint (47) was not active, *i.e.*, before $t = 2.658$ s and after $t = 3.575$ s, the desired compliant behavior of the robot, $K' = K$ and $D' = D$, was achieved. Additionally, in this experiment, the virtual stiffness K' in (49) was modified while leaving the virtual damping constant $D' = D$ in (50).

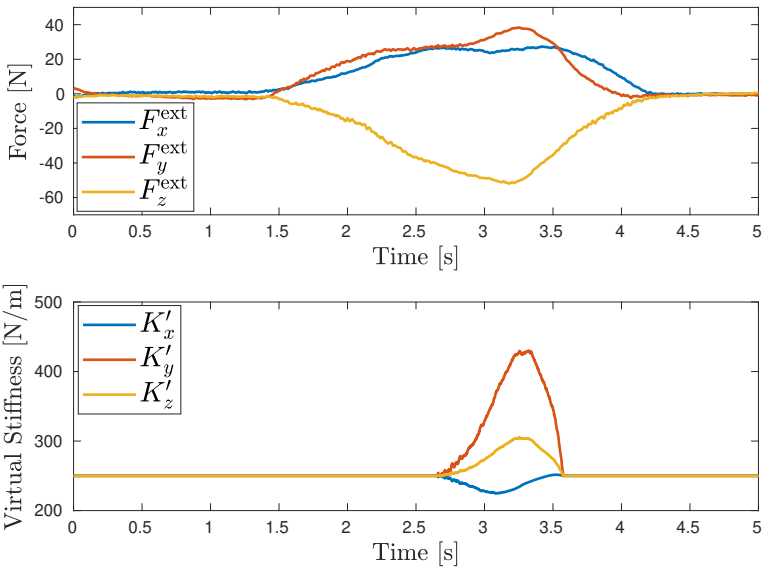


Figure 4. Temporal variation of the external force and the virtual stiffness throughout the experiment.

6. Discussion

In this paper, we have proposed a method to modify the Cartesian compliance parameters of a robot to avoid that human operators manually guide a robot to undesired parts of its workspace in the context of safe kinesthetic teaching. The proposed method modifies a nominal controller, whose goal is to achieve the desired compli-

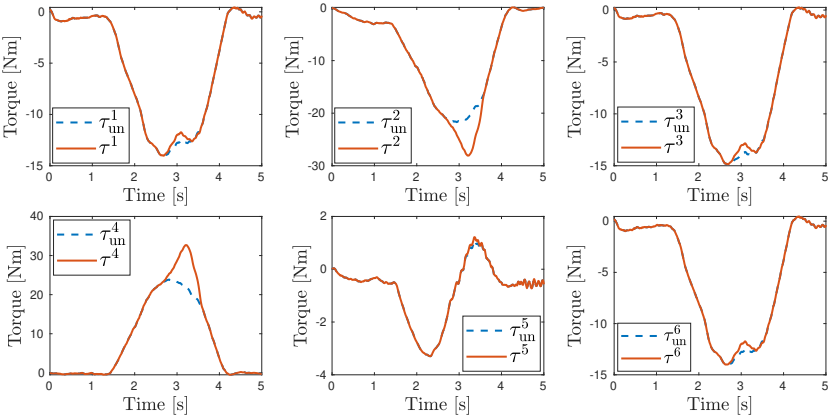


Figure 5. Temporal variation of the input torques τ^j, compared to the unmodified (*i.e.*, without SCBF-based compliance variation) input torques τ^j_{un} for each joint $j \in \{1,\dots,6\}$, throughout the experiment.

ant behavior of the robot, using an SCBF as an inequality constraint of a QP problem to ensure forward invariance of the safe set of robot states.

