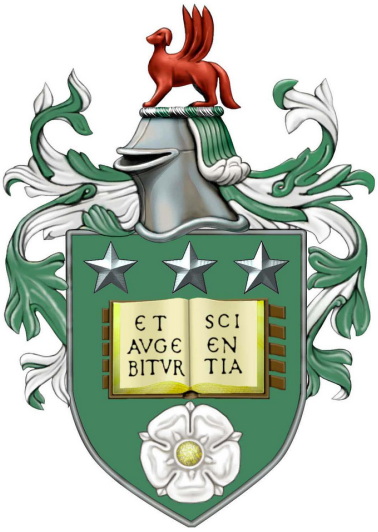
Manipulation Planning for Forceful Human Robot Collaboration

Lipeng Chen

Submitted in accordance with the requirements
for the degree of Doctor of Philosophy



Supervisors:

Prof. Mehmet R. Dogar, Main Supervisor (University of Leeds, UK)
Prof. Anthony G. Cohn (University of Leeds, UK)

Thesis Committee:

Prof. Netta Cohen, Chair (University of Leeds, UK)
Prof. Tamim Asfour, External Examiner (Karlsruhe Institute of Technology, DE)
Prof. Eric Atwell, Independent Chair (University of Leeds, UK)

School of Computing, University of Leeds
December 2019

The candidate confirms that the work submitted is his own. The contribution of the candidate to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Acknowledgements

I would like to express sincere gratitude to all those who have offered me help during my PhD studies.

I would like to thank my supervisor Prof. Mehmet Dogar, for his guidance, encouragement, support and friendship. I never expected a supervisor could be so helpful and supportive. You led me into the world of robotics.

I would like to thank Prof. Anthony Cohn, who was my co-supervisor and was always there whenever I needed advice and help. I learned so much from you.

I would like to thank my committee members Prof. Tamim Asfour and Prof. Netta Cohen for their invaluable comments and insightful guidance. It is such great honour to have you in my thesis. I would like to thank Prof. Eric Atwell for his arrangement of my viva.

I would like to thank Dr. Luis Figueredo, who have instructed and helped me so much in the past two years. It was so great to have you. My thanks also go to Prof. Samit Chakrabarty, Rafael Aguiar and Thomas Richards, whom we have been collaborating with on HRI. Thank you all for the discussions, the hard work, and the fun.

I would like to thank Judi Drew, Brandon Bennett and Gaynor Butterwick. Your great help has made every official business so much easier.

Thanks to all those whom I had worked with in the Robotic Manipulation Lab: Simon Obute, Francesco Foglino, Wissam Bejjani, Alexia Toumpa, Rafael Papallas, Logan Dunbar, Mohamed Hasan, Leo Pauly, Muhannad Al-Omari, Jawad Tayyub and Paul Duckworth. You all made this lab such an incredible place to work at.

Thanks to Baxter, who was the first robot I had ever known, was a friend that I have been staying with for the longest time in the lab, and was the one who gave me the most pain and joy in the past three years.

I would finally like to express my gratitude to my beloved parents, my sister Yuanyuan and my brother-in-law Peng Cao, who have been helping me out of difficulties and supporting with unconditional love. I would especially like to thank my nieces Chenxi and Yangyang. You little angels always bring me so much joy and happiness.

Abstract

This thesis addresses the problem of manipulation planning for forceful human-robot collaboration. Particularly, the focus is on the scenario where a human applies a sequence of changing external forces through forceful operations (e.g. cutting a circular piece off a board) on an object that is grasped by a cooperative robot. We present a range of planners that 1) enable the robot to stabilize and position the object under the human applied forces by exploiting supports from both the object-robot and object-environment contacts; 2) improve task efficiency by minimizing the need of configuration and grasp changes required by the changing external forces; 3) improve human comfort during the forceful interaction by optimizing the defined comfort criteria.

We first focus on the instance of using only robotic grasps, where the robot is supposed to grasp/regrasp the object multiple times to keep it stable under the changing external forces. We introduce a planner that can generate an efficient manipulation plan by intelligently deciding when the robot should change its grasp on the object as the human applies the forces, and choosing subsequent grasps such that they minimize the number of regrasps required in the long-term. The planner searches for such an efficient plan by first finding a minimal sequence of grasp configurations that are able to keep the object stable under the changing forces, and then generating connecting trajectories to switch between the planned configurations, i.e. planning regrasps. We perform the search for such a grasp (configuration) sequence by sampling stable configurations for the external forces, building an operation graph using these stable configurations and then searching the operation graph to minimize the number of regrasps. We solve the problem of bimanual regrasp planning under the assumption of no support surface, enabling the robot to regrasp an object in the air by finding intermediate configurations at which both the bimanual and unimanual grasps can hold the object stable under gravity. We present a variety of experiments to show the performance of our planner, particularly in minimizing the number of regrasps for forceful manipulation tasks and planning stable regrasps.

We then explore the problem of using both the object-environment contacts and object-robot contacts, which enlarges the set of stable configurations and thus boosts the robots capability in stabilizing the object under external forces. We present a planner that can intelligently exploit the environments and robots stabilization capabilities within a unified planning framework to search for a minimal number of stable contact configurations. A big computational bottleneck in this planner is due to the static stability analysis of a large

number of candidate configurations. We introduce a containment relation between different contact configurations, to efficiently prune the stability checking process. We present a set of real-robot and simulated experiments illustrating the effectiveness of the proposed framework. We present a detailed analysis of the proposed containment relationship, particularly in improving the planning efficiency.

We present a planning algorithm to further improve the cooperative robot behaviour concerning human comfort during the forceful human-robot interaction. Particularly, we are interested in empowering the robot with the capability of grasping and positioning the object not only to ensure the object stability against the human applied forces, but also to improve human experience and comfort during the interaction. We address human comfort as the muscular activation level required to apply a desired external force, together with the human spatial perception, i.e. the so-called peripersonal-space comfort during the interaction. We propose to maximize both comfort metrics to optimize the robot and object configuration such that the human can apply a forceful operation comfortably. We present a set of human-robot drilling and cutting experiments which verify the efficiency of the proposed metrics in improving the overall comfort and HRI experience, without compromising the force stability.

In addition to the above planning work, we present a conic formulation to approximate the distribution of a forceful operation in the wrench space with a polyhedral cone, which enables the planner to efficiently assess the stability of a system configuration even in the presence of force uncertainties that are inherent in the human applied forceful operations. We also develop a graphical user interface, which human users can easily use to specify various forceful tasks, i.e. sequences of forceful operations on selected objects, in an interactive manner. The user interface ties in human task specification, on-demand manipulation planning and robot-assisted fabrication together. We present a set of human-robot experiments using the interface demonstrating the feasibility of our system.

In short, in this thesis we present a series of planners for object manipulation under changing external forces. We show the object contacts with the robot and the environment enable the robot to manipulate an object under external forces, while making the most of the object contacts has the potential to eliminate redundant changes during manipulation, e.g. regrasp, and thus improve task efficiency and smoothness. We also show the necessity of optimizing human comfort in planning for forceful human-robot manipulation tasks. We believe the work presented here can be a key component in a human-robot collaboration framework.

Contents

1 Introduction **1**

1.1 Background . . . 1

1.2 Our Approach . . . 3

1.2.1 Forceful Operations . . . 4

1.2.2 Manipulation Planning under Changing External Forces . . . 5

1.2.3 Human Comfort . . . 6

1.3 Publication Note . . . 6

2 Literature Review **7**

2.1 Grasp Analysis and Cooperative Manipulation . . . 7

2.2 Multi-Step Manipulation Planning and Regrasping . . . 11

2.3 Human Comfort in fHRC . . . 14

3 Terminology, Notations and Problem Fundamentals **19**

3.1 Forceful Operations and Tasks . . . 19

3.2 Human, Robot, Environment and Object Contacts . . . 21

4 Modelling, Stability Check and Specification of Forceful Operations **23**

4.1 Modelling of a Forceful Operation . . . 23

4.1.1 Idealized Operation Model . . . 24

4.1.2 Conic Operation Model . . . 25

4.2 Stability Check of a forceful Operation . . . 27

4.2.1 Stability Check with Robot Grasps . . . 27

4.2.2 Stability Check with Environmental Contacts . . . 30

4.3 A Graphical Interface: Specification of Forceful Operations . . . 31

4.4 Experiments and Results . . . 33

4.4.1 Modelling Forceful Operations with Experimental Data . . . 33

4.4.2 Effect of Conic Model in Stability Check . . . 34

5 Manipulation Planning under Changing External Forces—Robot Grasps **37**
5.1 Manipulation Planning under Changing External Forces . . . 40
5.1.1 Problem Definition . . . 41
5.1.2 Approach Overview . . . 42
5.2 Planning Approach . . . 43
5.2.1 Stable Configurations . . . 43
5.2.2 Connectivity of Grasps . . . 45
5.2.3 Sampling Stable Intersections of Grasp Manifolds . . . 47
5.2.4 Connectivity of Sequence of Manifold Intersections . . . 47
5.3 Experiments and Results . . . 47
5.3.1 Analysis of Planning Performance in Simulated Experiments . . . 50
5.3.2 Analysis of Planning Performance in Real Human-Robot Experiments . . . 55

6 Manipulation Planning under Changing External Forces—Environmental Contacts **59**
6.1 Problem Formulation . . . 61
6.2 Planning Approach . . . 62
6.2.1 Manipulation Planning Using Operation Graph . . . 62
6.2.2 Finding Stable Configurations for an Operation . . . 64
6.3 Containment-Based Stability Checks . . . 64
6.3.1 Naive Stability Check of Q_{s} . . . 65
6.3.2 Containment-Based Stability Check of Q_{s} . . . 66
6.4 Experiments and Results . . . 68
6.4.1 Analysis of Minimizing Configuration Changes . . . 69
6.4.2 Analysis of Planning Efficiency . . . 70

7 Planning for Comfortable Forceful Human-Robot Collaboration **73**
7.1 Optimization Overview . . . 75
7.2 Muscular Comfort . . . 77
7.2.1 Human Arm Modelling . . . 77
7.2.2 Human Muscular Effort . . . 79
7.3 Peripersonal-Space Comfort . . . 80
7.4 Experiments and Results . . . 82
7.4.1 Consistency and Effectiveness of Muscular Comfort . . . 83
7.4.2 Effectiveness of Comfort Optimization . . . 86
7.4.3 Human Experiments . . . 87

8 Conclusion and Future Work **93**
8.1 Conclusion . . . 93
8.2 Future Work . . . 94

List of Figures

1.1 An example of forceful human-robot collaboration to fabricate a chair. 2
1.2 Our work towards stable, efficient and comfortable fHRC. 4

3.1 A circular cutting task consisting of a continuous sequence of cutting operations. . . . 19
3.2 A chair fabricating task involving a sequence of drilling, cutting and inserting operations. 20
3.3 Definitions in a forceful human-robot collaborative operation. 21

4.1 A forceful operation $\boldsymbol{F}$ ideally generates a deterministic operation force $\boldsymbol{f}$ on a target object along/about an expected operation axis (depicted as the blue axes). 24
4.2 The operation force $\boldsymbol{f}$ may deviate from the expected operation axis to some extent. . . 25
4.3 The spherical cone models the force deviation strictly, while the polyhedral cone conservatively approximates the spherical cone with a limited number of primitive forces. . 26
4.4 Left: We are interested in checking whether a composite configuration $\boldsymbol{q}$ (along with its corresponding grasp $\boldsymbol{g}$) is stable against a forceful operation $\boldsymbol{F}$. Right: We approximate the grasp wrench space of a grasp $\boldsymbol{g}$ with an axis-aligned box in the six-dimensional wrench space. 29
4.5 The environmental and robot contacts specify a group of contact region(s) on the object. 30
4.6 The workflow of human-robot collaboration in performing forceful tasks with the graphical user interface. 33
4.7 We collected experimental data from 30 trials to model the distribution of a forceful operation in the wrench space. The red polyhedral cones are the extracted models over 30 trials. The force plots on the right and the red dots inside the cones are the force data over one experimental trial. 34

5.1 The human is cutting a circular piece out of a board which is held by a dual-arm robot. 38
5.2 Task failures during cutting (a) and drilling (b). 39
5.3 A manipulation planning problem to keep an object stable under changing external forces. 40
5.4 Overview of the approach. 42
5.5 We build an operation graph to search for a minimal sequence of configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$ stable against $\{\boldsymbol{F}_i\}_{i=1}^{m}$. 44

5.6 A grasp graph $\mathcal{G}_{\mathrm{g}}$. Each node in the grasp graph represents a bimanual or a unimanual grasp. 46
5.7 A grasp sequence by the min-regrasp planner for a random-puncturing task. The dark points indicate the puncturing operations applied during the current grasp. The arrows indicate regrap actions. 51
5.8 A grasp sequence by the greedy planner for a V-puncturing task. The dark points indicate the puncturing operations applied during the current grasp. The arrows indicate regrap actions. 51
5.9 A plan by the min-regrasp planner for a V-puncturing task which contains two regrasps. 52
5.10 A grasp sequence by the min-regrasp planner for 40 circular puncturing operations on a round board. 52
5.11 Regrasping a heavy object: The robot moves the heavy object to some intermediate poses before regrasping. 52
5.12 Human-robot collaboration - A grasp sequence by the min-regrasp planner for the Drilling&Cutting task. 53
5.13 Human-robot collaboration - A cross cutting task. 54
5.14 Human-robot collaboration - A V-puncturing task. 54
5.15 Human-robot collaboration - A square cutting task. 55
5.16 A solution by the min-regrasp planner for the table assembly task in Fig. 3.2. 56

6.1 The robot holds chair sub-assemblies stable under a sequence of forceful operations, by exploiting deliberate object contacts with both the environment (e.g. a table surface) and robot (e.g. pressing or grasping). 60
6.2 The planner exploits deliberate environmental contacts to keep an object stable under changing external forces. 62
6.3 We build an operation graph to search for efficient manipulation solutions with a minimal number of configuration changes. We adopt a more fine-grained weighting scheme: from the node $\boldsymbol{q}^a$ to the node $\boldsymbol{q}^b$, the object contact with the environment changes. The robot contact changes by adding one more gripper on the object (The other is held constantly). Thus we weight the link from $\boldsymbol{q}^a$ to $\boldsymbol{q}^b$ in the operation graph as 2. 63
6.4 We build a containment graph over all system configurations in $\mathcal{Q}_{\mathrm{c}}$ to represent their containments. The red segments illustrate environmental contacts with the object. . . . 67
6.5 Two manipulation plans for a rectangular cutting task (Task 1) consisting of 20 forces. Top: Solution A contains 4 configuration changes, with a robot regrasp (from left arm in $\boldsymbol{q}_2$ to right arm in $\boldsymbol{q}_3$). Bottom: Solution B contains 3 configuration changes. 70
6.6 A manipulation plan for a stool fabricating task (Task 2) consisting of 20 forces. The solution contains 1 configuration change (environmental contact from $\boldsymbol{q}_1$ to $\boldsymbol{q}_2$). 71

7.1 Human-robot collaboratively drilling on a board. 74

7.2 Human-robot kinematic chain modeling: the human arm is modelled as a serial kinematic chain with seven degrees of freedom (DOFs). 78
7.3 We tested our metrics on 16 cutting operations forming the circular cutting task and nine drilling operations uniformly distributed on a foam board. 83
7.4 Two different arm configurations for a cutting operation with different muscular comfort. 84
7.5 Muscular comfort values for different cutting points along a circle. The circled letters on the graph correspond to the configurations in Fig. 7.4. 85
7.6 Comfortable human configurations for the circular cutting task. Circled letters correspond to configurations shown in Fig. 7.5. 85
7.7 Optimization results for a cutting operation. The comfort values are: (a) Musc. comfort: 8.34; Perip. comfort: 20.43. (b) Musc. comfort: 48.53; Perip. comfort: 23.38. (c) Musc. comfort: 3.98; Perip. comfort: 44.23. (d) Musc. comfort: 47.10; Perip. comfort: 43.30. 88
7.8 Optimization results for a drilling operation. The comfort values of the four planner are: (a) Musc. comfort: 3.70; Perip. comfort: 20.21. (b) Musc. comfort: 42.33; Perip. comfort: 21.88. (c) Musc. comfort: 4.72; Perip. comfort: 44.40. (d) Musc. comfort: 59.19; Perip. comfort: 44.75. 88
7.9 We conducted a set of human-robot experiments to evaluate the actual human comfort perception during forceful operations with regard the optimized configurations by the proposed comfort metrics. 89
7.10 The participants' perception of the peripersonal-space comfort (yellow dots) for drilling (left) and cutting (right) with respect to the **(a)** user preferred configurations (black), **(b)** the random configurations (red), and **(c)** the optimized configurations (blue). 89
7.11 The participants' perception of the muscular comfort (yellow dots) for drilling (left) and cutting (right) with respect to the **(a)** user preferred configurations (black), **(b)** the random configurations (red), and **(c)** the optimized configurations (blue). 90

List of Tables

3.1 Nomenclature. 22

5.1 Numbers of regrasps (with standard deviations in parentheses) of three planners on three categories of tasks. 50

5.2 Planning time for both heavy and light objects. Times are in seconds. Standard deviations are in parentheses. 54

6.1 Numbers of configuration changes for each task by the baseline and proposed planner with three sample sets of different sizes. 69

6.2 Average time (s) for building the operation graph and the overall planning time in parentheses for each task over 50 runs. 71

6.3 Average time (s) for building the containment graph. 72

7.1 Average results of four planners on 16 cutting and 9 drilling operations. Normalized with results of the Random Planner. The larger a value is, the more comfortable its corresponding configuration is. 87

List of Algorithms

1 Manipulation Planning under Changing Forces 48
2 Manipulation Planning Using Operation Graph 65
3 Naive Stability Check 66
4 Containment-Based Stability Check 68

Abbreviations

HRI	Human-Robot Interaction
HRC	Human-Robot Collaboration
pHRC	Physical Human-Robot Collaboration
fHRC	Forceful Human-Robot Collaboration
LP	Linear Programming
GWS	Grasp Wrench Space
TWS	Task Wrench Space
OWS	Object Wrench Space
CSP	Constraint Satisfaction Problem
EMG	Electromyography
DOF	Degree of Freedom
FME	Force Manipulability Ellipsoid

Chapter 1

Introduction

1.1 Background

Manipulation planning has been a central topic in robotics for decades for automatically generating motion sequences allowing robots to manipulate movable objects through geometrical constraints. Even though promising results have been obtained through extensive theoretical research and studies, robots in this body of research frequently demonstrate unsatisfying performance in a large set of practical applications, e.g. assembly. One dominant reason accounting for such failures is the presence of external forces applied onto the manipulated objects, particularly changing external forces, which can be required in various real-world robotic applications, but usually ignored or oversimplified from manipulation planning. An example chair assembly application is shown in Fig. 1.1, for which most studies focus on planning robot behaviours to manipulate target objects satisfying geometrical constraints, such as the relative poses among different chair sub-assemblies. The presence of changing external forces in such robotic applications leads to a more constrained but more realistic version of the classical manipulation planning problem.

Moreover, robots are nowadays playing a much more prominent role in human life than ever before. Even though many studies in human-robot interaction (HRI) have demonstrated the potential and feasibility of robot systems in collaboration with humans, they are mostly about humans and robots avoiding colliding each other while interacting in shared environments (Bauer *et al.*, 2008; Mainprice & Berenson, 2013; Park *et al.*, 2016; Pérez-D'Arpino & Shah, 2015). The emergence of more functional and skilled robots has greatly liberated humans from a variety of tedious and labouring work. However, there are still a large set of forceful tasks, e.g. cutting and assembling, requiring strong forceful interactions among humans and robots. The intensive *physical contacts* in close proximity expect a higher level of adaptability from the collaborative robots for, e.g. enhanced *human comfort and safety*. Despite growing efforts that have been made to convert this vision into a reality, there is still a lack of significant advances in robot grasping and manipulation planning, particularly for forceful human-robot collaboration (fHRC).

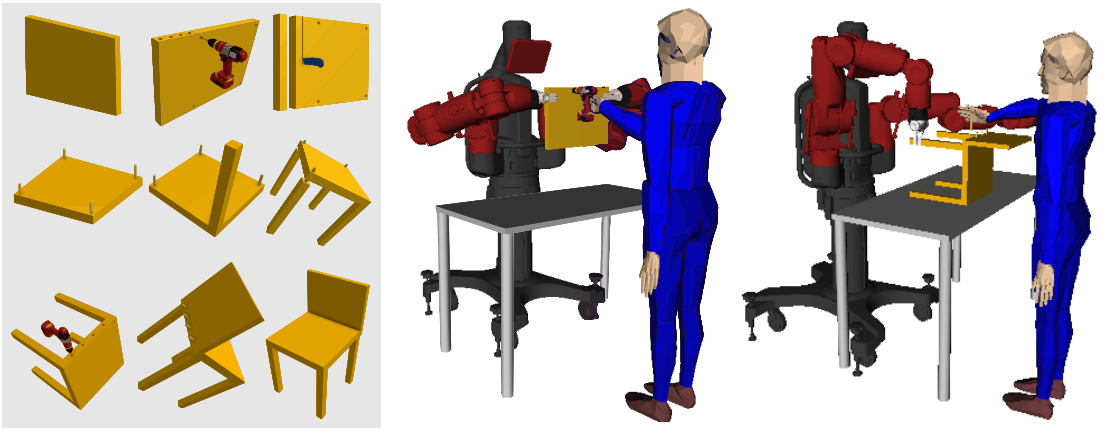


Figure 1.1: An example of forceful human-robot collaboration to fabricate a chair.

An Example of Forceful Human-Robot Collaboration: Consider the task illustrated in Fig. 1.1, where a human and a robot collaboratively fabricate a chair. During the task, the human applies a long sequence of *forceful operations*, e.g. cutting and drilling, which in essence produces a sequence of external operation forces changing position, direction and even magnitude onto the chair sub-assemblies.

Therefore, for such a forceful task, the robot is supposed to move the chair sub-assemblies not only to position them at preferred configurations meeting geometrical constraints, e.g. avoiding obstacles, but also to exploit deliberate *object contacts*, e.g. robot grasps and environmental contacts, to keep the target object stable under the application of changing external forces. Moreover, the robot might also need to move the objects to change their contacts *multiple times* due to the varying property of external forces.

Further, a forceful operation in the context of fHRC ties a robot and a human together via powerful physical contacts. Therefore, robots supporting humans in such forceful applications are expected not only to manipulate target objects to avoid humans, e.g. preventing potential injury upon contacts with humans, but also to proactively adapt their behaviours to achieve enhanced human experience and perception. Moreover, such forceful operation poses strong and lasting physical interactions between humans and robots, and therefore require a planner to take a comprehensive consideration of, in particular the human's physical limits in applying forceful operations, to plan collaborative robot behaviours allowing, for example, reduced physical efforts from humans. For example, for the drilling operation in Fig. 1.1-middle, if the robot holds the board at a higher position, the human would have to raise up his hand to reach the board, and therefore would require greater physical efforts.

Motivated by the potential of such a system, in this thesis, we investigate methods to explicitly address *changing external forces* in object grasping and manipulation for robots. We particularly focus on the context of *human-applied operation forces* onto objects held by collaborative robots. We develop a range of planners that enable robots to efficiently exploit *robot* and *environmental contacts* to manipulate objects under such changing-force constraints, while with due consideration of the *human comfort*

and *safety* during collaboration. We aim at empowering robots with the capability of assisting humans in performing elaborated and laborious forceful tasks in an efficient, stable and comfortable manner.

Such a framework leads to several key correlated questions addressed in this thesis:

Q1: Consider a collaborative forceful operation, e.g. cutting or drilling, how to model the operation force, particularly with the presence of *force uncertainties* inherent in human-applied forceful operations?

Q2: Consider a single forceful operation, can a pattern of object contacts with, e.g. the robot and/or structures in the shared environment, keep the target object stable under the application of the corresponding operation force?

Together with Q1, this question directly concerns *manipulation stability* under external disturbances, particularly in the presence of force uncertainties.

Q3: Consider a sequence of forceful operations forming a forceful task, how to choose appropriate object contacts to keep the target object stable under the changing external operation forces, particularly with enhanced *manipulation efficiency*, i.e. reducing the need of changing object contacts required throughout the task?

This question concerns robotic manipulation efficiency in sequential tasks, which also benefits human experience by reducing frequent task interruptions required for changing object contacts.

Q4: In the context of fHRC, how to adapt the robot behaviours, such that the human can perform collaborative forceful operations not only with guaranteed manipulation stability and efficiency, but also with a high level of *human comfort* and *safety*?

This question mainly concerns the modelling of human comfort in fHRC.

Q5: To further boost the collaboration smoothness, how to inform the robot of a human-desired forceful task, i.e. a sequence of forceful operations before collaboration, and how to inform the human of a manipulation plan during collaboration, e.g. when to apply a specific subsequence of operations and when to allow the robot to move the target object to change the object contacts?

This mainly concerns the automation of forceful robotic applications in the collaboration context, which in essence requires a certain level of communication between the human and robot.

1.2 Our Approach

This thesis addresses the problem of manipulation planning for forceful human-robot collaboration. Particularly, we focus on the context where a human applies a sequence of changing external forces through forceful operations (e.g. cutting a circular piece off from a board) on an object that is held by a collaborative robot.

As structured in Fig. 1.2, we devise a range of planners that: 1) address force uncertainties inherent in the human-applied forceful operations (i.e. changing external forces), and efficiently check the force

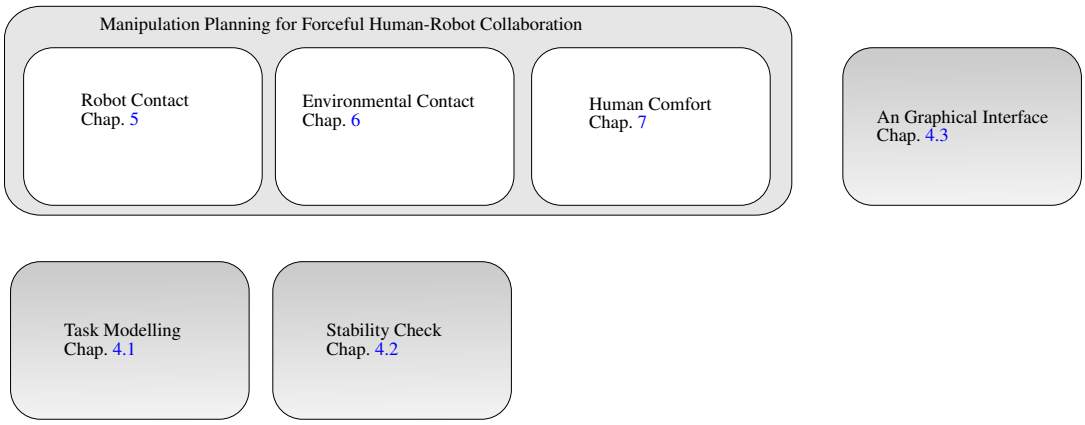


Figure 1.2: Our work towards stable, efficient and comfortable fHRC.

stability of a candidate object contact configuration under a specific forceful operation; 2) enable the robot to *manipulate an object under changing external forces*, i.e. to stabilize and position the target object under the application of a sequence of forceful operations, by exploiting supports from both environmental and robot contacts; 3) improve manipulation efficiency by minimizing the need of changing object contacts required by the sequential forceful operations in the long term; 4) improve *human comfort* and *safety* in applying the forceful operations concerning the human's physical efforts and spatial perception while collaborating with the robot in close proximity.

1.2.1 Forceful Operations

Task Modelling: The primary step towards manipulation planning for fHRC is to model the collaborative forceful task, which consists of a sequence of forceful operations. In the context of fHRC discussed in the thesis, a force operation is applied by a human with a specific tool/object, therefore force deviations are inevitably inherent in the operation force. Such force uncertainties pose substantial challenges for object manipulation due to the difficulties in finding robustly stable contact configurations against the operation force.

We introduce a data-driven conic representation to strictly model the distribution of operation force with the presence of deviations, and a conservative pyramid approximation of the conic distribution, relying on which we greatly simplify the process of check stable contact configurations (which we refer to as *stability check*) but with improved robustness (Chap. 4.1).

Stability Check: Task stability is an essential criterion for effective object manipulation, which requires a planner to find appropriate object contacts capable of providing sufficient supporting wrenches to stabilize the target object against external disturbances. We model the stability check of a candidate contact configuration under a certain forceful operation as a linear programming (LP) problem in the context of using only robot grasps for object manipulation, and as a constrained optimization problem

in the context of using both robot and environmental contacts. These formulations are integrated into our planners for a preliminary check to guarantee manipulation stability before planning (Chap. 4.2).

A Graphical User Interface: As an additional contribution, we develop a graphical interface for the planning work, which acts as a communication channel between human users and collaborative robots. In brief, a human user first employs the interface to specify and initiate a forceful task, i.e. a sequence of forceful operations. The interface then takes the human customized task and produces a manipulation plan with our proposed planners. According to the plan, the connecting robot uses the interface (together with a head monitor in our experiment setting) to instruct and collaborate with the human to perform the task (Chap. 4.3).

1.2.2 Manipulation Planning under Changing External Forces

We propose to exploit object contacts with both environment and robot for object grasping and manipulation under changing external forces.

Robot Grasps: We start from the scenario of using only object contacts with robots, particularly robot grasps, where a robot grasps and regrasps an object multiple times to keep it stable under changing external forces. In this context, we introduce a planner capable of generating a efficient manipulation plan by intelligently deciding when the robot should change its grasp on the object, i.e. regrasp, as the human applies changing operation forces on the object, and choosing subsequent grasps such that they minimize the number of regrasps required in the long-term. A regrasp means a task interruption, and therefore the number of regrasps directly affects manipulation efficiency. Our planner searches for such an efficient plan by first finding a minimal sequence of grasp configurations that are able to position and keep the object stable under the changing external forces, i.e. *minimizing regrasps*, and then generating connecting trajectories to switch between these planned grasp configurations, i.e. *planning regrasps*. We perform the search for such a grasp sequence by first sampling a set of stable configurations for each external force, building an *operation graph* using these stable configurations (which is similar to the manipulation graph (Alami *et al.*, 1990, 1994; Nielsen & Kavraki, 2000) widely used in general manipulation planners), and then searching the operation graph to minimize the number of required regrasps.

As an additional capability, our planner enables robots to regrasp objects *in the air*, by solving the problem of bimanual regrasping under the assumption of no supporting surfaces for intermediate placements. Briefly, this is achieved by finding *intermediate configurations* at which both bimanual and unimanual grasps can hold a target object stable under external forces, e.g. gravity (Chap. 5), and therefore allow the robot to release/regrasp the required griper from/on the object.

Environmental Contacts: We then explore a more general scenario, where the robot can use not only grasp contacts, but also object contacts with structures in the shared environment, as well as the object contacts with other robot bodies, for object manipulation under changing external forces. The exploita-

tion of general object contacts, particularly the environmental contacts, dramatically enlarges the set of candidate configurations stable against external disturbances. Even though it boosts the robot's capability of stabilizing objects under external forces, exploiting environmental contacts could impair the planning efficiency due to the explosion of available contact configurations and the computational characteristic of stability check for environmental contacts.

We present a planner that exploits the environment's and robot's stabilization capabilities within a unified planning framework, to search for an efficient manipulation plan, i.e. a minimal sequence of contact configurations and connecting motion trajectories that are able to keep an object stable under changing external forces. Further, we introduce a concept of *containment* relationship among different contact configurations pruning the redundant stability checks involved in searching for the manipulation plan, which guarantees enhanced planning efficiency in using environmental contacts (Chap. 6).

1.2.3 Human Comfort

We present a planning algorithm to further improve the collaborative robot behaviour concerning human comfort and safety in fHRC. Particularly, we are interested in empowering robots with the capability of grasping and positioning objects not only to ensure object stability against human-applied operation forces, but also to improve human comfort and safety during forceful interaction. We propose to quantify the human comfort in a collaborative forceful operation as the human's *muscular efforts* required to apply the end-effector operation force, together with the human's spatial perception regarding the human's so-called *peripersonal space* while in close proximity with the robot. The planner maximizes both comfort metrics in configuring the robot to grasp and position the target object such that the human can apply the forceful operation with enhanced comfort and safety perception, but without compromising force stability (Chap. 7).

1.3 Publication Note

The majority of Chap. 4 was published in Chen *et al.* (2018b, 2019). The majority of Chap. 5 was published in Chen *et al.* (2018b). The majority of Chap. 6 was published in Chen *et al.* (2019). The majority of Chap. 7 was published in Chen *et al.* (2018a).

Chapter 2

Literature Review

This work addresses manipulation planning for forceful human-robot collaboration. We are particularly concerned about using environmental contacts and robots contacts, such as robot grasps, to keep manipulated objects stable under changing external forces concerning manipulation stability and efficiency, which is mainly relevant to *grasp analysis and cooperative manipulation*, *multi-step manipulation planning and regrasping*. In addition, we are also concerned about the robot adaptability of improving *human comfort and safety* in forceful collaborative tasks.

2.1 Grasp Analysis and Cooperative Manipulation

The literature of *grasp analysis* investigates the question of how stable a grasp is. General methods in this body of research rely on two fundamental quality criteria *force closure* and *form closure* (Lakshminarayana, 1978; Reuleaux, 1875) to answer whether a grasp is able to completely or partially constrain motions of a manipulated object or to apply arbitrary contact wrenches on the object.

Briefly, under the assumption of rigid bodies, a form closure grasp relies on *frictionless contacts* to restrain motions of the grasped object:

$$\left.\begin{array}{l}\mathbf{G}\mathbf{h} = -\mathbf{h}_{\mathrm{e}} \\ \mathbf{h} \geq \mathbf{0}\end{array}\right\} \forall \mathbf{h}_{\mathrm{e}} \in \mathbb{R}^6 \tag{2.1}$$

where $\mathbf{h}$ is a vector of contact forces. $\mathbf{h}_{\mathrm{e}}$ is an arbitrary external disturbance applied onto the target object. $\mathbf{G}$ is a matrix mapping grasp forces/torques into a resultant wrench (i.e. *grasp wrench*) on the object and usually termed as the *grasp matrix* (Borst *et al.*, 2004; Ferrari & Canny, 1992; Mishra *et al.*, 1987).

Force closure is similar to form closure, but allows friction forces to help balance the external wrench:

$$\left.\begin{array}{l}\mathbf{G}\mathbf{h} = -\mathbf{h}_{\mathrm{e}} \\ \mathbf{h} \in \mathcal{F}\end{array}\right\} \forall \mathbf{h}_{\mathrm{e}} \in \mathbb{R}^6 \tag{2.2}$$

where $\mathcal{F}$ is the composite friction cone defined by $\mathcal{F} = \mathcal{F}_1 \times, \ldots, \times \mathcal{F}_{n_c}$, and $\mathcal{F}_i$ is the friction cone corresponding to the i-th contact point defined by, for example, the Coulomb's friction cone. n_c is the number of contact points. Eq. 2.2 can be interpreted physically that the robot hand can squeeze arbitrarily tightly to compensate for large external wrenches that are resistible by the corresponding frictional forces.

The primary difference between form closure and force closure grasps lies in the existence of *contact friction*, which in accordance allows fewer contacts to achieve force closure than form closure. For example, as proved in (Lakshiminarayana, 1978), a form closure grasp of a 3D object with six degrees of freedom requires at least seven different contacts (Mishra *et al.*, 1987), while to force close a grasp, at best, only three non-collinear distributed contacts are sufficient for *hard fingers* and two for *soft fingers*. In this regard, a form closure grasp is also force closure: if under form closure, the object is completely restrained from motions relative to the robot hand regardless of external disturbances. That is, the grasp contacts can keep the object in equilibrium under any external wrench, which is the exact force closure requirement.

Two important variables are frequently mentioned in grasp analysis in regard to the above closure properties: *grasp matrix* (denoted as $\mathbf{G}$ in Eq. 2.1 and Eq. 2.2) and *hand Jacobian* (usually denoted as $\mathbf{J}$) in the context of multi-fingered robot hand. In this thesis, we use parallel plate grippers for object grasping and manipulation. Therefore, we mainly focus on the former in our discussion and address the Jacobian matrix $\mathbf{J}$ later in the discussion of cooperative manipulation. The grasp matrix $\mathbf{G}$ maps the contact forces from robot hands to a resultant grasp wrench acting on the target object, and therefore usually acts as a handy tool for closure test and grasp classification (Bicchi, 1995; Murray, 2017). For example, for the form closure of a grasp of a 3D object, i.e. in order to meet Eq. 2.1, the rank of $\mathbf{G}$ must be six (the *full row rank condition*).

The *grasp wrench space* (GWS), i.e. the set of all external wrenches that can be resisted by a grasp, has also been widely used as a task-independent metric to measure the general quality of a grasp configuration (Borst *et al.*, 1999, 2004; Ferrari & Canny, 1992; Mishra *et al.*, 1987). For example, Ferrari & Canny (1992) proposed to approximate the GWS with the convex hull over the Minkowski sum of the friction cones. Kirkpatrick *et al.* (1992) used the largest wrench sphere centred at the origin that just fits within the GWS as a task-independent quality measure.

The *task-oriented grasping* literature explores the problem of grasping an object for a specific task (Bekiroglu *et al.*, 2011; El-Khoury *et al.*, 2015; Han *et al.*, 2000; Nikandrova & Kyrki, 2015; Trinkle, 1992). An important problem in this body of work is to model the task-oriented external wrench expected on the target object during a task. For example, Li & Sastry (1988) introduced the *Task Wrench Space* (TWS), i.e. a set of external wrenches to be resisted for a specific task, as a metric to measure how good a grasp is under the task-relevant external wrenches. The TWS was modelled as a six-dimensional *task ellipsoid* that just fits in the GWS. The issue with this approach is how to compute the task-oriented ellipsoid, which is complicated as stated by the authors. In addition to the task-based stability constraint, Dang & Allen (2012, 2014) proposed an example-based planning framework to generate so-called semantic grasps that are functionally suitable for specific object manipulation tasks.

The proposed planner treated the *semantic constraints* embedded into object geometry, tactile contacts and hand kinematics, e.g. relative hand orientation, as an additional planning criterion in searching for stable grasps. Specifically, the planner relies on a *semantic accordance map* to obtain the ideal hand approach direction, and predefined example grasps to obtain the ideal hand kinematics and tactile contact formation. In (Bohg *et al.*, 2012), the authors presented a grasping pipeline by integrating individual functional modules performing scene exploration through gaze shifts, segmentation, object categorization and task-based grasp selection. The pipeline allows the robot to transfer task-specific grasp knowledge between objects of the same category. Other work in this area mainly focuses on producing more efficient and robust quality metrics for task-oriented grasp synthesis (Borst *et al.*, 2004; Haschke *et al.*, 2005; Lin & Sun, 2016), which mainly concerns generating grasps of certain task-oriented properties.

Furthermore, for the case of unknown task specification, a widely accepted method is to model the unknown TWS as a unit sphere in the wrench space, under the assumption that external disturbances can happen uniformly along any direction in the wrench space. A more physically interpretable model is the *Object Wrench Space* (OWS) introduced by Pollard (1994), which treats the unknown TWS as a normalized distribution of disturbance forces acting anywhere on the geometric surface of the object. Huebner *et al.* (2009) proposed a grasping strategy for objects with known geometry, based on an offline *box-based grasp generation technique* on 3D shape representations (Geidenstam *et al.*, 2009). Borst *et al.* (2004) integrated the idea of the task ellipsoid (Li & Sastry, 1988) with OWS, approximating it with a smallest 6D enclosing ellipsoid, which was referred to as the *OWS approximation ellipsoid*. The ellipsoid approximation was then scaled to just fit in the GWS, so as to find the largest scaling factor as a quantitative measure of grasp quality.

Grasp analysis also concerns the properties of object contacts, particularly contact modelling (Salisbury & Roth, 1983), which plays a fundamental role in object grasping and manipulation. Three contact types of greatest interest in grasp analysis are commonly applied as *frictionless point contact*, *hard-finger contact* and *soft-finger contact* (Salisbury & Roth, 1983; Siciliano & Khatib, 2016). In this thesis, we adopt the model of *hard finger* in grasp analysis, while at each contact point, the contact force is constrained by a Coulomb's friction cone.

In general, given an external wrench, a set of contact points, and contact-models (i.e., the corresponding friction models, which provide constraints on the directions and magnitudes of the contact wrenches that can be applied at the contacts), the question of whether the set of contacts would be able to balance a known external wrench, which is usually referred to as the *force feasibility problem*, can be formulated as a linear matrix inequality problem (Han *et al.*, 2000), which can be further simplified as a linear programming (LP) problem by approximating quadratic friction cones with polyhedral cones.

Cooperative Manipulation: The grasp analysis literature focuses on the stability of a grasp under external disturbances, while our work is also related to *cooperative manipulation*, which addresses the problem of multiple manipulators cooperatively manipulating a common object. Cooperative manipulation, particularly of large and heavy objects, requires multiple agents to coordinate to hold an object such that a closed kinematic chain is formed with enhanced capacity of object manipulating and task precision (Alford & Belyeu, 1984; Zheng & Luh, 1989). This body of research can be traced back to

the early work by Fujii & Kurono (1975); Nakano (1974); Takase (1974), which addressed the control of multiple coordinated manipulators. Later on, extensive research efforts have been devoted onto important topics in cooperative manipulation such as kinematics, dynamics (McClamroch, 1986; Unseren, 1991; Walker *et al.*, 1989), control (Chen & Luh, 1994; Yoshikawa & Zheng, 1993), *load distribution and sharing* (Lee *et al.*, 2005; Zheng & Luh, 1989).

This thesis directly concerns the *kinematics and statics* of a two-manipulator system, which has been studied by Bonitz & Hsia (1994); Uchiyama & Dauchez (1988, 1992); Walker *et al.* (1991). Particularly, we focus on checking the load distribution at the grasp points and arm joints of two manipulators holding a common object under a certain external disturbance. In this regard, Uchiyama & Dauchez (1988, 1992) proposed the *symmetric formulation* modelling the kinematic and static relationships between the external wrench applied onto the object and their counterparts distributed over the cooperative manipulators, particularly at the grasp points:

$$\mathbf{W}\mathbf{h} = \mathbf{h}_{\mathrm{e}} \tag{2.3}$$

where $\mathbf{W} = (\mathbf{W}_1, \mathbf{W}_2)$ is the grasp matrix describing the grasp geometry (which is denoted as $\mathbf{G}$ in previous section), and $\mathbf{h} = \left(\mathbf{h}_1^{\mathsf{T}}, \mathbf{h}_2^{\mathsf{T}}\right)^{\mathsf{T}}$ is the vector of grasp wrench acting at the grasp points of end-effectors. Therefore, based on Eq. 2.3, to exert a resultant wrench on the target object, one can solve a set of linear equations to find the force/torque efforts required at the grasp points.

It is also worth mentioning that cooperative manipulation can also learn from multi-fingered grasping, which shares a conceptually similar pattern of multiple coordinated structures grasping and manipulating a common object but differs in relying on contact points for object manipulation compared with the grasp points used by cooperative manipulation. In particular, an important and interesting work by Chiacchio *et al.* (1996) proposed to integrate both cooperative manipulation and multi-fingered grasping together into a formulation for direct kinematics of a two-manipulator system. Encompassed by the formulation, *non-tight grasps*, particularly the rolling and sliding motion of grasp contacts on the object, are modelled as *virtual prismatic and rotational joints* in line with the number of degrees of freedom, and then added to the direct kinematics equations as virtual joint variables. The preferred manipulator/finger joints trajectories are then derived from the object motion with numerical inverse kinematics algorithms. Similarly, in this thesis, we model *grasp contacts* as a group of virtual prismatic and rotational joints and check their force/torque limits in stability analysis.

Another inspiring work from Hunawar & Uchiyama (1997); Munawar & Uchiyama (1999) addressed the robust holding of an object in the presence of end-effector slippage on the object. The proposed work focuses on the *contact slip* problem specifically, presenting a geometrical strategy based on the coordination of multiple manipulators/fingers to detect the occurrence of end-effector lip and compensation of its effects without employment of additional sensors but the information of finger-tip position only.

We build on the formulations of grasp stability in Eq. 2.2 and cooperative manipulation in Eq. 2.3 to propose our grasp stability check (Sec. 4.2), which also integrates force/torque limits at both grasp

contacts and joints of cooperative manipulators.

2.2 Multi-Step Manipulation Planning and Regrasping

Manipulation planning addresses the motion planning problem to manipulate moveable objects among obstacles. In a typical *multi-step manipulation planning* problem, the target object is either transported by a robot or lies at some stable placement, while the robot grasps and regrasps the object multiple times through geometrical obstacles.

The need for regrasping objects was recognized even in the earliest manipulation systems (Lozano-Pérez *et al.*, 1987; Tournassoud *et al.*, 1987). The work by Lozano-Pérez *et al.* (1987) introduced a robot system capable of manipulating objects in an unstructured and cluttered environment. The integrated grasping module can choose grasps on the object that are both stable and accessible by the robot, and plan regrasping motions for illegal grasps due to the presence of geometrical obstacles in the cluttered environment. Tournassoud *et al.* (1987) explicitly defined the regrasping problem as constructing a sequence of ungrasping and grasping operations to connect an initial grasp and a final grasp. The proposed planner employed a table surface to place the object at intermediate positions for regrasping. Following the idea of using stable placements as intermediate configurations for regrasping, a large number of planners in this line of research have been put forward towards more efficient and adaptive regrasping, which we will review later in this section.

Later, Siméon *et al.* (2004) presented a sampling-based planner solving the manipulation problem with an alternating sequence of *transfer* and *transit* sub-paths. Specifically, robot motions of holding the target object at a fixed grasp are called "*transfer paths*", and robot motions are called "*transit paths*" while the object stays at a stable placement. Based on a so-called *reduction property* (Alami *et al.*, 1994), the proposed planner builds a *manipulation graph* (Alami *et al.*, 1990, 1994; Nielsen & Kavraki, 2000), i.e. a probabilistic roadmap, to capture connectivities of the foliated manifolds in the composite configuration space, and then searches the graph for feasible paths once receiving manipulation queries.

The graphical formulation of manipulation planning problem, i.e. the manipulation graph, has been widely utilized later and further developed in various contexts (Chang *et al.*, 2010; Dogar & Srinivasa, 2011; Geraerts & Overmars, 2006; Lozano-Pérez & Kaelbling, 2014). For example, Harada *et al.* (2012, 2014) built a similar graphical representation accounting for the special topology of the *manipulation space* for *dual-arm manipulation*, which is structured into four foliated manifolds. The proposed planner decides if both hands are used simultaneously or not according to the context, whereas the objective is to minimize the number of regrasping. Herein, we also explicitly address the problem of *regrasp minimization* as an objective in manipulation planning, which we will discuss later in Chap. 5.

More recently, planners have been proposed for the manipulation problem in the context of multiple manipulators for assembly-like tasks (Dogar *et al.*, 2019; Sierla *et al.*, 2018; Wan & Harada, 2016; Wan *et al.*, 2018). The work by Dogar *et al.* (2019) addressed the problem of finding multi-robot configurations to grasp assembly parts for a sequence of collaborative assembly operations. The problem was formulated as a constraint satisfaction problem (CSP), with the so-called *collision constraints* within

an operation and *transfer constraints* between operations. The proposed planner can quickly find a *first-feasible* solution by first dividing the CSP into independent smaller problems under the assumption of feasible regrasps, and then use local search techniques to improve this solution by removing a gradually increasing number of regrasps from it. In addition to improved planning efficiency, the divide strategy also enables real-time planning as a trade-off of more regrasps required in the plan.

Most existing work on manipulation planning focuses on dealing with geometrical constraints, and generates *collision-free* robot motions to manipulate movable objects. Our planner goes beyond geometrical constraints, taking into account the sequential-force constraint, which can be required in a large variety of sequential manipulation tasks, such as assembly. In our task, for example, a robot is required not only to move a target object to expected pose(s) under geometrical constraints, but also to keep the object stable under changing external forces.

There are some recent works explicitly addressing force constraints in multi-step manipulation tasks. Lipton *et al.* (2017, 2018) presented a system for multi-robot assisted carpentry. The system controls a team of mobile robots to fabricate human-customised parts with standard carpentry tools, e.g. a saw, and assumes two specialized powerful stands to stabilize lumber against fabricating forces. In another recent work by Moriyama *et al.* (2019), a sampling-based assembly planner was proposed to generate stable assembly poses under the gravitational constraint. Our work differs in the existence of changing external forces applied on the target object manipulated by a multi-arm robot. Inspired by our work (Chen *et al.*, 2018b), Holladay *et al.* (2019) studied a different planning problem in the context of robots using hand tools. The authors developed a system allowing robots to reason about force and motion constraints in order to complete complex tool-using tasks like wielding a screwdriver. From the perspective of control, Lin *et al.* (2018) proposed a model-based control framework for multi-arm manipulation of a rigid object subject to external disturbances.

Our problem can also be interpreted as an instance of *multi-modal manipulation planning* (Bretl, 2006; Hauser & Latombe, 2010; Lee *et al.*, 2015), where each modality corresponds to a bimanual or unimanual grasp. In developing planners, we follow a similar strategy of first identifying intersections among different modalities/manifolds, and then planning motion trajectories to connect these intersections (Chap. 5).

Regrasping: As stated previously, this work is related to regrasp planning, particularly the case of dual-arm or multi-arm regrasping. Regrasp planning involves finding a connected path over a sequence of sub-manifolds in the composite configuration space. Roughly, the basic flow of regrasp planning follows the pattern of first building a graphical representation of the foliated configuration space, e.g. the manipulation graph (Alami *et al.*, 1990, 1994; Nielsen & Kavraki, 2000), and then searching the graph for available regrasp sequences. Early studies (Rohrdanz & Wahl, 1997; Stoeter *et al.*, 1999) employed a *grasp-placement* table to generate a sequence of pick-and-place operations for regrasping. More recent studies propose other graph-based representations, such as the *regrasp graph* (Wan & Harada, 2016, 2017) as an offline hierarchical graph of feasible grasps and their corresponding stable placements, and *certificate* (Lertkultanon & Pham, 2018), which was defined as a set of transfer paths that spans all the placement classes.

Most existing algorithms on regrasp planning rely on stable placements on extra supports as intermediate configurations for regrasping, such as support surfaces (Chavan-Dafie & Rodriguez, 2018; Wan & Harada, 2016, 2017) and other complex structures (Cao *et al.*, 2016; Ma *et al.*, 2018). For example, rather than flat surfaces, Cao *et al.* (2016) proposed to use a vertical pin on the working surface as a support for pick-and-place regrasp. Experiment results on reorientation and assembly tasks verified that using support pins can benefit regrasping.

Different from the existing regrasping work mentioned above, our work particularly focuses on a different scenario where the robot can not place an object down on an extra support surface, but only use its manipulators to cooperatively regrasp the object under external forces, e.g. gravity (Chap. 5).

Manipulation with Environmental Constraints: The idea of exploiting environmental constraints or contacts can be traced back to the influential work by Erdmann & Mason (1988); Lozano-Perez *et al.* (1984); Mason (1985), where the task mechanics were utilized to eliminate object uncertainty and to improve robustness in performing sensor-less manipulation. The evidence of complex interaction between multi-step object manipulation and a shared environment can also be found in the earliest approaches (Lozano-Pérez *et al.*, 1987; Tournassoud *et al.*, 1987) for automatic generation of robot motion sequences to manipulate movable objects in a cluttered environment. This line of research, however, usually simply addresses the environment as a constraint, in particular as obstacles to avoid, concerning generating collision-free robot motions allowing manipulating objects through structures in the shared environment.

Instead, many recent advances in robotics focus on exploiting deliberate interactions with the shared environment to achieve increased robustness and enhanced capabilities of robotic systems, particularly for object grasping (Babin & Gosselin, 2018; Eppner *et al.*, 2015) and manipulation (Chavan-Dafie & Rodriguez, 2018; Jorda *et al.*, 2019; Lee *et al.*, 2015). For example, rather than regarding a cutter in a manipulation task as a constraint to avoid, Dogar & Srinivasa (2011) presented a planning framework that enables robots to actively intact with and rearrange the clutter around the target objects in manipulation tasks, using a library of actions inspired by human strategies, such as *push-grasping* (Dogar & Srinivasa, 2010). Eppner et al. (Eppner & Brock, 2015; Eppner *et al.*, 2015) proposed to increase robustness in robotic grasping through the deliberate exploitation of environmental contacts. The proposed planner produced a grasp of an object by generating a sequence of *environmental constraint exploitations*, i.e. consecutive motions constrained by features in the environment. Promising results can be seen in the paper demonstrating the effectiveness and generality of the planner in a set of real-robot experiments with a variety of environments. A multi-step planning problem similar to our work was studied by Bretl (2006), which produces contact modes for a spider robot climbing a wall subject to equilibrium constraints. The paper presented a multi-step planning framework of first searching in the intersections of manifolds for a candidate sequence of hand and foot placements and searching in these manifolds to refine the obtained sequence into a feasible, continuous trajectory by finding paths between subsequent transitions. Particularly, the former search (generating a candidate sequence of hand and foot placements) guides the latter (generating continuous motions to reach them) by restraining the search into a limited number of manifolds. In this thesis, we exploit a similar hierarchical planning framework,

but focus on the problem where there are changing external forces applied onto the manipulated objects.

Regrasp planning in the context of using object placements on additional support structures can also be interpreted as an instance of using environmental constraints (Cao *et al.*, 2016; Chavan-Dafle & Rodriguez, 2018; Ma *et al.*, 2018; Wan & Harada, 2017; Wan *et al.*, 2015). An interesting work by Chavan-Dafle & Rodriguez (2018) presents a *fixtureless fixturing* strategy for regrasping, which is based on the idea that to regrasp an object in a gripper, a robot can push the object against an external contact in the environment such that the external contact keeps the object stationary while the fingers slide over the object. Another recent work by Ma *et al.* (2018) proposed a planner for regrasp planning using stable object placements supported by complex structures, which have a high variety of contact elements providing not only flat surface supports but also point supports, line supports, and a combination of them. The proposed planner relies on a dynamic simulator to compute immediate object stable placements on such given supporting structures.

The main body of this research can be explained by a simple insight: rather than a constraint, the environment can be an opportunity if used properly. In this work, we draw inspirations from this insight, using object-environment contacts as additional supports to robot contacts in manipulating and stabilizing objects under changing external forces (Chap. 6).

2.3 Human Comfort in fHRC

Thanks to the prominent advances achieved in areas like mechatronics, control, learning and planning, physical human-robot collaboration (pHRC) has become a central topic in robotics since the last decades, leading towards a rapid rise of more skilled and specialized robots venturing into human life, such as the Baxter robot from Rethink Robotics (which we have been using mostly in this thesis), Atlas from Boston Dynamics and the humanoid robot ARMAR from KIT (Asfour *et al.*, 2013, 2018).

pHRC has been interpreted in the literature from various perspectives. The major body of these research in *human-robot shared manipulation* focuses on **controlling**. In particular, these works usually assume the manipulated objects to be already stably grasped, solving for necessary stiffness of robot manipulator joints with external disturbances applied onto the objects (Abi-Farraj *et al.*, 2017; Rozo *et al.*, 2016). The early work by Kosuge & Kazamura (1997) presented and compared several control methods, like *damping control* and *impedance control*, for the motion generation problem of a robot supporting and moving an object in cooperation with a human under the effect of external disturbances such as friction force. In another work (Kosuge *et al.*, 1998), the authors proposed a controller for a multiple-robot system assisting humans in *flexible object handling*. The robot system played two roles of supporting and deforming a flexible object by adjusting internal forces onto the object, so that the human could easily handle the object by only applying intentional forces. Most control schemes in this line of research fall into some form of *impedance* and the related *admittance* control (Hogan, 1985a,b). For example, in Ikeura *et al.* (2002); Rahman *et al.* (2002), *human characteristics*, particularly the impedance characteristics of the human arm, were explicitly investigated and employed into the controlling of a robot cooperating with a human in collaborative tasks like object lifting. Kosuge *et al.*

(2000) applied a similar method but with *fixed virtual impedance* to control the horizontal movement of a mobile robot in response to human-applied intentional forces on a cooperatively carried load.

The **planning work** in this line of research mostly focuses on *handover* (Cakmak *et al.*, 2011; Peternel *et al.*, 2017; Sisbot & Alami, 2012; Strabala *et al.*, 2013b), *object carrying* (Rozo *et al.*, 2016; Solanes *et al.*, 2018), or other joint actions where robots are expected to avoid colliding humans in a shared workspace (Luo *et al.*, 2018; Maeda *et al.*, 2017).

Handover: Handover resembles our collaborative forceful tasks discussed in this thesis, in requiring a robot to position an object at acceptable, and in particular comfortable configurations for a human and to adapt its behaviours, e.g. to avoid harming the human, to move the object to reach the preferred object configurations. In Sisbot & Alami (2012), the authors prosed a three-stage manipulation planner to select such object transfer configurations (so-called *object transfer point* (OTP)), object path and robot motions for robot-human handovers, by optimizing devised quality metrics (as cost functions) as *safety*, *visibility* and *accessibility* in line with the humans kinematics, vision field and arm comfort. Parastegari *et al.* (2017) drew an *ergonomic* model by learning from human-human handovers to characterize human preferred handover configurations and as a bonus to generate human-like *legible* reaching motions for robots.

This line of planning work mostly takes some form of kinematic optimization on task-oriented criteria like *smoothness*, *visibility* and *legibility* (Aleotti *et al.*, 2012; Cakmak *et al.*, 2011; Mainprice *et al.*, 2010; Parastegari *et al.*, 2017; Strabala *et al.*, 2013a) in handovers. However, the physical contacts in a handover are usually momentary and trivial, without strong interactions among the robot and human, therefore most relevant work focuses on geometric preferences, like legibility. Our forceful tasks differ in the existence of intensive and persistent forceful interactions among the robot, object and human, particularly in the context that the human applies strong forceful operations in close proximity of the robot, thus planning for fHRC has different priorities, e.g. the human's muscular stress and safety.

Carrying: Similar to handovers, collaborative *transport*, *lifting* or *carrying*, particularly of bulky, heavy or deformable objects, is a common joint action in the community of pHRC research. A collaborative carrying task connects a human and a robot together via a common manipulated object but for a longer duration. Most existing planning work in this line focuses on the problem of *load sharing* among robot and human partners. In Lawitzky *et al.* (2010), the authors introduced three *effort sharing policies* for human-robot cooperation from a system-theoretic analysis of geometric and dynamic task properties. The effort sharing policies were parametrized with respect to the degree of assistance and therefore the physical load can be dynamically shared depending on the task state and the assistance degree. These policies were embedded into a two-level hierarchical architecture for motion planning and control in joint manipulation under *environmental constraints*. In a recent work (Muthusamy *et al.*, 2018; Tariq *et al.*, 2018), the authors highlighted the importance of relative locations of *grasps* in collaborative manipulation load sharing. This work proposed a grasp analysis approach for collaborative manipulation of unknown objects that allows optimal load sharing by minimizing exerted grasp wrenches in a task-oriented manner. This body of research, in particular the problem of load sharing, is similar to our

problem of finding appropriate grasps and load distributions over robot manipulators to keep manipulated objects stable under external disturbances.

Other joint actions: There is also some other planning work on collaborative manipulation, which mainly focuses on the handling of collisions between robots and humans in joint actions to limit possible human injury due to physical contacts (De Luca *et al.*, 2006; Suita *et al.*, 1995; Yamada *et al.*, 1997). In Mainprice & Berenson (2013), the authors proposed a framework allowing a human and a robot to perform simultaneous manipulation tasks safely in close proximity based on the early prediction of human motion. The prediction was modelled as the *human workspace occupancy* by computing the swept volume of learned human motion trajectories, with robot trajectories minimizing a *penetration cost* in the human workspace occupancy returned by motion planning. In devising our planners capable of dealing with spatial proximity, we address the human workspace occupancy in a collaborative forceful interaction by minimizing the robot's penetration in the human's peripersonal space (Chap. 7).

Human Safety: Human safety is another relevant concern in planning comfortable fHRC and a central topic in pHRC due to the potential human injury and threats originating from physical contacts and spatial proximity. Human comfort and safety have been well explored in planning, for example, for object handover and transport (Parastegari *et al.*, 2017; Sisbot & Alami, 2012; Solanes *et al.*, 2018). With regard to forceful collaborative applications, Mansfeld *et al.* (2018) extended the safety analysis to general HRI and proposed a new concept for global safety assessment in practical HRC applications, namely *safety map*. In (Shingarey *et al.*, 2019), the authors integrated *velocity* and *torque* control into one joint-level control mode and proposed a so-called *torque-based velocity controller* (TBVC). By taking merits from both sides, the TBVC could provide the ease-of-use of joint-level velocity control for trajectory tracking combined with the safe interaction characteristics of compliant torque control when in contact with humans and environment.

In addition to human safety, some other physical or psychological human natures demonstrated in collaborative manipulation tasks have also been explored to address human comfort in pHRC, such as *fatigue* and *ergonomics* (Busch *et al.*, 2018). In a recent work by Peternel *et al.* (2016), the authors proposed to adapt and modulate the robot's physical behaviour and assistance degree in human-robot co-manipulation as a function of the *human motor fatigue*, which was estimated using an online model based on observed *muscle activity* measured by the *electromyography* (EMG). In an extension work of this thesis, we also collected EMG data to verify our proposed muscular comfort metric. Another recent work (Peternel *et al.*, 2017) proposed to address human ergonomics, particularly the *overloading joint torques*, i.e. the torque induced into human joints by an external load, in human-robot co-manipulation through an online optimisation process for comfortable and ergonomic human body poses in forceful interactions and handover. Despite the fact that joint torques were taken into account through an offline identification technique, this work still lacks proper knowledge on muscle activity—as well as safety perception and the robot grasp stability.

Human Intention and Preferences: Planning for human-friendly robot motion and collaborative be-

haviours can also benefit from understanding *human intention* and *preference* (Huber *et al.*, 2008; Li & Ge, 2013; Prez-D'Arpino & Shah, 2015; Sakita *et al.*, 2004; Welke *et al.*, 2013). In Duchaine & Gosselin (2007), the authors presented a method enhancing the transparency of human-robot movements based on an online variable impedance control, which used *force differentiation* as a prediction of human intention. This work also reported that compared with typical position control, velocity control fits better in the controlling of a human-friendly robot. Thobbi *et al.* (2011) proposed a control framework that enables a robot to switch its role autonomously and dynamically between leading and following during a cooperative manipulation task. The framework governs robot behaviours with *reactive* and *proactive* controllers each giving the robot follower and leader characteristics respectively. This work was built on the idea that the robot's role in a human-robot collaborative task can be determined by its *confidence* in predictions of the humans intentions. In another study, Cakmak *et al.* (2011) exploited *human preferences* in selecting preferable robot-human handover configurations. Direct human preferences were collected by human subjects evaluating different handover configurations simulated for a HERB robot, which the author compared against planned configurations using a kinematic model of the human. It was stated that learned configurations were preferred, while planned configurations provided better object reachability.

Despite the advances and contributions aforementioned, pHRC still lacks effective metrics for evaluating and ensuring the human comfort in planning for forceful human-robot co-manipulation, particularly in terms of the human's physical efforts and the comfort perception with close proximity to robots in forceful interactions. In this thesis, we explicitly consider the human's physical efforts in applying forceful operations, and propose a muscular comfort metric to quantify the human's muscular efforts based on the observation of muscle activations. Learning from the concept of *peripersonal space*, we introduce a distance-based comfort metric to evaluate the human spatial comfort in applying forceful operations in the context of human-robot collaborative manipulation. By integrating both metrics in a constrained (i.e. grasp stability) optimization process, our planner can instruct a robot to grasp and position an object for a human-applied forceful operation at configurations which are not only stable against the operation force, but also comfortable for the human operator (Chap. 7).

Chapter 3

Terminology, Notations and Problem Fundamentals

This chapter presents general terminology, notations and problem fundamentals, particularly of the forceful tasks and human-robot collaborative system. Throughout this work, we assume rigid bodies and quasi-static movements.

3.1 Forceful Operations and Tasks

As illustrated in Fig. 3.1, we refer to a complete forceful interaction as a *forceful task*, and each forceful task comprises a continuous and/or discrete sequence of *forceful operations*.

For example, the circular cutting task in Fig. 3.1 consists of a continuous sequence of cutting operations tangential to a desired circle pattern on a rectangular board. Similarly, another example illustrated in Fig. 3.2, the chair fabricating task on a raw wooden board (Fig. 3.2(a)), involves a discrete subsequence of drilling operations for eight holes on the board (Fig. 3.2(b)), a continuous subsequence of cutting operations for four table legs (Fig. 3.2(c)), and a discrete subsequence of four inserting operations (Fig. 3.2(d)) for four fixture pegs and four table legs. In the context of fHRC, human operators

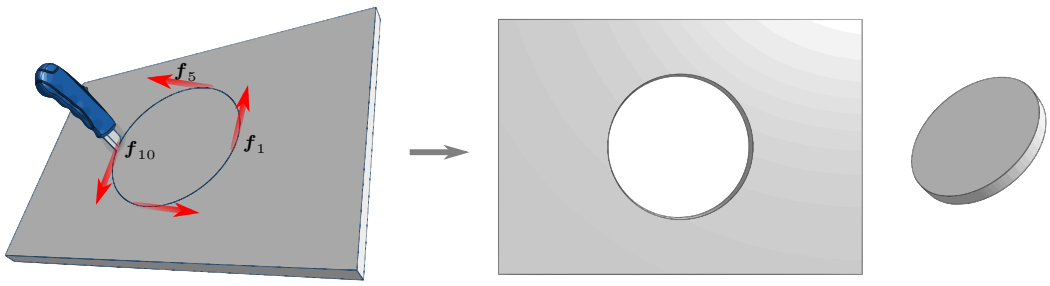
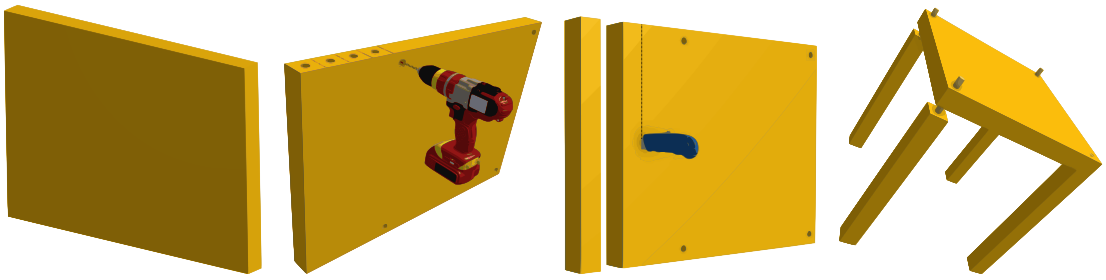


Figure 3.1: A circular cutting task consisting of a continuous sequence of cutting operations.



(a) A wooden board (b) Drilling 8 holes (c) Cutting 4 legs (d) Inserting 4 pegs

Figure 3.2: A chair fabricating task involving a sequence of drilling, cutting and inserting operations.

apply such forceful operations, e.g. cutting and drilling, on target objects, while robots hold the objects for humans. Later in Chap. 4, we present a graphical user interface, which human users can easily use to specify such forceful tasks, i.e. sequences of forceful operations on selected objects, in an interactive manner.

Mathematically, as illustrated in Fig. 3.3, we define a *forceful operation* $\boldsymbol{F}$ as a generalized operation force[1] (wrench) ${}^{\mathrm{t}}\boldsymbol{f}$ w.r.t. a tool frame Σ_{t}, which is applied at a pose ${}^{\mathrm{o}}\boldsymbol{p}$ w.r.t. the target object. Hereafter we omit the superscripts in above denotations for the sake of simplicity. Then, a *forceful operation* can be denoted as:

$$\boldsymbol{F} : (\boldsymbol{f}, \boldsymbol{p}) \tag{3.1}$$

Accordingly, a *forceful task* comprising a sequence of forceful operations, can be denoted as:

$$\{\boldsymbol{F}_i\}_{i=1}^{m} = \{(\boldsymbol{f}_i,\ \boldsymbol{p}_i)\}_{i=1}^{m} \tag{3.2}$$

where m indicates the number of involved forceful operations in the task. For example, as illustrated in Fig. 3.1, we address the circular cutting task as a sequence of 20 continuous cutting operations via discretization.

Note that during the course of a forceful operation $\boldsymbol{F}$, the target object is positioned by a robot at a pose $\boldsymbol{p}_{\mathrm{o}} \in \mathbb{SE}(3)$ w.r.t. a robot/world frame[2] Σ_{r}. The object pose $\boldsymbol{p}_{\mathrm{o}}$ can be pre-specified empirically or by optimizing certain quality metrics, such as *human preferences* in the context of HRC. Specifically, in this thesis, we assume the object pose(s) during a forceful task to be pre-specified in Chap. 5, focusing on using grasp contacts to stabilize objects. In Chap. 6, object contacts with the shared environment are exploited for stable object manipulation, therefore the constraint on object pose is released to allow the exploration of various environmental contacts. In the context of fHRC discussed in Chap. 7, we propose to optimize the object pose in a collaborative forceful operation such that the human comfort and safety

[1] Ideally, a forceful operation $\boldsymbol{F}$ would simply impose a single deterministic operation force $\boldsymbol{f}$ onto a target object. However, in actual applications, such a human-applied forceful operation $\boldsymbol{F}$ would inevitably deviate from the expected force direction to some extent. Later in Sec. 4.1, we take such force uncertainties into consideration and model the operation force $\boldsymbol{f}$ as a distribution of deviated forces that can possibly happen in actual applications.

[2] Without loss of generality, we assume the robot frame Σ_{r} coincides with the world frame.

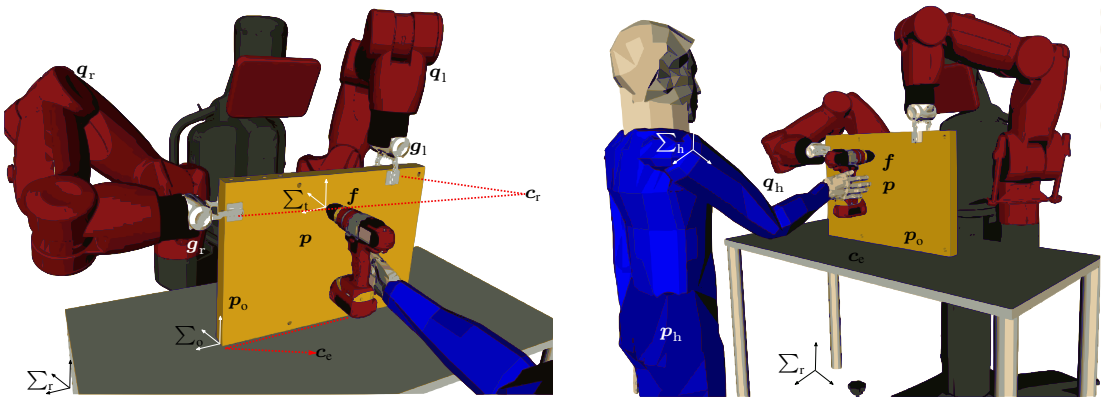


Figure 3.3: Definitions in a forceful human-robot collaborative operation.

can be maximized.

3.2 Human, Robot, Environment and Object Contacts

We assume the robot has two manipulators, and each manipulator is equipped with a parallel plate gripper[3]. Let $\mathcal{C}_\text{l}, \mathcal{C}_\text{r}$ be the configuration space of the left and right arm respectively, and $\mathcal{C}_\text{o} \subseteq SE(3)$ be the configuration space of the object. The composite configuration space $\mathcal{C}$ of the system can be then defined as the Cartesian product of the three aforementioned spaces $\mathcal{C} = \mathcal{C}_\text{l} \times \mathcal{C}_\text{r} \times \mathcal{C}_\text{o}$. Each composite configuration $\boldsymbol{q} \in \mathcal{C}$ can be denoted as $\boldsymbol{q} = (\boldsymbol{q}_\text{l},\ \boldsymbol{q}_\text{r},\ \boldsymbol{p}_\text{o})$, where $\boldsymbol{q}_\text{l} \in \mathcal{C}_\text{l}$, $\boldsymbol{q}_\text{r} \in \mathcal{C}_\text{r}$, and $\boldsymbol{p}_\text{o} \in \mathcal{C}_\text{o}$.

We define a robot *grasp* $\boldsymbol{g}$, or more generally, *grasp contact*, using the relative pose(s) of gripper(s) on the target object. A bimanual grasp (contact) $(\boldsymbol{g}_\text{l},\ \boldsymbol{g}_\text{r})$ specifies poses of both left and right grippers, while the unimanual grasps $(\boldsymbol{g}_\text{l})$, $(\boldsymbol{g}_\text{r})$ specify poses of only left and right gripper respectively.

Note that there is redundancy in this definition. Specifically, a system configuration $\boldsymbol{q} = (\boldsymbol{q}_\text{l}, \boldsymbol{q}_\text{r}, \boldsymbol{p}_\text{o})$ can be mapped to its corresponding grasp $\boldsymbol{g}$ via forward kinematics. In this regard, the composite configuration space $\mathcal{C}$ can be regarded as a collection of lower-dimensional *grasp manifolds*, in which each manifold corresponds to a particular robot grasp on the object. We denote the manifold in $\mathcal{C}$ corresponding to the grasp $\boldsymbol{g}$ as $\mathcal{M}(\boldsymbol{g})$.

Environmental and Robot Contact: Geometrically, a system configuration $\boldsymbol{q}$ specifies *environmental contact* $\boldsymbol{c}_\text{e}$ and/or *robot contact* $\boldsymbol{c}_\text{r}$, including the grasp contact $\boldsymbol{g}$, onto the surface of the target object, which the environment and robot rely on to provide reactive forces capable of stabilizing the object against gravity and forceful operations. In this thesis, we assume the robot can provide such object contacts by *pressing* or *grasping* with its grippers, while the environment includes rigid structures that allow the object to be in contact with. We aim to explore more general and diverse contact patterns of environmental and robot contacts in future work.

[3]This is for clarity of explanation and because the robot we use in our experiments has two arms. However, our formulation is general and can be easily extended to systems with more manipulators.

Table 3.1: Nomenclature.

${}^{\mathrm{t}}\boldsymbol{f}$, $\boldsymbol{f}$	An operation force w.r.t. the tool frame Σ_{t}
${}^{\mathrm{o}}\boldsymbol{p}_{\mathrm{f}}$, $\boldsymbol{p}_{\mathrm{f}}$	An operation pose w.r.t. the object frame Σ_{o}
${}^{\mathrm{r}}\boldsymbol{p}_{\mathrm{o}}$, $\boldsymbol{p}_{\mathrm{o}}$	An object pose w.r.t. the robot/world frame Σ_{r}
$\boldsymbol{F}$	A forceful operation
$\{\boldsymbol{F}_i\}_{i=1}^{m}$	A forceful task
m	The number of involved forceful operations in a task
$\mathcal{C}$	The composite system configuration space
$\boldsymbol{q}$	A composite system configuration
$\boldsymbol{g}$	A robot grasp (grasp contact) on the target object
$\mathcal{M}(\boldsymbol{g})$	The grasp manifold corresponding to grasp $\boldsymbol{g}$
$\boldsymbol{c}_{\mathrm{e}}$	Environmental contact on the target object
$\boldsymbol{c}_{\mathrm{r}}$	Robot contact on the target object
$\boldsymbol{q}_{\mathrm{h}}$	The human arm configuration
$\boldsymbol{p}_{\mathrm{h}}$	The human body pose w.r.t. the robot frame Σ_{r}

Human: As illustrated in Fig. 3.3, during a collaborative forceful operation, a human applies an external force with a specific tool (e.g. cutter and drill) onto an object held by a robot. In accordance, the geometry and kinematics of the human and robot are tightly coupled through the tool-object system. In this thesis we assume a fixed grasp configuration for the human to hold the tool in applying a forceful operation, and thus a fixed kinematic transformation between the human hand and tool.

We assume a human frame attached at the human shoulder Σ_{h}. We denote the human arm configuration[4] as $\boldsymbol{q}_{\mathrm{h}}$. We use $\boldsymbol{p}_{\mathrm{h}} \in \mathbb{SE}(3)$ to denote the human body pose w.r.t. the robot frame. These notations will be used later in Chap. 7 in planning for human comfort and safety in fHRC.

In this thesis, we assume the geometrical models of the robot, object and environment are given to the planner. Other physical parameters, e.g. the object mass, the centre of gravity and the friction coefficients associated with contacts, are also specified and fed to the planners. All notations mentioned above are list in Table. 3.1.

[4] In this work we use q_{h} to refer to human arm configuration. However, the optimization problem for human comfort (Chap. 7) we define in this thesis is general to the whole body configuration.

Chapter 4

Modelling, Stability Check and Specification of Forceful Operations

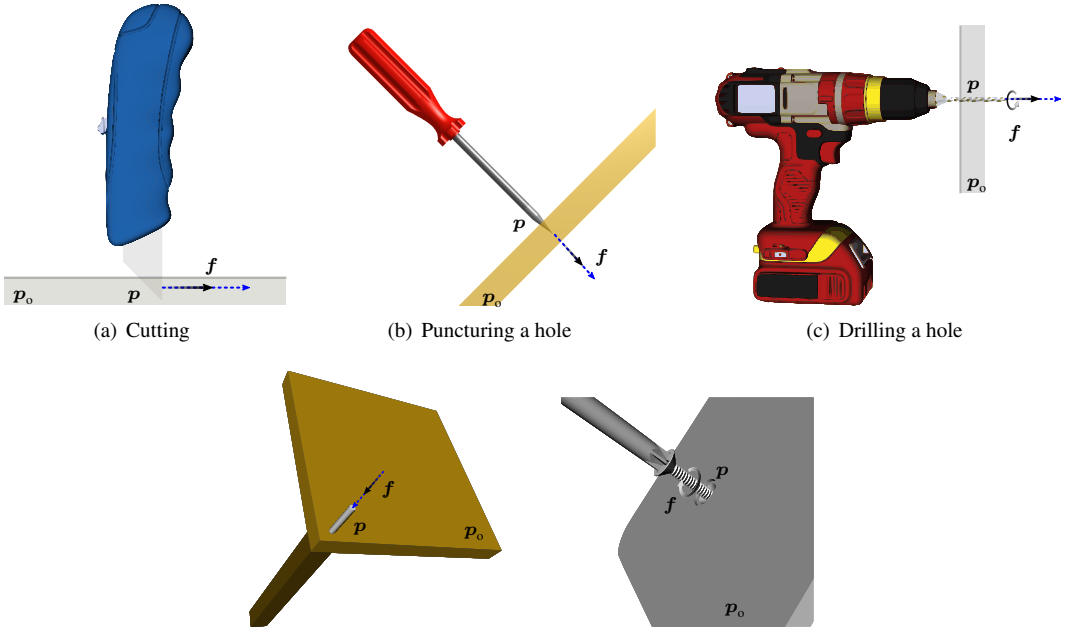
This section presents our work on modelling human-robot collaborative forceful operations under the presence of force deviations, and explains in detail how we check the system's static stability while at a candidate configuration $\boldsymbol{q}$ under the application of a forceful operation $\boldsymbol{F}$, which is referred to as the *stability check* problem throughout this thesis. Additionally, we present a graphical user interface, which human users can rely on to specify desired forceful tasks, i.e. sequences of forceful operations on selected objects, interactively.

4.1 Modelling of a Forceful Operation

The primary step towards manipulation planning under changing external forces, i.e. a sequence of collaborative forceful operations, is to model the force distribution of each involved forceful operation. Previously in Chap. 3, we conceptually define a forceful operation $\boldsymbol{F}$ as an operation force $\boldsymbol{f}$ to be applied at a pose $\boldsymbol{p}$ on the target object. In line with the definition, herein we study the distribution of the operation force $\boldsymbol{f}$ in the wrench space systematically.

We first present an *idealized operation model* representing the expected distribution of the operation force $\boldsymbol{f}$, and then introduce a *conic operation model*, which takes into account the force uncertainties inherent in human-applied operations, strictly representing the probabilistic/predictive distribution of the operation force $\boldsymbol{f}$ in the presence of deviations with a generalized spherical force (wrench) cone.

Further, we propose to approximate the spherical cone with a circumscribed multi-edged pyramid, i.e. a limited number of primitive forces, which dramatically reduces the computational complexity on account of robust stability check.



(a) Cutting (b) Puncturing a hole (c) Drilling a hole

(d) Interting a peg (e) Screwing a fastener

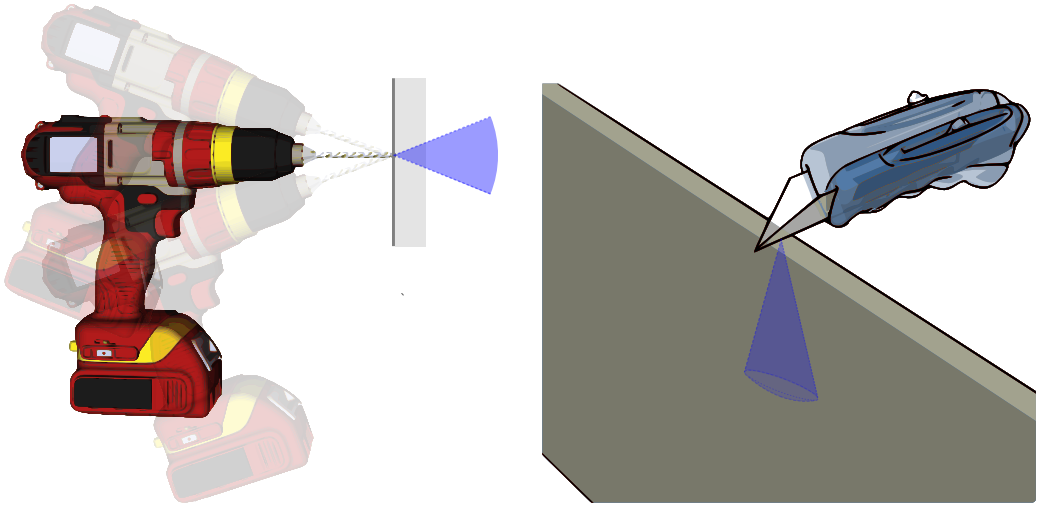
Figure 4.1: A forceful operation $\boldsymbol{F}$ ideally generates a deterministic operation force $\boldsymbol{f}$ on a target object along/about an expected operation axis (depicted as the blue axes).