Prior to the formulation of SCBF-based methods, artificial potential fields have been used for robot obstacle avoidance [Khatib, 1985]. However, SCBFs have recently gained popularity, since they ensure formal guarantees for obstacle avoidance. Also, while potential fields do not emphasize optimality [Liu and Tomizuka, 2016], SCBFs are minimally invasive and only modify the nominal controller behavior if needed [Ames et al., 2019], as illustrated in Sec. 5. In addition, the main appeal of artificial potential fields is the low computational loads needed, but fast problem-solving is also guaranteed for our method, since the proposed QP problem (47) is a convex problem with positive definite quadratic term, $Q \in S^n_{++}$: using a convex optimization solver such as CVXGEN [Mattingley and Boyd, 2012] with C++ to solve (47) took on average 5.2 μs with an standard deviation of 3.1 μs using a single PC (Intel Xeon CPU E3-1245, 3.7 GHz, 4 cores, 64-bit).

Moreover, several authors have formulated SCBFs as inequality constraints of a QP problem for obstacle avoidance in robot manipulators [Landi et al., 2019; Ferraguti et al., 2020; Rauscher et al., 2016; Singletary et al., 2021]. However, it is a novelty of our proposed method to explicitly take the rigid-body dynamics of the robot into consideration: [Landi et al., 2019] and [Ferraguti et al., 2020] considered the robot kinematics, [Rauscher et al., 2016] included a simplified version of the dynamics that neglects the Coriolis and centripetal forces, and [Singletary et al., 2021] performed a purely kinematic implementation of a SCBF but guarantees safety at the level of dynamics by incorporating kinetic energy to the SCBF. The benefit of considering the robot dynamics when formulating our SCBF is that adherence to the

constraints can be guaranteed [Rauscher et al., 2016], as illustrated by the experiments performed (see Fig. 3). In contrast, SCBF-based constraint violations may occur for a kinematic formulation depending on the choice of the optimization parameters, as illustrated in [Singletary et al., 2021]. Also, slight constraint violations were observed in [Rauscher et al., 2016] for a simplified-dynamics formulation.

Furthermore, an additional benefit of using an explicit formulation of the dynamic model of the robot is that it allows to quantify the additional Cartesian force, Δu in (48), required to modify the nominal state-feedback controller k^d to ensure safety. It is a novelty of the proposed method to calculate the required variation of the Cartesian compliant behavior of the system (as shown in Fig. 4) that is necessary to achieve this additional force (49)–(51), so that SCBF-based constraints are satisfied. This is relevant for kinesthetic teaching applications, *e.g.*, in the scenario shown in Sec. 4, since it indicates the changes in the robot's compliant behavior toward external force that human operators would feel when manually guiding the robot. Another example of a kinesthetic teaching scenario where our method may be relevant is for avoiding potential collisions occurring when an operator guides a robot with a sensitive object grasped in its end-effector.

Finally, previous works [Landi et al., 2019; Ferraguti et al., 2020; Rauscher et al., 2016; Singletary et al., 2021] where a robot nominal controller was modified using SCBFs focused on the stability guarantees of the nominal controller. In addition, we provided global asymptotic stability guarantees of convergence to the robot's desired state for the modified controller obtained from the QP problem. We used a passivity-based energy-storage formulation to ensure that the variation of the Cartesian compliance parameters determined by the proposed method is stable. This formulation has previously been used for a robot puncturing task through a three-layers box that simulated the varying stiffness of a human body [Ferraguti et al., 2013], and also to allow stable robot controller-switching between position control and compliance control [Landi et al., 2018]. Therefore, its use in showing stability for SCBF-based modifications of a nominal controller is novel. In addition, our contribution to this energy-storage formulation, as presented in Lemma 3.2 (Sec. 3.5) and its proof (Appendix A), is to replace the nonstrict Lyapunov function used in [Ferraguti et al., 2013; Landi et al., 2018] by a Lyapunov function with strictly negative time-derivative to ensure the strict stability of our method. As a trade-off, the power available to fill the energy storage, P_D in (58), is smaller for our method.