4.1.1 Idealized Operation Model

As illustrated in Fig. 4.1, a collaborative forceful operation $\boldsymbol{F}$, such as cutting (Fig. 4.1(a)) and drilling (Fig. 4.1(c)), qualitatively can be stated as moving a tool, e.g. the cutter in Fig. 4.1(a) or the peg in Fig. 4.1(d), along/about an *operation axis* (depicted as blue axes in Fig. 4.1), which is specified together by the expected operation pose $\boldsymbol{p}$ and the object pose $\boldsymbol{p}_o$, to interact with a target object. Accordingly, the operation $\boldsymbol{F}$ generates and transmits an operation force $\boldsymbol{f}$ along/about the operation axis onto the object.

In this sense, if the forceful operation $\boldsymbol{F}$ is applied as expected, i.e. the operation $\boldsymbol{F}$ generates a deterministic force $\boldsymbol{f}$ on the target object along/about the expected operation axis on the object, we can simply model $\boldsymbol{f}$ as a single generalized force. For example, ideally, a puncturing operation (Fig. 4.1(b)) applies a pure intruding force along an expected puncturing direction perpendicular to the object surface. Similarly, a drilling operation (Fig. 4.1(c)) applies an element of pure drilling force along an expected drilling axis perpendicular to the object surface, together with an element of pure torque about the axis.

Herein, we refer to this simplified formulation as the *idealized operation model*. The model assumes the operation $\boldsymbol{F}$ to be exactly applied along/about its expected operation axis on the target object with a deterministic operation force $\boldsymbol{f}$, namely a single point in the wrench space, which can be parametrized by the maximum operation force (f_z and/or τ_z).



(a) Deviation of a drilling operation.

(b) Deviation of a cutting operation.

Figure 4.2: The operation force $\boldsymbol{f}$ may deviate from the expected operation axis to some extent.

4.1.2 Conic Operation Model

Clearly, the idealized model is only applicable to cases where forceful operations can be precisely applied as expected. However, in practice, such a collaborative forceful operation in its actual applications would inevitably deviate to some extent from the expected operation axis. For example, as illustrated in Fig. 4.2(a), for a drilling operation, rather than along the expected drilling axis, the actual applied drilling force may possibly deviate towards any direction within a certain range. Fig. 4.2(b) shows a similar observation of a cutting operation.

The existence of such uncertainties, as a result, pose a challenge, particularly for *robust stability check*. To be more specific, for a forceful operation $\boldsymbol{F}$, since the operation force can inevitably deviate, even if a candidate configuration $\boldsymbol{q}$ can keep the object stable under the expected operation force $\boldsymbol{f}$, there is no guarantee that the configuration $\boldsymbol{q}$ can also keep the object stable under all other deviated forces, which is critical for the robustness of stability check. That is, with the existence of operation deviations, in addition to the expected force $\boldsymbol{f}$, a robustly stable configuration is also required to be stable against any possible deviated force, e.g. the forces within the illustrated cone in Fig. 4.2(a) for the drilling operation.

We formulate a *conic model* to address such operation deviations in modelling the forceful operation $\boldsymbol{F}$. Specifically, consider a forceful operation $\boldsymbol{F}$, we assign a tool frame Σ_{t} at the tool-tip as shown in Fig. 4.3, with its $+z$ axis aligned with the operation axis. The force deviation can then be modelled equally as the deviation of the operation axis. Geometrically, the operation force $\boldsymbol{f}$ can be applied towards any direction within a range, which is a spherical cone centred with the expected operation axis as illustrated in Fig. 4.3 with the grey cone. In this regard, the distribution of the operation force $\boldsymbol{f}$ can

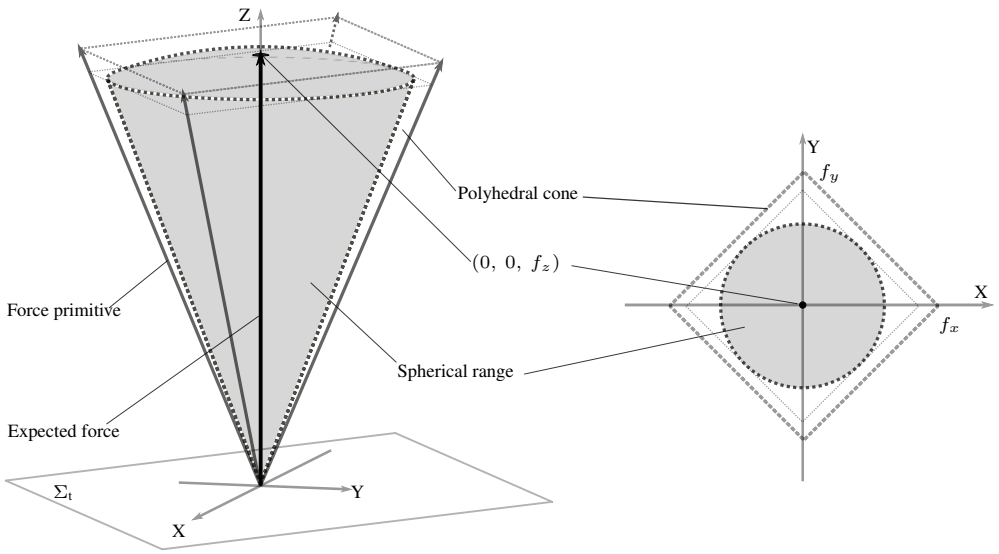


Figure 4.3: The spherical cone models the force deviation strictly, while the polyhedral cone conservatively approximates the spherical cone with a limited number of primitive forces.

then be modelled as a generalized spherical cone in the wrench space, which we refer to as the *spherical cone model* hereafter.

The spherical cone model strictly constrains the force distribution of a forceful operation $\boldsymbol{F}$ in the presence of force deviations. However, to ensure the robustness of stability check and thus the effectiveness of manipulation planning under the operation $\boldsymbol{F}$, this model requires a planner to fully consider all possible forces within the spherical cone. That is, to guarantee the force stability of a candidate configuration $\boldsymbol{q}$ under the operation $\boldsymbol{F}$, all possible forces within its corresponding spherical cone must be separately checked to be resistible by the configuration $\boldsymbol{q}$, which is extremely computationally expensive due to the cone continuity.

To reduce the computational complexity for robust stability check, we propose to relax the geometric constraint defined by the spherical cone, and conservatively approximate it with a $n_{\boldsymbol{F}}$-edged *circumscribed* polyhedral cone, which is illustrated as the outer pyramid in Fig. 4.3. We call this model as the *polyhedral cone model*. As illustrated, the polyhedral cone represents the force distribution of the operation $\boldsymbol{F}$ in a loose manner, but bounds all possible deviated forces in the spherical cone conservatively. This is advantageous in the sense of producing no false positives while the cost we pay may be false negatives.

More importantly, by exploiting *convex analysis*, any force within the polyhedral cone can be denoted as a convex linear combination of its edges, for example, the four grey arrowed lines in Fig. 4.3. We call these edges as *primitive forces* hereafter. That is, any deviated force within the polyhedral cone, and therefore all forces within the spherical cone, can be represented by a limited number of deterministic primitive forces. This property would simplify the stability check using the conic model dramatically, since checking only a limited number of primitive forces is sufficient to guarantee force stability under

all possible forces within the polyhedral cone and therefore within the spherical cone. This is the exact requirement for robust stability check, and can be easily proved by convex analysis.

The number $n_{\boldsymbol{F}}$ can be chosen empirically. Note that, a larger $n_{\boldsymbol{F}}$ would make the polyhedral cone closer to the spherical cone. However, this may require more time for stability check, since for a forceful operation $\boldsymbol{F}$ and a candidate configuration $\boldsymbol{q}$, in the worst case, stability check involves checking all $n_{\boldsymbol{F}}$ primitive forces. A smaller $n_{\boldsymbol{F}}$, in contrast, would make the polyhedral cone more conservative by containing additional forces outside the spherical cone. This might lead to the loss of feasible solutions, since the polyhedral cone imposes a relatively stronger constraint by covering additional forces into stability check. In this sense, the choice of $n_{\boldsymbol{F}}$ is more or less a trade-off between planning efficiency and the loss of feasible solutions due to conservative approximation.

In the next section, we discuss the stability check using both the idealized model and the conic model.

4.2 Stability Check of a forceful Operation

Consider a forceful operation $\boldsymbol{F}$, the *stability check* refers to checking whether a candidate configuration $\boldsymbol{q}$ is able to provide sufficient wrenches, particularly through object contacts with both the robot and environment, to keep the target object stable under the application of operation force $\boldsymbol{f}$.

Specifically, we focus on two manipulation scenarios discussed later in Chap. 5 and 6 respectively, which differ in the *contact patterns* that the system relies on for object manipulation under changing external forces:

1. Object manipulation with robot grasps: A scenario in line with the planning work in Chap. 5 where the robot can only exploit grasp contacts to keep the object stable under forceful operations;

2. Object manipulation with environmental contacts: A scenario in line with the planning work in Chap. 6 where the robot can use object contacts with both the environment and robot, including robot grasps, to stabilize the object under forceful operations.

4.2.1 Stability Check with Robot Grasps

In the case of manipulating objects with only robot grasps, given a forceful operation $\boldsymbol{F}$ and a candidate system configuration $\boldsymbol{q}$ (and its corresponding grasp $\boldsymbol{g}$ on the object), the focus of stability check is on whether:

– The robot manipulators are able to provide sufficient stiffness to keep itself and the target object stable against the operation $\boldsymbol{F}$. This requires the planner to check whether the required torques distributed at the manipulator joints exceed their corresponding joint torque limits.

– The robot grippers are able to provide sufficient wrenches at the grip points to stabilize the object in hands, which requires the external wrench induced by the operation force $\boldsymbol{f}$ on the object is

inside the grasp wrench space (Mishra *et al.*, 1987) of the grasp $\boldsymbol{g}$, namely the set of all external wrenches the robot grasp $\boldsymbol{g}$ can resist.

In this context, we discuss the stability check using both the idealized operation model and the conic operation model respectively.

Stability Check Using Idealized Model:

As aforementioned, the idealized model addresses a forceful operation $\boldsymbol{F}$ as a deterministic force $\boldsymbol{f}$ to be applied at a pose $\boldsymbol{p}$ on the target object. In this context, consider a generalized force $\boldsymbol{f}_{\mathrm{g},i}$ acting at the grip point of the i-th robot manipulator, by invoking the principle of virtual work (Takase, 1974), the required efforts $\boldsymbol{\tau}_i$ at the manipulator joints can be derived as

$$\boldsymbol{\tau}_i = \boldsymbol{J}_i^{\mathsf{T}} \boldsymbol{f}_{\mathrm{g},i} \tag{4.1}$$

where $\boldsymbol{J}_i$ is the geometric Jacobian matrix of the i-th manipulator at configuration $\boldsymbol{q}$.

Furthermore, the symmetric formulation proposed by Uchiyama & Dauchez (1988, 1992), generalized this relationship (Eq. 4.1) to multiple manipulators of cooperative manipulation, representing the kinematic and static relationships between the external force $\boldsymbol{f}$ applied at the common manipulated object and its counterparts $\boldsymbol{\tau}$ acting at the manipulator joints (which we review in Sec. 2.1).

The symmetric formulation, however, leaves the force/torque $\boldsymbol{f}_{\mathrm{g}}$ at grip points unconstrained. Specifically, for the case of parallel-plate gripper we use in this work, as illustrated in Fig. 4.4-Right, we approximate the grasp wrench space of a grip with an *axis-aligned wrench box* in the wrench space. We use the maximum resistible forces/torques along/about the three principal axes (XYZ) at each grip point as its grip limits $\boldsymbol{f}_i^+$, $\boldsymbol{f}_i^-$, where $\boldsymbol{f}_i^+ = \left[\mathrm{P}_x^+, \mathrm{P}_y^+, \mathrm{P}_z^+, \mathrm{R}_x^+, \mathrm{R}_y^+, \mathrm{R}_z^+\right]_i^{\mathsf{T}}$ and $\boldsymbol{f}_i^- = \left[\mathrm{P}_x^-, \mathrm{P}_y^-, \mathrm{P}_z^-, \mathrm{R}_x^-, \mathrm{R}_y^-, \mathrm{R}_z^-\right]_i^{\mathsf{T}}$ are the vectors of estimated limits at the i-th grip point, in the $+$ direction and $-$ direction respectively. $\mathrm{P}_{x/y/z}^{+/-}$ and $\mathrm{R}_{x/y/z}^{+/-}$ are the maximum resistible forces and torques respectively.

Imposing the additional constraints onto the above formulations, we model the stability check as finding a distribution of manipulator torques $\boldsymbol{\tau}$ and grasp wrenches $\boldsymbol{f}_{\mathrm{g}}$ that satisfies:

$$\begin{cases} \boldsymbol{J}^{\mathsf{T}} \boldsymbol{f}_{\mathrm{g}} = \boldsymbol{\tau} \\ \boldsymbol{G} \boldsymbol{f}_{\mathrm{g}} = -\boldsymbol{R}(\boldsymbol{p}) \boldsymbol{f} \end{cases} \\ \boldsymbol{\tau}^- \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}^+, \quad \boldsymbol{f}^- \leq \boldsymbol{f}_{\mathrm{g}} \leq \boldsymbol{f}^+ \tag{4.2}$$

where

- $\boldsymbol{J}$=$diag(\boldsymbol{J}_1, \ldots, \boldsymbol{J}_n)$ is the composite Jacobian matrix at configuration $\boldsymbol{q}$;
- $\boldsymbol{f}_{\mathrm{g}}$=$\left[\boldsymbol{f}_{\mathrm{g},1}^{\mathsf{T}}, \boldsymbol{f}_{\mathrm{g},2}^{\mathsf{T}}, \ldots, \boldsymbol{f}_{\mathrm{g},n}^{\mathsf{T}}\right]^{\mathsf{T}}$ and $\boldsymbol{f}_{\mathrm{g},i}^{\mathsf{T}}$ is the generalized force acting at the i-th robot gripper;
- $\boldsymbol{\tau}$=$\left[\boldsymbol{\tau}_1^{\mathsf{T}}, \boldsymbol{\tau}_2^{\mathsf{T}}, \ldots, \boldsymbol{\tau}_n^{\mathsf{T}}\right]^{\mathsf{T}}$and $\boldsymbol{\tau}_i$ is the vector of joint torques at i-th manipulator;

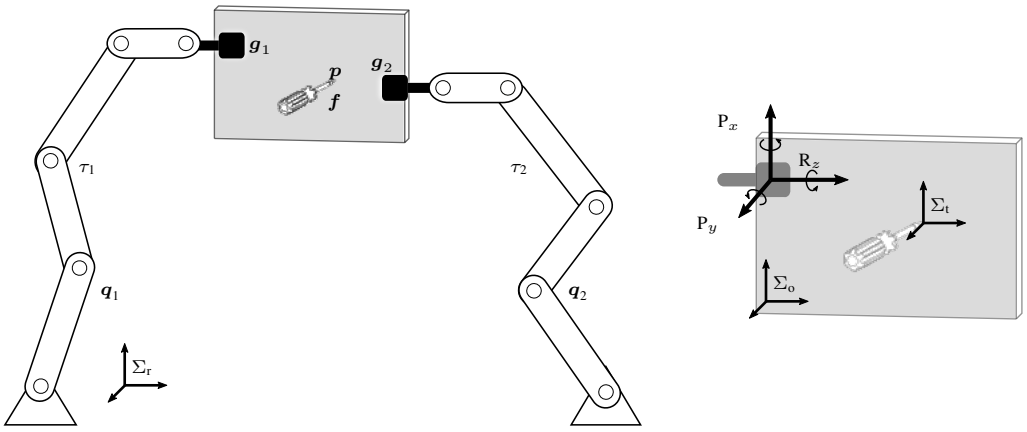


Figure 4.4: Left: We are interested in checking whether a composite configuration $\boldsymbol{q}$ (along with its corresponding grasp $\boldsymbol{g}$) is stable against a forceful operation $\boldsymbol{F}$. Right: We approximate the grasp wrench space of a grasp $\boldsymbol{g}$ with an axis-aligned box in the six-dimensional wrench space.

- $\boldsymbol{G} = [\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_n]$ (usually termed as the grasp matrix (Borst *et al.*, 2004; Ferrari & Canny, 1992; Mishra *et al.*, 1987)) is a $(6 \times 6n)$ matrix mapping the forces/torques acting at grips to a resultant wrench onto the object w.r.t. the robot frame;
- $\boldsymbol{\tau}^{+/-}$ are the force/torque limits of all manipulator joints;
- $\boldsymbol{f}^{+/-}$ denotes the estimate of the force/torque limits at grip points (i.e. our estimate of the grasp wrench space).
- $\boldsymbol{R}(\boldsymbol{p})$ transforms the external wrench from the tool frame to the robot frame.

Eq. 4.2 models the stability check problem as a linear programming problem, and thus can be solved, e.g. using the Simplex method, to see if there exists any feasible solution of the torques $\boldsymbol{\tau}$ at the manipulator joints and grasp wrenches $\boldsymbol{f}_{\mathrm{g}}$ at the grip points meeting Eq. 4.2. If this process fails, we consider the configuration $\boldsymbol{q}$ (and its corresponding grasp $\boldsymbol{g}$) unstable against the operation $\boldsymbol{F}$.

Stability Check Using Conic Model

As aforementioned, using the conic model, theoretically, all possible forces within the spherical cone should be separately checked for robust stability check. However, we can use a limited number of primitive forces to conservatively approximate the spherical cone and therefore simplify this computationally expensive procedure.

Therefore, given a configuration $\boldsymbol{q}$ and a forceful operation $\boldsymbol{F}$, the stability check using the conic model involves checking the force stability of the configuration $\boldsymbol{q}$ under $n_{\boldsymbol{F}}$ primitive forces, while each of the $n_{\boldsymbol{F}}$ checks follows the same form of Eq. 4.2.

That is, the planner needs to perform *at most* $n_{\boldsymbol{F}}$ basic stability checks defined by Eq. 4.2 for a forceful operation $\boldsymbol{F}$ and a configuration $\boldsymbol{q}$. If all $n_{\boldsymbol{F}}$ stability checks succeed, we then regard $\boldsymbol{q}$ as a

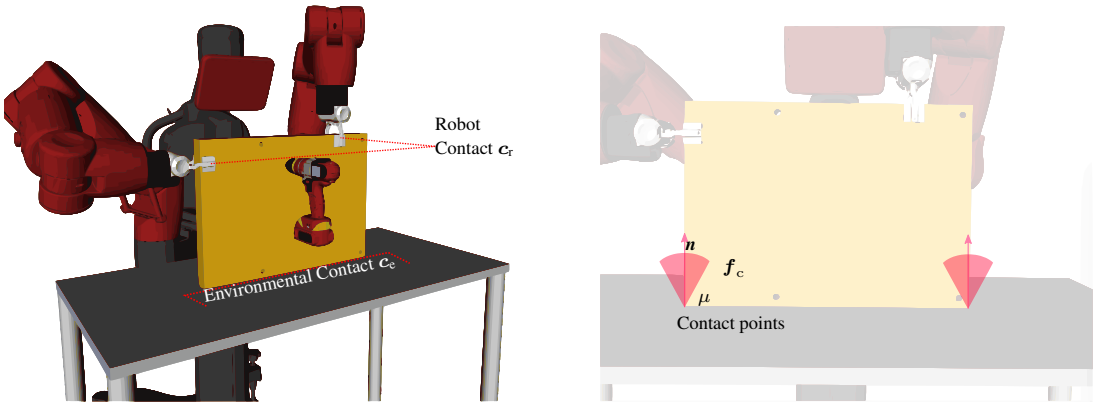


Figure 4.5: The environmental and robot contacts specify a group of contact region(s) on the object.

stable configuration for the operation $\boldsymbol{F}$, while if any of the $n_{\boldsymbol{F}}$ stability checks fails, the planer can stop and regard $\boldsymbol{q}$ as unstable without further checking.

4.2.2 Stability Check with Environmental Contacts

In the case of manipulating objects using object contacts with both robot grippers and rigid structures in the shared environment, we focus on checking whether the system can maintain the contacts under the application of a forceful operation $\boldsymbol{F}$, i.e. whether these object contacts can provide sufficient wrenches to stabilize the object under the operation force $\boldsymbol{f}$. Generally, in the context of using environmental contacts for object manipulation, the environmental contact plays the major role in resisting external forces, while the robot contact provides mobility to the manipulated object, which can also be observed in real forceful applications by humans and our real robot experiments. Thus, we ignore the torque limits on the manipulator joints in this context and mainly focus on constraints on maintaining the object contacts under external disturbances.

We discuss the stability check using both the idealized operation model and the conic operation model in this context below.

Stability Check Using Idealized Model

As illustrated in Fig 4.5, under the assumption of rigid bodies, both environmental and robot contacts specify a group of contact region(s) on the target object, denoted as $\boldsymbol{c}_{\mathrm{e}}$ and $\boldsymbol{c}_{\mathrm{r}}$ respectively, which can be further represented as a set of contact points. At each such contact point, the contact force $\boldsymbol{f}_c$ that can be applied onto the object is constrained within a friction cone $\boldsymbol{f}_{\mathrm{c}} \in FC(\boldsymbol{n}, \mu)$ characterized by a contact normal $\boldsymbol{n}$ and a friction coefficient μ.

In this context, given a forceful operation $\boldsymbol{F}$, and a candidate configuration $\boldsymbol{q}$ which generates a set of total n_{p} point contacts (including uniformly discretized surfaces of both environmental contacts and robot contacts) on the object, stability check requires finding a distribution of contact forces $\boldsymbol{h} = [\boldsymbol{f}_{\mathrm{c},1}^{\mathrm{T}},\ \boldsymbol{f}_{\mathrm{c},2}^{\mathrm{T}},\ \ldots,\ \boldsymbol{f}_{\mathrm{c},n_{\mathrm{p}}}^{\mathrm{T}}]^{\mathrm{T}}$, such that:

1. The resultant wrench provided by $\boldsymbol{h}$ is able to keep the object stable, i.e. in static equilibrium, under the operation force $\boldsymbol{f}$;

2. The vector of contact forces $\boldsymbol{h}$ lies within the composite friction cone $\mathrm{H} = FC_1 \times ... \times FC_{n_\mathrm{p}}$.

Therefore, similar to Lee *et al.* (2015); Murooka *et al.* (2017); Wang & Pelinescu (2003), we formulate the problem of stability check with environmental contacts as a constrained optimization problem, in particular using quadratic programming:

$$\min_{\boldsymbol{h}\in\mathrm{H}} \quad \boldsymbol{h}^{\mathrm{T}}\boldsymbol{h} \tag{4.3a}$$

$$\text{s.t.} \quad \boldsymbol{G}\boldsymbol{h} + \boldsymbol{h}_{mg} + R(\boldsymbol{p})\boldsymbol{f} = 0 \tag{4.3b}$$

$$\boldsymbol{h} \in \mathrm{H} \tag{4.3c}$$

where $\boldsymbol{G} = [\boldsymbol{G}_1, ..., \boldsymbol{G}_{n_\mathrm{p}}]$ is a $(6 \times 6n_\mathrm{p})$ grasp matrix mapping the contact wrenches to a resultant wrench onto the object. $\boldsymbol{h}_{mg}$ is the wrench by the object gravity. $\boldsymbol{R}(\boldsymbol{p})$ transforms the external wrench by $\boldsymbol{f}$ from the tool frame to the robot frame.

Given a candidate configuration $\boldsymbol{q}$ and a forceful operation $\boldsymbol{F}$, if the planner can find a solution of contact forces $\boldsymbol{h}$ satisfying the constraints in Eq. 4.3, we say that the configuration $\boldsymbol{q}$ can be stable against the operation $\boldsymbol{F}$. Otherwise, we say that the configuration $\boldsymbol{q}$ cannot be stable against the operation $\boldsymbol{F}$. The stability constraints (Eq. 4.3(b)-(c)) guarantee that there exists a distribution of contact forces $\boldsymbol{h}$ that can keep the object stable under the operation $\boldsymbol{F}$, while optimizing the objective 4.3(a) ensures $\boldsymbol{h}$ to be close to the actual distribution of the contact forces (Wang & Pelinescu, 2003).

Stability Check Using Conic Model

Likewise, in the case of stability check using the conic model, the operation force $\boldsymbol{f}$ represents a continuous spherical cone and can be approximated with a number of $n_{\boldsymbol{F}}$ primitive forces. Thus, using the conic model, stability check with environmental contacts concerns checking the force stability of the contact configuration $\boldsymbol{q}$ under all the $n_{\boldsymbol{F}}$ primitive forces, based on the optimization problem defined in Eq. 4.3.

Specifically, if all the $n_{\boldsymbol{F}}$ stability checks based on Eq. 4.3 succeed, we regard the candidate configuration $\boldsymbol{q}$ as a feasible solution for the operation $\boldsymbol{F}$. Otherwise, any failed check among the $n_{\boldsymbol{F}}$ stability checks terminates the checking process with a failure.

4.3 A Graphical Interface: Specification of Forceful Operations

In this section, we present *a graphical user interface* to close the loop of robot-assisted forceful manipulation:

- ☐ Using the interface, the human users can easily specify forceful tasks, i.e. sequences of forceful operations on selected objects;

- ☐ The robot and our planners assist the humans in performing the customized tasks by manipulating the selected objects stable for humans and providing operation instructions;

- ☐ The expert developers pre-specify available forceful operations using experimental data.

The interface is designed as three components: 1) As part of interface specification, expert developers input the geometry of available objects, tools and accordingly applicable forceful operations into the interface. For example, the expert developers can collect experimental data to specify available forceful operations. In the following section, we show how this was achieved for forceful operations involved in this work. 2) Human users specify desired forceful tasks using the tools and objects provided by the interface. The forceful tasks will be sent to the planner as inputs to generate effective and efficient plans for object manipulation. 3) The robot manipulates the target object for the human users in performing collaborative forceful tasks.

Fig. 4.6 illustrates the overall workflow of the robot-assisted fabrication with the user interface using the circular cutting task. First, a human user specifies an expected forceful task by choosing a tool(s) (e.g. a cutter or a drill) to draw on a selected object. For example in Fig. 4.6, the human first selects a cutter and a rectangular board, and then draws a circle on the board to specify the circular cutting task (Fig. 4.6-Task Specification). Once receiving confirmation, the interface triggers a planning process (with our planners acting as the underlying planners) to generate efficient manipulation plans which will be discussed in following sections (Fig. 4.6-Manipulation Planning).

After planning, the interface controls the robot to assist the human in performing the specified forceful operations, as well as providing *operation instructions* to the human according to the manipulation plan (Fig. 4.6-Fabricating). During fabrication, the robot assistant manipulates the target object to the planned configurations in sequence and stabilizes it under the application of forceful operations.

At each planned configuration, the interface instructs the human user to apply *a subsequence of resistible forceful operations* checked by our planners, by visually displaying the subsequence in both the interface and the robot head monitor (Fig. 4.6-Operations to Be Applied). After completing the instructed operations, the human presses a Regrasp button provided by the interface to command the robot to the next planned configuration(s) (Fig. 4.6-Regrasp). The regrasp button is how the human notifies the system that the subsequence of forceful operations are applied. In this manner, the interface connects the robot assistant and the human user to perform forceful tasks interactively.

Besides, the interface allows expert developers to pre-specify available target objects, tools and accordingly forceful operations that can be applied under a current workshop setting, using experimental data or expert knowledge. We will discuss how we capture the distributions of forceful operations involved in this thesis using experimental data in the next section.

We realized the limitation of the communication and coordination through the interface: compared with other channels, like natural language, the interface may affect the fluidness of collaboration. In future work, we aim to improve the system by automatic perception of human operations. In addition, we aim to explore other communication manners, e.g. implicit communication (Gildert *et al.*, 2018; Giuliani *et al.*, 2018; Hazbar, 2019; Knepper, 2016; Knepper *et al.*, 2017; Kulkarni *et al.*, 2019; Wortham

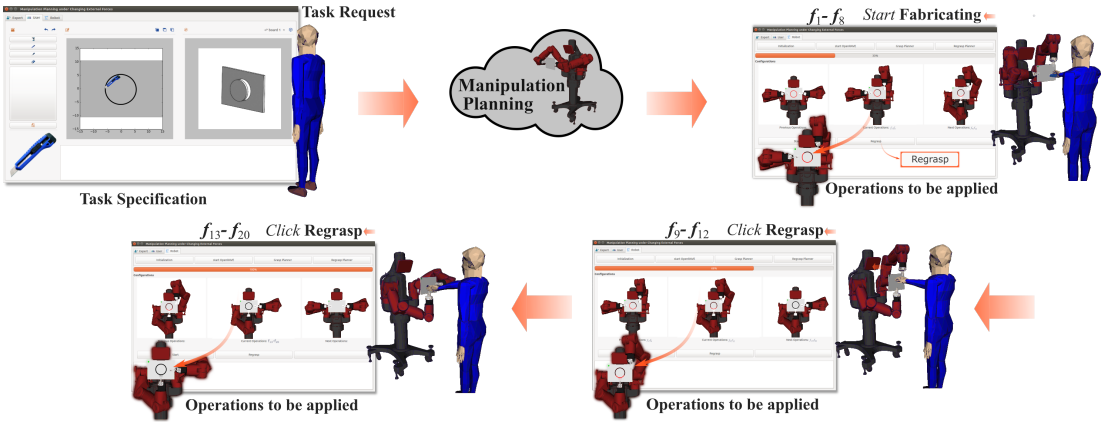


Figure 4.6: The workflow of human-robot collaboration in performing forceful tasks with the graphical user interface.

& Theodorou, 2017) and channels, e.g. vision-based communication, to future improve the overall collaboration performance.

4.4 Experiments and Results

In this section, we collected experimental data to capture distributions of the forceful operations involved in our experiments and studied the effect of conic model in stability check and manipulation planning in comparison with the idealized model.

4.4.1 Modelling Forceful Operations with Experimental Data

We tested our proposed planners on a variety of collaborative forceful operations. Herein we show three of them, including *cutting*, *puncturing* and *drilling* on rigid foam boards as illustrated in Fig. 4.1(a)-4.1(c). We collected experimental data to capture their distributions in the wrench space.

As discussed in Sec. 4.1, according to the idealized operation model:

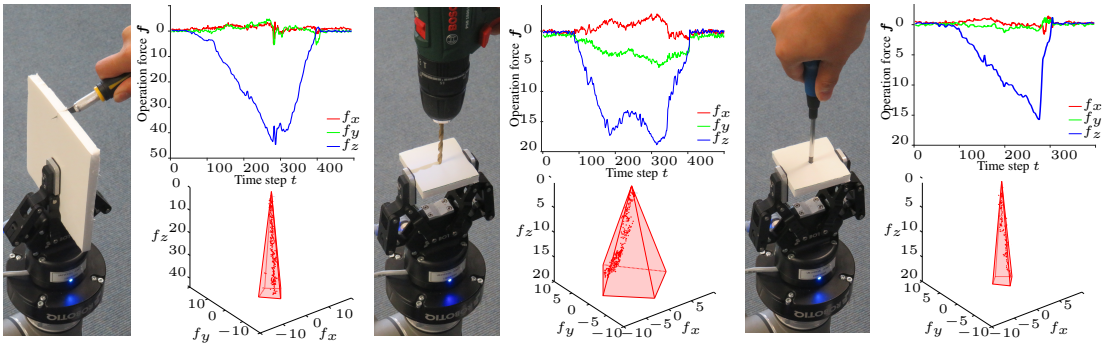
– A cutting operation applies a pure cutting force along an expected cutting axis;

– A puncturing operation applies a pure intruding force along an expected puncturing axis;

– A drilling operation applies a torque about an expected drilling axis plus a force along the axis.

Therefore, the idealized model requires identifying the maximum operation forces, extracting values of parameters f_z and τ_z from experimental data.

While using the conic model, we need to further determine their deviation ranges in the wrench space, i.e. extracting values of parameters f_x, f_y and τ_x, τ_y. To do this, we applied each category of



(a) Distrib. of a cutting operation. (b) Distrib. of a drilling operation. (c) Distrib. of a puncturing operation.

Figure 4.7: We collected experimental data from 30 trials to model the distribution of a forceful operation in the wrench space. The red polyhedral cones are the extracted models over 30 trials. The force plots on the right and the red dots inside the cones are the force data over one experimental trial.

operations 30 times separately, collecting data using a six-dimensional force/torque sensor (FT150 from Robotiq) as shown in Fig. 4.7.

During each trial (30 for each category), we recorded the force and torque values at different time steps at the rate reported by the sensor. This means, for each trial, we collected between 400-500 data points. For each category of forceful operations, we computed the polyhedral cone as discussed in Sec. 4.1 which contains the force distributions over all time steps over the 30 trials.

Specifically, for the cutting operations, we tested that $f_z = 45\,N$, $f_x = 4\,N$ and $f_y = 6\,N$. For the drilling operations, $f_z = 19\,N$, $f_x = 6\,N$ and $f_y = 6\,N$. Note that the torque element generated by the drill bit rotation was much smaller than the ones generated by drilling forces, thus we simply neglect it and assume $\tau_x = \tau_y = \tau_z = 0\,Nm$. For the puncturing operations, $f_z = 16\,N$, $f_x = 2\,N$ and $f_y = 2\,N$. Fig. 4.7 shows the distributions of these three operations in one experiment trial. The red polyhedral cone in each figure is the extracted conic model over 30 trials.

4.4.2 Effect of Conic Model in Stability Check

To test the effect of the conic model in stability check, for each category of operations above, we tested 20 forceful operations evenly distributed on the object surface.

For each forceful operation, we first generated 50 stable configurations (i.e. 50 different complete robot configurations grasping the object stably against the idealized operation force) while using the idealized model. That is, these 50 configurations were checked to be stable against the operation using the idealized operation model.

We then checked the stability of these configurations again (50×20 for the 20 operations, giving a total of 1000 additional stability checks for each category) *but* using the conic model, to see if they could still remain stable if operation deviations were considered.

The results showed that, on average, there were **39** among the 50 configurations remaining feasible

using the conic model for the cutting operations, **41** configurations remaining feasible for the drilling operations and **49** configurations remaining feasible for puncturing operations. This is reasonable, since the drilling forces and the cutting forces deviate more dramatically compared with the puncturing forces.

We are also concerned about the extra checking time required by the conic model. As discussed in Sec. 4.2, each stability check involves a linear programming problem based on Eq. 4.2 or a constrained optimization problem based on Eq. 4.3. Thus, in comparison with the idealized model, the conic model requires the planner to solve at most $(n_{\boldsymbol{F}}-1)$ more linear programmings or constrained optimizations, for each combination of a forceful operation and a candidate contact configuration, which may result in extra checking time cost. However, for example, for the case of object manipulation using robot grippers, we will show later in Table 5.2 that the time for generating the stable grasp sequences, which includes the total time of checking the force stability over all forceful operations and candidate configurations, is negligible in the overall planning. Similar time efficiency can also be observed in the case of manipulation using environmental contacts in Table. 6.2 and 6.3.

In this context, we can sum up that using the conic model can greatly benefit the robustness of stability check, but without imposing significant computational load. In the experiments shown in following chapters, we used the conic model for stability check and set the number of primitive forces $n_{\boldsymbol{F}} = 4$ in modelling forceful operations. In the following chapters, we will systematically discuss our planning work to achieve efficient, smooth and human-friendly collaborative forceful manipulation.

Chapter 5

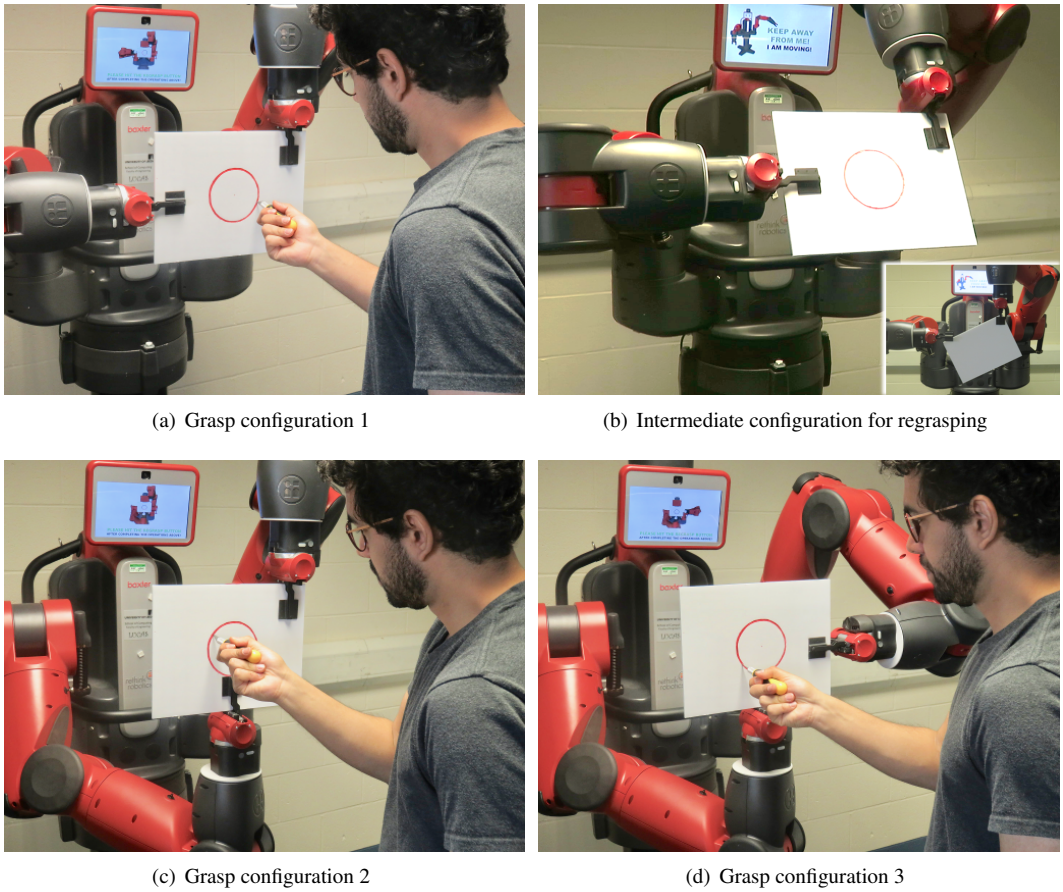
Manipulation Planning under Changing External Forces—Robot Grasps

This chapter presents our work on manipulation planning to generate collaborative robot behaviours, particularly using *robot grasps*, to keep an object stable under changing external forces. We aim at empowering robots with the capability of manipulating objects not only under geometric constraints, but also under the application of changing externals forces while explicitly considering both task *stability* and *efficiency* in object manipulation.

Take the cutting task in Fig. 5.1 for example , where a human is cutting a circular piece out from a rectangular board, while with the assistance of a robot system. Before the task, the human informs the robot system of the operation type (cutting) and the desired cutting pattern (a circle) with the interface previously introduced in Sec. 4.3. During the cutting task, the human applies external cutting forces on the board which change position, direction, and even magnitude along the circular path. To assist the human to perform the task, the robot changes its grasp on the object multiple times (Fig. 5.1(a)-5.1(d)) to position the object at desired pose(s) for cuttings and keep it stable against the changing cutting forces. In this context, we propose a planner that enables the robot to manipulate objects under changing external forces like this.

Specifically, there are two key problems our planner solves:

First, our planner produces **efficient** manipulation plans by minimizing the number of times the robot needs to change its grasp on the object, namely *regrasp*. For example in Fig. 5.1, the robot uses three different grasp configurations to stabilize the object and accordingly changes its grippers' position on the object only two times (counting each gripper move separately) throughout the whole task. This is also a capability demonstrated by humans in sequential manipulation tasks: we regrasp when we need to, but we are also able to choose grasps which are useful for long durations during a task.



(a) Grasp configuration 1

(b) Intermediate configuration for regrasping

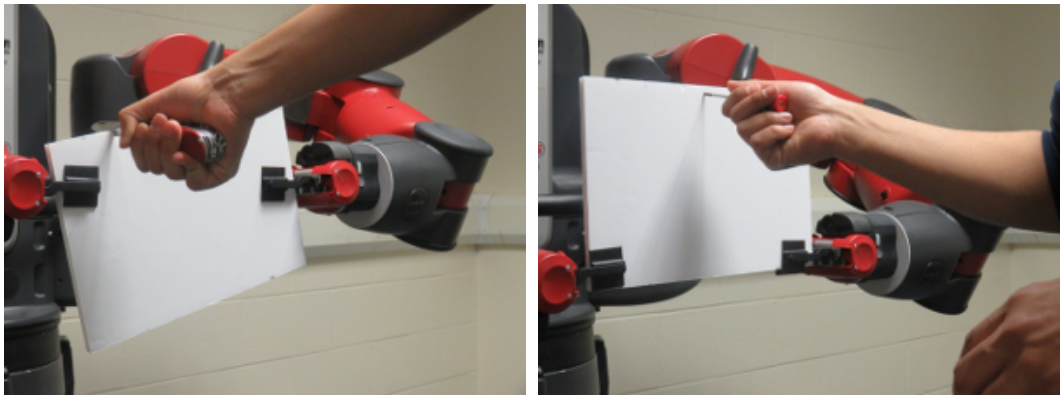
(c) Grasp configuration 2

(d) Grasp configuration 3

Figure 5.1: The human is cutting a circular piece out of a board which is held by a dual-arm robot.