7. Conclusion

Safety control barrier functions have been used to online modify the Cartesian compliant behavior of a robot in a strictly stable manner (global asymptotic stability), so as to avoid that human operators manually guide a robot's end-effector to undesired parts of its workspace in the context of safe kinesthetic teaching. The rigid-body dynamics of the robot is considered in our method to guarantee adherence to the

safety constraints. The proposed method has been successfully evaluated through experiments using a Franka Emika Panda robot for a kinesthetic teaching application.

References

Ames, A. D., S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada (2019). "Control barrier functions: theory and applications". In: *European Control Conference (ECC)*. Jun. 25–28. Naples, Italy, pp. 3420–3431.

Ferraguti, F., M. Bertuletti, C. T. Landi, M. Bonfè, C. Fantuzzi, and C. Secchi (2020). "A control barrier function approach for maximizing performance while fulfilling to iso/ts 15066 regulations". *IEEE Robotics and Automation Letters* **5**:4, pp. 5921–5928.

Ferraguti, F., C. Secchi, and C. Fantuzzi (2013). "A tank-based approach to impedance control with variable stiffness". In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 6–10. Karlsruhe, Germany, pp. 4948–4953.

Franka Emika (2019). *Franka Emika Panda – Data Sheet*. `https://www.generationrobots.com/media/panda-franka-emika-datasheet.pdf`. (Visited on 2022-09-13).

Hogan, N. (1985). "Impedance control: an approach to manipulation: Parts I–III". *Journal of Dynamic Systems, Measurement, and Control* **107**:1, pp. 1–24.

Kakani, V., V. H. Nguyen, B. P. Kumar, H. Kim, and V. R. Pasupuleti (2020). "A critical review on computer vision and artificial intelligence in food industry". *Journal of Agriculture and Food Research* **2**, p. 100033.

Karlsson, M., A. Robertsson, and R. Johansson (2017). "Autonomous interpretation of demonstrations for modification of dynamical movement primitives". English. In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 29–Jun. 2. Singapore, pp. 316–321.

Khalil, H. K. (2014). *Nonlinear control*. Pearson Higher Ed., New York.

Khatib, O. (1985). "Real-time obstacle avoidance for manipulators and mobile robots". In: *IEEE International Conference on Robotics and Automation (ICRA)*. Vol. 2. Mar. 25–28. St. Louis, USA, pp. 500–505.

Khatib, O. (1987). "A unified approach for motion and force control of robot manipulators: the operational space formulation". *IEEE Journal on Robotics and Automation* **3**:1, pp. 43–53.

Landi, C. T., F. Ferraguti, S. Costi, M. Bonfè, and C. Secchi (2019). "Safety barrier functions for human-robot interaction with industrial manipulators". In: *European Control Conference (ECC)*. Jun. 25–28. Naples, Italy, pp. 2565–2570.

Landi, C. T., F. Ferraguti, C. Fantuzzi, and C. Secchi (2018). "A passivity-based strategy for coaching in human-robot interaction". In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 21–25. Brisbane, Australia, pp. 3279–3284.

Liu, C. and M. Tomizuka (2016). "Algorithmic safety measures for intelligent industrial co-robots". In: *IEEE International Conference on Robotics and Automation (ICRA)*. May 16–21. Stockholm, Sweden, pp. 3095–3102.

Mattingley, J. and S. Boyd (2012). "CVXGEN: a code generator for embedded convex optimization". *Optimization and Engineering* **12**:1, pp. 1–27.

Ott, C. (2008). *Cartesian impedance control of redundant and flexible-joint robots*. Springer, Berlin, Germany.

Rauscher, M., M. Kimmel, and S. Hirche (2016). "Constrained robot control using control barrier functions". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Oct. 9–14. Daejeon, Korea, pp. 279–285.

Santibáñez, V. and R. Kelly (1997). "Strict Lyapunov functions for control of robot manipulators". *Automatica* **33**:4, pp. 675–682.