This capability poses two closely related challenges to the planner: *grasp planning* and *regrasp minimization*. Specifically, the planner must decide not only how to grasp the object, but also when to regrasp the object during the course of forceful interaction, such that the object can be always stable under the changing external forces. A *good* choice of robot grasp on the object may enable the robot to stabilize the object against multiple sequential external forces, and thus reduce the need of regrasping throughout the interaction. A *bad* grasp, however, would lead to frequent regrasps and thus task interruptions. Even worse, an inappropriate grasp may not be able to stabilize the object against certain external forces, thus bringing about failures and even risks during task execution. For example, the object may slip through the gripper fingers under a cutting operation (Fig. 5.2(a)) due to insufficient frictional forces between the grippers and object. Similarly, a drilling operation may exert excessive torque deforming the grippers and object due to a bad choice of grasp (Fig. 5.2(b)).

Second, our planner plans each regrasp. A regrasp requires the robot to release its grippers off the object and then to grasp the object at different positions. However, when the robot releases a gripper,



(a) The object slides between fingers due to insufficient frictional forces.

(b) The object bends due to excessive torque.

Figure 5.2: Task failures during cutting (a) and drilling (b).

the object may become unstable due to the existence of external forces. Even if we assume the human in Fig. 5.1(a) stops applying cutting forces during regrasps, the object may still become unstable due to gravity. For example, to regrasp the object from the grasp configuration in Fig. 5.1(a) to the one in Fig. 5.1(c), if the robot directly releases its right gripper as shown in the small figure at the right bottom of Fig. 5.1(b), a heavy object may slip within the remaining gripper. Alternatively, the robot may move the object to an intermediate pose first before releasing one of its grippers, such that the remaining gripper(s) can still hold the object stable until the robot completes the regrasp. Fig. 5.1(b) shows such an intermediate pose, where the object can be stable even when the right robot gripper releases from it.

We build the planner on the following key contributions:

– A graph-based formulation of the problem of manipulation planning under changing external forces, which is referred to as the *operation graph* hereafter and accordingly a planning algorithm which can simultaneously (i) decide on a sequence of grasp configurations to position and stabilize the object, and (ii) minimize the need of regrasping during manipulation (Sec. 5.2.1).

– An algorithm to plan stable *regrasps in the air* using multiple cooperative manipulators. Different from most existing work in regrasp planning, which we reviewed in Sec. 2.2), we focus on the context of bimanual regrasping without placing an object on extra supports. This is achieved by further evaluating the object stability under gravity while going through a sequence of intermediate unimanual and bimanual grasps (Sec. 5.2.2-5.2.4).

– A variety of simulated and real human-robot experiments to verify the performance of our proposed planner in terms of minimizing the number of regrasps and planning stable regrasps (Sec. 5.3).

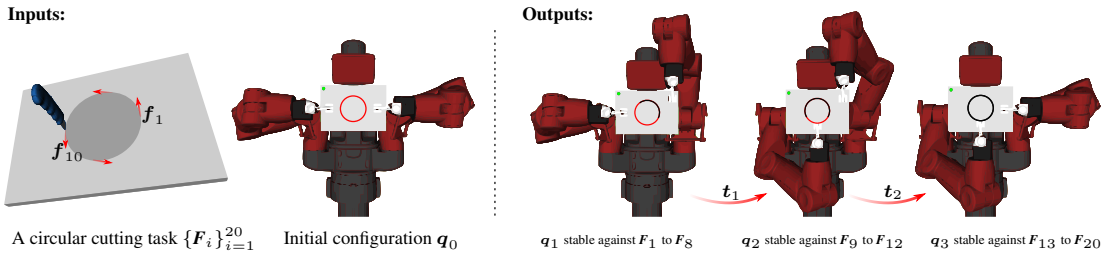


Figure 5.3: A manipulation planning problem to keep an object stable under changing external forces.

5.1 Manipulation Planning under Changing External Forces

Fig. 5.3 illustrates the manipulation planning problem to keep an object stable under changing external forces using the circular cutting task. In this chapter, we particularly focus on the context of using only *robot grasps* to achieve this manipulation goal.

Broadly, consider a forceful manipulation task, the robot is supposed to position and stabilize a target object under the application of a sequence of forceful operations. Accordingly, it requires the planner to find an appropriate system configuration, which generates a set of robot and/or environmental contacts on the object, for each involved forceful operation, such that the system stability can be always maintained by the object contacts under the changing operation forces.

In this context, given a **single** forceful operation $\boldsymbol{F}$, the planner can find such a stable grasp configuration $\boldsymbol{q}$ for the operation by first searching for a kinematically valid configuration $\boldsymbol{q}$ and then checking if the corresponding object contacts can provide sufficient supporting wrenches keeping the object stable under the external operation force $\boldsymbol{f}$, which we discuss previously in Sec. 4.2 as stability check. Note that, in this chapter, the object contacts involve only robot grasps, while in the context of using the environment discussed in the next chapter, the object contacts can be environmental and/or robot contacts on the target object.

Then, given **a sequence of** forceful operations $\{\boldsymbol{F}_i\}_{i=1}^{m}$, i.e. a complete forceful task, the planner can simply find one feasible configuration $\boldsymbol{q}_i$ for each operation $\boldsymbol{F}_i \in \{\boldsymbol{F}_i\}_{i=1}^{m}$ and accordingly, impose a need for changing the system configuration, e.g. *regrasp* in the context of using grasp contacts, between every two sequential operations. Such a straightforward strategy, however, would require the planner to use a large sequence of different system configurations and therefore configuration transfers during object manipulation. Herein, we refer to such a configuration transfer as a *configuration change*.

Alternatively, to avoid frequent task interruptions due to required configuration changes, the robot can make the utmost of one configuration $\boldsymbol{q}$, using it against a larger sequence of forceful operations, which in turn would reduce the number of configuration changes required in the long term. This, ***configuration change minimization***, imposes an additional but practically necessary requirement to achieve efficient and smooth manipulation, particularly in the context of fHRC. The need of reducing task interruptions, e.g. regrasps, has been recognized as an important demand by many existing studies in various contexts (Dobson & Bekris, 2015; Dogar *et al.*, 2019; Harada *et al.*, 2014; Saut *et al.*, 2010). Note that, in addition to the number of regrasps, some other quality metrics may play signifiant roles

in object manipulation, especially in the context of fHRC. For example, human comfort is one of the critical criteria in collaborative forceful operations. In this thesis, we explicitly address the configuration change minimization as a main objective, building a graph-based planning framework to find a sequence of grasp/contact configurations minimizing the configuration changes while satisfying the stability constraint under changing external forces, while in this chapter, we interpret configuration change minimization as ***regrasp minimization***.

5.1.1 Problem Definition

Here we define the problem mathematically. We say a system configuration $\boldsymbol{q}$ is *stable against* a sequence of k forceful operations $\{\boldsymbol{F}_i\}_{i=1}^{k}$ if, at $\boldsymbol{q}$, the corresponding object contacts can provide sufficient wrenches to keep the object stable under any forceful operation in $\{\boldsymbol{F}_i\}_{i=1}^{k}$. Further, we say that a sequence of system configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$ is stable against a sequence of forceful operations $\{\boldsymbol{F}_i\}_{i=1}^{m}$, if the configurations in $\{\boldsymbol{q}_j\}_{j=1}^{n}$ cover all forceful operations in $\{\boldsymbol{F}_i\}_{i=1}^{m}$ in order, i.e. if $\boldsymbol{q}_1$ is stable against $\{\boldsymbol{F}_1, \boldsymbol{F}_2, ..., \boldsymbol{F}_k\}$, $\boldsymbol{q}_2$ is stable against $\{\boldsymbol{F}_{k+1}, \boldsymbol{F}_{k+2}, ..., \boldsymbol{F}_p\}$, and so on, until $\boldsymbol{q}_n$ is stable against $\{\boldsymbol{F}_{q+1}, \boldsymbol{F}_{q+2}, ..., \boldsymbol{F}_m\}$, where $1 \leq k < p \leq ... \leq q < m$.

For example, the three grasp configurations $\{\boldsymbol{q}_1,\ \boldsymbol{q}_2,\ \boldsymbol{q}_3\}$ shown in Fig. 5.3 are stable against the 20 circular cutting operations ($\boldsymbol{q}_1$ is stable against $\boldsymbol{F}_1$ to $\boldsymbol{F}_8$. $\boldsymbol{q}_2$ is stable against $\boldsymbol{F}_9$ to $\boldsymbol{F}_{12}$. $\boldsymbol{q}_3$ is stable against $\boldsymbol{F}_{13}$ to $\boldsymbol{F}_{20}$). Note that different configurations correspond to different grasps. In this sense, regrasp minimization can be achieved by finding a minimal sequence of configurations, denoted as $\{\boldsymbol{q}_j\}_{j=1}^{n}$, stable against the required task $\{\boldsymbol{F}_i\}_{i=1}^{m}$.

In addition to minimizing the number of regrasps, the robot also needs to move the object to go through the planned configurations in $\{\boldsymbol{q}_j\}_{j=1}^{n}$ successively, using *collision-free* and *stable* trajectories $\{\boldsymbol{t}_j\}_{j=1}^{n}$. Specifically, each trajectory $\boldsymbol{t}_j \in \{\boldsymbol{t}_j\}_{j=1}^{n}$ moves the system from $\boldsymbol{q}_{j-1}$ to $\boldsymbol{q}_j$ ($\boldsymbol{q}_0$ is the initial system configuration), which corresponds to a constrained motion planning or regrasping task. The two arrowed lines in in Fig. 5.3 illustrate such regrasping trajectories.

In this context, we define a ***manipulation query*** as a forceful task consisting of a sequence of forceful operations $\{\boldsymbol{F}_i\}_{i=1}^{m}$ to be applied on the object, together with a starting system configuration $\boldsymbol{q}_0$. Then, the problem can be stated as:

Definition 1 (**Manipulation Planning to Keep an Object Stable under Changing External Forces**). Given the description of the system and a manipulation query $(\{\boldsymbol{F}_i\}_{i=1}^{m},\ \boldsymbol{q}_0)$, find a minimal sequence of system configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$ and connecting trajectories $\{\boldsymbol{t}_j\}_{j=1}^{n}$ to position and stabilize the object under the the application of forceful operations in $\{\boldsymbol{F}_i\}_{i=1}^{m}$ in order.

Note that in this chapter finding a minimal sequence of system configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$ will minimize the number of regrasps required in the solution, which is in line with the general objective, configuration change minimization. Later in Chap. 6, we will discuss the planning context with environmental contacts, but still under the same problem definition.

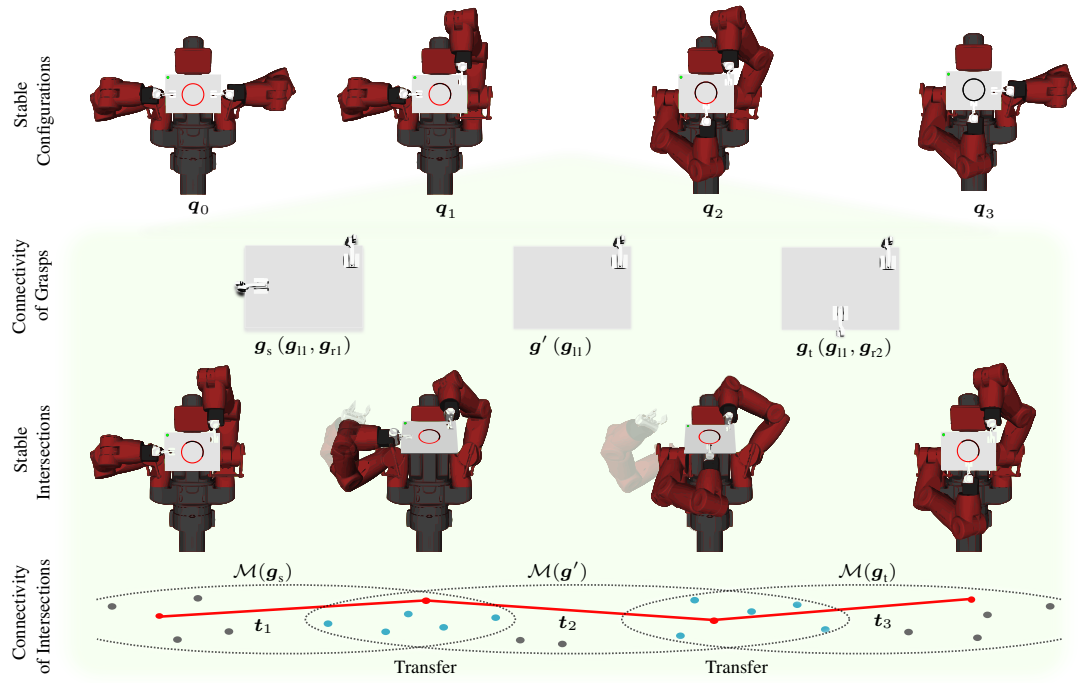


Figure 5.4: Overview of the approach.

5.1.2 Approach Overview

This section outlines our planning approach to solve a manipulation query in the context of minimizing regrasps, in Fig. 5.4 with four layers using the circular cutting task, and explain how these layers fit together. We present the details of each layer in Sec. 5.2.

Stable Configurations: Given a manipulation query $(\{\boldsymbol{F}_i\}_{i=1}^{m}, \boldsymbol{q}_0)$, the planner first identifies a sequence of configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$ stable against the task, while minimizing the number of regrasps required to position and stabilize the object. In Fig. 5.4, the three configurations $\{\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3\}$ shown in the top layer is such an example candidate configuration sequence.

The configurations generated by this layer are discrete over the configuration space. The lower layers of the planner try to generate a sequence of collision-free and stable trajectories $\{\boldsymbol{t}_j\}_{j=1}^{n}$ to connect every two subsequent configurations in $\{\boldsymbol{q}_j\}_{j=1}^{n}$ starting from $\boldsymbol{q}_0$, which corresponds to a sequence of constrained regrasping tasks.

Connectivity of Grasps: Given any two subsequent configurations $\boldsymbol{q}_s$, $\boldsymbol{q}_t \in \{\boldsymbol{q}_j\}_{j=1}^{n}$ generated in the previous layer (e.g. $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ in Fig. 5.4), the planner identifies a sequence of intermediate grasps $\{\boldsymbol{g}_i\}_{i=1}^{n_g}$, which moves the robot grippers from the grasp $\boldsymbol{g}_s$ in $\boldsymbol{q}_s$ to the grasp $\boldsymbol{g}_t$ in $\boldsymbol{q}_t$ (denoted as $\boldsymbol{g}_1$ and $\boldsymbol{g}_{n_g}$ respectively in the sequence $\{\boldsymbol{g}_i\}_{i=1}^{n_g}$).

The grasp sequence acts as an abstract plan to guide the subsequent search. The second layer in Fig. 5.4 shows such an example grasp sequence $\{\boldsymbol{g}_s, \boldsymbol{g}', \boldsymbol{g}_t\}$. It connects the grasps in configurations

$\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ of the previous layer. Note that there might be other feasible grasp sequences, which go through different intermediate gripper contacts as shown in Fig. 5.6.

Sampling Stable Intersections of Grasp Manifolds: Given any two neighbouring grasps $\boldsymbol{g}_i$, $\boldsymbol{g}_{i+1} \in \{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ generated in the previous layer, the planner identifies a set of candidate configurations by sampling within the intersection of their grasp manifolds $\mathcal{M}(\boldsymbol{g}_i) \bigcap \mathcal{M}(\boldsymbol{g}_{i+1})$ (illustrated as the blue points in Fig. 5.4). These samples are checked for stability against gravity such that at each configuration in the set, the *transition* from $\boldsymbol{g}_i$ to $\boldsymbol{g}_{i+1}$ can be performed stably. The second configuration in the third layer is such an example, at which both the unimanual and the bimanual grasps can hold the object stable under gravity.

Connectivity of Manifold Intersections: After obtaining a sequence of stable configurations in the intersections of the sequence of grasp manifolds, the fourth layer performs collision-free and stability-constrained motion planning within these manifolds, namely generating $\{\boldsymbol{t}_j\}_{j=1}^{n}$ to connect these configurations (illustrated as red solid lines in Fig. 5.4).

Overall, the layered structure enables the planner to minimize the number of regrasps at the top layer. Besides, it enables the upper layers to provide significant search guidance to lower layers, and leaves the time-consuming motion planning to the final layer. If lower layers of the planner fail to find a plan, the system goes back to higher layers to generate new and different high-level plans.

5.2 Planning Approach

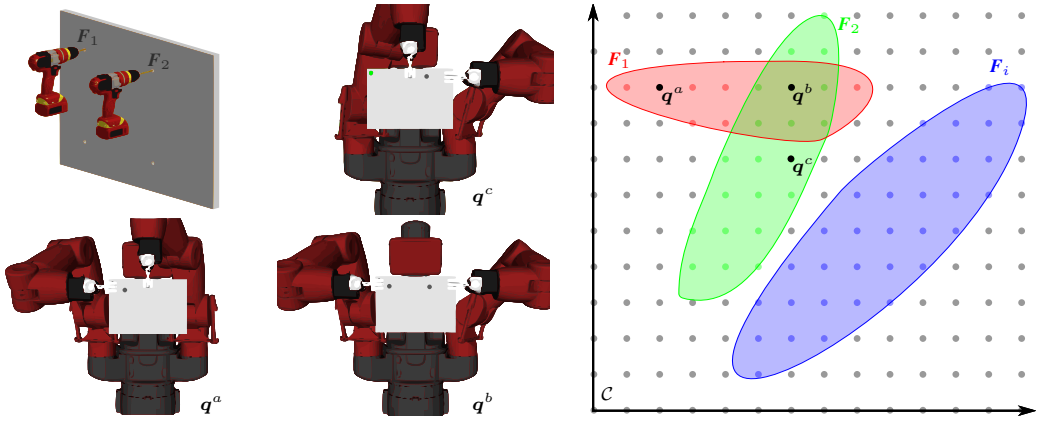
This section presents details of our planning approach in generating an efficient manipulation plan $(\{\boldsymbol{q}_j\}_{j=1}^{n}, \{\boldsymbol{t}_j\}_{j=1}^{n})$ to position and stabilize an object under a sequence of forceful operations $\{\boldsymbol{F}_i\}_{i=1}^{m}$.

5.2.1 Stable Configurations

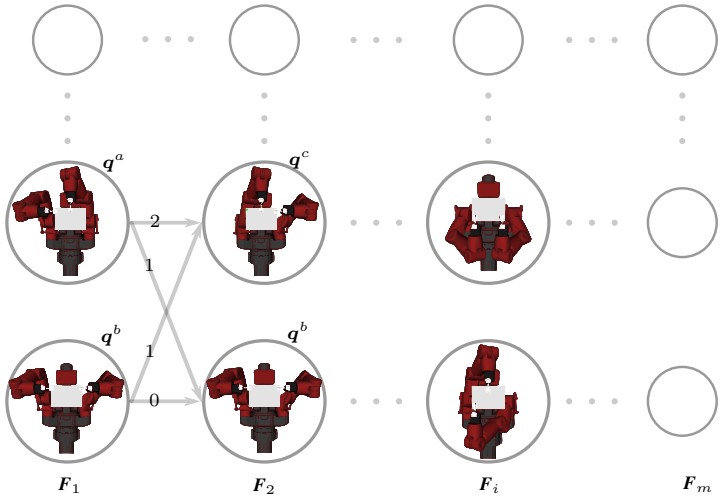
The planner starts by generating a minimal sequence of configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$ that are stable against the forceful operations $\{\boldsymbol{F}_i\}_{i=1}^{m}$.

Given a forceful operation $\boldsymbol{F}$, theoretically, there exists a set of configurations in the configuration space $\mathcal{C}$, i.e. a *stable region*, in which all configurations are stable against the operation $\boldsymbol{F}$. For example, in Fig. 5.5(a), we show two subsequent drilling operations ($\boldsymbol{F}_1$ and $\boldsymbol{F}_2$) of the table assembly task in Fig. 3.2, while the red and green regions in the configuration space $\mathcal{C}$ illustrate such stable regions for $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$ respectively. In this sense, finding a sequence of system configurations stable against $\{\boldsymbol{F}_i\}_{i=1}^{m}$ can be regarded as finding a sequence of configurations to visit all stable regions for the operations in $\{\boldsymbol{F}_i\}_{i=1}^{m}$ in order.

Furthermore, there might be *intersections* between these stable regions. Within each such intersection, any configuration would be stable against the corresponding multiple operations. For example, the configuration $\boldsymbol{q}^b$ in Fig. 5.5(a) is stable against both $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$, since $\boldsymbol{q}^b$ is inside the intersection of stable regions for $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$. Relying on this property, we use these intersections to minimize the number of regrasps.



(a) There exists a stable region in the configuration space $\mathcal{C}$ for a forceful opearation $\boldsymbol{F}$.



(b) An operation graph $\mathcal{G}_o$.

Figure 5.5: We build an operation graph to search for a minimal sequence of configurations $\{\boldsymbol{q}_j\}_{j=1}^n$ stable against $\{\boldsymbol{F}_i\}_{i=1}^m$.

Specifically, to create a minimal sequence of configurations $\{\boldsymbol{q}_j\}_{j=1}^n$, our planner first samples a set of candidate configurations in $\mathcal{C}$. To sample configurations that are likely to be stable against a variety of operations, i.e. configurations in the intersections, the planner starts by sampling grasps uniformly on the object. Then, using such a sampled grasp $\boldsymbol{g}$ and the desired object pose $\boldsymbol{p}_o$, the planner solves the inverse-kinematics problem, which may output multiple solutions, and randomly picks one configuration $\boldsymbol{q}$.

For such a sampled configuration $\boldsymbol{q}$, the planner identifies the operations in $\{\boldsymbol{F}_i\}_{i=1}^m$ that the configu-

ration $\boldsymbol{q}$ is stable against (Details of stability check is explained in Sec. 4.2). We then build an **operation graph** $\mathcal{G}_{\text{o}}$ using these stable configurations as shown in Fig. 5.5(b). The operation graph is an acyclic directed weighted graph. Specifically, in the operation graph, each column corresponds to a forceful operation. That is, the nodes in the i_{th} column are the sampled configurations that are stable against the operation $\boldsymbol{F}_i$. Further, we define a link between every two nodes in the neighbouring columns, and associate the link with a weight using *the number of gripper moves* required from one configuration to the other. For example, the weight for the link between the node $\boldsymbol{q}^b$ in the first column and the node $\boldsymbol{q}^b$ in the second column is zero, since these two nodes correspond to the same configuration and thus no regrasp is required. Similarly, if two configurations differ only by one gripper location on the object, the weight for their link is set as one (e.g. $\boldsymbol{q}^b$ and $\boldsymbol{q}^c$). Otherwise, the weight would be two. Note that one can come up with other weighting schemes, e.g. one that takes the overall motion trajectories into account to switch grasps.

At this point, our problem in this layer is formulated as a graph search problem. Given a manipulation query, the expected output is a path that starts from one node in the leftmost column for the operation $\boldsymbol{F}_1$ and ends with a node in the rightmost column for the operation $\boldsymbol{F}_m$.

By searching the entire operation graph $\mathcal{G}_{\text{o}}$, e.g. using Dijkstra's algorithm, the planner can generate a candidate sequence $\{\boldsymbol{q}_j\}_{j=1}^{n}$ with the least number of gripper moves based on the current set of samples. Hereafter, we call this planner the ***min-regrasp*** planner.

We provide the pseudo-code for this layer of the planner in Alg. 1 in the procedure PlanStableSequence. In line 1, the planner constructs the operation graph $\mathcal{G}_{\text{o}}$ as described above. In line 2, the planner searches the graph $\mathcal{G}_{\text{o}}$ (e.g. using Dijkstra's algorithm) for a candidate sequence $\{\boldsymbol{q}_j\}_{j=1}^{n}$. Then the planner iterates over every subsequent pair of configurations in $\{\boldsymbol{q}_j\}_{j=1}^{n}$ (line 4), attempting to plan a regrasp between them, which is explained below. If the regrasp planning fails between any two configurations (line 6), the planner removes the failing link from the graph in Fig. 5.5(b) (line 7), and then re-searches the graph to generate a new candidate sequence $\{\boldsymbol{q}_j\}_{j=1}^{n}$ (line 8).

Note that, building the whole graph $\mathcal{G}_{\text{o}}$ requires knowing the whole task $\{\boldsymbol{F}_i\}_{i=1}^{m}$ in advance. However, there may be cases for which the forceful operations are revealed progressively, e.g. one by one. In such cases, the operation graph $\mathcal{G}_{\text{o}}$ can be constructed as the next operation(s) is specified, and then be searched greedily. We call this version the ***greedy*** planner.

In addition, it is notable the operation graph is transferable. Specifically, the transferability can be achieved by the *similarity* among forceful tasks, which share similar forceful operations in terms of both operation order and operation feature. For example, an operation graph for assembling a stool can be directly extended for assembling a chair, as both tasks involve inserting four legs. Obviously, the similarity can also be defined for forceful tasks of different task scales, e.g. assembling a large chair versus a small chair. We aim to explore the diverse interpretations of task similarity in future work.

5.2.2 Connectivity of Grasps

Hereafter, we explain how our planner generates collision-free and stable trajectories $\{\boldsymbol{t}_j\}_{j=1}^{n}$ to move the target object to go through the planned configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$.

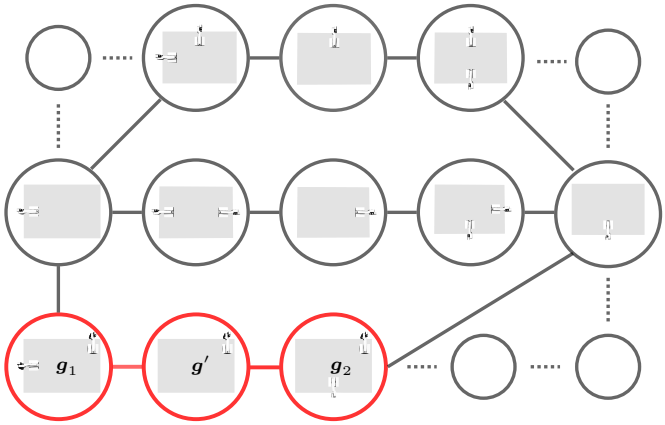


Figure 5.6: A grasp graph $\mathcal{G}_\mathrm{g}$. Each node in the grasp graph represents a bimanual or a unimanual grasp.

Given any two subsequent configurations, $\boldsymbol{q}_\mathrm{s}$, $\boldsymbol{q}_\mathrm{t} \in \{\boldsymbol{q}_j\}_{j=1}^{n}$ generated above, rather than directly searching in the high-dimensional combined configuration space $\mathcal{C}$, the planner first plans in the grasp level. Specifically, the planner starts by identifying a sequence of intermediate grasps $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ on the object, which moves the system from the grasp $\boldsymbol{g}_\mathrm{s}$ to the grasp $\boldsymbol{g}_\mathrm{t}$ ($\boldsymbol{g}_\mathrm{s}$ and $\boldsymbol{g}_\mathrm{t}$ are denoted as $\boldsymbol{g}_1$ and $\boldsymbol{g}_{n_\mathrm{g}}$ in $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ respectively). Here, $\boldsymbol{g}_\mathrm{s}$ is the grasp at the configuration $\boldsymbol{q}_\mathrm{s}$, and $\boldsymbol{g}_\mathrm{t}$ is the grasp at the configuration $\boldsymbol{q}_\mathrm{t}$. For example, consider $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ in the top layer of Fig. 5.4, the robot must go through a number of intermediate grasps ($\boldsymbol{g}'$) to switch from the grasp $\boldsymbol{g}_\mathrm{s}$ to the grasp $\boldsymbol{g}_\mathrm{t}$ on the object.

In the case of the dual-arm system used in this chapter, these intermediate grasps are either bimanual or unimanual. We represent the connectivity of these grasps as a *grasp graph* $\mathcal{G}_\mathrm{g}$ as illustrated in Fig. 5.6. Each node in the graph $\mathcal{G}_\mathrm{g}$ represents a grasp on the object. A bimanual and a unimanual grasp are connected if the unimanual grasp is contained by the bimanual grasp. For example, we say a bimanual grasp $(\boldsymbol{g}_\mathrm{l},\ \boldsymbol{g}_\mathrm{r})$ contains a unimanual grasp $(\boldsymbol{g}_\mathrm{l})$, since they share a common left gripper placement on the object. Building such a grasp graph requires the generation of feasible grasps on the object, which can be prespecified or can be generated using a grasp planner, e.g. Miller & Allen (2004).

Then, the planner searches the grasp graph $\mathcal{G}_\mathrm{g}$ to get a sequence of intermediate grasps $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$, which connects the grasps $\boldsymbol{g}_\mathrm{s}$ and $\boldsymbol{g}_\mathrm{t}$ (denoted as $\boldsymbol{g}_1$ and $\boldsymbol{g}_{n_\mathrm{g}}$ in the sequence) with a sequence of alternating bimanual and unimanual grasps. Fig. 5.6 highlights in red the shortest grasp sequence. Note that there might be other longer grasp sequences to connect the grasps $\boldsymbol{g}_\mathrm{s}$ and $\boldsymbol{g}_\mathrm{t}$ as well.

The grasp sequence $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ acts as an abstract plan to guide the search in the lower layers of the planner, constraining the motion planning into a concrete and limited sequence of grasp manifolds. In Alg. 1, the procedure PlanRegrasp outlines this process. On lines 1-2, the planner builds the grasp graph $\mathcal{G}_\mathrm{g}$ and searches it to obtain a sequence of grasps $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ as described above.

The planner then tries to plan the motion from $\boldsymbol{q}_\mathrm{s}$ to $\boldsymbol{q}_\mathrm{t}$ through the planned grasps in $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ (line 3). If the lower layers of the planner return with a failure to connect any two grasps $\boldsymbol{g}_i$ and $\boldsymbol{g}_{i+1}$ in $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ (line 4), then the planner removes the link between these grasps (line 7), and perform the search again to generate a new sequence of grasps (line 8). If the connection is successful, the planner returns the

re-grasp motion trajectory $\boldsymbol{t}$ (line 10).

5.2.3 Sampling Stable Intersections of Grasp Manifolds

A grasp sequence $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ provides necessary but not sufficient conditions of connectivity of the corresponding grasp manifolds. To check this connectivity, given any two subsequent grasps $\boldsymbol{g}_i$, $\boldsymbol{g}_{i+1} \in \{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$, we need to identify configurations at which both grasps $\boldsymbol{g}_i$, $\boldsymbol{g}_{i+1}$ are kinematically feasible and stable against external forces (e.g. gravity). Particularly in our task, the transition from a bimanual grasp to a unimanual grasp may fail, as the object might be unstable against the gravity and thus slide between the remaining gripper. Fig. 5.1(b) shows one such configuration in the bottom right corner, and another configuration, at which the same transition is stable due to an appropriate choice of the system configuration.

In this sense, our planner searches for stable configurations at the intersection of manifolds $\mathcal{M}(\boldsymbol{g}_i)$ and $\mathcal{M}(\boldsymbol{g}_{i+1})$ by random sampling. Specifically, in Alg. 1, the procedure SampleIntersection samples a set of k such configurations. To generate one such configuration, the planner first samples an object pose in the reachable space of the robot (line 4). Then, it solves the inverse-kinematics for the bimanual grasp at the sampled object pose to get a fully-assigned configuration $\boldsymbol{q}$ (line 5). The planner checks (line 6) whether both grasps $\boldsymbol{g}_i$ and $\boldsymbol{g}_{i+1}$ are stable against gravity at $\boldsymbol{q}$, using the stability check described in Sec. 4.2. The stable configuration $\boldsymbol{q}$ is then returned as a candidate point connection in the final solution path (line 7).

5.2.4 Connectivity of Sequence of Manifold Intersections

In this layer, given two configurations $\boldsymbol{q}_\mathrm{s}$ and $\boldsymbol{q}_\mathrm{t}$, and stable configurations sampled at the intersections of a sequence of manifolds (i.e. the grasp manifolds of the grasp sequence $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$), the planner attempts to generate a collision-free and stable trajectory $\boldsymbol{t}$ that connects $\boldsymbol{q}_\mathrm{s}$ to $\boldsymbol{q}_\mathrm{t}$ through these manifolds.

In Alg. 1, the procedure Connect implements this process as depth-first-search. Given a current configuration $\boldsymbol{q}_\mathrm{s}$ and a sequence of grasps $\{\boldsymbol{g}_i\}_{i=1}^{n_\mathrm{g}}$ (where $\boldsymbol{g}_1$ is the grasp in $\boldsymbol{q}_\mathrm{s}$), the planner samples the intersection of the first two grasp manifolds in the sequence for a set of stable configurations S (line 7). Then it tries to plan a motion from $\boldsymbol{q}_\mathrm{s}$ to a sampled configuration $\boldsymbol{q} \in S$ (line 9). Note that this is a motion plan within a single manifold and thus can be solved by existing closed-chain or single-arm motion planners. However, the object motion must be also stable against gravity, for which the constrained motion planners (Berenson *et al.*, 2011; Jaillet & Porta, 2013) can be used. If the motion planning is successful, the trajectory is returned along with a recursive call to the depth-first-search. Lines 1-6 handle the simple case where $\boldsymbol{q}_\mathrm{s}$ and $\boldsymbol{q}_\mathrm{t}$ are already in the same manifold.

5.3 Experiments and Results

This section presents a variety of experiments to verify the performance of our proposed planners.

Algorithm 1 Manipulation Planning under Changing Forces

PlanStableSequence $\left(\{\boldsymbol{F}_i\}_{i=1}^{m}, \boldsymbol{q}_0\right)$:

1: $\mathcal{G}_\text{o} \leftarrow$ Sample stable configurations in $\mathcal{C}$ and build an operation graph as shown in Fig. 5.5(b)
2: $\{\boldsymbol{q}_j\}_{j=1}^{n} \leftarrow$ GraphSearch$(\mathcal{G}_\text{o})$
3: $\{\boldsymbol{q}_j\}_{j=0}^{n} \leftarrow$ Insert $\boldsymbol{q}_0$ to the beginning of $\{\boldsymbol{q}_j\}_{j=1}^{n}$
4: **for** each subsequent $\boldsymbol{q}_j$ and $\boldsymbol{q}_{j+1}$ in $\{\boldsymbol{q}_j\}_{j=0}^{n}$ **do**
5: $\quad \boldsymbol{t}_{j+1} \leftarrow$ PlanRegrasp$(\boldsymbol{q}_j, \boldsymbol{q}_{j+1})$
6: $\quad$ **if** PlanRegrasp failed **then**
7: $\quad\quad \mathcal{G}_\text{o} \leftarrow$ Remove failing edge from graph $\mathcal{G}_\text{o}$
8: $\quad\quad$ Go to line 2
9: $\quad$ **end if**
10: **end for**
11: **return** $(\{\boldsymbol{q}_j\}_{j=1}^{n}, \{\boldsymbol{t}_j\}_{j=1}^{n})$

PlanRegrasp $(\boldsymbol{q}_\text{s}, \boldsymbol{q}_\text{t})$:

1: $\mathcal{G}_\text{g} \leftarrow$ Sample grasps and build graph in Fig. 5.6
2: $\{\boldsymbol{g}_i\}_{i=1}^{n_\text{g}} \leftarrow$ GraphSearch$(\mathcal{G}_\text{g}, \boldsymbol{q}_\text{s}, \boldsymbol{q}_\text{t})$
3: $\boldsymbol{t} \leftarrow$ Connect$(\boldsymbol{q}_\text{s},\ \{\boldsymbol{g}_i\}_{i=1}^{n_\text{g}},\ \boldsymbol{q}_\text{t})$
4: **if** Connect failed or t is None **then**
5: $\quad$ **if** maximum number of attempts reached **then**
6: $\quad\quad$ **return** failure
7: $\quad$ **end if**
8: $\quad \mathcal{G}_\text{g} \leftarrow$ Remove failing edge from graph $\mathcal{G}_\text{g}$
9: $\quad$ Go to line 2
10: **else**
11: $\quad$ **return** $\boldsymbol{t}$
12: **end if**

Connect $\left(\boldsymbol{q}_\text{s},\ \{\boldsymbol{g}_i\}_{i=1}^{n_\text{g}},\ \boldsymbol{q}_\text{t}\right)$:

1: **if** $n_\text{g} = 1$ **then**
2: $\quad t \leftarrow$ MotionPlan$(\boldsymbol{q}_\text{s}, \boldsymbol{q}_\text{t})$ using grasp $\boldsymbol{g}_{n_\text{g}}$
3: $\quad$ **if** MotionPlan successful **then**
4: $\quad\quad$ **return** $\boldsymbol{t}$
5: $\quad$ **else**
6: $\quad\quad$ **return** failure
7: $\quad$ **end if**
8: **end if**
9: $S \leftarrow$ SampleIntersection$(\boldsymbol{g}_1, \boldsymbol{g}_2)$

for each $\boldsymbol{q}$ in S **do**
    $\boldsymbol{t} \leftarrow$ MotionPlan$(\boldsymbol{q}_{\text{s}}, \boldsymbol{q})$ using grasp $\boldsymbol{g}_1$
    if MotionPlan successful **then**
        return $\boldsymbol{t}$ + Connect$(\boldsymbol{q},\ \{\boldsymbol{g}_i\}_{i=2}^{n_{\text{g}}},\ \boldsymbol{q}_{\text{t}})$
    end if
end for
return failure

SampleIntersection $(\boldsymbol{g},\ \boldsymbol{g}')$:

One of $\boldsymbol{g}$ and $\boldsymbol{g}'$ must be bimanual. Assuming $\boldsymbol{g}$.
$S \leftarrow \{\}$
while S contains less than k elements **do**
    $\boldsymbol{x} \leftarrow$ Sample pose for object
    $\boldsymbol{q} \leftarrow$ Solve IK with object at $\boldsymbol{x}$ and grippers at $\boldsymbol{g}$
    if $\boldsymbol{q}$ is stable against gravity with both $\boldsymbol{g}$ and $\boldsymbol{g}'$ **then**
        Add $\boldsymbol{q}$ to S
    end if
end while
return S

Experimental Setting:

We applied our planners to the Baxter Robot from Rethink Robotics for real robot experiments. Baxter has two 7-DOF manipulators, each equipped with a parallel plate gripper. We tested the planners in an OpenRAVE environment (Diankov & Kuffner, 2008) for simulated experiments.

For Alg. 1, we used the NetworkX (Hagberg *et al.*, 2008) for graph construction and search, and BiRRT (Kuffner Jr & LaValle, 2000) for motion planning. In our implementation of Alg. 1, we set the maximum number of planning attempts to be 3 for the procedure PlanRegrasp and the number of samples to be 20 for the procedure SampleIntersection.

We measured the grip force/torque limits (as explained previously in Sec. 4.2) of the Baxter grippers on foam boards. Specifically, along each principal axis, we applied an incremental amount of forces and torques on the foam board gripped by a Baxter gripper, to find the point when the object started to slide between the parallel gripper plates or when the object tilted more than 5^{o} due to finger structure deformation. In this way, we tested the limits as $\boldsymbol{f}_{\text{g},i}^{+} = [13\,N, 40\,N, 100\,N, 0.5\,Nm, 0.1\,Nm, 0.15\,Nm]$ and $\boldsymbol{f}_{\text{g},i}^{-} = [-13\,N, -40\,N, -13\,N, -0.5\,Nm, -0.1\,Nm, -0.15\,Nm]$[1].

We implemented the planners on three types of forceful operations, cutting, puncturing and drilling on rigid foam boards. We discuss how we model the operation force of these forceful operations, particularly with the existence of force deviations previously in Sec. 4.1 and in accordance formulate the stability check in Sec. 4.2. We present a series of experiments and analysis in Sec. 4.4 to capture the force distributions of these forceful operations, with extracted operation models used in this chapter.

[1] Along the $+z$ direction, the object can rest against the gripper palm, therefore the planner adopted a large force limit ($100\,N$) for P_z^{+} during the stability check.

Table 5.1: Numbers of regrasps (with standard deviations in parentheses) of three planners on three categories of tasks.

	Random-Puncturing	V-Puncturing	Drilling&Cutting
Random	19.7(0.7)	52.9(10.1)	5.8(2.1)
Greedy	8.2(2.1)	5.3(1.9)	3.1(0.8)
Min-regrasp	5.4(1.3)	1.6(0.6)	2.0(0.5)

Experiments Overview:

We conducted two categories of experimental studies, including:

- Simulated Experiments: We tested our planners on a variety of forceful tasks to verify their performance in minimizing the number of regrasps, planning stable regrasps and time efficiency (Sec. 5.3.1);
- Real Experiments: We did a set of real human-robot experiments to further study the feasibility of our planners in the real implementations. We used a graphical user interface presented previously in Sec. 4.3 for task specification in these experiments (Sec. 5.3.2).

5.3.1 Analysis of Planning Performance in Simulated Experiments

In simulated experiments, we created three categories of forceful tasks:

- Random-Puncturing: Each task contains 10 puncturing operations randomly distributed on the surface of a foam board. An example is shown in Fig. 5.7;
- V-Puncturing: Each task consists of 40 puncturing operations along two random line segments meeting at a common point. An example is shown in Fig. 5.8 and 5.9;
- Drilling&Cutting: Each task involves four drilling operations and a subsequence of cutting operations as shown in Fig. 5.12.

We generated 100 random forceful tasks for each category above.

Analysis of Number of Regrasps:

First, we compared the performance of our planners, *min-regrasp* and *greedy*, with a *random* planner in reducing the number of regrasps. The random planner acts as a baseline approach. For the first forceful operation, the random planner performs sampling in the configuration space until it finds the first stable configuration against the operation. Then, for any subsequent operations, it first checks whether the current configuration is still stable. If not, it falls back to random sampling.

Table 5.1 shows the average number of regrasps generated by the three planners over 100 random forceful tasks. For the random-puncturing tasks, the random planner generates almost a new grasp and

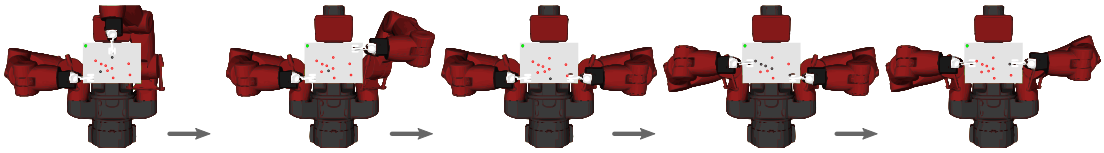


Figure 5.7: A grasp sequence by the min-regrasp planner for a random-puncturing task. The dark points indicate the puncturing operations applied during the current grasp. The arrows indicate regrap actions.

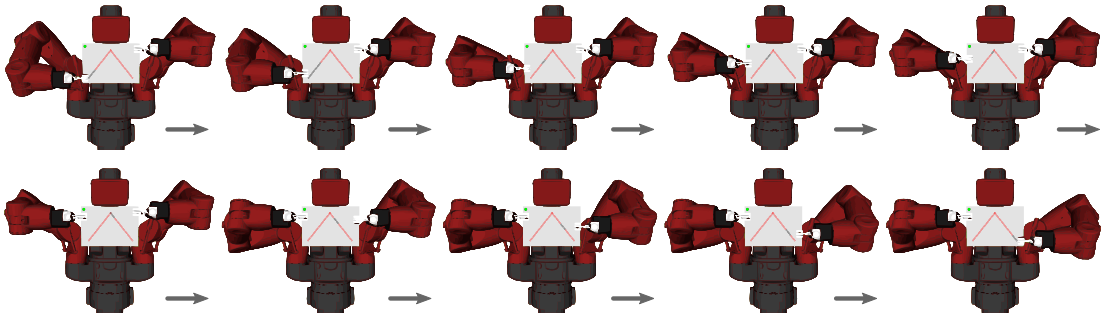


Figure 5.8: A grasp sequence by the greedy planner for a V-puncturing task. The dark points indicate the puncturing operations applied during the current grasp. The arrows indicate regrap actions.

thus one bimanual regrasp for every forceful operation (maximum 20 regrasps for 10 operations). The min-regrasp dramatically reduces the number of regrasps (5.4 regrasps on average for 10 operations, an example solution is shown in Fig. 5.7). The greedy planner also performs well in terms of reducing regrasps (8.2 regrasps on average).

Similarly, for the V-puncturing tasks, the random planner generates plans with a large number of regrasps (52.9 regrasps for the 40 operations of a V-puncturing task on average), while the min-regrasp planner just needs 1.6 regrasps on average (an example solution is shown in Fig. 5.9). The greedy planner shows a much better performance compared with the random planner, but still worse than the min-regrasp planner. For example, as shown in Fig. 5.8, one solution generated by the greedy planner requires the grippers to climb along the edges of the board up and down frequently to follow the movement of the puncturing operations, while the min-regrasp planner comes up with a plan of just two regrasps in Fig. 5.9. Similar results can also be found for the drilling&cutting tasks.

We also counted the number of samples the random planner needed before it found a stable grasp. On average, the random planner needed **41.1** samples for each forceful operation above, showing that planning is necessary, since random grasps have little chance of being feasible.

Our planners are generalized to common objects, not limited to grasping only rectangular objects. To demonstrate this, we tested the min-regrasp planner with a sequence of 40 circular puncturing operations applied on a round board. A plan with three grasps (two regrasps) is shown in Fig. 5.10.

Analysis of Planning Stable Regrasps:

We also tested the performance of our planner on *light* and *heavy* objects respectively. For light objects, the robot can perform regrasps by directly releasing and re-placing its grippers, whereas the robot might

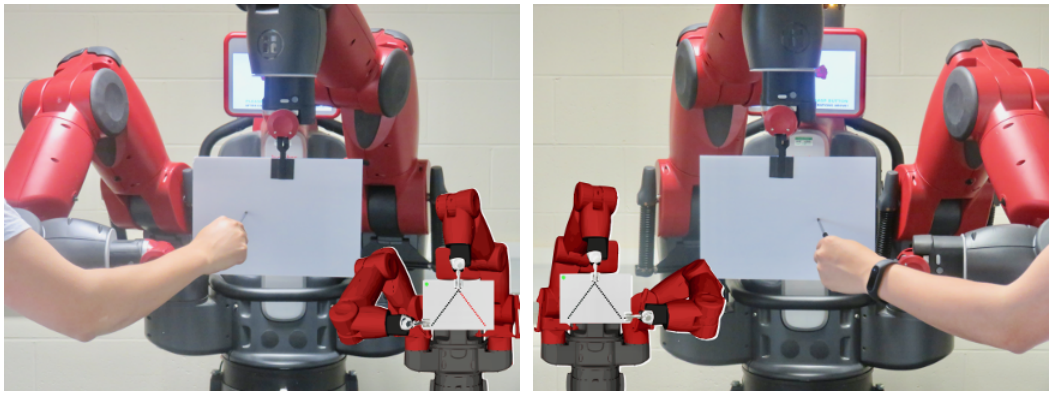


Figure 5.9: A plan by the min-regrasp planner for a V-puncturing task which contains two regrasps.

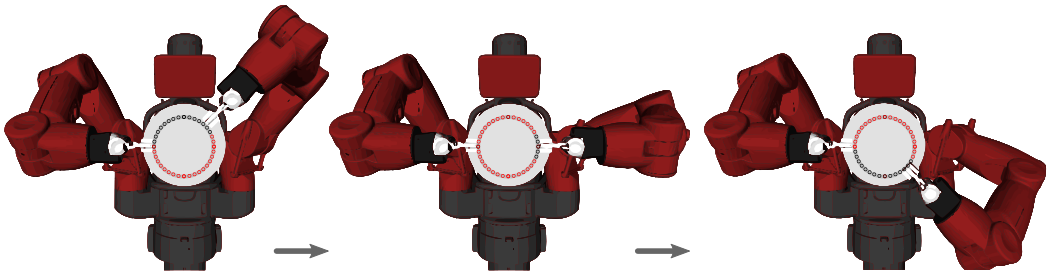
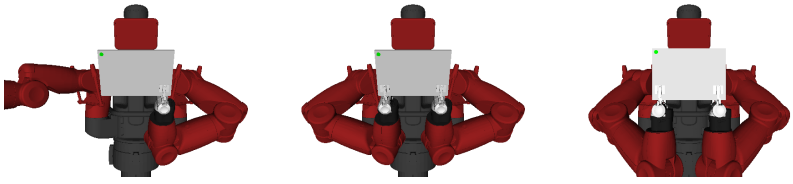


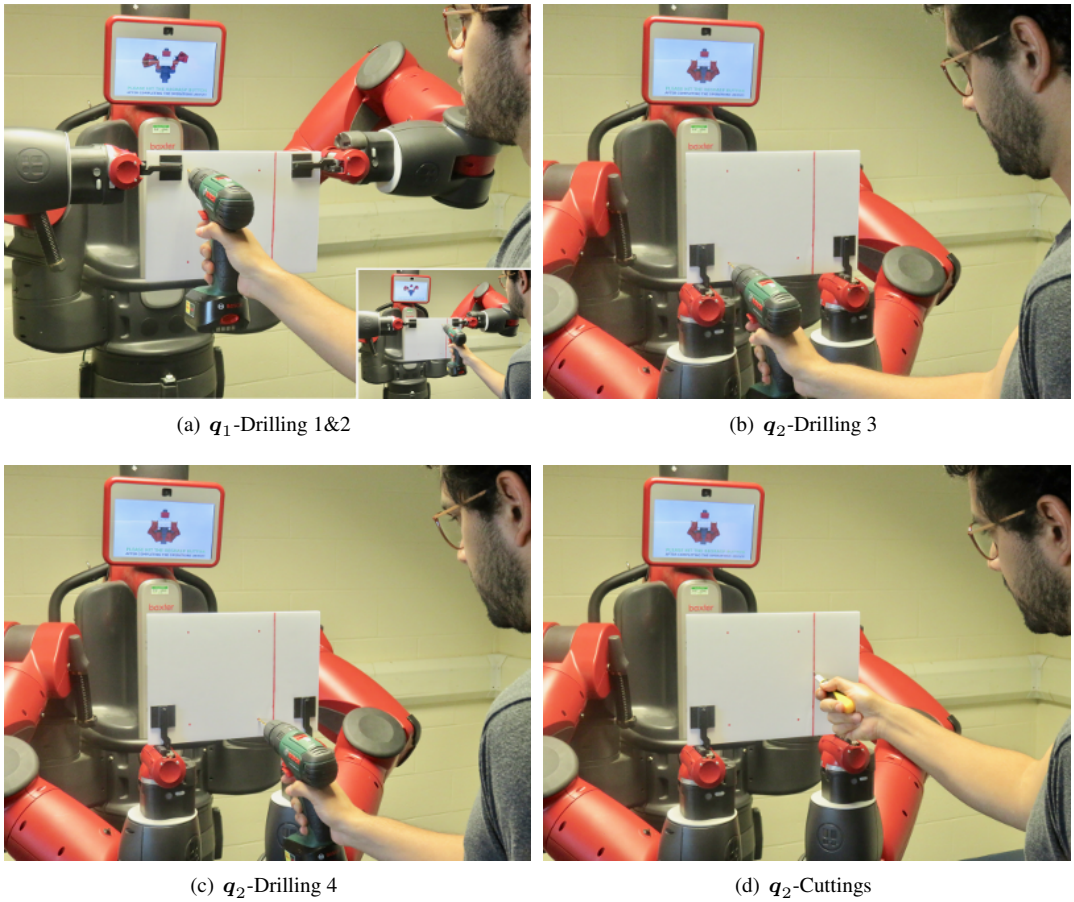
Figure 5.10: A grasp sequence by the min-regrasp planner for 40 circular puncturing operations on a round board.

(a) Start configuration (b) Intermediate configuration (c) Release (d) Regrasp



(e) Release (f) Regrasp (g) Target configuration

Figure 5.11: Regrasping a heavy object: The robot moves the heavy object to some intermediate poses before regrasping.



(a) $\boldsymbol{q}_1$-Drilling 1&2

(b) $\boldsymbol{q}_2$-Drilling 3

(c) $\boldsymbol{q}_2$-Drilling 4

(d) $\boldsymbol{q}_2$-Cuttings

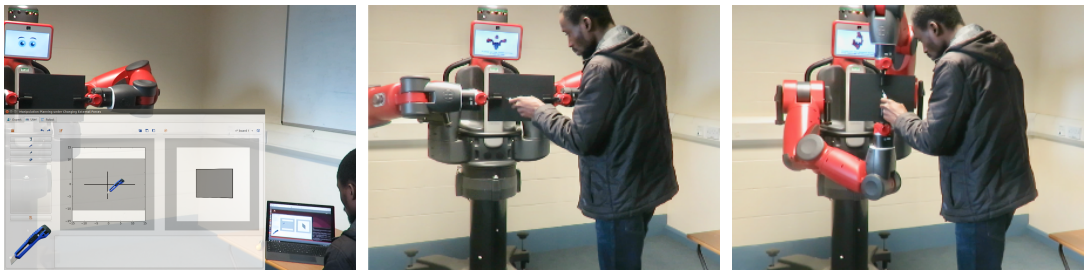
Figure 5.12: Human-robot collaboration - A grasp sequence by the min-regrasp planner for the Drilling&Cutting task.

need to move a heavy object to certain intermediate poses before regrasping. Similarly, we ran the planner on the 100 forceful tasks as discussed above.

Fig. 5.11 shows an example regrasp sequence to regrasp a heavy object. For a light object, the robot can stably grasp and move the object using just a single gripper at most reachable configurations. Thus, mostly, the robot can directly release off to regrasp the object, without the need of reorienting it to intermediate configurations. However, for a heavy object, as discussed previously, the object may slip down between gripper fingers if the robot directly releases one gripper. That is, the robot needs to move it to intermediate configurations at which one single gripper is still enough to keep the object stable. In Fig. 5.11, the robot first transfers the object to configurations in Fig. 5.11(b) and 5.11(d) before releasing one gripper. After releasing, most object weight will be resisted by the forces arising from gripper finger structure as shown in Fig. 5.11(c) and 5.11(e), which are much larger than the frictional forces between the object and finger surfaces.

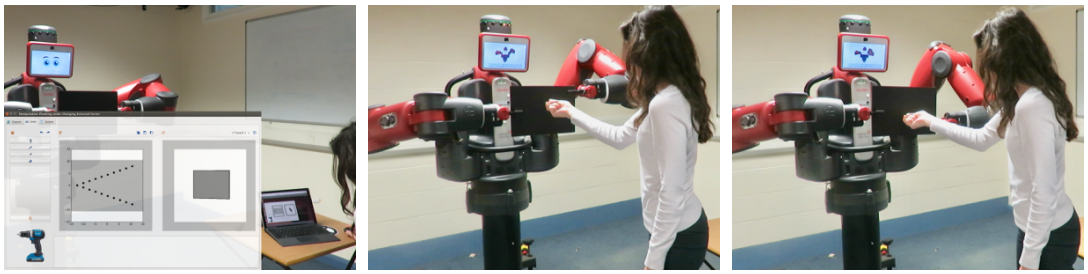
Table 5.2: Planning time for both heavy and light objects. Times are in seconds. Standard deviations are in parentheses.

	Random-Puncturing			V-Puncturing			Drilling&Cutting		
	StabSeq	SampInt	Connect	StabSeq	SampInt	Connect	StabSeq	SampInt	Connect
heavy	1.8(0.1)	12.6(0.9)	299.1(40.3)	5.1(0.5)	3.3(0.3)	77.2(11.5)	0.7(0.1)	3.5(0.2)	94.1(16.2)
light	1.6(0.2)	\	107.5(10.1)	4.9(0.6)	\	29.8(5.6)	0.8(0.1)	\	47.5(7.8)



(a) Task Specification (b) q_1-Horizon. cutting (c) q_1-Vertical cutting

Figure 5.13: Human-robot collaboration - A cross cutting task.

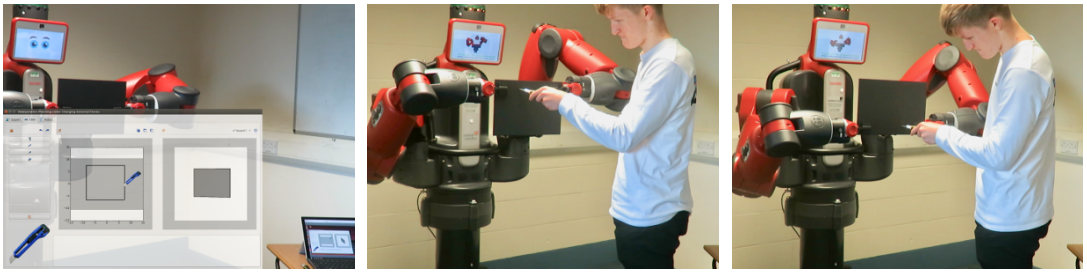


(a) Task Specification (b) q_1- Upper holes (c) q_2-Lower holes

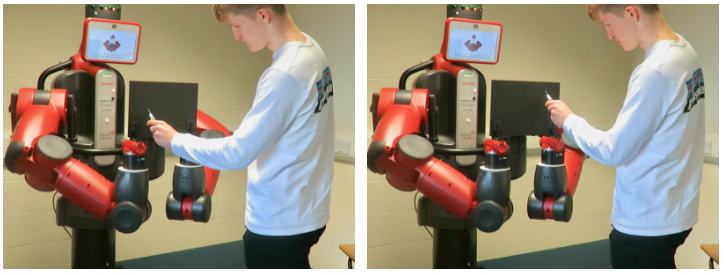
Figure 5.14: Human-robot collaboration - A V-puncturing task.

Analysis of Planning Time:

Table 5.2 shows the average planning time each layer of the planner takes, including time for generating stable sequences (StabSeq for short in Table 5.2), time for generating and searching the grasp graph combined with sampling intersections (SampInt, for short) and motion planning (Connect, for short). As the table shows, most time is spent on motion planning, while the time for planning stable configuration sequences and sampling intersections is negligible. Planning for the heavy objects takes much longer time since finding stable regrasp configurations and motion trajectories is more difficult.



(a) Task Specification (b) $\boldsymbol{q}_1$-Cutting top side (c) $\boldsymbol{q}_2$-Cutting bottom side



(d) $\boldsymbol{q}_3$-Cutting left side (e) $\boldsymbol{q}_3$-Cutting right side

Figure 5.15: Human-robot collaboration - A square cutting task.

Computational Complexity:

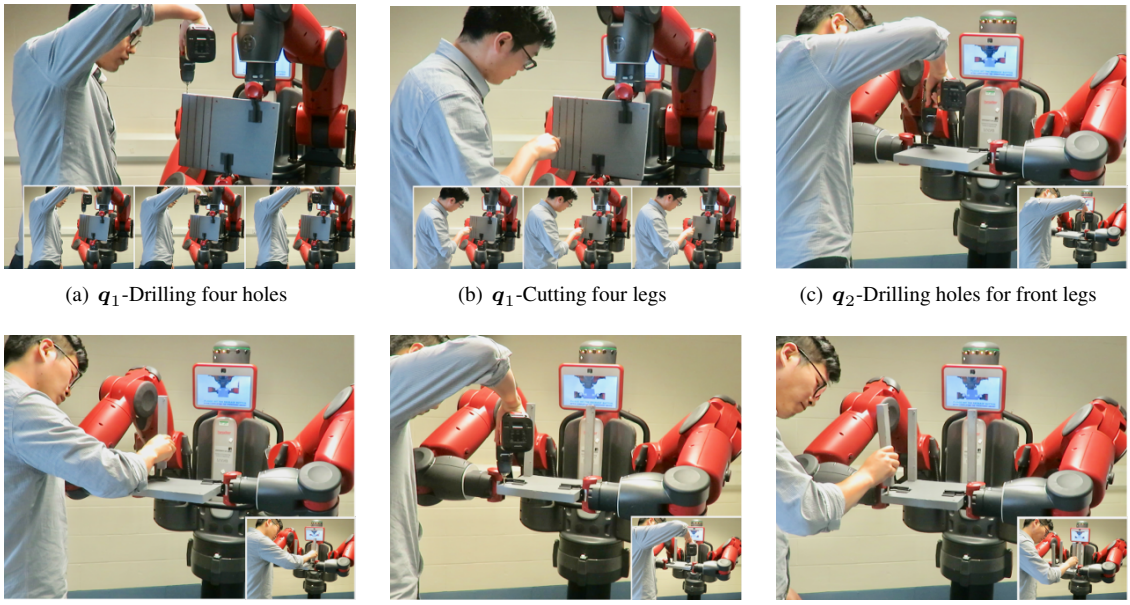
Suppose a forceful task consisting of a sequence of M forceful operations and a set of N sampled configurations. According to the definition of operation graph, the number of stability checks required to build an operation graph for the forceful task is bounded by $O(MN)$. Furthermore, the number of different configurations involved in a manipulation plan from searching the operation graph is bounded by $O(M)$. Suppose the robot has two manipulators, then the number of general motion plannings required to execute a manipulation plan is bounded by $O(2M)$.

5.3.2 Analysis of Planning Performance in Real Human-Robot Experiments

We did a variety of real robot experiments to further verify the feasibility of our system, with a recorded video which can be seen from the link `https://youtu.be/X6eTlqSKiKk`.

Fig. 5.1, 5.9 and 5.12 show the real human-robot implementations of the forceful tasks discussed above. Fig. 5.16 shows a solution generated by the min-regrasp planner for the table assembly task in Fig. 3.2, which consists of a large sequence of drilling, cutting and inserting operations. As shown, the solution involves only three different grasp configurations.

We also performed 10 human-robot experiments using a graphical user interface introduced in Sec. 4.3. Before these experiments, the 10 human participants were fully explained the usage of the interface and the robot system. Then they specified and performed their customized forceful tasks with the assistance of the interface as explained in Sec. 4.3. During the experiments, we regarded an in-



(a) q_1-Drilling four holes (b) q_1-Cutting four legs (c) q_2-Drilling holes for front legs

(d) q_2-Inserting front legs (e) q_3-Drilling holes for back legs (f) q_3-Inserting back legs

Figure 5.16: A solution by the min-regrasp planner for the table assembly task in Fig. 3.2.

teraction as failure if the interaction had any unexpected interruption, e.g. unstable operations due to inappropriate grasp. Fig. 5.13-5.15 show three of these experiments. Among the ten experiments, nine interactions succeeded with a small number of regrasps changing from 1 to 4. One interaction failed due to a collision between the robot gripper and the object during regrasping, which can be seen from time 13:18 to 13:24 in the linked video. This is mainly because of the uncertainty in the robot system and can be improved by the automatic perception of the object and system motion.

We also collected the interaction time of each part during interactions. Over the 10 experiments, on average, the Task Specification took 39.5(3.5) s (standard deviation is in parentheses). The Manipulation Planning took 44.5(9.3) s and the Fabrication took 191.3(44.5) s.

Limitations:

Our experiments show that the planning time for forceful tasks can still take tens of seconds, while the time efficiency of our planners is limited mostly by the speed of the low-level constrained motion planners. This leaves room for improvement in future work to either speed up these individual motion plans, or to reduce the number of such motion plan queries, i.e. the number of regrasps.

We also realized the limitation of regrasp minimization: even though minimizing the number of regrasps can minimize the task interruptions and human waits in the context of HRC, optimizing only this metric is not sufficient to achieve effective object manipulation under changing external forces. For

example, human factors, e.g. comfort and preference, should also be considered into planning.

In the next chapter, we continue to discuss the problem of planning collaborative robot behaviours, but focus on the exploitation of environmental contacts, together with more general robot contacts, in object manipulation under changing external force.

Chapter 6

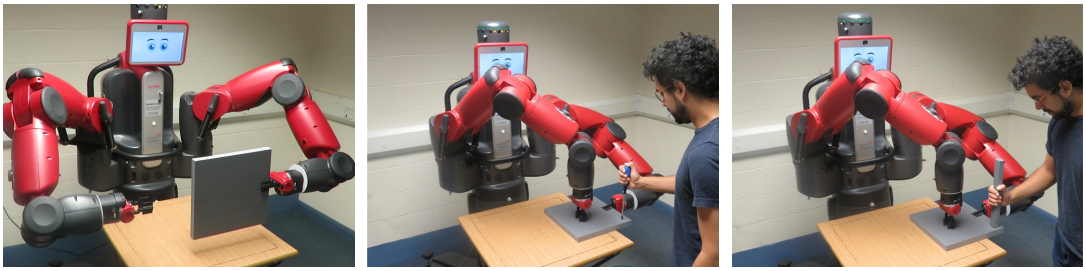
Manipulation Planning under Changing External Forces—Environmental Contacts

0In this chapter, we still focus on addressing manipulation planning problem to keep an object stable under changing external forces. Particularly, in addition to grasp contacts which have been studied in Chap. 5, we aim at enabling robots to exploit deliberate object contacts with rigid structures in the shared environment and other robot contacts, to further boost their capacity of manipulating objects under changing external forces.

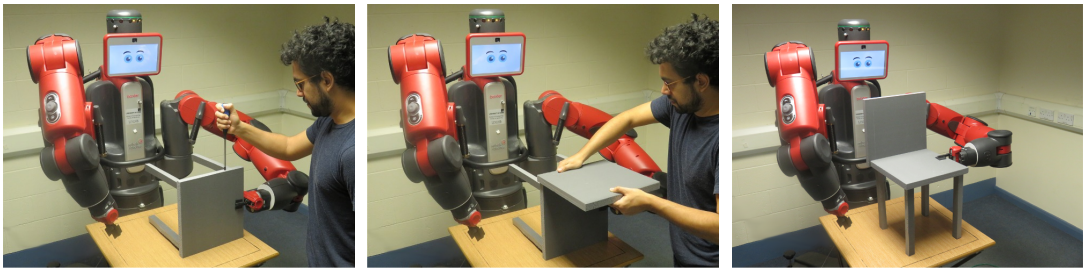
Take the example in Fig. 6.1, where a human and a robot collaborate to assemble a chair. Throughout the chair assembly task, the human applies changing external forces on the chair sub-assemblies via a sequence of drilling and (peg/leg) inserting operations. The robot is supposed to assist the human by moving and keeping the chair assemblies stable as these forceful operations are applied onto them.

The robot itself lacks the capacity of holding the chair sub-assemblies stable against the *varying* and *strong* sequential operation forces. However, the shared environment, particularly the various structures in the environment, provide robots with the chance of performing such forceful tasks, despite its innate incompetence in dealing with large external forces. Specifically, as shown in the Fig. 6.1, by resting the chair sub-assemblies on the table surface to exploit supports from deliberate object contacts with the environment (e.g. the table surface in Fig. 6.1) and the robot (e.g. robot grasps or gripper pressing as shown in Fig. 6.1(b)), the robot succeeds in stabilizing the chair sub-assemblies under the human-applied changing external forces.

This suggests that rather than a constraint, the environment can be an opportunity, particularly for object grasping and manipulation under external disturbances. Motivated by such potential of using environmental contacts in object manipulation, we present a planner that exploits the stabilization capabilities of both environment and robot within a unified planning framework. The planner allows robots



(a) Initial configuration (b) Drilling four holes (c) Inserting four legs



(d) Driling four holes (e) Inserting chair back (f) Task complete

Figure 6.1: The robot holds chair sub-assemblies stable under a sequence of forceful operations, by exploiting deliberate object contacts with both the environment (e.g. a table surface) and robot (e.g. pressing or grasping).

to use deliberate environmental and robot contacts to keep objects stable under changing external forces, while with explicit consideration of manipulation stability and efficiency. Specifically, the planner addresses three major challenges:

First, the planner identifies appropriate object contacts with the environment and/or robot, relying on which the object and robot can stay stable under forceful operations. This requires the planner to choose *where* to move and position the object w.r.t. the environment and robot, particularly among a large number of available contact configurations. For example, in the configuration shown in Fig. 6.1(b), the object-table contact and the object-robot contact (the gripper presses on the object) together can stabilize the chair-top under the shown drilling operation. However, if the gripper is removed from the object, the chair-top would possibly flip over and even slip down from the table due to an imbalanced torque incurred by the operation force.

Second, to improve manipulation efficiency, the planner reasons over a large number of available contact configurations to produce an optimal solution requiring a minimal number of contact adjustments, i.e. configuration changes, and therefore less robot motion during the object manipulation. For example, in the configuration shown in Fig. 6.1(b) and 6.1(c), the chair-top is stable under the first four drilling and four following leg-inserting operations. Then at the configuration shown in Fig. 6.1(d), the object is stable under the following four drilling and four peg-inserting operations. Therefore, in total, there are only two different contact configurations involved in the displayed solution and only one con-

figuration change. Finding such an efficient manipulation plan requires the planner to decide on *when* to change the object contact with the environment and/or robot as forceful operations are applied onto the object.

Finally, for each change in the object contacts, the planner needs to decide on *how* to move the object to implement the configuration change, e.g. from the one in Fig. 6.1(c) to the one in Fig. 6.1(d).

To address the above challenges, the planner starts with searching for a sequence of contact configurations, i.e. robot configuration and object pose w.r.t. the environment, which can keep the object stable under the sequential external forces. The planner minimizes the number of different configurations in the sequence, such that the robot can move the object as minimally as possible throughout the task. Then, the planner generates continuous motion trajectories to connect these planned configurations. We describe details of this hierarchical planning approach in Sec. 6.2.

Note that a key source of computational cost in the planning process comes from the stability check and the explosion of available contact configurations: to find a minimal sequence of contact configurations to keep an object stable under changing external forces, the planner explores from a large set of candidate configurations which act as representatives of the augmented high-dimensional configuration space. What is more, each such candidate configuration needs to be separately checked for stability against all involved external forces. Given the frictional constraints at contact points, such a stability check takes the form of constrained optimization, a computationally expensive procedure.

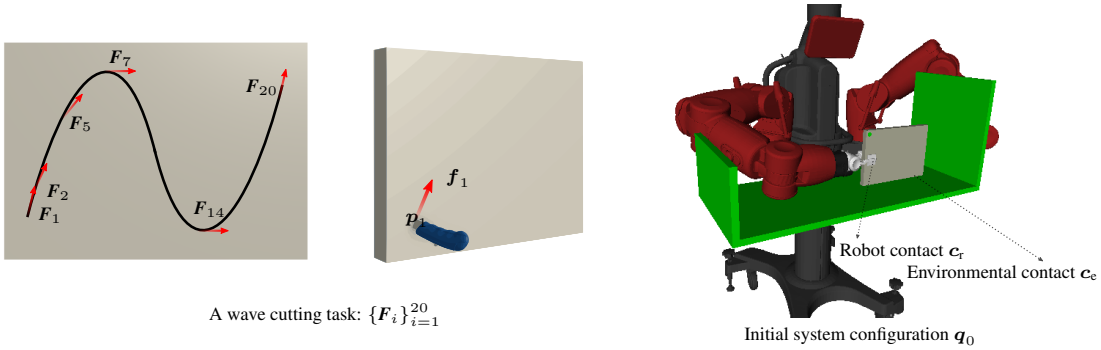
In this regard, as another contribution, we propose a novel strategy to efficiently perform stability checks of a large number of configurations. We introduce a concept of *containment* relationship among different contact configurations based on their capabilities of resisting external wrenches, which the planner can use to quickly compare the stabilization capabilities among different configurations and accordingly cut off redundant stability checks involved for the set of candidate configurations. We provide details of the containment-based stability check in Sec. 6.3.

6.1 Problem Formulation

This section briefly outlines the manipulation planning problem under changing external forces but using environmental contacts, i.e. another interpretation of Def. 1 in the context of using environmental contacts for object manipulation under changing external forces.

We use a wave cutting task consisting of 20 cutting operations as shown in Fig. 6.2(a) to illustrate the problem. For such a forceful task $\{\boldsymbol{F}_i\}_{i=1}^{m}$, as stated in Def. 1, the planner generates a sequence of system configurations $\{\boldsymbol{q}_j\}_{j=1}^{n}$ and motion trajectories $\{\boldsymbol{t}_j\}_{j=1}^{n}$ to keep the object stable under changing operation forces as shown in Fig. 6.2(b). Each configuration in $\{\boldsymbol{q}_j\}_{j=1}^{n}$ corresponds to a combination of environmental and robot contacts providing resisting wrenches onto the object against a subsequence of forceful operations in $\{\boldsymbol{F}_i\}_{i=1}^{m}$.

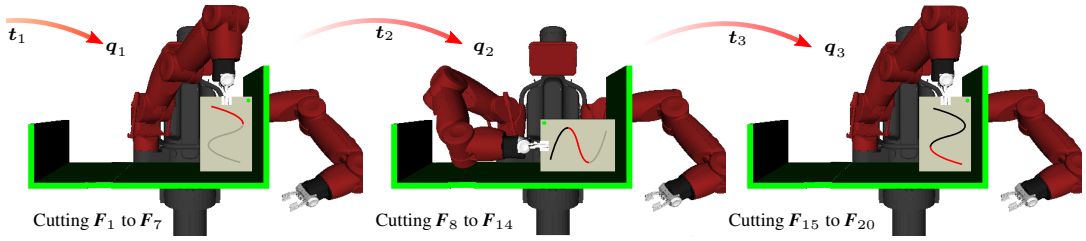
Further, as previously referred to as ***configuration change minimization*** in Chap. 5, the planner minimizes the need of changing object contacts in the solution for the sake of manipulation efficiency. For example, as shown in Fig. 6.2(b), the planner uses only three contact configurations to keep a board



A wave cutting task: $\{F_i\}_{i=1}^{20}$

Initial system configuration q_0

(a) Inputs: A wave cutting task $\{F_i\}_{i=1}^{20}$ and the initial system configuration q_0.



Cutting F_1 to F_7 | Cutting F_8 to F_{14} | Cutting F_{15} to F_{20}

(b) Outputs: A sequence of system configurations $\{q_j\}_{j=1}^{3}$ and motion trajectories $\{t_j\}_{j=1}^{3}$.

Figure 6.2: The planner exploits deliberate environmental contacts to keep an object stable under changing external forces.

stable under 20 wave cutting operations.

6.2 Planning Approach

This section explains in detail the overall planning procedure.

6.2.1 Manipulation Planning Using Operation Graph

Similarly to our previous work in Chap. 5, formally, the planner takes a hierarchical planning framework. At the high-level it builds an operation graph over a sequence of forceful operations $\{F_i\}_{i=1}^{m}$ and a set of candidate contact configurations $\mathcal{Q}_c$. Then, the planner searches along the operation graph for a minimal sequence of configurations $\{q_j\}_{j=1}^{n}$ that are stable against the forceful operations $\{F_i\}_{i=1}^{m}$. Then, the planner generates robot motion trajectories $\{t_j\}_{j=1}^{n}$ to connect these configurations. We provide pseudo-code of the planner in Alg. 2.

Operation Graph: Our planner starts by building an operation graph, using a set of checked-stable comfiguration from among a set of sampled condidate configurations Q_s. The set Q_s acts as a rep-

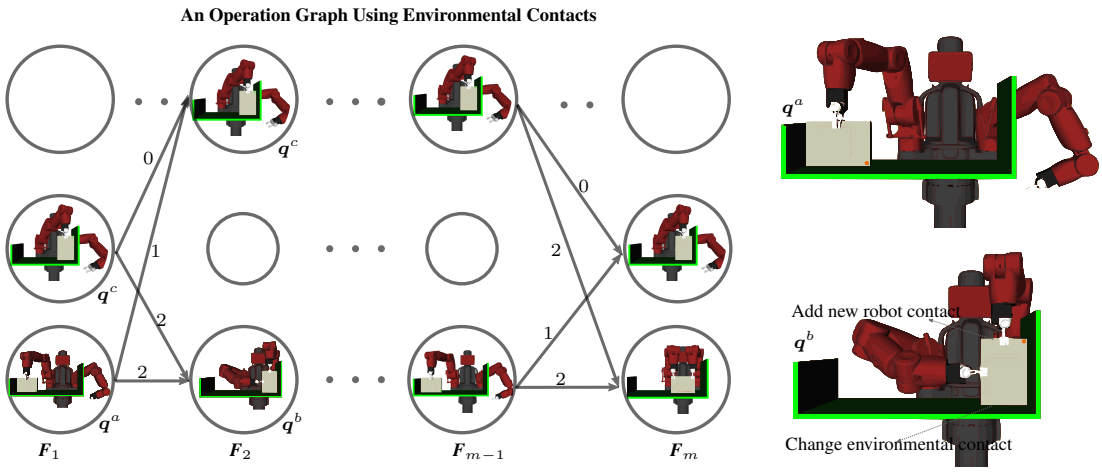


Figure 6.3: We build an operation graph to search for efficient manipulation solutions with a minimal number of configuration changes. We adopt a more fine-grained weighting scheme: from the node $\boldsymbol{q}^a$ to the node $\boldsymbol{q}^b$, the object contact with the environment changes. The robot contact changes by adding one more gripper on the object (The other is held constantly). Thus we weight the link from $\boldsymbol{q}^a$ to $\boldsymbol{q}^b$ in the operation graph as 2.

resentative of the augmented configuration space, and can be prespecified, e.g. using existing contact generators (Debus *et al.*, 2004; Lefebvre, 2003).

Similarly, as illustrated in Fig. 6.3, the operation graph a directed acyclic weighted graph. In an operation graph, the i_{th} column of the graph corresponds to the forceful operation $\boldsymbol{F}_i \in \{\boldsymbol{F}_i\}_{i=1}^m$, while the nodes in the i_{th} column represent a subset of configurations in $\mathcal{Q}_{\text{c}}$, which are checked stable against $\boldsymbol{F}_i$ (We discuss how this stability check is performed previously in Sec. 4.2).

Weighting Links in the Operation Graph: We further define a link between every two nodes in neighbouring columns of the graph and weight the link with a weighting scheme related to the number of configuration changes.

As illustrated in Fig 6.3, rather than weighting the link of two different configurations (e.g. from $\boldsymbol{q}^a$ to $\boldsymbol{q}^b$) as one, here we adopt a more fine-grained weighting scheme, computing the weight as the number of total changes in the environmental and robot contacts. For example, in Fig. 6.3, from the node $\boldsymbol{q}^a$ in the first column to the node $\boldsymbol{q}^b$ in the second column, we say the number of configuration changes is two, including one robot contact change (one extra gripper is added onto the object) and one environmental contact change (the contact region changes). Thus, we weight the link with two. However, the weight between the node $\boldsymbol{q}^c$ in the first column and the node $\boldsymbol{q}^c$ in the second column is zero, since they represent the same sample configuration.

Searching the Operation Graph: At this point, using the operation graph, finding a minimal sequence of system configurations $\{\boldsymbol{q}_j\}_{j=1}^n$ stable against $\{\boldsymbol{F}_i\}_{i=1}^m$ is reformulated as a graph search problem. The expected output is a path that starts from one node in the leftmost column for the operation $\boldsymbol{F}_1$ and ends with a node in the rightmost column for the operation $\boldsymbol{F}_m$, with a smallest total weight. By

searching the graph, e.g. using Dijkstra's algorithm, the planner can easily find an optimal solution, i.e. an efficient manipulation plan with a minimal number of configuration changes. In Fig. 6.3, the red path illustrates such a solution.

The procedure PlanStableSequence in Alg. 2 provides pseudo-code of the planning process using an operation graph. The procedure BuildOperationGraph builds the operation graph (line 2) as described above. The procedure GraphSearch (line 3) searches the operation graph to generate a candidate configuration sequence $\{\boldsymbol{q}_j\}_{j=1}^{n}$. Then for every two subsequent pair of the configurations in the sequence (line 5 - 9), the procedure PlanConfigChange attempts to plan motions using general motion planners, e.g. RRT-based planners, to implement the configuration changes.

6.2.2 Finding Stable Configurations for an Operation

As described above, building an operation graph requires the planner to check and find a subset of stable configurations for each forceful operation in $\{\boldsymbol{F}_i\}_{i=1}^{m}$, from a set of sampled candidate configurations Q_{s}. This is achieved by FindStableConfigs in Alg. 1 (line 3 in BuildOperationGraph).

The planner starts from sampling a set of candidate configurations $\mathcal{Q}_{\mathrm{c}}$, which acts as a representative to the high-dimensional composite configuration space (line 1 in PlanStableSequence). The set $\mathcal{Q}_{\mathrm{c}}$ includes a large number of candidate configurations corresponding to a variety of object-environment and object-robot contacts. Given the environment model, the problem of environmental contact generation has been extensively studied in the literature based on geometric computation (Lefebvre, 2003; Xiao & Ji, 2001), learning (Debus *et al.*, 2004) and kinematics simulators (Ma *et al.*, 2018; Pan *et al.*, 2012). Likewise, robot contacts can be computed via general grasp planners, e.g. Miller and Allen (Miller & Allen, 2000). Such techniques lie outside the scope of this work. Our planner is, in fact, agnostic to the contact generation strategy and thus can use any existing method in the literature for this step. In this work, we assume the existence of such a representative contact configuration set $\mathcal{Q}_{\mathrm{c}}$. Later in Sec. 6.4, we explain how we generated such a set for the experiments involved in this work.

Note that, to find efficient manipulation plans that minimize configuration changes, the planner needs a large set of candidate configurations Q_{s}, which, however, makes the procedure FindStableConfigs and as a result the procedure BuildOperationGraph computationally expensive. This is because, given an operation $\boldsymbol{F}$, the procedure requires the planner to perform a separate stability check for each sampled configuration $\boldsymbol{q} \in \mathcal{Q}_{\mathrm{c}}$, while the stability check itself of a single configuration is already a computationally expensive process (as explained in Sec. 4.2). Therefore, after presenting a naive approach in Sec. 6.3.1, we present, in Sec. 6.3.2, a containment-based strategy to implement the procedure FindStableConfigs efficiently.

6.3 Containment-Based Stability Checks

This section introduces a *containment relationship*, which we use to present an efficient strategy to find all configurations in $\mathcal{Q}_{\mathrm{c}}$ that are stable against a forceful operation $\boldsymbol{F}$, i.e. the procedure FindStableConfigs in Alg. 2.