Schou, C., J. Damgaard, S. Bøgh, and O. Madsen (2013). "Human-robot interface for instructing industrial tasks using kinesthetic teaching". In: *44th IEEE International Symposium on Robotics (ISR)*. Oct. 19–27. Seoul, Korea, pp. 1–6.

Siciliano, B. and O. Khatib (2016). *Springer Handbook of Robotics*. Springer, Berlin, Germany.

Singletary, A., S. Kolathaya, and A. D. Ames (2021). "Safety-critical kinematic control of robotic systems". *IEEE Control Systems Letters* **6**, pp. 139–144.

Wang, L., A. D. Ames, and M. Egerstedt (2017). "Safety barrier certificates for collisions-free multirobot systems". *IEEE Transactions on Robotics* **33**:3, pp. 661–674.

Zhu, L., P. Spachos, E. Pensini, and K. N. Plataniotis (2021). "Deep learning and machine vision for food processing: a survey". *Current Research in Food Science* **4**, pp. 233–249.

A. Proof of Lemma 3.2

It is noted in [Santibáñez and Kelly, 1997] that the time-varying Lyapunov function

$$V_1(\xi, \Delta\xi, t) = \frac{1}{2}\dot{\xi}^{\mathrm{T}} M_\xi(q)\dot{\xi} + \frac{1}{2}\Delta\xi^{\mathrm{T}} K \Delta\xi \tag{68}$$

that is often used to show the stability of a Cartesian impedance controller [Ott, 2008, Ch. 3], such as the nominal state-feedback controller k^d (22), is a nonstrict Lyapunov function, *i.e.*, its time derivative is a globally negative-semidefinite function. Then, the authors in [Santibáñez and Kelly, 1997] have proposed the following

alternative Lyapunov candidate to obtain a globally negative-definite time derivative [Santibáñez and Kelly, 1997]:

$$V_2(x,t) = \frac{1}{2}\dot{\xi}^{\mathrm{T}} M_\xi(q)\dot{\xi} + \frac{1}{2}\Delta\xi^{\mathrm{T}} K \Delta\xi - \alpha f(\Delta\xi)^{\mathrm{T}} M_\xi(q)\dot{\xi} \tag{69}$$

where

$$f(\Delta\xi) = \frac{1}{1+||\Delta\xi||}\Delta\xi \tag{70}$$

and $\alpha > 0$ must satisfy

$$\min\left(\sqrt{\frac{\lambda_{m,K}}{\lambda_{M,M_\xi}}},\ \frac{2\lambda_{m,K}}{\lambda_{M,D}},\ \frac{\lambda_{m,D}}{2(k_C + 2\lambda_{M,M_\xi})}\right) > \alpha \tag{71}$$

However, using a scaling factor $f(\Delta\xi)$ in the cross-term of the Lyapunov candidate function can cause slow convergence to the equilibrium point, $\left[\Delta\xi^{\mathrm{T}},\quad \dot{\xi}^{\mathrm{T}}\right] = 0 \in \mathbb{R}^{2n}$. Therefore, we present a solution based on the work by [Santibáñez and Kelly, 1997], but removing the scaling factor $f(\Delta\xi)$:

$$V(x,t) = \frac{1}{2}\dot{\xi}^{\mathrm{T}} M_\xi(q)\dot{\xi} + \frac{1}{2}\Delta\xi^{\mathrm{T}} K \Delta\xi - \alpha\Delta\xi^{\mathrm{T}} M_\xi(q)\dot{\xi} \tag{72}$$

The Lyapunov candidate (72) is equivalent to

$$\begin{aligned} V(x) \quad = \quad & \frac{1}{2}\left[\dot{\xi} - \alpha\Delta\xi\right]^{\mathrm{T}} M_\xi(q)\left[\dot{\xi} - \alpha\Delta\xi\right] \\ & + \frac{1}{2}\Delta\xi^{\mathrm{T}}\left[K - \alpha^2 M_\xi(q)\right]\Delta\xi \end{aligned} \tag{73}$$

Therefore, the Lyapunov candidate is strictly positive ($V(x \neq 0) > 0$ and $V(x=0) = 0$) for

$$\alpha < \sqrt{\frac{\lambda_{m,K}}{\lambda_{M,M_\xi}}} \tag{74}$$

which ensures $K - \alpha^2 M_\xi(q) > 0$.

Moreover, the time-derivative of the Lyapunov candidate (72) is equal to

$$\begin{aligned} \dot{V}(x) \quad = \quad & -\alpha\Delta\xi^{\mathrm{T}}\left[\dot{M}_\xi(q) - C_\xi(q,\dot{q})\right]\dot{\xi} + \alpha\dot{\xi}^{\mathrm{T}} M_\xi(q)\dot{\xi} \\ & -\dot{\xi}^{\mathrm{T}} D\dot{\xi} - \alpha\Delta\xi^{\mathrm{T}} K\Delta\xi + \alpha\Delta\xi^{\mathrm{T}} D\dot{\xi} \end{aligned} \tag{75}$$

Considering that the matrix $\dot{M}_\xi(q) - 2C_\xi(q,\dot{q})$ is skew symmetric [Ott, 2008, Ch. 2]:

$$\begin{aligned} \dot{V}(x) \quad = \quad & -\alpha\Delta\xi^{\mathrm{T}} C_\xi(q,\dot{q})\dot{\xi} + \alpha\dot{\xi}^{\mathrm{T}} M_\xi(q)\dot{\xi} \\ & -\dot{\xi}^{\mathrm{T}} D\dot{\xi} - \alpha\Delta\xi^{\mathrm{T}} K\Delta\xi + \alpha\Delta\xi^{\mathrm{T}} D\dot{\xi} \end{aligned} \tag{76}$$

Then, defining the upper bound on certain terms:

$$-\dot{\xi}^{\mathrm{T}} D\dot{\xi} \leq -\frac{1}{2}\dot{\xi}^{\mathrm{T}} D\dot{\xi} - \frac{1}{2}\lambda_{m,D}\left|\left|\dot{\xi}\right|\right|^2 \tag{77}$$

$$\alpha\dot{\xi}^{\mathrm{T}} M_{\xi}(q)\dot{\xi} \leq \alpha\lambda_{M,M_{\xi}}\left|\left|\dot{\xi}\right|\right|^2 \tag{78}$$

$$-\alpha\Delta\xi^{\mathrm{T}} C_{\xi}(q,\dot{q})\dot{\xi} \leq \alpha k_C\left|\left|\Delta\xi\right|\right|\left|\left|\dot{\xi}\right|\right|^2 \tag{79}$$

it follows that

$$\begin{aligned}\dot{V}(x) \leq\ & -\frac{1}{2}\left[\dot{\xi} - \alpha\Delta\xi\right]^{\mathrm{T}} D\left[\dot{\xi} - \alpha\Delta\xi\right] + \frac{1}{2}\alpha^2\Delta\xi^{\mathrm{T}} D\Delta\xi \\ & -\frac{1}{2}\lambda_{m,D}\left|\left|\dot{\xi}\right|\right|^2 - \alpha\Delta\xi^{\mathrm{T}} K\Delta\xi \\ & +\alpha k_C\left|\left|\Delta\xi\right|\right|\left|\left|\dot{\xi}\right|\right|^2 + \alpha\lambda_{M,M_{\xi}}\left|\left|\dot{\xi}\right|\right|^2\end{aligned} \tag{80}$$

which can be rewritten as

$$\begin{aligned}\dot{V}(x) \leq\ & -\frac{1}{2}\left[\dot{\xi} - \alpha\Delta\xi\right]^{\mathrm{T}} D\left[\dot{\xi} - \alpha\Delta\xi\right] \\ & +\alpha\Delta\xi^{\mathrm{T}}\left[\tfrac{\alpha}{2}D - K\right]\Delta\xi - \frac{1}{2}\lambda_{m,D}\left|\left|\dot{\xi}\right|\right|^2 \\ & +\alpha k_C\left|\left|\Delta\xi\right|\right|\left|\left|\dot{\xi}\right|\right|^2 + \alpha\lambda_{M,M_{\xi}}\left|\left|\dot{\xi}\right|\right|^2\end{aligned} \tag{81}$$