Algorithm 2 Manipulation Planning Using Operation Graph

PlanStableSequence $\left(\{\boldsymbol{F}_i\}_{i=1}^{m}, \boldsymbol{q}_0\right)$:

1: $\mathcal{Q}_{\mathrm{c}} \leftarrow$ Generate a set of candidate system configurations
2: $\mathcal{G}_{\mathrm{O}} \leftarrow$ BuildOperationGraph $\left(\{\boldsymbol{F}_i\}_{i=1}^{m}, \mathcal{Q}_{\mathrm{c}}\right)$
3: $\left\{\boldsymbol{q}_j\right\}_{j=1}^{n} \leftarrow$ GraphSearch $(\mathcal{G}_{\mathrm{O}})$
4: $\left\{\boldsymbol{q}_j\right\}_{j=0}^{n} \leftarrow$ Add $\boldsymbol{q}_0$ to the beginning of $\left\{\boldsymbol{q}_j\right\}_{j=1}^{n}$
5: **for** each subsequent $\boldsymbol{q}_j$ and $\boldsymbol{q}_{j+1}$ in $\left\{\boldsymbol{q}_j\right\}_{j=0}^{n}$ **do**
6: $\quad \boldsymbol{t}_{j+1} \leftarrow$ PlanConfigChange$(\boldsymbol{q}_j, \boldsymbol{q}_{j+1})$
7: $\quad$ **if** PlanConfigChange failed **then**
8: $\quad\quad$ Remove failing edge from graph $\mathcal{G}_{\mathrm{O}}$
9: $\quad\quad$ Go to line 3
10: $\quad$ **end if**
11: **end for**
12: **return** $(\left\{\boldsymbol{q}_j\right\}_{j=1}^{n}, \{\boldsymbol{t}_j\}_{j=1}^{n})$

BuildOperationGraph $\left(\{\boldsymbol{F}_i\}_{i=1}^{m}, \mathcal{Q}_{\mathrm{c}}\right)$:

1: $\mathcal{G}_{\mathrm{O}} \leftarrow \emptyset$
2: **for** each forceful operation $\boldsymbol{F}_i$ in $\{\boldsymbol{F}_i\}_{i=1}^{m}$ **do**
3: $\quad S \leftarrow$ FindStableConfigs$(\boldsymbol{F}_i, \mathcal{Q}_{\mathrm{c}})$
4: $\quad$ **if** $i = 1$ **then**
5: $\quad\quad$ Add S into $\mathcal{G}_{\mathrm{O}}$ as the first column
6: $\quad$ **else**
7: $\quad\quad$ **for** each configuration $\boldsymbol{q}'$ in previous column of $\mathcal{G}_{\mathrm{O}}$ **do**
8: $\quad\quad\quad$ **for** each configuration $\boldsymbol{q}''$ in S **do**
9: $\quad\quad\quad\quad$ $w \leftarrow$ ComputeWeight $(\boldsymbol{q}', \boldsymbol{q}'')$
10: $\quad\quad\quad\quad$ Create a link from $\boldsymbol{q}'$ to $\boldsymbol{q}''$ with a weight w
11: $\quad\quad\quad$ **end for**
12: $\quad\quad$ **end for**
13: $\quad$ **end if**
14: **end for**
15: **return** $\mathcal{G}_{\mathrm{O}}$

6.3.1 Naive Stability Check of Q_{s}

As previously mentioned, given a forceful operation $\boldsymbol{F}$ and a candidate configuration $\boldsymbol{q}$, stability check means checking if the configuration $\boldsymbol{q}$ is stable against the forceful operation $\boldsymbol{F}$, while the procedure FindStableConfigs involves performing stability checks for all configurations in $\mathcal{Q}_{\mathrm{c}}$.

In this context, for the procedure FindStableConfigs, the planner can simply perform a stability check for each candidate configuration $\boldsymbol{q}$, and return a set of stable ones in $\mathcal{Q}_{\mathrm{c}}$ against the forceful operation $\boldsymbol{F}$. We call this implementation the *Naive Stability Check* and provide the pseudo-code of this implementation in Alg. 3.

Nonetheless, as aforementioned, due to the large size of the set Q_{s} and the property of stability check, Alg. 3 can be computationally expensive, which would degrade the planning efficiency of Alg. 2. To address this issue, we propose an efficient strategy in the next section.

Algorithm 3 Naive Stability Check

FindStableConfigs($\boldsymbol{F}$, $\mathcal{Q}_c$):

1: $S = \emptyset$
2: **for** each configuration $\boldsymbol{q}$ in $\mathcal{Q}_c$ **do**
3: Solve the constrained optimization problem in Eq. 4.3
4: **if** Eq. 4.3 has a solution **then**
5: Add $\boldsymbol{q}$ to S
6: **end if**
7: **end for**
8: **return** S

6.3.2 Containment-Based Stability Check of Q_s

Rather than performing excessive stability checks by directly solving optimization problems over all sample configurations in $\mathcal{Q}_c$, we propose a *containment-based* strategy to implement FindStableConfigs efficiently. The algorithm relies on the containment relationship among different configurations to quickly eliminate redundant stability checks. We present pseudo-code of this implementation in Alg. 4.

Containment Relationship: We define the containment among different system configurations over their capability of resisting external forces, i.e. contact wrench space (Borst *et al.*, 2004; Hertkorn *et al.*, 2012). A system configuration $\boldsymbol{q}$ describes the geometric relationships among the object, the robot and the environment, while the contact region(s) corresponding to the configuration indicates its capability of resisting external forces. For example in Fig. 6.4[1], since the environmental contact region $\boldsymbol{c}_e^b$ contains $\boldsymbol{c}_e^a$ and the robot contact $\boldsymbol{c}_r^b$ contains $\boldsymbol{c}_r^a$, any forceful operation resistible by the configuration $\boldsymbol{q}^a$ is also resistible by the configuration $\boldsymbol{q}^b$. That is, $\boldsymbol{q}^b$ *contains* $\boldsymbol{q}^a$ in terms of the capacity of resisting external wrenches. In this context, we define the containment relationship (described with $\subseteq$) as:

Definition 2 (**Containment Relationship**). Let $\boldsymbol{q}^a$ and $\boldsymbol{q}^b$ be two system configurations, we say $\boldsymbol{q}^b$ contains $\boldsymbol{q}^a$ ($\boldsymbol{q}^a \subseteq \boldsymbol{q}^b$), iff

- $\boldsymbol{c}_e^a \subseteq \boldsymbol{c}_e^b$ and
- $\boldsymbol{c}_r^a \subseteq \boldsymbol{c}_r^b$

where $\boldsymbol{c}_e^a$, $\boldsymbol{c}_e^b$ are the environmental contact regions corresponding to the configuration $\boldsymbol{q}^a$ and $\boldsymbol{q}^b$, and $\boldsymbol{c}_r^a$ and $\boldsymbol{c}_r^b$ are the corresponding robot contacts.

Note that the containment among the environmental contact regions holds only for the same surface contacts, as the friction coefficients may be different for different materials. The containment relationship can be further constrained and defined in more subtle degree, e.g. by comparing environmental contacts and robot contacts in an integrated manner, which, as a trade-off, would result in high computational complexity.

Containment Graph: We then build a directed acyclic *containment graph* $\mathcal{T}$ over all sampled configurations in $\mathcal{Q}_c$ to represent their containments (line 2 in Alg. 4). Specifically, as illustrated in Fig. 6.4(c),

[1] Fig. 6.4 is illustrated in 2D for clarity of presentation. In our implementation, contact regions and containment are defined in 3D.

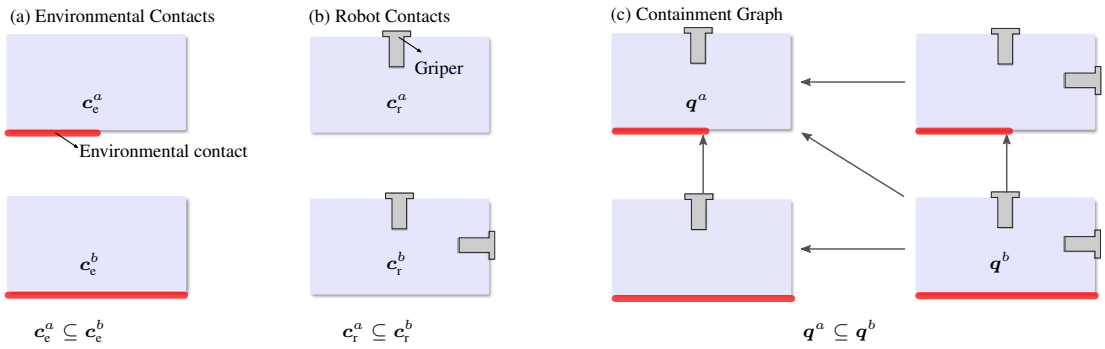


Figure 6.4: We build a containment graph over all system configurations in $\mathcal{Q}_c$ to represent their containments. The red segments illustrate environmental contacts with the object.

each node in the graph represents a contact configuration $\boldsymbol{q} \in \mathcal{Q}_c$. For every pair of nodes, if there exists a containment relationship between them as defined in Def. 2, they are connected with a directed link. For example, the node $\boldsymbol{q}^a$ in Fig. 6.4(c) is contained and thus becomes a successor to the other three nodes.

Stability Check Using Containment Graph: Based on the characteristics of the containment graph, we introduce two properties to simplify the procedure FindStableConfigs.

Property 1 (**Activating Property**). Given a forceful operation $\boldsymbol{F}$ and a containment graph $\mathcal{T}$, if a node $\boldsymbol{q}$ is stable against $\boldsymbol{F}$, then all its predecessors in $\mathcal{T}$ are stable against the operation $\boldsymbol{F}$.

The property can be easily proved via Def. 2. We call this property the '*activating*' property, as based on this property, if a configuration $\boldsymbol{q}$ is checked to be stable against a forceful operation $\boldsymbol{F}$, all its predecessor configurations can be directly 'activated' to be feasible without solving an optimization problem in Eq. 4.3 (line 6-9 in Alg. 4). For example, if $\boldsymbol{q}^a$ in Fig. 6.4(c) is stable against an operation $\boldsymbol{F}$, then all other nodes in the graph can be directly regarded as stable configurations against $\boldsymbol{F}$.

Property 2 (**Blocking Property**). Given a forceful operation $\boldsymbol{F}$ and a containment graph $\mathcal{T}$, if a node $\boldsymbol{q}$ is not stable against $\boldsymbol{F}$, then all its successors in $\mathcal{T}$ are not stable against the operation $\boldsymbol{F}$.

Similarly, this property can also be easily proved via Def. 2. We call this property the '*blocking*' property, as based on it, if a configuration $\boldsymbol{q}$ is checked to be not stable against a forceful operation $\boldsymbol{F}$, all its successor configurations can be directly 'blocked' (line 10-12 in Alg. 4). For example, if $\boldsymbol{q}^b$ in Fig. 6.4 is not stable against an operation $\boldsymbol{F}$, then all other nodes in the graph can be directly regarded as infeasible.

By utilizing the activating and blocking properties, the algorithm does not need to solve a constrained optimization problem for each configuration in $\mathcal{Q}_c$ separately, thus reducing the computational complexity of procedure FindStableConfigs greatly. We proposed experiments to verify the effectiveness of using the containment graph in the overall planning in Sec. 6.4.2.

Algorithm 4 Containment-Based Stability Check

FindStableConfigs($\boldsymbol{F}$, $\mathcal{Q}_c$):

1: $S \leftarrow \emptyset$
2: $\mathcal{T} \leftarrow$ build a containment graph as Fig. 6.4
3: **while** $\mathcal{T}$ is not empty **do**
4: $\quad \boldsymbol{q} \leftarrow$ randomly pick a configuration in $\mathcal{T}$
5: $\quad$ Solve the constrained optimization problem in Eq. 4.3
6: $\quad$ **if** Eq. 4.3 has a solution **then**
7: $\quad\quad S \leftarrow$ Add $\boldsymbol{q}$ and its predecessors in $\mathcal{T}$ to S
8: $\quad\quad \mathcal{T} \leftarrow$ Remove $\boldsymbol{q}$ and its predecessors from $\mathcal{T}$
9: $\quad\quad$ go to line 3
10: $\quad$ **else**
11: $\quad\quad \mathcal{T} \leftarrow$ Remove $\boldsymbol{q}$ and its successors from $\mathcal{T}$
12: $\quad\quad$ go to line 3
13: $\quad$ **end if**
14: **end while**
15: **return** S

6.4 Experiments and Results

This section presents a variety of simulated and real robot experiments, using a Baxter robot in both cases, to validate and quantitatively assess the performance of our proposed planning approach.

Experimental Setting:

The planning approach was implemented in OpenRAVE (Diankov & Kuffner, 2008) with the flexible collision library (FCL) (Pan *et al.*, 2012) for collision checking and contact detection. We used the Scipy.optimize library for the optimization based stability check (Eq. 4.3), the NetworkX (Hagberg *et al.*, 2008) for graph construction and search, and the BiRRT (Kuffner Jr & LaValle, 2000) for motion planning in the procedure PlanConfigChange of Alg. 2.

We implemented the planner on three types of forceful operations, drilling, cutting and inserting on foam boards. We captured experimental data via a force/torque sensor (FT150 from Robotiq) to model the distributions of these forceful operations as discussed in Sec. 4.4.

Generating a Set of Candidate Configurations Q_s:

We sampled three sets of candidate system configurations (line 1 of the procedure PlanStableSequence in Alg. 2) and fed them to the planner.

To obtain such sets with higher sample diversity, we evenly discretized and generated a set of contact regions $\boldsymbol{c}_e$ on the object surface with a fixed step size for environmental contacts, and discretized the object surface as a set of contact points $\boldsymbol{c}_r$ for robot contacts. A combination $(\boldsymbol{c}_e, \boldsymbol{c}_r)$ of such an environmental and robot contact regions defines a contact profile a system configuration $\boldsymbol{q}$ may have. Then, to map the contact regions $(\boldsymbol{c}_e, \boldsymbol{c}_r)$ into a fully-assigned system configuration $\boldsymbol{q}$, we evenly discretized the structure surfaces in the environment into a set of placement positions. At each position, we checked whether there exists a kinematically valid configuration $\boldsymbol{q}$ meeting the contact profile $(\boldsymbol{c}_e, \boldsymbol{c}_r)$.

In this manner, we sampled three sets of candidate configurations with different set size ($|Q_s| = 144,\ 560,\ 1172$) for the following experimental studies.

Table 6.1: Numbers of configuration changes for each task by the baseline and proposed planner with three sample sets of different sizes.

Method	Set Size $\lvert Q_s \rvert$	# of Configuration Changes			
		Task 1	Task 2	Task 3	Task 4
Baseline	144	23	19	20	27
Proposed	144	4	10	7	9
	560	3	4	3	3
	1172	3	1	1	2

6.4.1 Analysis of Minimizing Configuration Changes

First, we assessed the performance of our planner in minimizing the number of configuration changes (consequently, task interruptions) along four different forceful tasks in comparison with a baseline planner.

- Task 1: A *rectangular cutting* task consisting of 20 continuous cuttings as shown in Fig. 6.5 to cut a rectangular piece off from a board. The environment has a flat surface in front of the robot;
- Task 2: A *stool fabricating* task involving cutting four legs (discretized as twenty cutting operations), drilling four holes and inserting four legs in sequence (28 forceful operations in total) as shown in Fig. 6.6. The environment has an L-shaped structure in front of the robot;
- Task 3: A *chair assembly* task consisting of four hole-drillings, four peg-insertings, four leg-insertings, four hole-drillings and four peg-insertings in sequence (20 forceful operations in total) as shown in Fig. 6.1. The environment has only a flat surface in front of the robot;
- Task 4: A *wave cutting* task discretized into 20 cuttings as shown in Fig. 6.2. The environment has a ⊔-shaped supporting structure in front of the robot.

To the best of our knowledge, there exists no planner in the literature directly capable of solving such complex tasks. Existing strategies would, in the best scenario, need to plan each forceful operation individually, neglecting the sequential property that defines a task. We define such a scenario as the *baseline* planner. For each operation in a task, the baseline planner iterates over the available candidate configurations in Q_s until it finds the first stable one.

Table 6.1 summarizes the results of configuration changes by the baseline planner and our proposed planner. As shown in the table, compared with the baseline planner, our planner reduces the number of configuration changes dramatically. Specifically, with $\lvert Q_s \rvert = 144$, for Task 1 (the rectangular cutting task), the baseline planner finds a solution with 23 configuration changes and therefore generates a new configuration for almost every involved operation (20 operations in total). Our planner generates a more efficient solution (Fig. 6.5-Top, Solution A), which involves only four different system configurations and four configuration changes in total. Similarly, Fig. 6.6 shows a solution generated by our planner

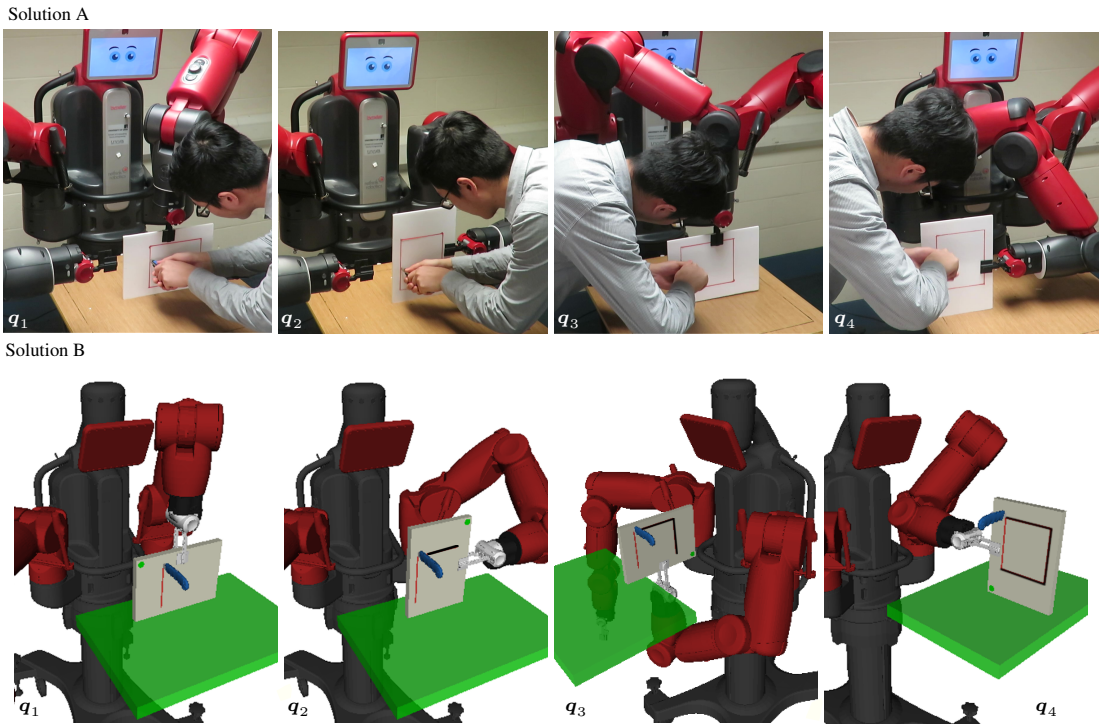


Figure 6.5: Two manipulation plans for a rectangular cutting task (Task 1) consisting of 20 forces. Top: Solution A contains 4 configuration changes, with a robot regrasp (from left arm in $\boldsymbol{q}_2$ to right arm in $\boldsymbol{q}_3$). Bottom: Solution B contains 3 configuration changes.

for Task 2, which involves only one configuration change. Fig. 6.1 and 6.2 show an efficient solution for Task 3 and 4 respectively.

It is also notable that as we increase the number of sampled configurations in Q_s, i.e. the set size $|Q_s|$, the planner may come up with better solutions, that is, manipulation plans with a further reduced number of configuration changes, as shown in Table 6.1 for all tasks. To better illustrate the difference, take for instance Fig. 6.5-Bottom which shows a different solution for Task 1 generated by our planner but with $|Q_s| = 1172$. As shown, when the set size $|Q_s|$ increased, the planner came up with a more efficient solution: in Solution A (real robot experiments), the planner requires a regrasping (from right arm in $\boldsymbol{q}_2$ to left arm in $\boldsymbol{q}_3$) whereas in Solution B (from the simulator)[2] the robot is capable to perform the task only with the right hand.

6.4.2 Analysis of Planning Efficiency

We further verified the performance of our planner in terms of time efficiency. More specifically, we compared our planner (Alg. 2) using the naive stability check in Alg. 3 (Plan-N, for brevity) with the

[2]Note both scenarios were implemented in simulation and in the real robot, but they are presented separately in Fig. 6.5 to aid the discussion.

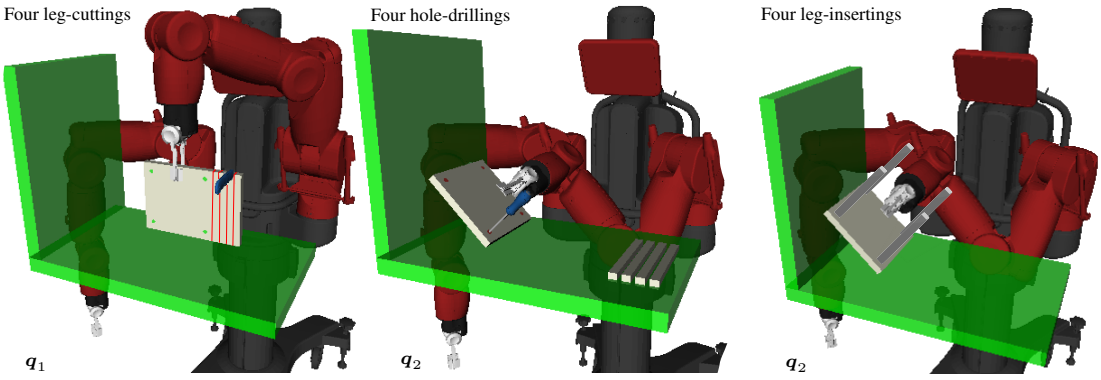


Figure 6.6: A manipulation plan for a stool fabricating task (Task 2) consisting of 20 forces. The solution contains 1 configuration change (environmental contact from q_1 to q_2).

Table 6.2: Average time (s) for building the operation graph and the overall planning time in parentheses for each task over 50 runs.

$\lvert Q_s \rvert$	Task 1		Task 2		Task 3		Task 4	
	Plan-N	Plan-Cont	Plan-N	Plan-Cont	Plan-N	Plan-Cont	Plan-N	Plan-Cont
144	198.5(223.1)	**20.1**(37.7)	280.1(335.0)	**31.0**(81.9)	211.8 (246.8)	**22.9**(56.7)	201.3(256.1)	**22.2**(74.6)
560	773.2(792.0)	**69.7**(86.6)	1100.6(1121.1)	**77.8**(96.0)	821.5(837.5)	**72.4**(87.9)	799.1(820.0)	**70.8**(91.3)
1172	1551.2(1570.1)	**98.3**(115.4)	2087.8(2092.5)	**135.8**(138.2)	1611.7(1615.4)	**112.0**(115.8)	1605.4(1616.7)	**109.9**(120.0)

(same) planner but using the containment-based strategy in Alg. 4 (Plan-Cont, for brevity)[3]. Note that the two planners differ in the strategy of implementing the procedure FindStableConfigs in Alg. 2, i.e. building the operation graph which is the most computationally complex procedure in Alg. 2, as it involves perform stability checks over all forceful operations and candidate configurations.

Taking the same four tasks as previous, Table 6.2 summarizes the average time for building the operation graph over 50 runs for each task, with the *total planning time* listed in parentheses. As shown, the containment-based strategy (bold in Table 6.2) increases the planning efficiency significantly for about ten-fold compared to a more straightforward, but naive approach. For example, for Task 2 with $|Q_s| = 1172$, it takes about 2087.8 s for the Plan-N to build the operation graph compared to 135.8 s by the containment-based planner (an improvement of about $15\times$). Similar analysis can also be made for the other three tasks and configuration sets.

Table 6.2 also highlights the time cost of building the operation graph in the total planning time (in parentheses). As shown in the Table, with lower values of $|Q_s|$, i.e. smaller configuration sets, the planner would generate solutions with more configuration changes, and therefore more motion planning iterations would be required. This results in a larger difference between the time for building the operation graph and the total planning time.

It is also important to mention that the containment-based planner requires extra construction of the containment graph as described in Sec. 6.3.2. Nonetheless, this is a low computational complexity task

[3] In Sec. 6.4.1, the proposed planner refers to Plan-Cont, but both strategies could have been used as they return the same plans and configuration changes.

Table 6.3: Average time (s) for building the containment graph.

$\|Q_s\|$	Task 1	Task 2	Task 3	Task 4
124	0.04	0.06	0.04	0.07
560	0.26	0.40	0.31	0.59
1172	0.97	1.25	1.05	1.52

as illustrated in Table 6.3, which shows the average time for building the containment graph over 50 runs for each task. As shown, the time for building the containment graph increases proportionally to $|Q_s|$, but it still represents less than 1% of the building time required for the operation graph in Table 6.2 and thus negligible in the overall planning. For example, for Task 2 with $|Q_s| = 1172$, it took only 1.05s for the planner to build the containment graph, but 135.8 s to build the operation graph and 138.2 s for total planning.

Based on the above analysis, we can conclude that compared with performing naive stability checks (Eq. 4.3) for all operations and sampled configurations, building the containment graph can greatly improve planning efficiency.

Limitations:

One limitation of this work is the specification of the candidate contact configuration set $\mathcal{Q}_c$. In this chapter, we mainly focus on the robot kinematic and geometric-collision constraints in generating such a configuration set. However, this might result in object configurations which are not acceptable to humans in the context of fHRC due to, e.g. object visibility and reachability. Regarding to this issue, *human preferences* can be integrated into the specification as a preliminarily filter of inappropriate contact configurations. In addition, from the perspective of planning efficiency, a separate process can be created to maintain and update a set of high-quality candidate contact configurations based on human preferences.

Till now, we have discussed the manipulation planning problem using both environmental and robot contacts to keep objects stable under changing external forces, concerning task stability and efficiency explicitly in planning cooperative robot behaviours. In the next chapter, we move our focus to the human partner, further improving robot behaviours for enhanced human comfort in fHRC.

Chapter 7

Planning for Comfortable Forceful Human-Robot Collaboration

This chapter presents our work on planning for human comfort in fHRC. In addition to task stability and efficiency addressed in previous chapters, we aim at further improving robot behaviours by explicitly taking *human comfort* in performing collaborative forceful operations into account, generating robot and object configurations that maximizes the human's *muscular* comfort and *peripersonal-space* comfort.

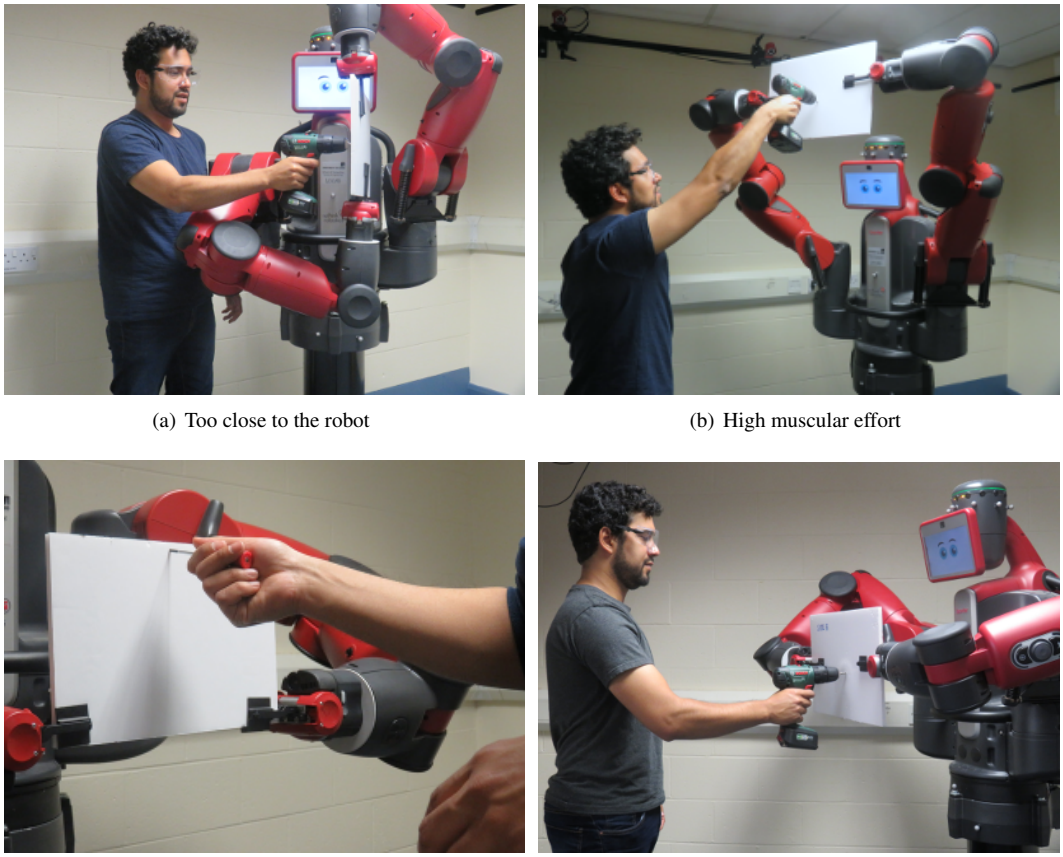
We particularly focus on the context where a robot manipulates an object in close proximity to a human, who applies a collaborative forceful operation, such as drilling and cutting, on the object. An instance is shown in Fig. 7.1, where a human is drilling on a wooden board held by a robot assistant at different configurations.

Theoretically, there may be infinite ways for the robot to grasp and position the board and accordingly for the human to apply the desired forceful operation. Among these candidate configurations, however, a large proportion would pose the human at configurations which are uncomfortable (e.g. the one in Fig. 7.1(b)) and even unsafe (e.g. the one in Fig. 7.1(a)) for the human to apply the operation.

Human comfort and safety in a collaborative forceful operation highly depends on where the human body is positioned relative to the object and robot **and** how the active human arm is configured in applying the required operation force, and therefore, where and how the robot configures its manipulators to grasp and position the target object for the human during the operation. In this sense, to achieve a high level of human comfort in performing a collaborative forceful operation, it is critical for the robot to find appropriate configurations to grasp and position the object which are not only stable against the operation force, but also comfortable for the human.

In this chapter, we propose a planner to empower robots with this capability, which explicitly concerns both human comfort and force stability so as to achieve an enhanced fHRC experience with improved performance from both sides. To quantify human comfort during forceful interaction, we formulate two quality metrics:

First, we formualte a ***muscular comfort*** metric based on the human's muscular features, kinematics



(a) Too close to the robot

(b) High muscular effort

(c) Unstable grasp

(d) Comfortable and stable

Figure 7.1: Human-robot collaboratively drilling on a board.

and the involved forces. This metric quantifies the human's muscular efforts in performing an operation force at a specific arm configuration. For example, it predicts a higher muscular comfort value for the human configuration in Fig. 7.1(a) and Fig. 7.1(d), but a lower muscular comfort value for the human configuration in Fig. 7.1(b), for a same collaborative drilling operation. The muscular comfort focuses on the physical sensation of the human body in applying an operation force at a certain configuration. It predicts and proactively instructs the human to configurations which require less physical joint-torque efforts to perform a forceful operation.

Nonetheless, through our initial experiments, we discovered that optimizing only muscular comfort is not sufficient to guarantee human comfort. For example, while the human in Fig. 7.1(a) may have better muscular comfort in comparison with Fig. 7.1(b), he is dangerously close to the robot and is even obstructed by the robot limbs. Human behaviours vary according to their assessments of robot counterparts while collaborating for forceful operations in close proximity, in particular the level of trust and safety perception (Lasota & Shah, 2015; Lasota *et al.*, 2014). Furthermore, the human and

robot may need to move before, during and after the forceful operation. Therefore, having enough space between them is important. In this regard, to improve human awareness and safety perception, we exploit a concept of the human's ***peripersonal-space comfort***, which will increase the overall spatial distances between body parts of the human and robot. The idea is to allow humans to move and act within their peripersonal space[1] minimizing the perceived risk of robot intervention. Fig. 7.1(d) shows a configuration optimizing both the human's muscular and peripersonal-space comforts: the human is drilling at a physically comfortable pose, with the distance to the robot being large enough to reduce spatial discomfort.

In our previous work introduced in Chap. 5 and 6, from the robot perspective, we mainly focus on searching for efficient and robust plans for object manipulation under changing external force. In that body of research, we assumed the object pose to be prespecified (Chap. 5) or to be any feasible one (Chap. 6) during collaborative forceful operations, and therefore robots can hold objects at any kinematically accessible configuration under the constraint of force stability. This chapter we plan from the human perspective, aiming at generating configurations for robots to position and grasp objects, to guarantee both manipulation stability and human comfort in collaborative forceful operations. In this sense, the work in this chapter can be regarded as a preliminary step towards stable, comfortable and efficient fHRC.

To achieve this, the planner needs as inputs the collaborative forceful operation and the geometrical models for the human, the robot and the target object. It maximizes a cost of human's muscular comfort and peripersonal-space comfort, which generates an optimal configuration for the robot to grasp and position the target object for the human.

7.1 Optimization Overview

This section briefly outlines our optimization framework for improving human comfort in fHRC. Particularly, we are interested in optimizing a combined cost of the human's *muscular comfort* and *peripersonal-space comfort*, to search for an optimal configuration at which the robot can configure its manipulators to grasp an object such that:

- The human can perform a collaborative forceful operation $\boldsymbol{F}$ with a high level of comfort;
- The object can remain stable while the human applies the operation force $\boldsymbol{f}$ onto it.

Given a forceful operation $\boldsymbol{F}$, the planner optimizes a cost Comfort $(\boldsymbol{q}_{\mathrm{r}}, \boldsymbol{p}_{\mathrm{o}}, \boldsymbol{f}, \boldsymbol{p})$ to search for a robot configuration $\boldsymbol{q}_{\mathrm{r}}$ and an object pose $\boldsymbol{p}_{\mathrm{o}}$, such that the object can be stably grasped against the operation force $\boldsymbol{f}$ *and* the human comfort during the operation can be maximized:

[1] Peripersonal space is the space immediately surrounding our body, or the sector of space that closely surrounds a certain body part, in which multi-sensory and sensorimotor integration is enhanced (Bartolo *et al.*, 2014; Rizzolatti *et al.*, 1981).

$$\begin{aligned} \boldsymbol{q}_{\mathrm{r}}{}^{*},\, \boldsymbol{p}_{\mathrm{o}}{}^{*} &= \underset{\boldsymbol{q}_{\mathrm{r}},\, \boldsymbol{p}_{\mathrm{o}}}{\arg\max}\ \mathrm{Comfort}\,(\boldsymbol{q}_{\mathrm{r}},\, \boldsymbol{p}_{\mathrm{o}},\, \boldsymbol{f},\, \boldsymbol{p}) \\ \text{s.t.}\ \ &\mathrm{Is\_stable}\,(\boldsymbol{q}_{\mathrm{r}},\, \boldsymbol{p}_{\mathrm{o}},\, \boldsymbol{f},\, \boldsymbol{p}) \end{aligned} \tag{7.1}$$

The function $\mathrm{Is\_stable}\,(\boldsymbol{q}_{\mathrm{r}},\, \boldsymbol{p}_{\mathrm{o}},\, \boldsymbol{f},\, \boldsymbol{p})$ checks via *static equilibrium* whether the robot at the configuration $\boldsymbol{q}_{\mathrm{r}}$ is able to keep the object stable under the operation force $\boldsymbol{f}$ with the corresponding grasp $\boldsymbol{g}$, which has been discussed in detail as *stability check* in Sec. 4.2.

Eq. 7.1 attempts to maximize a cost of Comfort, which we use to quantity the overall human comfort in performing a forceful operation. Herein we model the cost Comfort with two components: the human's *muscular* comfort and *peripersonal-space* comfort. We assume the existence of corresponding cost functions which quantify both comfort values. Specifically,

– $\mathrm{Muscular}\,(\boldsymbol{q}_{\mathrm{h}},\, \boldsymbol{f},\, \boldsymbol{p},\, \boldsymbol{p}_{\mathrm{h}},\, \boldsymbol{p}_{\mathrm{o}})$ returns a scalar value quantifying the human's muscular comfort in performing a forceful operation $(\boldsymbol{f},\, \boldsymbol{p})$ at a configuration $\boldsymbol{q}_{\mathrm{h}}$ (which is defined by the human body pose $\boldsymbol{p}_{\mathrm{h}}$ and object pose $\boldsymbol{p}_{\mathrm{o}}$ due to the geometric and kinematic coupling). We explain how we formulate this cost in Sec. 7.2.

– $\mathrm{Peripersonal}\,(\boldsymbol{q}_{\mathrm{h}},\, \boldsymbol{p}_{\mathrm{h}},\, \boldsymbol{q}_{\mathrm{r}})$ returns a scalar value quantifying the human's peripersonal-space comfort at a configuration $\boldsymbol{q}_{\mathrm{h}}$, a body pose $\boldsymbol{p}_{\mathrm{h}}$ and a robot configuration $\boldsymbol{q}_{\mathrm{r}}$. We present how we formulate this cost in Sec. 7.3.

Note that both comfort metrics take the human configuration $\boldsymbol{q}_{\mathrm{h}}$ and body pose $\boldsymbol{p}_{\mathrm{h}}$ as inputs, which are undetermined until the human applies the specific forceful operation. Theoretically, given a forceful operation $(\boldsymbol{f},\, \boldsymbol{p})$, an object pose $\boldsymbol{p}_{\mathrm{o}}$ and a robot configuration $\boldsymbol{q}_{\mathrm{r}}$ to hold the object, the human can choose any accessible arm configuration and body pose $(\boldsymbol{q}_{\mathrm{h}},\, \boldsymbol{p}_{\mathrm{h}})$. This raises the question that how the human will perform a forceful operation $\boldsymbol{F}$, i.e. which arm configuration $\boldsymbol{q}_{\mathrm{h}}$ and body pose $\boldsymbol{p}_{\mathrm{h}}$ the human will choose for $(\boldsymbol{f},\, \boldsymbol{p})$, which we need to answer for solving the optimization problem in Eq. 7.1.

Given an object pose $\boldsymbol{p}_{\mathrm{o}}$ and a robot configuration $\boldsymbol{q}_{\mathrm{r}}$ to hold the object for a forceful operation $(\boldsymbol{f},\, \boldsymbol{p})$, we denote the set of all *feasible* human choices, i.e. the arm configuration $\boldsymbol{q}_{\mathrm{h}}$ and body pose $\boldsymbol{p}_{\mathrm{h}}$, as

$$\mathrm{Q_h}\,(\boldsymbol{q}_{\mathrm{r}},\, \boldsymbol{p}_{\mathrm{o}},\, \boldsymbol{f},\, \boldsymbol{p})$$

By feasible, here we mean that the arm configuration $\boldsymbol{q}_{\mathrm{h}}$ and the body pose $\boldsymbol{p}_{\mathrm{h}}$ satisfy the stability and kinematic constraints. In other words, $\mathrm{Q_h}$ includes all the human poses and configurations the human can choose from to successfully perform the forceful operation $(\boldsymbol{f},\, \boldsymbol{p})$, if the robot grasps the object at a configuration $(\boldsymbol{q}_{\mathrm{r}},\, \boldsymbol{p}_{\mathrm{o}})$.

Then, the cost Comfort can be formulated with different options:

Option One is to take the average comfort value of all feasible human choices. Assuming $\mathrm{Q_h}$ is a

discretized set,

$$\text{Comfort}\,(\boldsymbol{q}_\text{r},\,\boldsymbol{p}_\text{o},\,\boldsymbol{f},\,\boldsymbol{p}) = \sum_{\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h}\in\text{Q}_\text{h}} \{w_\text{M}\text{Muscular}\,(\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h},\,\boldsymbol{f},\,\boldsymbol{p},\,\boldsymbol{p}_\text{o}) + w_\text{P}\text{Peripersonal}\,(\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h},\,\boldsymbol{q}_\text{r})\}/|\text{Q}_\text{h}| \tag{7.2}$$

where w_M and w_P are the weighting coefficients for muscular and peripersonal-space comfort respectively. $|\text{Q}_\text{h}|$ indicates the size of the set Q_h[2].

Option Two is to assume the human will choose the most comfortable arm configuration $\boldsymbol{q}_\text{h}$ and body pose $\boldsymbol{p}_\text{h}$ from the feasible set Q_h,

$$\text{Comfort}\,(\boldsymbol{q}_\text{r},\,{}_\text{o}\boldsymbol{T},\,{}^\text{o}\boldsymbol{f},\,{}^\text{o}_\text{t}\boldsymbol{T}) = \max_{\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h}\in\text{Q}_\text{h}} \{w_\text{M}\text{Muscular}\,(\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h},\,\boldsymbol{f},\,\boldsymbol{p},\,\boldsymbol{p}_\text{o}) + w_\text{P}\text{Peripersonal}\,(\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h},\,\boldsymbol{q}_\text{r})\} \tag{7.3}$$

In the rest of this chapter we assume the human choice is optimal and therefore adopt the second option. We solve the optimization problem defined by Eq. 7.1 and 7.3. The problem can alternatively also be solved for Eq. 7.1 and 7.2. In Sec. 7.4, we present human subject studies to further investigate and verify this assumption.