It can be ensured that the term

$$\alpha\Delta\xi^{\mathrm{T}}\left[\tfrac{\alpha}{2}D - K\right]\Delta\xi \tag{82}$$

is strictly negative for

$$\alpha < \frac{2\lambda_{m,K}}{\lambda_{M,D}} \tag{83}$$

and that the term

$$-\frac{1}{2}\lambda_{m,D}\left|\left|\dot{\xi}\right|\right|^2 + \alpha k_C\left|\left|\Delta\xi\right|\right|\left|\left|\dot{\xi}\right|\right|^2 + \alpha\lambda_{M,M_{\xi}}\left|\left|\dot{\xi}\right|\right|^2 \tag{84}$$

is strictly negative for

$$\alpha < \frac{\lambda_{m,D}}{2(\lambda_{M,M_{\xi}} + k_C||\Delta\xi||)} \tag{85}$$

Therefore, if $\alpha > 0$ satisfies (53)

$$\min\left(\sqrt{\frac{\lambda_{m,K}}{\lambda_{M,M_{\xi}}}},\ \frac{2\lambda_{m,K}}{\lambda_{M,D}},\ \frac{\lambda_{m,D}}{2(\lambda_{M,M_{\xi}} + k_C||\Delta\xi||)}\right) > \alpha \tag{86}$$

the Lyapunov candidate function $V(x)$ is strictly positive ($V(x \neq 0) > 0$ and $V(x = 0) = 0$) and its time-derivative is strictly negative ($\dot{V}(x) < 0$).

A

Notation Overview

Some symbols have been used with different meanings in each of the papers included in this thesis. This section shows the symbols that have been used differently, and also their meaning for each of the papers.

Table A.1 Lowercase Latin letters used differently in each of the papers included in this thesis.

Symbol	**Meaning**		
	Paper I	**Paper II**	**Paper III**
a	-	-	Acceleration
c	-	-	Linear term of cost function
g	Gravity-induced torque	-	-
h	Sample period		Barrier function
k	Discrete-time step	-	State-feedback controller
x	State vector	-	State vector
	X-coordinate		
y	Output vector	-	-
	Y-coordinate		
z	-	-	Reservoir state
	Z-coordinate		

Table A.2 Uppercase Latin letters used differently in each of the papers included in this thesis.

Symbol	**Meaning**		
	Paper I	**Paper II**	**Paper III**
A	State matrix	-	State matrix
B	Input matrix	Virtual damping matrix	Input matrix
C	Coriolis and centripetal forces matrix		
	Output matrix	-	-
D	Feedforward matrix	-	Virtual damping matrix
F	-	Cartesian force vector	
G	-	-	Gravity vector
H	Number of control horizon steps	-	Lyapunov function
I	Identity matrix	Virtual inertia matrix	Identity matrix
L	-	Norm	Cost function
	-	-	Lie derivative
N	Null-space projection matrix	Number of samples	-
Q	State weighting function	-	Quadratic term of cost function
T	Time horizon	-	Stored energy
V	Cost function	-	Lyapunov function
Z	-	Z-coordinate	Odd constant

Table A.3 Lowercase Greek letters used differently in each of the papers included in this thesis.

Symbol	**Meaning**		
	Paper I	**Paper II**	**Paper III**
α	Amplitude of sinusoidal signal	-	-
γ	End-effector twist	Threshold coefficient	Positive constant
ζ	Discrete-state variable	-	-
η	-	-	Sample time
ρ	-	-	Position vector
σ	Standard deviation	-	Parameters controlling injection/extraction of stored energy
ω	End-effector angular velocity	Frequency	-