7.2 Muscular Comfort

This section[3] introduces the metric of muscular comfort, using which we quantify the human's muscular effort required in applying a forceful operation $(\boldsymbol{f},\,\boldsymbol{p})$ while at an arm configuration $\boldsymbol{q}_\text{h}$, i.e. the cost function $\text{Muscular}\,(\boldsymbol{q}_\text{h},\,\boldsymbol{f},\,\boldsymbol{p},\,\boldsymbol{p}_\text{h},\,\boldsymbol{p}_\text{o})$ in Eq. 7.3.

7.2.1 Human Arm Modelling

As shown in Fig. 7.2, like a robot manipulator, we model the kinematics and dynamics of a human limb as a serial-link kinematic chain with 7 DOFs —Two spherical joints at the shoulder and wrist and one revolute joint at the elbow[4]. The resemblance between the human arm and the robot manipulator enables a direct mapping from the human arm joint configuration space to the task space where the operation force is applied, that is, the human-arm forward kinematics

$$\boldsymbol{x}_\text{h} = \text{FKM}\,(\boldsymbol{q}_\text{h}) \tag{7.4}$$

[2]Here we omit the arguments to Q_h for clarity, but it is important to note that the set Q_h is determined by the same arguments as the cost function Comfort.

[3]The work in this section is led by Dr. Figueredo (L.Figueredo@leeds.ac.uk) from the School of Computing, University of Leeds, UK. Briefly, Dr. Figueredo proposed the formulation of muscular comfort, while Lipeng Chen did most of the experiments and optimization work.

[4]An important limitation of this current formulation is that it only considers the human arm in measuring the human's muscular effort, ignoring the rest of the body. It is our intention to extend the formulation and our planner to the whole body in the future.

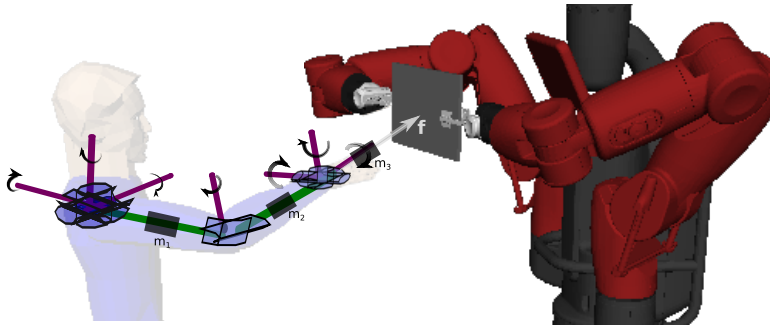


Figure 7.2: Human-robot kinematic chain modeling: the human arm is modelled as a serial kinematic chain with seven degrees of freedom (DOFs).

where $\boldsymbol{x}_{\mathrm{h}}$ denotes the pose of the human hand w.r.t. the robot frame. Then, the differential forward kinematics of the human arm can be written as

$$\dot{\boldsymbol{x}}_{\mathrm{h}} = \boldsymbol{J}_{\mathrm{h}}(\boldsymbol{q}_{\mathrm{h}})\,(\dot{\boldsymbol{q}}_{\mathrm{h}}) \tag{7.5}$$

where $\boldsymbol{J}_{\mathrm{h}}(\boldsymbol{q}_{\mathrm{h}})$ is the geometric Jacobian of the human arm at the configuration $\boldsymbol{q}_{\mathrm{h}}$.

Then, given a human applied operation force $\boldsymbol{f}_{\mathrm{h}}$ with respect to the robot frame, the corresponding force/torque response $\boldsymbol{\tau}_{\mathrm{h}}$ distributed at the arm joints can be denoted as

$$\boldsymbol{M}_{\mathrm{h}}\,(\boldsymbol{q}_{\mathrm{h}}) + \boldsymbol{C}_{\mathrm{h}}\,(\boldsymbol{q}_{\mathrm{h}}, \dot{\boldsymbol{q}}_{\mathrm{h}}) + \boldsymbol{G}_{\mathrm{h}}\,(\boldsymbol{q}_{\mathrm{h}}) = \boldsymbol{\tau}_{\mathrm{h}} - \boldsymbol{J}_{\mathrm{h}}^{\mathsf{T}}(\boldsymbol{q}_{\mathrm{h}})\boldsymbol{f}_{\mathrm{h}} \tag{7.6}$$

where $\boldsymbol{M}_{\mathrm{h}}$, $\boldsymbol{C}_{\mathrm{h}}$, and $\boldsymbol{G}_{\mathrm{h}}$ represent the inertia matrix, the vector of centrifugal and Coriolis forces, and the vector of gravitational effects respectively.

Assuming quasi-static movements, the effects of higher order dynamics can be neglected and therefore the arm joint torques $\boldsymbol{\tau}_{\mathrm{h}}$ depend solely on the applied operation force $\boldsymbol{f}_{\mathrm{h}}$ and gravity effects $\mathbf{G}_{\mathrm{h}}$.

The gravitational effects result from contributions of the arm weight, the arm centre of mass, the tool mass and its pose. Generally, the gravitational *forces* are well-defined in the inertial frame and are independent of the arm configuration $\boldsymbol{q}_{\mathrm{h}}$, whereas the corresponding gravitational *torque* closely depends on the joint configuration $\boldsymbol{q}_{\mathrm{h}}$. Given an arm configuration $\boldsymbol{q}_{\mathrm{h}}$, we denote the corresponding generalized gravitational effects as $\boldsymbol{\rho}_{\mathrm{h}}(\boldsymbol{q}_{\mathrm{h}})$[5].

Then, given a forceful operation $(\boldsymbol{f}, \boldsymbol{p})$ and a human arm configuration $\boldsymbol{q}_{\mathrm{h}}$, the required forces/torques $\boldsymbol{\tau}_{\mathrm{h}}$ at the arm joints can then be computed as:

$$\boldsymbol{\tau}_{\mathrm{h}} = \boldsymbol{J}_{\mathrm{h}}^{\mathsf{T}}({}^{\mathrm{r}}_{\mathrm{t}}\mathrm{T}\boldsymbol{f} + \boldsymbol{\rho}_{\mathrm{h}}) \tag{7.7}$$

where the matrix ${}^{\mathrm{r}}_{\mathrm{t}}\mathrm{T}$ is a norm-preserving congruence transformation which converts the operation force $\boldsymbol{f}$ from the tool frame to the inertia frame $\boldsymbol{f}_{\mathrm{h}} = {}^{\mathrm{r}}_{\mathrm{t}}\mathrm{T}\boldsymbol{f}$.

[5]We drop the argument $\boldsymbol{q}_{\mathrm{h}}$ in the rest of the section for clarity.

7.2.2 Human Muscular Effort

This section presents our methodology of the measuring human's muscular comfort in applying a forceful operation.

The existing studies in robotics usually quantitatively assess a configuration of a robotic manipulator, particularity its manipulability (Yoshikawa, 1985), via the force manipulability ellipsoid (FME), which is defined as the pre-image of the unit sphere in the joint space of the manipulator $\|\boldsymbol{\tau}\|^2 = \boldsymbol{f}^\mathsf{T}\boldsymbol{J}\boldsymbol{J}^\mathsf{T}\boldsymbol{f} \leq 1$. An alternative quality metric to the FME is the force polytopes (Chiacchio *et al.*, 1997) which replaces the FME L_2-norm problem with an L_∞-norm problem with n more constraints (n denotes the size of $\boldsymbol{\tau}$). Although not being as precise as the force polytopes, the FME is more commonly used due to the reduced computational difficulties. Vahrenkamp & Asfour (2015) also proposed an *extended manipulability measure* incorporating constraining factors, such as joint limits, obstacle or the self-distance between manipulator and other parts of the robot.

Or, assume knowing the human joint limits $\boldsymbol{\tau}_\mathrm{h}^\mathrm{max}$, if the human arm worked exactly like a robotic manipulator, given a forceful operation $(\boldsymbol{f}, \boldsymbol{p})$ and an arm configuration $\boldsymbol{q}_\mathrm{h}$, intuitively, one planner could compute the corresponding forces/torques $\boldsymbol{\tau}_\mathrm{h}$ required at the arm joints using Eq. 7.7, and then evaluate the human's muscular effort simply by measuring how close the required forces/torques $\boldsymbol{\tau}_\mathrm{h}$ are to their corresponding joint limits $\boldsymbol{\tau}_\mathrm{h}^\mathrm{max}$, i.e. the further they are from the limits, the more comfortable the corresponding arm configuration is for the human to apply the operation force $\boldsymbol{f}$.

Nevertheless, these assessments cannot properly measure the force generation capability at end-effectors for biological systems, particularly for human arms, since no biological motor properties, such as the nonlinear feature of torque limits $\boldsymbol{\tau}_\mathrm{h}^\mathrm{max}$ at the human arm joints, are considered in modelling. A better modelling of how human arm muscles work is presented by Tanaka *et al.* (2014), based on which we devise our metric of muscular comfort in quantifying human's muscular effort. Compared with a kinematic chain model, there are two key differences in this model. Specifically, different from the robot manipulator, for a human arm,

- The torque limits $\boldsymbol{\tau}_\mathrm{h}^\mathrm{max}$ at the arm joints for a human depend on arm configuration $\boldsymbol{q}_\mathrm{h}$, i.e. the joint force/torque limits do not stay constant at different arm configurations as they do for a robot manipulator;

- The torque limits $\boldsymbol{\tau}_\mathrm{h}^\mathrm{max}$ also depend on the arm motion direction $\dot{\boldsymbol{q}}_\mathrm{h}$, e.g. the torque limit of the elbow should be modelled differently if it is flexing versus if it is extending.

This suggests that, for a human arm, rather than being fixed values at different configurations, the joint torque limits $\boldsymbol{\tau}_\mathrm{h}^\mathrm{max}$ should be modelled and represented as a function of the arm configuration $\boldsymbol{q}_\mathrm{h}$ and the arm moving direction $\dot{\boldsymbol{q}}_\mathrm{h}$ (i.e. the derivative of the arm configuration) $\boldsymbol{\Pi}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h})$. The matrix $\boldsymbol{\Pi}$ is a diagonal matrix

$$diag\{\tau_1^\mathrm{max}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h}), \ldots, \tau_7^\mathrm{max}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h})\}$$

where each diagonal element $\tau_i^\mathrm{max}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h})$ represents the torque limit of the i-th joint, a nonlinear function of the human arm configuration $\boldsymbol{q}_\mathrm{h}$ and its derivative $\dot{\boldsymbol{q}}_\mathrm{h}$. Experimental data to capture the values of

the matrix $\mathbf{\Pi}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h})$ for different human arm configurations can be found in the excel work of Tanaka *et al.* (2014) and the values therein are used throughout this work. Although being computed empirically, these values and proportions between joints and maximum torques follow a similar pattern across human individuals, even for different body weights whereby results differ only in magnitude. More details can be found in Tanaka *et al.* (2014).

As for the arm moving direction $\dot{\boldsymbol{q}}_\mathrm{h}$, herein we assume the human arm motion follows a Jacobian-transpose controller $\dot{\boldsymbol{q}}_\mathrm{h} = \boldsymbol{J}_\mathrm{h}^T(\boldsymbol{q}_\mathrm{h})\dot{\boldsymbol{x}}_\mathrm{h}$ while applying a specific operation force $\boldsymbol{f}_\mathrm{h}$, where $\dot{\boldsymbol{x}}_\mathrm{h}$ denotes the motion direction of the human hand. In this work, we assume the human hand moves towards the same direction as the operation force $\boldsymbol{f}_\mathrm{h}$.

Using the joint torque limits $\mathbf{\Pi}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h})$ and the kinematic model in Eq. 7.7, given that *muscle tension* is nearly proportional to the muscle activation levels, the joint torque distribution $\boldsymbol{\tau}_\mathrm{h}$ can be reformulated as

$$\boldsymbol{\tau}_\mathrm{h} = \mathbf{\Pi}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h})\boldsymbol{\alpha}_\mathrm{h} \tag{7.8}$$

where $\boldsymbol{\alpha}_\mathrm{h}$=$[\alpha_1, \ldots, \alpha_7]^T$ is the vector of *joint torque activation levels*, while $\alpha_i \in [0, 1]$ represents the activation ratio of the i-th joint torque τ_i to its maximum torque $\tau_i^{\max}$.

Accordingly, we devise the metric to evaluate human's muscular comfort based on the norm of the activation level vector $||\boldsymbol{\alpha}_\mathrm{h}||$

$$\text{Muscular}\,(\boldsymbol{q}_\mathrm{h},\ \boldsymbol{f},\ \boldsymbol{p},\ \boldsymbol{p}_\mathrm{h},\ \boldsymbol{p}_\mathrm{o}) = 1/||\boldsymbol{\alpha}_\mathrm{h}||^2 \tag{7.9}$$

where

$$\boldsymbol{\alpha}_\mathrm{h} = \mathbf{\Pi}^{-1}(\boldsymbol{q}_\mathrm{h}, \dot{\boldsymbol{q}}_\mathrm{h})\boldsymbol{J}_\mathrm{h}^\mathrm{T}({}^\mathrm{r}_\mathrm{t}\mathrm{T}\boldsymbol{f} + \boldsymbol{\rho}_\mathrm{h}) \tag{7.10}$$

7.3 Peripersonal-Space Comfort

In this section, we define the metric of peripersonal-space comfort, which we use to further improve the human's comfort experience while collaborating with a robot to perform a forceful operation, i.e. the cost function $\text{Peripersonal}\,(\boldsymbol{q}_\mathrm{h},\ \boldsymbol{p}_\mathrm{h},\ \boldsymbol{q}_\mathrm{r})$ in Eq. 7.3. The planner maximizes the human's peripersonal-space comfort so that the human and robot bodies are configured and positioned with a relatively comfortable distance during the forceful collaboration.

Similar to human to human interactions, in the context of fHRC, the human and robot have to stand in close proximity to perform forceful operations together. In such cases, if the robot stands too close to the human or even surrounds the human body as the one shown in Fig. 7.1(a), the robot would intrude and occupy a large proportion of the human's peripersonal space, which as a consequence would cause a sense of discomfort, such as stress and anxiety, to the human. Furthermore, the inappropriate spatial occupancy, in some cases may block the human's view and affect the visibility of object held by the robot. Besides, unlike most human-robot collaborations well studied in the literature (Cakmak *et al.*, 2011; Peternel *et al.*, 2017; Rozo *et al.*, 2016; Sisbot & Alami, 2012; Solanes *et al.*, 2018; Strabala

et al., 2013b), e.g. handovers and object carrying, in which the human and robot simply reach and hold some object(s) momentarily, the forceful operations discussed in the context of fHRC require strong and persistent forceful interactions among the human, robot and target object, which poses an additional spatial requirements in configuring and positioning the robot and object. For example, a too-short distance would inevitably pose a potential danger to the human, especially when the human is involved in forceful operations.

To achieve a high level of spatial comfort and safety experience, particularly the peripersonal-space comfort and safety, one would want the human and robot to get as far from each other as possible during forceful interactions. Therefore, we formulate the cost of peripersonal-space comfort in the optimization problem using the distance between the human and robot. Specifically, given a human arm configuration $\boldsymbol{q}_{\mathrm{h}}$ and a body pose $\boldsymbol{p}_{\mathrm{h}}$, we assume the existence of a set of points distributed on the human body and represent the set as

$$P_{\mathrm{h}}(\boldsymbol{q}_{\mathrm{h}}, \boldsymbol{p}_{\mathrm{h}}) = \left\{\boldsymbol{p}_{\mathrm{h}}^{1}, \boldsymbol{p}_{\mathrm{h}}^{2}, ..., \boldsymbol{p}_{\mathrm{h}}^{n_{\mathrm{h}}}\right\}$$

where $\boldsymbol{p}_{\mathrm{h}}^{i} \in \mathbb{R}^3$ refers to a point on the human body. The distribution of model points on the human body can be specified according to the operation property. For example, for the case of forceful operations involved in this work, more points can be arranged on the human arms as they are relatively more close to the robot. The number of points n_{h} can be specified empirically. Specifically, a larger n_{h} would lead to a more fine-grained control of the spatial distance between the human and the robot, while a smaller n_{h} would make the optimization process more efficient.

Similarly, given a robot configuration $\boldsymbol{q}_{\mathrm{r}}$, we represent the set of points on the robot body as

$$P_{\mathrm{r}}(\boldsymbol{q}_{\mathrm{r}}) = \left\{\boldsymbol{p}_{\mathrm{r}}^{1}, \boldsymbol{p}_{\mathrm{r}}^{2}, ..., \boldsymbol{p}_{\mathrm{r}}^{n_{\mathrm{r}}}\right\}$$

where $\boldsymbol{p}_{\mathrm{r}}^{j} \in \mathbb{R}^3$ refers to a point on the robot body. Similarly, the distribution of model points on the robot body can be specified according to the operation property. The number of points n_{r} can be chosen empirically.

We use these two sets P_{h} and P_{r} to represent the body geometry of the human and robot respectively. Then, we formulate the peripersonal-space comfort using the distances between points in this two set:

One option is to consider the minimum distance between the human and robot, and hypothesize the cooperation would be as comfortable as the minimum distance allows:

$$\text{Peripersonal}\left(\boldsymbol{q}_{\mathrm{h}}, \boldsymbol{p}_{\mathrm{h}}, \boldsymbol{q}_{\mathrm{r}}\right) = \min_{\forall \boldsymbol{p}_{\mathrm{h}} \in P_{\mathrm{h}}, \forall \boldsymbol{p}_{\mathrm{r}} \in P_{\mathrm{r}}} ||\boldsymbol{p}_{\mathrm{h}} - \boldsymbol{p}_{\mathrm{r}}|| \tag{7.11}$$

Then, the optimization process in Sec. 7.1 would maximize this minimum distance between the human and robot. While this metric may work well for some types of HRI, e.g. handover, it does not work well for the context of fHRC, since the minimum distance between the human and robot has a tight upper bound in our tasks: The robot grippers and the human hand on the tool are naturally the closest points. Thus, maximizing the minimum distance will mostly just maximize the distance between the human

hand and robot grippers, completely ignoring any other part of the human and robot bodies.

To address this issue and to take all points into account, one can instead consider the average distance over all points in the human body:

$$\text{Peripersonal}\,(\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h},\,\boldsymbol{q}_\text{r}) = \frac{1}{|P_\text{h}|}\sum_{\boldsymbol{p}_\text{h}\in P_\text{h}}\min_{\forall \boldsymbol{p}_\text{r}\in P_\text{r}}||\boldsymbol{p}_\text{h}-\boldsymbol{p}_\text{r}|| \tag{7.12}$$

While this metric takes all human body points into account, it brings about another problem: the larger a distance is, the more it dominates the computation of Eq. 7.12. For example, if there exists a point $\boldsymbol{p}_\text{h} \in P_\text{h}$ on the human body that is far away from the robot but some other points are much closer to the robot, then, these points would have less effect on Eq. 7.12 which as a result would provide misleading results.

This suggests that we need a metric that takes all points on the human body into account, but also give more weights to the points with smaller distances to the robot and less weight to the points with larger distances to the robot. Therefore, we formulate the peripersonal-space comfort as

$$\text{Peripersonal}\,(\boldsymbol{q}_\text{h},\,\boldsymbol{p}_\text{h},\,\boldsymbol{q}_\text{r}) = \frac{1}{|P_\text{h}|}\sum_{\boldsymbol{p}_\text{h}\in P_\text{h}} w(\boldsymbol{p}_\text{h})\min_{\forall \boldsymbol{p}_\text{r}\in P_\text{r}}||\boldsymbol{p}_\text{h}-\boldsymbol{p}_\text{r}|| \tag{7.13}$$

where $w(p_\text{h})$ is the weight of the point p_h defined as

$$w(\boldsymbol{p}_\text{h}) = 1 - \frac{\min_{\forall \boldsymbol{p}_\text{r}\in P_\text{r}}||\boldsymbol{p}_\text{h}-\boldsymbol{p}_\text{r}||}{\sum_{\boldsymbol{p}_\text{h}\in P_\text{h}}\min_{\forall \boldsymbol{p}_\text{r}\in P_\text{r}}||\boldsymbol{p}_\text{h}-p_\text{r}||} \tag{7.14}$$

It associates a small distance with a large weight, thus controlling all distances efficiently. Eq. 7.13 and 7.14 together define the metric of peripersonal-space comfort used in this work.

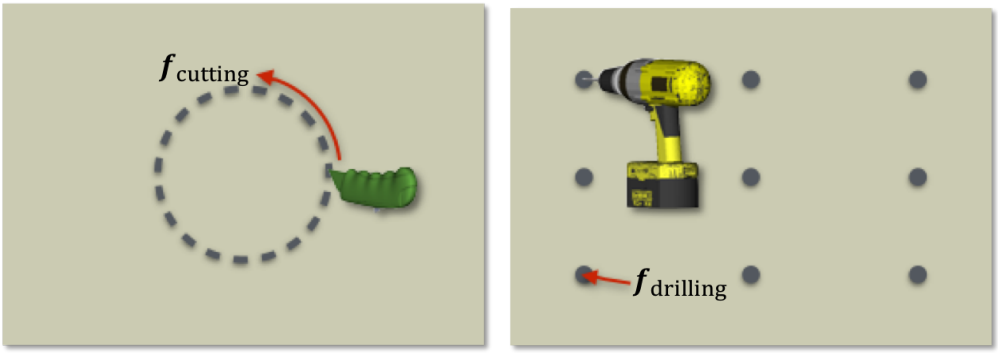
7.4 Experiments and Results

This section presents a series of experiments to assess the effectiveness of our proposed comfort metrics in planning for comfortable fHRC.

Experimental Setting:

We implemented the comfort metrics in Python and used the SciPy library for the optimization process, particularly we used the SLSQP method[6] to solve the constrained optimization problem defined by Eq. 7.1 and 7.3. We adopted the Baxter robot from Rethink Robotics and a human model in OpenRAVE (Diankov & Kuffner, 2008) for the simulated experiments. We took a set of 64 points P_h distributed over the human body and a set of 56 points P_r distributed over the Baxter robot for computing the peripersonal-space comfort in Eq. 7.13 and 7.14. For the optimization process in Eq. 7.1, We weighted the muscular and peripersonal-space comfort equally, yet it is still worth noting that different weighting schemes may be chosen according to the task property and human preferences, e.g. expert knowledge.

[6]https://docs.scipy.org/doc/scipy/reference/optimize.minimize-slsqp.html



(a) 16 cutting Operations (b) 9 drilling Operations

Figure 7.3: We tested our metrics on 16 cutting operations forming the circular cutting task and nine drilling operations uniformly distributed on a foam board.

We tested the comfort metrics on two kinds of forceful operations, cutting and drilling. Particularly, we focused on 16 discretized cutting operations on a foam board which form the circular cutting task as shown in Fig. 7.3(a), and nine drilling operations which are uniformly distributed on the surface of a foam board as shown in Fig. 7.3(b). For these operations, we used a 6D force/torque sensor to measure the operation forces as previously discussed in Chap 4. For each category of forceful operations, we assume an *operation-based grasp configuration* for the human to hold a specific tool as discussed in Sec. 7.1.

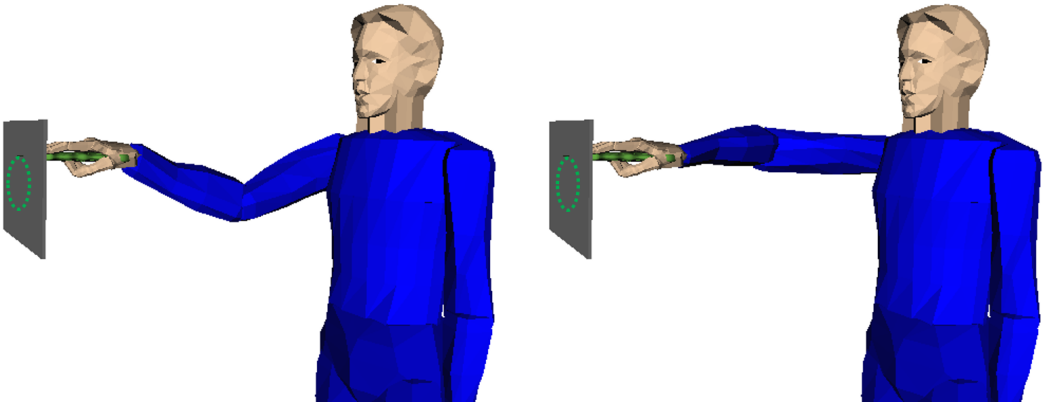
Experimental Studies:

In this setting, we did a variety of experimental studies to verify the performance of our proposed comfort metrics. Specifically, we focus on evaluating:

- The consistency and effectiveness of the muscular comfort metric in assessing the human's muscle effort in performing forceful operations;
- The effectiveness of the optimization framework using our proposed comfort metrics to cope with both the muscular and peripersonal-space comforts in planning for comfortable fHRC;
- The actual human comfort perception in real human-robot interactions with regard to the predicted optimal configurations from optimizing our proposed metrics.

7.4.1 Consistency and Effectiveness of Muscular Comfort

This section presents our experiments to verify the consistency and effectiveness of the muscular comfort metric. We compared the *quality measure* of a set candidate configurations for the 16 cutting operations in Fig. 7.3(a), predicted by the muscular comfort metric, with their *known* muscular comfort sensation, to check whether they are consistent with each other.



(a) The arm configuration ⓐ for a cutting operation.

(b) The arm configuration ⓑ for a cutting operation.

Figure 7.4: Two different arm configurations for a cutting operation with different muscular comfort.

Specifically, for the 16 cutting operations, we assumed a fixed transformation the human, robot and object, i.e. the human could only move his arm to apply the cutting operations while the robot was assumed to tightly hold the object at a fixed configuration. This assumption constrains motions on the human arm, which is in line with our formulation of the muscular comfort.

The robot in this setting only played the role of stabilizing the object, since the muscular comfort defined in Eq. 7.9 only concerns the human arm configuration and the external force.

We focused on checking two variation trends in the known human muscular comfort sensation in these 16 cutting operations and its corresponding reflection in the measured muscular comfort value, specifically,

– Throughout the sequence of cutting operations, the human applies continuous cutting forces along the circular pattern (with a fixed grasp configuration) and thus changes his arm configuration constantly. Different arm configurations correspond to different muscular comfort values. Therefore, as the human applies the circular cutting forces, the corresponding muscular comfort predicted by the proposed metric is expected to fluctuate constantly with the arm configurations, to reflect the comfort variation of the real human comfort sensation. Particularly, the arm configurations near the proximity of joint limits, especially at the wrist spherical joint, are known to be very uncomfortable and have less force generation capabilities, which should also be reflected with lower muscular comfort predictions in the fluctuation of measured muscular comfort.

– For a single cutting operation, there might be multiple feasible arm configurations due to the IK redundancy (even ignoring freedoms of the human body). For example, Fig. 7.4 shows two different arm configurations for the human to perform a same cutting operation. Therefore, the human's muscular comfort can even vary for one single forceful operation. For example, the configuration in Fig. 7.4(a) is clearly much more comfortable than the one in Fig. 7.4(b), which should be also seen in the change of measured muscular comfort.

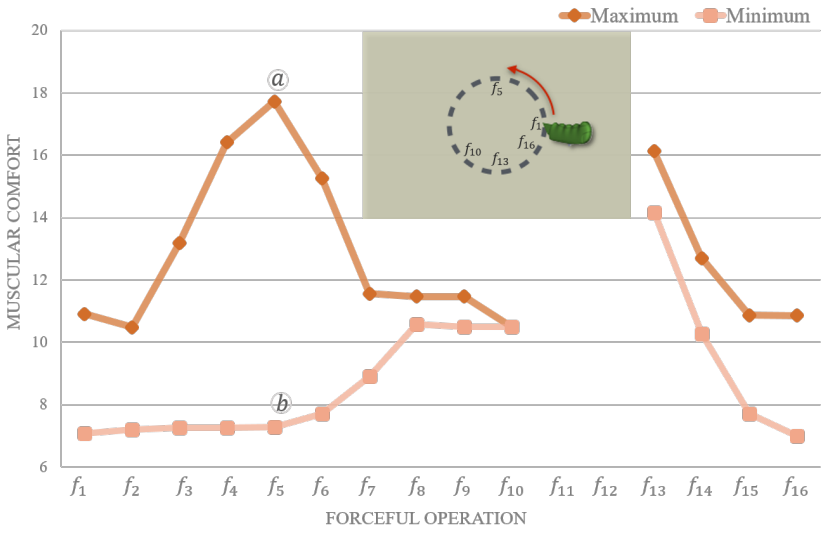
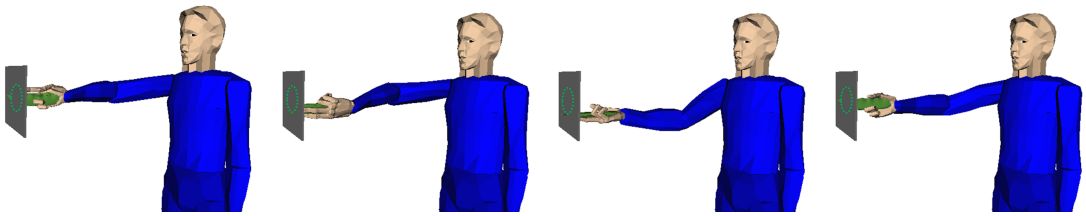


Figure 7.5: Muscular comfort values for different cutting points along a circle. The circled letters on the graph correspond to the configurations in Fig. 7.4.



(a) config.ⓐ (b) config.ⓓ (c) config.ⓔ (d) config.ⓕ

Figure 7.6: Comfortable human configurations for the circular cutting task. Circled letters correspond to configurations shown in Fig. 7.5.

For each of the 16 cutting operations, we generated a set of feasible IK solutions for the human arm and computed their corresponding muscular comfort values. We present the results in Fig. 7.5, in which the dark orange curve depicts the fluctuation of the measured muscular comfort values when the human applied the cutting operations always at the most comfortable predicted configurations, whilst the light orange line chart depicts the fluctuation of measured muscular comfort values when the human applied the operations always at the worst (most uncomfortable) configurations. From the results, we can see:

The measured muscular comfort over the circular cutting task coincides with the actual known human comfort perception. Specifically, the top comfort curve in Fig. 7.5 indicates that in the best case, consistent with the real human-comfort perception, the muscular comfort first increases to an optimum till the operation $\boldsymbol{f}_5$ (the corresponding human configuration is shown in Fig. 7.4(a)), and then decreases till the operation $\boldsymbol{f}_{10}$, where the human arm nearly reaches its joint limits (see the configuration

in Fig. 7.6(b)). From the operation $\boldsymbol{f}_{13}$, the muscular comfort decreases from a local peak value again till the last operation $\boldsymbol{f}_{16}$ (see the configuration in Fig. 7.6(d)). Similar consistency between the muscular comfort measured by the muscular comfort metric and the actual human comfort perception is also reflected in the below comfort curve for the worst case.

The measured muscular comfort over configurations for one single forceful operation also coincides with the actual human comfort perception. For example, Fig. 7.4 shows two arm configurations for the operation $\boldsymbol{f}_5$. Our experience is that configuration ⓐ is much more comfortable than configuration ⓑ, which is exactly reflected by the measured muscular comfort values. Further, Fig. 7.5 shows that the comfort range between the two comfort curves, i.e. the gap between the measured muscular comfort values of the most and least comfortable configurations, increases till the operation $\boldsymbol{f}_5$, where the human applies a horizontal cutting operation and thus enjoys more flexibilities. The comfort range then decreases to a single value at the operation $\boldsymbol{f}_{10}$ (see the configuration in Fig. 7.6(c)) where the human arm reaches the joint limits, and decreases to none from the operation $\boldsymbol{f}_{11}$ to $\boldsymbol{f}_{12}$ due to the loss of feasible IK solutions. In Fig. 7.5, we can also see the consistency at the point of circular closing from the operation $\boldsymbol{f}_{16}$ to the operation $\boldsymbol{f}_1$.

In this context, we can summarize that the muscular comfort can effectively reflect the human's actual muscular effort in performing forceful operations, and therefore optimizing the muscular comfort in Eq. 7.1 and 7.9 is effective in improving human's muscular comfort for fHRC. Besides, the circular cutting example also suggests that it is critical to plan optimal robot and object configurations for comfortable fHRC. For example, for the circular cutting task, if the robot keeps rotating the object to the optimal configuration in Fig. 7.4(a), the human can then perform all the cutting operations with a high level of muscular comfort, e.g. at the configuration shown in Fig. 7.4(a).

7.4.2 Effectiveness of Comfort Optimization

Here we present experiments to evaluate the effectiveness of optimizing our proposed comfort metrics in planning for comfortable fHRC.

We implemented and compared the comfort metrics with four different planners. Specifically,

- Comfort planner: The comfort planner refers to the optimization process defined in Eq. 7.1, namely the planner which optimizes a weighted sum of both human's muscular comfort and peripersonal-space comfort to find the optimal object pose and robot configuration;

- Random planner: Given a forceful operation, the random planner randomly picks an object pose within the reachable space of the robot. Then it searches for the first feasible robot configuration satisfying the constraint of force stability to hold the object, and then chooses the optimal human arm configuration and body pose to perform the operation in terms of maximizing both the muscular and peripersonal-space comforts. The random planner acts as the baseline;

- Muscular planner: The muscular planner takes only the muscular comfort as the cost in optimizing the human comfort, i.e. we set $w_{\mathrm{P}} = 0$ in Eq. 7.3;

Table 7.1: Average results of four planners on 16 cutting and 9 drilling operations. Normalized with results of the Random Planner. The larger a value is, the more comfortable its corresponding configuration is.

	Random Planner		Comfort Planner		Peripersonal Planner		Muscular Planner	
	Musc. Comf.	Perip. Comf.	Musc. Comf.	Perip. Comf.	Musc. Comf.	Perip. Comf.	Musc. Comf.	Perip. Comf.
Cutting	1	1	11.19	2.08	0.34	**2.09**	**15.66**	1.02
Drilling	1	1	19.24	**2.55**	5.78	2.51	**22.45**	0.94

– Peripersonal planner: Similarly, the peripersonal planner maximizes only the human's peripersonal-space comfort (i.e. $w_{\mathrm{M}} = 0$) in optimization.

We ran the above planners on the 25 forceful operations (16 cutting operations and 9 drilling operations). The average results are shown in Table 7.1. To improve readability, we used the results of the random planner as the baseline and normalized results in the table with the baseline.

As shown in Table 7.1, for the 16 cutting operations, the peripersonal planner outperforms the other three planners in maximizing the human's peripersonal-space comfort with an average improvement of 2.09 times over the random planner, yet it still performs poorly in optimizing the human's muscular comfort (0.34 times of the random planner on average). The muscular planner shows similar performance: It performs well in improving human's muscular comfort (15.66) but poorly for the peripersonal-space comfort (1.02). In contrast, the comfort planner can optimize both comfort metrics together to the similar levels as achieved by the peripersonal and muscular planners, which only optimize an individual metric respectively. Similar results can also be found for the 9 drilling operations.

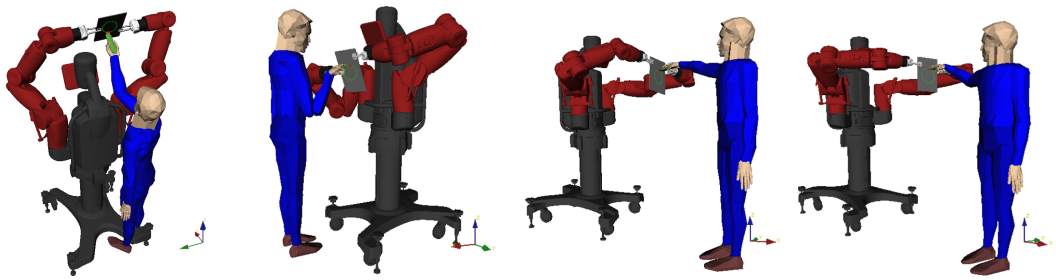
Fig. 7.7 and Fig. 7.8 show configurations found by the four planners for a cutting and a drilling operation respectively. As shown, even though the muscular and peripersonal planners can effectively optimize the corresponding comfort metrics individually, but still limited in improving both comforts together. For example, the configuration generated by the muscular planner in Fig. 7.8(b) has a high level of muscular comfort, but obviously its corresponding peripersonal-space comfort is low (as the human has to stand close next to the robot).

Through the above comparisons, we can sum up that it is necessary and effective to optimize the muscular and peripersonal-space comfort metrics in planning for comfortable fHRC. The results also highlight that it is possible to cope with both muscular and peripersonal-space comforts together without much compromise.

7.4.3 Human Experiments

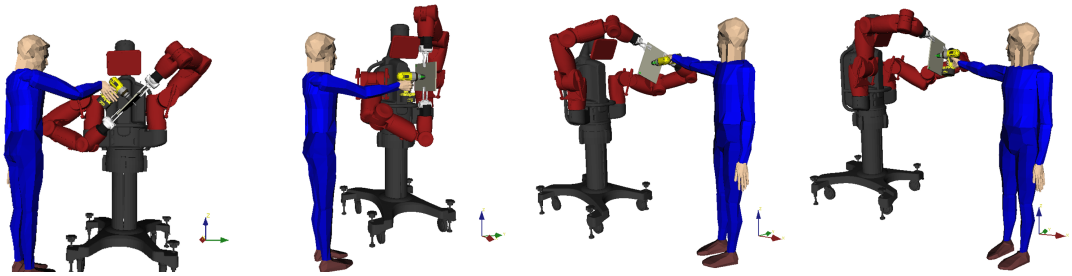
We conducted a set of human-robot experiments to evaluate the actual human comfort perception during fHRC with regard to the optimized configurations generated by optimizing our comfort metrics.

Specifically, a number of five human participants were recruited randomly without knowing the research background. Before the experiments, the participants were briefed about the goal of the study and the forceful tasks they were expected to perform with the robot during the experiments.



(a) Random Planner (b) Muscular planner (c) Peripersonal planner (d) Comfort planner

Figure 7.7: Optimization results for a cutting operation. The comfort values are: (a) Musc. comfort: 8.34; Perip. comfort: 20.43. (b) Musc. comfort: 48.53; Perip. comfort: 23.38. (c) Musc. comfort: 3.98; Perip. comfort: 44.23. (d) Musc. comfort: 47.10; Perip. comfort: 43.30.



(a) Random planner (b) Muscular planner (c) Perip. planner (d) Comfort planner

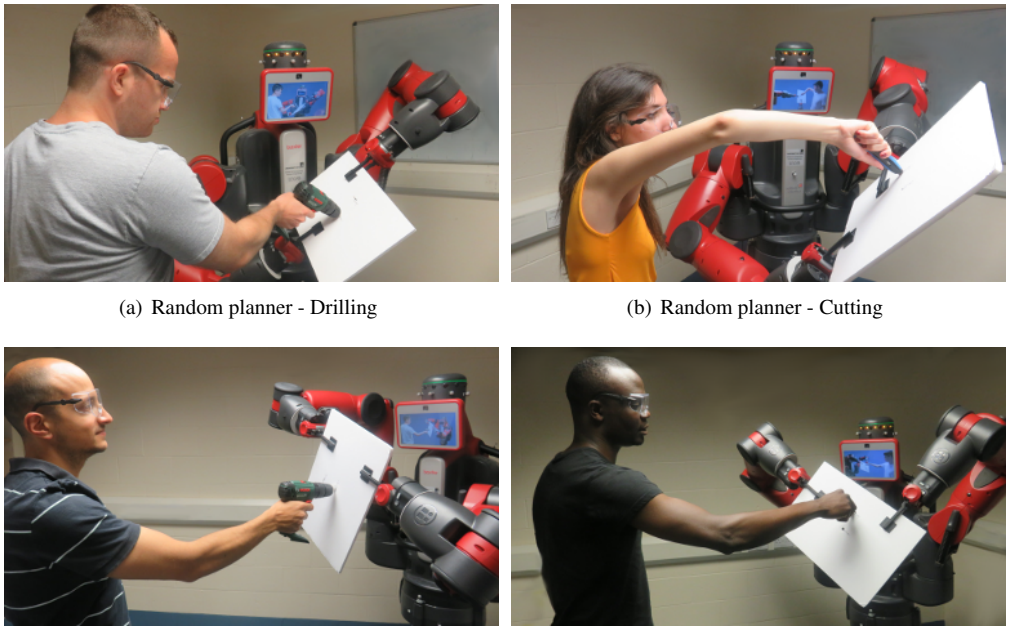
Figure 7.8: Optimization results for a drilling operation. The comfort values of the four planner are: (a) Musc. comfort: 3.70; Perip. comfort: 20.21. (b) Musc. comfort: 42.33; Perip. comfort: 21.88. (c) Musc. comfort: 4.72; Perip. comfort: 44.40. (d) Musc. comfort: 59.19; Perip. comfort: 44.75.

During the experiments, we activated the *gravity compensation mode* provided by the Baxter SDK, which the human participants can use to easily configure the robot manipulators to any preferred reachable configurations by holding and moving the manipulator cuff. Using this feature, the human participants were asked to engage in motion activities with the Baxter robot, so that they could familiarize themselves with the robot before the collaborated study with the robot.

The experiments consist of three comparison groups. Each group involves a set of drilling and cutting tasks on foam boards which were tightly held by the Baxter robot:

– User-Preference: For each experiment in the user-preference group, all participants were asked to (i) move the robot arms using the gravity compensation mode to a configuration they felt comfortable and safe according to their own preferences, and then (ii) drill through the centre of a foam board held by the robot; then (iii) they were asked to do the same but for a cutting operation.

– Random: For each experiment in the random group, (i) we first moved the robot manipulators and the object to configurations generated by the random planner (one for drilling and one for cutting



(a) Random planner - Drilling

(b) Random planner - Cutting

(c) Comfort based planner - Drilling

(d) Comfort based planner - Cutting

Figure 7.9: We conducted a set of human-robot experiments to evaluate the actual human comfort perception during forceful operations with regard the optimized configurations by the proposed comfort metrics.

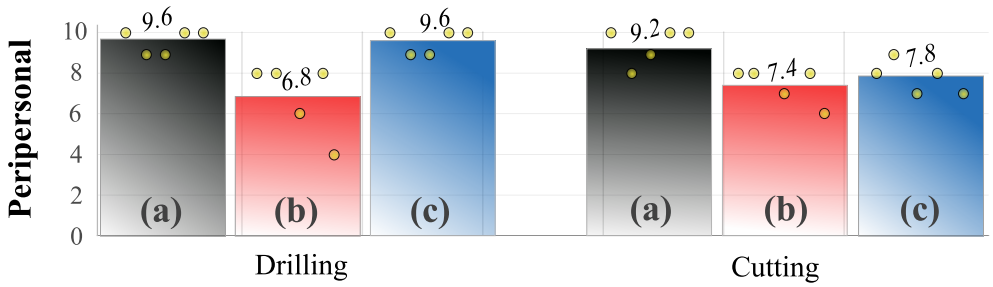


Figure 7.10: The participants' perception of the peripersonal-space comfort (yellow dots) for drilling (left) and cutting (right) with respect to the **(a)** user preferred configurations (black), **(b)** the random configurations (red), and **(c)** the optimized configurations (blue).

respectively). Participants were then asked to (ii) drill and then (iii) cut the board.

– Optimized: For each experiment in the optimized group, (i) we first moved the Baxter arms and the grasped object to the optimized configurations generated by the comfort planner (one optimized configuration for drilling and one for cutting respectively). Participants were then asked to (ii) drill and (iii) cut the board.

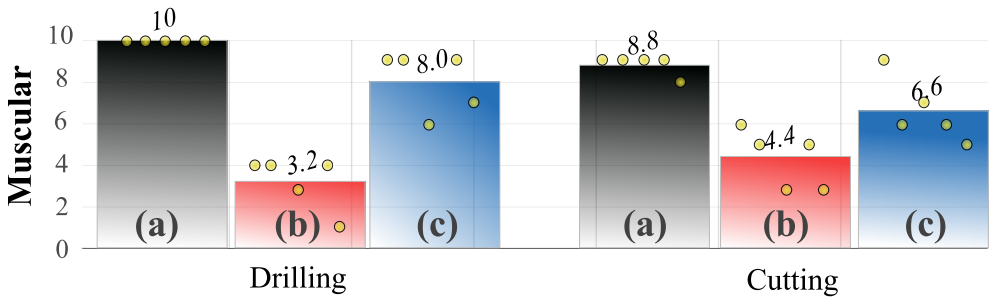


Figure 7.11: The participants' perception of the muscular comfort (yellow dots) for drilling (left) and cutting (right) with respect to the **(a)** user preferred configurations (black), **(b)** the random configurations (red), and **(c)** the optimized configurations (blue).

Fig. 7.9 shows some of these experiments. Throughout the user-preference and optimized experiments, human participants were also instructed to follow the arm configurations and relative body poses suggested by the planners, reducing their flexibility over the redundant configuration space. We then asked them to use self-motion (mostly the redundancy over elbow) to see whether they could find better postures to execute the tasks. This experiment analysis has been particularly devised to investigate our hypothesis of choosing the most comfortable human posture among the redundant space stressed in Eq. 7.3. Among the five participants, only one could find a better configuration for the drilling operation, but only through exploiting the drilling *task redundancy* that the drill could rotate within the drilling axis. None of the participants was able to find a better arm posture for the cutting operations.

After these experiments, for each interaction, the participants were asked to grade from 1 to 10 (higher better), their perception on the safety and peripersonal-space comfort, and their perception on the muscular efforts required during the forceful interaction, i.e. their perception on the muscular comfort. The scores are summarized in Fig. 7.10 for the peripersonal-space comfort and in Fig. 7.11 for the muscular comfort. In each figure, the results of the drilling operations are shown in the left, while the results for the cutting operations are shown in the right.

From both figures, it is easy to see the participants' perceptions regarding the peripersonal-space and muscular comfort agree with our expectations, particularly for the drilling operations. Cutting operations were clearly not as easy as drilling and thus reflected in the figure with a bit lower grades, even at the configurations chosen by the participants themselves. Significant difference between the random planner and the comfort planner can be seen on the participants' perception of the muscular comfort for the drilling operations. This particular result also highlights the importance of explicitly considering and planning for improving the human comfort during forceful operations, which is crucial to foster real-world fHRC.

With regard to the peripersonal-space comfort, as shown in Fig. 7.10, while the participants showed strong preferences for the configurations generated by the comfort planner compared with the ones generated by the random planner, the difference was not as significant. Especially, during these experiments some participants expressed that they did not prefer to be very far away from the robot, possibly sug-

gesting that, while the peripersonal distance must be maximized up to a certain degree, there may be a specific distance after which it starts to negatively affect the human's comfort perception.

Limitations:

One limitation of the proposed optimization process is the time it takes for iterative optimization. To compute one configuration, the comfort planner took 9.4 seconds on average over the 25 operations (with a standard deviation of 1.8), while most of this time was spent on computing IK solutions. These time results indicate that faster schemes should be integrated if the robot is to perform fluent interaction with a human during a continuous task. We aim to investigate more methods, e.g. based on a precomputed IK dataset (Rodríguez *et al.*, 2016; Sisbot & Alami, 2012; Vahrenkamp *et al.*, 2009, 2013) to further improve the time efficiency of the optimization process in the future.

We also realized one limitation of the human experiments is the number of involved human subjects. In one extension work of this thesis, a larger number of human participants were recruited in the comparison experiments, where similar results were obtained. In addition, this work can also be improved by adding more quantitative analysis, which we extended by collected EMG data to further verify the metric of muscular comfort.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

In this thesis, we addressed the problem of manipulation planning for forceful human-robot collaboration. We presented a series of planners allowing stable, efficient and human-comfortable object grasping and manipulation under changing external forces, particularly in a context of forceful human-robot collaboration.

The work in this thesis has given us a few inspirations and lessons:

– We believe that manipulation planning should go beyond geometric constraints, taking into account forceful constraints which are generally required for object grasping and manipulation.

  Through a large variety of our initial experiments, we have shown that object manipulation can easily fail due to the presence of external disturbances. In this thesis, we modelled these external disturbances and presented a range of planners to systematically address them in object manipulation. We explicitly made manipulation *stability* and *efficiency* as objectives in devising these planners. We have shown that task stability can be guaranteed by choosing appropriate powerful object contacts, such as robot grasps. We have also demonstrated the significance of task efficiency in object grasping and manipulation, and proposed to achieve this goal by reducing need for changing contact configurations. However, as we have pointed out, this work can easily integrate some other interpretations of manipulation efficiency, e.g. overall robot motions, as performance indices for manipulation planning.

– We showed that the shared environment can be an opportunity for object grasping and manipulation under external forces.

  In this work, we exploited deliberate object contacts with structures in the shared environment as additional supports to robot grasps in forceful object manipulation. We assumed structures in the environment were rigid and used their reactive wrenches to stabilize objects under external disturbances. We have pointed out that this work is limited in the diversity of environmental and

robot contacts we used. In future work, we hope to explore more diverse object contacts, e.g. face-edge contact, for object manipulation under external forces.

– Our initial experiments showed that the human-applied operation forces would deviate to some extent from the expected operation axes. We proposed to use a spherical conic model to deal with such force uncertainties in modelling the external forces in forceful object manipulation, particularly the collaborative operation forces.

  We showed that, by conservatively approximating the spherical force cone with a circumscribed pyramid, both robustness and efficiency can be guaranteed in checking the force stability of candidate object contacts.

– We showed the necessity of improving human comfort and safety in planning for forceful human-robot manipulation tasks. We have demonstrated that manipulation planning can easily produce robot behaviours that are uncomfortable and unsafe for humans. In particular for forceful collaborative tasks, we showed that human's physical efforts and spatial perception can be critical in enabling robots to collaborate forcefully in the human's close proximity.

  In this thesis, we have taken an analytical approach by optimizing the human's muscular comfort and peripersonal-space comfort. Promising results and strong consistencies have been proved in a large range of comparison studies.

– We presented a graphical interface allowing controlled communication between humans and robots in fHRC. We showed that the interface enables the automation of forceful human-robot fabrications to a relatively satisfying degree.

  We have pointed out the limitations of communication through the interface: compared with other channels, like natural language, the interface may affect the fluidness of collaboration.

We believe empowering robots with the capability of assisting humans in performing forceful collaborative tasks will get them closer to be a part of human life, and our work presented in this thesis can be a key component in a human-robot collaboration framework.

8.2 Future Work

In this section, we discuss the **limitations** of this work and present ideas on how it can be extended and improved in our **future work**.

1. Integrated Manipulation Planning for fHRC

We believe manipulation planning for fHRC can be framed in a more integrated manner:

First, in this thesis, we built a planning framework for forceful human-robot co-manipulation, which can be roughly divided into two parts: From Chap. 4 to 6, we focus on robot behaviours concerning the stability and efficiency of robotic manipulation under changing external forces, while in Chap. 7,

we mainly address human comfort and safety in collaborative forceful object grasping and manipulation. Even though promising results have been achieved by each individual part and fully demonstrated through a variety of experiments, there is still a lack of integration between these works.

In this regard, a potentially more powerful approach can integrate task stability, manipulation efficiency, human comfort and safety, and possibly some other task-oriented planning criteria into one planner, which may be able to generate further *adapted* and *balanced* robot behaviours. Similar to many existing studies on physical human-robot collaboration (Aleotti *et al.*, 2012; Parastegari *et al.*, 2017; Sisbot & Alami, 2012), such a planner may take some form of constrained optimization.

Second, in the context of fHRC discussed in this thesis, we assumed the robot can move an object, e.g. for regrasping, only after the human stops applying forceful operations on the grasped object. Such an *alternate-action* scheme is applicable, but in some cases, may affect, e.g. human experience due to the required waits during collaboration. Even through configuration change minimization can reduce the need of such intentional waits, a more ideal collaboration scheme may allow the human and robot to move simultaneously, with or without physical contacts. For example, the robot should be allowed to move, e.g. reposition the target object, when the human changes his hand tool. Furthermore, in a joint forceful task, the robot and human can move together to proactively adapt to each other's motion and intention. For example, as stated in Chap. 7, in the circular cutting task shown in Fig. 3.1, while the human cuts the board, the robot can rotate the board to reduce human motion, as well as improving human comfort.

Such a collaboration scheme would lead towards more adaptations from both the robot and human, and therefore requires a more integrated control and planning framework for human-robot joint actions. Regarding to this issue, there is a large body of research work reviewed in Chap. 2.3, which we can learn from to develop our collaborative system, for example, based on impedance control (Kosuge & Kazamura, 1997; Lin *et al.*, 2018; Rahman *et al.*, 2002).

Third, another immediate work which is not addressed in this work but important is to integrate human factors, e.g. human comfort, in choosing environmental contacts for collaborative object grasping and manipulation under external forces. In Chap. 6, we discussed the exploitation of environmental contacts for object stabilization, where we mainly focused on the robot kinematic and geometric-collision constraints. However, this might result in object configurations which are not acceptable to humans due to, e.g. object visibility and reachability. Regarding to this issue, *human preferences* can be integrated into the selection as a preliminarily filter of inappropriate contact configurations. Furthermore, a separate process can be created to maintain and update a set of high-quality candidate contact configurations based on human preferences, and be fed into the planning process also for the sake of planning efficiency.

Fourth, we employed the BiRRT (Kuffner Jr & LaValle, 2000) planner to generate *robot motions* for object manipulation, which treated the human as collision for the robot to avoid. However, robot motions should also proactively adapt to the human preferences, particularly in the presence of physical contacts. In this thesis, we have explicitly addressed the problem of object grasping and positioning concerning human comfort and safety. In regard to robot motions, a hierarchical planning framework

can be developed of first generating preferable object paths and then robot motions in line with the human preferences.

Fifth, in planning for comfortable fHRC, we assumed a fixed hand configuration for the human in tool grasping and therefore a fixed kinematic transformation between the human hand and tool. Human grasping plays an critical role in tool use and has been well explored in the literature (Cutkosky & Howe, 1990; Dipietro *et al.*, 2008; Feix *et al.*, 2015; Santello *et al.*, 1998; Schieber & Santello, 2004). Furthermore, as stated previously in Chap. 7, our formulation of the muscular comfort is limited to arm configuration. A more fine-grained formulation can integrate grasping into planning, where learning-based methods, e.g. learning from demonstration (LfD) (Amor *et al.*, 2013; Kim *et al.*, 2019; Vogt *et al.*, 2016) can be explored, e.g. from motion and force data (Schmidts *et al.*, 2011).

2. Implicit Communication for fHRC

In this thesis, we presented a graphical interface in Sec. 4.3 as a communication channel between humans and robots for forceful human-robot collaboration. Specifically, in a collaborative forceful task, a human needs to inform a robot partner of a task, such that the underlying planners can generate appropriate manipulation plans. Then the robot and human collaboratively perform the task in an interactive and communicative manner, where the human needs instructions from the robot on executable operations and the robot wait for the human commands of regrasp.

Even a high level of automation has been achieved via the interface, more effective methods are expected to make this process, e.g. more user-friendly and efficient. In particular, a more helpful communication system in the context of fHRC is required to convey not only task-specific information, but also intentions, motions and even opinions of both parts, e.g. whether a collaboration is safe or dangerous. Further, fHRC requires more powerful and easy-to-use communication channels, not only verbal but also non-verbal, such as gestures and emotional feedback, to enhance system usability and interpretability.

In this regard, *implicit communication* can be a potentially handy tool to achieve effective and adaptive communication for fHRC, which has been attracting increasing attention from the robotics community for *joint human-robot actions* (Gildert *et al.*, 2018; Giuliani *et al.*, 2018; Hazbar, 2019; Knepper, 2016; Knepper *et al.*, 2017; Kulkarni *et al.*, 2019; Wortham & Theodorou, 2017).

Particularly, in the context of fHRC discussed in this thesis, the unique external forces applied by human users can be an additional channel for human-robot communication. In comparison with general channels like natural language and motion, forces can be more efficient and lightweight in conveying human intentions thanks to the physical contacts between humans and robot in fHRC. In addition, such a communication system would allow more adaptations between humans and robots. For example, humans performing collaborative tasks, like dancing, can switch between or share the leader-follower roles effortlessly even in the absence of direct communication, since humans are capable of developing a mutual understanding only using haptic perceptions, which in essence are communicative forces.

Such a communication system leads towards a highly automated collaborative system. For example, in this thesis, we assumed forceful tasks were revealed to robots completely or progressively in advance, which might be not true or difficult in actual practice. Using the interface, this could be done as part of

communication between humans and robots in the course of collaboration.

3. Other Work

Our experiments show that the planning time for forceful tasks in both Chap. 5 and 6 can still take tens of seconds. Our quantitative assessment of these simulated experiments has shown that the time efficiency of our planners is limited mostly by the speed of the low-level constrained motion planners, which leaves room for improvement in future work to either speed up these individual motion plans, or to reduce the number of such motion plan queries. Meanwhile, we have pointed out the limitation of our approach in Chap. 7 in terms of time efficiency, which was mainly due to the required IK computation. Regarding to this issue, a precomputed IK dataset with preferred grasping and manipulation qualities, e.g. manipulability and reachability, can be created to speed up this process, which is also widely used by existing studies (Przybylski *et al.*, 2013; Rodríguez *et al.*, 2016; Sisbot & Alami, 2012; Vahrenkamp *et al.*, 2009, 2011, 2013).

In terms of system integrity, in our experiment setting, robot regrasping may fail due to the uncertainties of robot system, e.g. unexpected object slippage. In this regard, a perception system for object and robot motion, e.g. a vision-based tracking system, can be integrated to improve the motion accuracy for regrasping. Some other functional modules, e.g. force/torque sensation, can also be integrated into the manipulation system to improve overall performance.

In addition to above ideas, we hope to explore more extension studies in future work to develop a more skilled and competent robot system for forceful human-robot collaboration.

Bibliography

ABI-FARRAJ, F., OSA, T., PETERS, N.P.J., NEUMANN, G. & GIORDANO, P.R. (2017). A learning-based shared control architecture for interactive task execution. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 329–335. 14

ALAMI, R., SIMEON, T. & LAUMOND, J.P. (1990). A geometrical approach to planning manipulation tasks. the case of discrete placements and grasps. In *The fifth international symposium on Robotics research*, 453–463, MIT Press. 5, 11, 12

ALAMI, R., LAUMOND, J.P. & SIMÉON, T. (1994). Two manipulation planning algorithms. In *WAFR Proceedings of the workshop on Algorithmic foundations of robotics*, 109–125, AK Peters, Ltd. Natick, MA, USA. 5, 11, 12

ALEOTTI, J., MICELLI, V. & CASELLI, S. (2012). Comfortable robot to human object hand-over. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, 771–776, IEEE. 15, 95

ALFORD, C. & BELYEU, S. (1984). Coordinated control of two robot arms. In *Proceedings. 1984 IEEE international conference on robotics and automation*, vol. 1, 468–473, IEEE. 9

AMOR, H.B., VOGT, D., EWERTON, M., BERGER, E., JUNG, B. & PETERS, J. (2013). Learning responsive robot behavior by imitation. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3257–3264, IEEE. 96

ASFOUR, T., VAHRENKAMP, N., SCHIEBENER, D., DO, M., PRZYBYLSKI, M., WELKE, K., SCHILL, J. & DILLMANN, R. (2013). Armar-iii: Advances in humanoid grasping and manipulation. *Journal of the Robotics Society of Japan*, **31**, 341–346. 14

ASFOUR, T., KAUL, L., WÄCHTER, M., OTTENHAUS, S., WEINER, P., RADER, S., GRIMM, R., ZHOU, Y., GROTZ, M., PAUS, F. *et al.* (2018). Armar-6: A collaborative humanoid robot for industrial environments. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, 447–454, IEEE. 14

BABIN, V. & GOSSELIN, C. (2018). Picking, grasping, or scooping small objects lying on flat surfaces: A design approach. *The International Journal of Robotics Research*, **37**, 1484–1499. 13

BARTOLO, A., CARLIER, M., HASSAINI, S., MARTIN, Y. & COELLO, Y. (2014). The perception of peripersonal space in right and left brain damage hemiplegic patients. *Frontiers in Human Neuroscience*, **8**. 75

BAUER, A., WOLLHERR, D. & BUSS, M. (2008). Human–robot collaboration: a survey. *International Journal of Humanoid Robotics*, **5**, 47–66. 1

BEKIROGLU, Y., LAAKSONEN, J., JORGENSEN, J.A., KYRKI, V. & KRAGIC, D. (2011). Assessing grasp stability based on learning and haptic data. *IEEE Transactions on Robotics*, **27**, 616–629. 8

BERENSON, D., SRINIVASA, S. & KUFFNER, J. (2011). Task space regions: A framework for pose-constrained manipulation planning. *The International Journal of Robotics Research*, **30**. 47

BICCHI, A. (1995). On the closure properties of robotic grasping. *The International Journal of Robotics Research*, **14**, 319–334. 8

BOHG, J., WELKE, K., LEÓN, B., DO, M., SONG, D., WOHLKINGER, W., MADRY, M., ALDÓMA, A., PRZYBYLSKI, M., ASFOUR, T. *et al.* (2012). Task-based grasp adaptation on a humanoid robot. *IFAC Proceedings Volumes*, **45**, 779–786. 9

BONITZ, R.G. & HSIA, T.C. (1994). Force decomposition in cooperating manipulators using the theory of metric spaces and generalized inverses. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 1521–1527, IEEE. 10

BORST, C., FISCHER, M. & HIRZINGER, G. (1999). A fast and robust grasp planner for arbitrary 3d objects. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 3, 1890–1896, IEEE. 8

BORST, C., FISCHER, M. & HIRZINGER, G. (2004). Grasp planning: How to choose a suitable task wrench space. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 1, 319–325, IEEE. 7, 8, 9, 29, 66

BRETL, T. (2006). Motion planning of multi-limbed robots subject to equilibrium constraints: The free-climbing robot problem. *The International Journal of Robotics Research*, **25**, 317–342. 12, 13

BUSCH, B., TOUSSAINT, M. & LOPES, M. (2018). Planning ergonomic sequences of actions in human-robot interaction. In *IEEE International Conference on Robotics and Automation*, 1916–1923. 16

CAKMAK, M., SRINIVASA, S.S., LEE, M.K., FORLIZZI, J. & KIESLER, S. (2011). Human preferences for robot-human hand-over configurations. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1986–1993, IEEE. 15, 17, 80

CAO, C., WAN, W., PAN, J. & HARADA, K. (2016). Analyzing the utility of a support pin in sequential robotic manipulation. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 5499–5504, IEEE. 13, 14

CHANG, L.Y., SRINIVASA, S.S. & POLLARD, N.S. (2010). Planning pre-grasp manipulation for transport tasks. In *2010 IEEE International Conference on Robotics and Automation*, 2697–2704, IEEE. 11

CHAVAN-DAFIE, N. & RODRIGUEZ, A. (2018). Regrasping by fixtureless fixturing. In *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, 122–129, IEEE. 13, 14

CHEN, L., FIGUEREDO, L. & DOGAR, M. (2018a). Planning for muscular and peripersonal-space comfort during human-robot forceful collaboration. In *Proceedings of Humanoids 2018*, IEEE. 6

CHEN, L., FIGUEREDO, L.F. & DOGAR, M. (2018b). Manipulation planning under changing external forces. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3503–3510, IEEE. 6, 12

CHEN, L., FIGUEREDO, L.F. & DOGAR, M. (2019). Manipulation planning using environmental contacts to keep objects stable under external forces. In *Proceedings of 2019 IEEE-RAS International Conference on Humanoid Robots*, IEEE. 6

CHEN, Q. & LUH, J. (1994). Coordination and control of a group of small mobile robots. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 2315–2320, IEEE. 10

CHIACCHIO, P., CHIAVERINI, S. & SICILIANO, B. (1996). Direct and inverse kinematics for coordinated motion tasks of a two-manipulator system. *Journal of dynamic systems, measurement, and control*, **118**, 691–697. 10

CHIACCHIO, P., BOUFFARD-VERCELLI, Y. & PIERROT, F. (1997). Force polytope and force ellipsoid for redundant manipulators. *Journal of Robotic Systems*, **14**, 613–620. 79

CUTKOSKY, M.R. & HOWE, R.D. (1990). Human grasp choice and robotic grasp analysis. In *Dextrous robot hands*, 5–31, Springer. 96

DANG, H. & ALLEN, P.K. (2012). Semantic grasping: Planning robotic grasps functionally suitable for an object manipulation task. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1311–1317, IEEE. 8

DANG, H. & ALLEN, P.K. (2014). Semantic grasping: planning task-specific stable robotic grasps. *Autonomous Robots*, **37**, 301–316. 8

DE LUCA, A., ALBU-SCHAFFER, A., HADDADIN, S. & HIRZINGER, G. (2006). Collision detection and safe reaction with the dlr-iii lightweight manipulator arm. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1623–1630, IEEE. 16

DEBUS, T.J., DUPONT, P.E. & HOWE, R.D. (2004). Contact state estimation using multiple model estimation and hidden markov models. *The International Journal of Robotics Research*, **23**. 63, 64

DIANKOV, R. & KUFFNER, J. (2008). Openrave: A planning architecture for autonomous robotics. *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-34*, **79**. 49, 68, 82

DIPIETRO, L., SABATINI, A.M. & DARIO, P. (2008). A survey of glove-based systems and their applications. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, **38**, 461–482. 96

DOBSON, A. & BEKRIS, K.E. (2015). Planning representations and algorithms for prehensile multi-arm manipulation. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 6381–6386, IEEE. 40

DOGAR, M. & SRINIVASA, S. (2011). A framework for push-grasping in clutter. *Robotics: Science and systems VII*, **1**. 11, 13

DOGAR, M., SPIELBERG, A., BAKER, S. & RUS, D. (2019). Multi-robot grasp planning for sequential assembly operations. *Autonomous Robots*, **43**, 649–664. 11, 40

DOGAR, M.R. & SRINIVASA, S.S. (2010). Push-grasping with dexterous hands: Mechanics and a method. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2123–2130, IEEE. 13

DUCHAINE, V. & GOSSELIN, C.M. (2007). General model of human-robot cooperation using a novel velocity based variable impedance control. In *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*, 446–451. 17

EL-KHOURY, S., DE SOUZA, R. & BILLARD, A. (2015). On computing task-oriented grasps. *Robotics and Autonomous Systems*, **66**, 145–158. 8

EPPNER, C. & BROCK, O. (2015). Planning grasp strategies that exploit environmental constraints. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 4947–4952, IEEE. 13

EPPNER, C., DEIMEL, R., ALVAREZ-RUIZ, J., MAERTENS, M. & BROCK, O. (2015). Exploitation of environmental constraints in human and robotic grasping. *The International Journal of Robotics Research*, **34**, 1021–1038. 13

ERDMANN, M.A. & MASON, M.T. (1988). An exploration of sensorless manipulation. *IEEE Journal on Robotics and Automation*, **4**, 369–379. 13

FEIX, T., ROMERO, J., SCHMIEDMAYER, H.B., DOLLAR, A.M. & KRAGIC, D. (2015). The grasp taxonomy of human grasp types. *IEEE Transactions on Human-Machine Systems*, **46**, 66–77. 96

FERRARI, C. & CANNY, J.F. (1992). Planning optimal grasps. In *IEEE International Conference on Robotics and Automation, 1992. Proceedings.*, vol. 3, 2290–2295, IEEE. 7, 8, 29

FUJII, S. & KURONO, S. (1975). Coordinated computer control of a pair of manipulators. In *Proceeding of 4th IFToMM World Congress*, 411–417. 10

GEIDENSTAM, S., HUEBNER, K., BANKSELL, D. & KRAGIC, D. (2009). Learning of 2d grasping strategies from box-based 3d object approximations. In *Robotics: Science and Systems*, vol. 2008. 9

GERAERTS, R. & OVERMARS, M.H. (2006). Sampling and node adding in probabilistic roadmap planners. *Robotics and Autonomous Systems*, **54**, 165–173. 11

GILDERT, N., MILLARD, A.G., POMFRET, A. & TIMMIS, J. (2018). The need for combining implicit and explicit communication in cooperative robotic systems. *Frontiers in Robotics and AI*, **5**, 65. 32, 96

GIULIANI, M. *et al.* (2018). Beyond to me, to you: Combining implicit and explicit communication for cooperative transport in humanoid robotic systems. In *Towards Autonomous Robotic Systems: 19th Annual Conference, TAROS 2018, Bristol, UK July 25-27, 2018, Proceedings*, vol. 10965, 455, Springer. 32, 96

HAGBERG, A., SWART, P. & S CHULT, D. (2008). Exploring network structure, dynamics, and function using networkx. Tech. rep., Los Alamos National Lab.(LANL), Los Alamos, NM (United States). 49, 68

HAN, L., TRINKLE, J.C. & LI, Z.X. (2000). Grasp analysis as linear matrix inequality problems. *IEEE Transactions on Robotics and Automation*, **16**, 663–674. 8, 9

HARADA, K., TSUJI, T., NAGATA, K., YAMANOBE, N., ONDA, H., YOSHIMI, T. & KAWAI, Y. (2012). Object placement planner for robotic pick and place tasks. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 980–985, IEEE. 11

HARADA, K., TSUJI, T. & LAUMOND, J.P. (2014). A manipulation motion planner for dual-arm industrial manipulators. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 928–934, IEEE. 11, 40

HASCHKE, R., STEIL, J.J., STEUWER, I. & RITTER, H.J. (2005). Task-oriented quality measures for dextrous grasping. In *CIRA*, 689–694, Citeseer. 9

HAUSER, K. & LATOMBE, J.C. (2010). Multi-modal motion planning in non-expansive spaces. *The International Journal of Robotics Research*, **29**, 897–915. 12

HAZBAR, T. (2019). Task planning and execution for human robot team performing a shared task in a shared workspace. 32, 96

HERTKORN, K., ROA, M.A., PREUSCHE, C., BORST, C. & HIRZINGER, G. (2012). Identification of contact formations: Resolving ambiguous force torque information. In *2012 IEEE International Conference on Robotics and Automation*, 3278–3284, IEEE. 66

HOGAN, N. (1985a). Impedance control: An approach to manipulation: Part iiimplementation. *Journal of dynamic systems, measurement, and control*, **107**, 8–16. 14

HOGAN, N. (1985b). Impedance control: An approach to manipulation: Part itheory. *Journal of dynamic systems, measurement, and control*, **107**, 1–7. 14

HOLLADAY, R., LOZANO-PEREZ, T. & RODRIGUEZ, A. (2019). Force-and-motion constrained planning for tool use. In *International Conference on Intelligent Robots and Systems (IROS)*. 12

HUBER, M., LENZ, C., RICKERT, M., KNOLL, A., BRANDT, T. & GLASAUER, S. (2008). Human preferences in industrial human-robot interactions. In *Proceedings of the international workshop on cognition for technical systems*. 17

HUEBNER, K., WELKE, K., PRZYBYLSKI, M., VAHRENKAMP, N., ASFOUR, T., KRAGIC, D. & DILLMANN, R. (2009). Grasping known objects with humanoid robots: A box-based approach. In *2009 International Conference on Advanced Robotics*, 1–6. 9

HUNAWAR, K. & UCHIYAMA, M. (1997). Slip compensated manipulation with cooperating multiple robots. In *Proceedings of the 36th IEEE Conference on Decision and Control*, vol. 2, 1918–1923, IEEE. 10

IKEURA, R., MORIGUCHI, T. & MIZUTANI, K. (2002). Optimal variable impedance control for a robot and its application to lifting an object with a human. In *Proceedings. 11th IEEE International Workshop on Robot and Human Interactive Communication*, 500–505. 14

JAILLET, L. & PORTA, J.M. (2013). Path planning under kinematic constraints by rapidly exploring manifolds. *IEEE Transactions on Robotics*, **29**, 105–117. 47

JORDA, M., HERRERO, E.G. & KHATIB, O. (2019). Contact-driven posture behavior for safe and interactive robot operation. In *2019 International Conference on Robotics and Automation (ICRA)*, 9243–9249, IEEE. 13

KIM, W., LORENZINI, M., BALATTI, P., NGUYEN, D.H.P., PATTACINI, U., TIKHANOFF, V., PETERNEL, L., FANTACCI, C., NATALE, L., METTA, G. *et al.* (2019). Adaptable workstations for human–robot collaboration: A reconfigurable framework for improving worker ergonomics and productivity. *IEEE Robotics & Automation Magazine*. 96

KIRKPATRICK, D., MISHRA, B. & YAP, C.K. (1992). Quantitative steinitz's theorems with applications to multifingered grasping. *Discrete & Computational Geometry*, **7**, 295–318. 8

KNEPPER, R.A. (2016). On the communicative aspect of human-robot joint action. In *the IEEE International Symposium on Robot and Human Interactive Communication Workshop: Toward a Framework for Joint Action, What about Common Ground*. 32, 96

KNEPPER, R.A., MAVROGIANNIS, C.I., PROFT, J. & LIANG, C. (2017). Implicit communication in a joint action. In *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI*, 283–292, IEEE. 32, 96

KOSUGE, K. & KAZAMURA, N. (1997). Control of a robot handling an object in cooperation with a human. In *Proceedings 6th IEEE International Workshop on Robot and Human Communication. RO-MAN'97 SENDAI*, 142–147, IEEE. 14, 95

KOSUGE, K., HASHIMOTO, S. & YOSHIDA, H. (1998). Human-robots collaboration system for flexible object handling. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 2, 1841–1846 vol.2. 14

KOSUGE, K., SATO, M. & KAZAMURA, N. (2000). Mobile robot helper. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1, 583–588 vol.1. 14

KUFFNER JR, J.J. & LAVALLE, S.M. (2000). Rrt-connect: An efficient approach to single-query path planning. In *ICRA*, vol. 2. 49, 68, 95

KULKARNI, A., SRIVASTAVA, S. & KAMBHAMPATI, S. (2019). A unified framework for planning in adversarial and cooperative environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2479–2487. 32, 96

LAKSHIMINARAYANA, K. (1978). Mechanics of form closure. *ASME paper, 78-DET-32*. 7, 8

LASOTA, P.A. & SHAH, J.A. (2015). Analyzing the effects of human-aware motion planning on close-proximity human-robot collaboration. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, **57**, 21–33. 74

LASOTA, P.A., ROSSANO, G.F. & SHAH, J.A. (2014). Toward safe close-proximity human-robot interaction with standard industrial robots. In *IEEE Intern Conf on Automation Science & Engineering*. 74

LAWITZKY, M., MÖRTL, A. & HIRCHE, S. (2010). Load sharing in human-robot cooperative manipulation. In *RO-MAN, IEEE*, 185–191. 15

LEE, G., LOZANO-PÉREZ, T. & KAELBLING, L.P. (2015). Hierarchical planning for multi-contact non-prehensile manipulation. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 264–271, IEEE. 12, 13, 31

LEE, S.H., LEE, J.H., YI, B.J., KIM, S.H. & KWAK, Y.K. (2005). Optimization and experimental verification for the antagonistic stiffness in redundantly actuated mechanisms: a five-bar example. *Mechatronics*, **15**, 213–238. 10

LEFEBVRE, T. (2003). Contact modelling, parameter identification and task planning for autonomous compliant motion using elementary contacts. *PhD thesis, KU Leuven, Department of Mechanical Engineering*. 63, 64

LERTKULTANON, P. & PHAM, Q.C. (2018). A certified-complete bimanual manipulation planner. *IEEE Transactions on Automation Science and Engineering*, **15**, 1355–1368. 12

LI, Y. & GE, S.S. (2013). Human–robot collaboration based on motion intention estimation. *IEEE/ASME Transactions on Mechatronics*, **19**, 1007–1014. 17

LI, Z. & SASTRY, S.S. (1988). Task-oriented optimal grasping by multifingered robot hands. *IEEE Journal on Robotics and Automation*, **4**, 32–44. 8, 9

LIN, H.C., SMITH, J., BABARAHMATI, K.K., DEHIO, N. & MISTRY, M. (2018). A projected inverse dynamics approach for multi-arm cartesian impedance control. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 1–5, IEEE. 12, 95

LIN, Y. & SUN, Y. (2016). Task-oriented grasp planning based on disturbance distribution. In *Robotics Research*, 577–592, Springer. 9

LIPTON, J.I., MANCHESTER, Z. & RUS, D. (2017). Planning cuts for mobile robots with bladed tools. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE. 12

LIPTON, J.I., SCHULZ, A., SPIELBERG, A., TRUEBA, L.H., MATUSIK, W. & RUS, D. (2018). Robot assisted carpentry for mass customization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 1–8, IEEE. 12

LOZANO-PÉREZ, T. & KAELBLING, L.P. (2014). A constraint-based method for solving sequential manipulation planning problems. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3684–3691, IEEE. 11

LOZANO-PEREZ, T., MASON, M.T. & TAYLOR, R.H. (1984). Automatic synthesis of fine-motion strategies for robots. *The International Journal of Robotics Research*, **3**, 3–24. 13

LOZANO-PÉREZ, T., JONES, J., MAZER, E., O'DONNELL, P., GRIMSON, W., TOURNASSOUD, P. & LANUSSE, A. (1987). Handey: A robot system that recognizes, plans, and manipulates. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, vol. 4, 843–849, IEEE. 11, 13

LUO, R., HAYNE, R. & BERENSON, D. (2018). Unsupervised early prediction of human reaching for human–robot collaboration in shared workspaces. *Autonomous Robots*, **42**, 631–648. 15

MA, J., WAN, W., HARADA, K., ZHU, Q. & LIU, H. (2018). Regrasp planning using stable object poses supported by complex structure. *IEEE Transactions on Cognitive and Developmental Systems*. 13, 14, 64

MAEDA, G.J., NEUMANN, G., EWERTON, M., LIOUTIKOV, R., KROEMER, O. & PETERS, J. (2017). Probabilistic movement primitives for coordination of multiple human–robot collaborative tasks. *Autonomous Robots*, **41**, 593–612. 15

MAINPRICE, J. & BERENSON, D. (2013). Human-robot collaborative manipulation planning using early prediction of human motion. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 299–306, IEEE. 1

MAINPRICE, J. & BERENSON, D. (2013). Human-robot collaborative manipulation planning using early prediction of human motion. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 299–306. 16

MAINPRICE, J., SISBOT, E.A., SIMÉON, T. & ALAMI, R. (2010). Planning safe and legible hand-over motions for human-robot interaction. 15

MANSFELD, N., HAMAD, M., BECKER, M., MARIN, A.G. & HADDADIN, S. (2018). Safety map: A unified representation for biomechanics impact data and robot instantaneous dynamic properties. *IEEE Robotics and Automation Letters*, **3**, 1880–1887. 16

MASON, M. (1985). The mechanics of manipulation. In *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 544–548, IEEE. 13

MCCLAMROCH, N. (1986). Singular systems of differential equations as dynamic models for constrained robot systems. In *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, vol. 3, 21–28, IEEE. 10

MILLER, A.T. & ALLEN, P.K. (2000). Graspit!: A versatile simulator for grasp analysis. In *in Proc. of the ASME Dyn Syst & Contr Div*. 64

MILLER, A.T. & ALLEN, P.K. (2004). Graspit! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, **11**, 110–122. 46

MISHRA, B., SCHWARTZ, J.T. & SHARIR, M. (1987). On the existence and synthesis of multifinger positive grips. *Algorithmica*, **2**, 541–558. 7, 8, 28, 29

MORIYAMA, R., WAN, W. & HARADA, K. (2019). Dual-arm assembly planning considering gravitational constraints. *arXiv preprint arXiv:1903.00646*. 12

MUNAWAR, K. & UCHIYAMA, M. (1999). Slip compensated manipulation of an object with cooperating multiple robots. *Robotica*, **17**, 543–551. 10

MUROOKA, M., UEDA, R., NOZAWA, S., KAKIUCHI, Y., OKADA, K. & INABA, M. (2017). Global planning of whole-body manipulation by humanoid robot based on transition graph of object motion and contact switching. *Advanced Robotics*, **31**, 322–340. 31

MURRAY, R.M. (2017). *A mathematical introduction to robotic manipulation*. CRC press. 8

MUTHUSAMY, R. *et al.* (2018). Decentralized grasp coordination and kinematic control for cooperative manipulation. 15

NAKANO, E. (1974). Cooperational control of the anthropomorphous manipulator" melarm". In *Proc. of 4th International Symposium on Industrial Robots*, 251–260. 10

NIELSEN, C.L. & KAVRAKI, L.E. (2000). A two level fuzzy prm for manipulation planning. In *Proceedings. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)(Cat. No. 00CH37113)*, vol. 3, 1716–1721, IEEE. 5, 11, 12

NIKANDROVA, E. & KYRKI, V. (2015). Category-based task specific grasping. *Robotics and Autonomous Systems*, **70**, 25–35. 8

PAN, J., CHITTA, S. & MANOCHA, D. (2012). Fcl: A general purpose library for collision and proximity queries. In *2012 IEEE International Conference on Robotics and Automation*, IEEE. 64, 68

PARASTEGARI, S., ABBASI, B., NOOHI, E. & ZEFRAN, M. (2017). Modeling human reaching phase in human-human object handover with application in robot-human handover. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 3597–3602, IEEE. 15, 16, 95

PARK, C., ONDŘEJ, J., GILBERT, M., FREEMAN, K. & O'SULLIVAN, C. (2016). Hi robot: Human intention-aware robot planning for safe and efficient navigation in crowds. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3320–3326, IEEE. 1

PÉREZ-D'ARPINO, C. & SHAH, J.A. (2015). Fast target prediction of human reaching motion for cooperative human-robot manipulation tasks using time series classification. In *2015 IEEE international conference on robotics and automation (ICRA)*, 6175–6182, IEEE. 1

PETERNEL, L., TSAGARAKIS, N., CALDWELL, D. & AJOUDANI, A. (2016). Adaptation of robot physical behaviour to human fatigue in human-robot co-manipulation. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 489–494, IEEE. 16

PETERNEL, L., KIM, W., BABI, J. & AJOUDANI, A. (2017). Towards ergonomic control of human-robot co-manipulation and handover. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 55–60. 15, 16, 80

POLLARD, N.S. (1994). Parallel methods for synthesizing whole-hand grasps from generalized prototypes. Tech. rep., MASSACHUSETTS INST OF TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE LAB. 9

PRZYBYLSKI, M., VAHRENKAMP, N., ASFOUR, T. & DILLMANN, R. (2013). Grasp and motion planning for humanoid robots. In *Grasping in Robotics*, 329–359, Springer. 97

PREZ-D'ARPINO, C. & SHAH, J.A. (2015). Fast target prediction of human reaching motion for cooperative human-robot manipulation tasks using time series classification. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 6175–6182. 17

RAHMAN, M.M., IKEURA, R. & MIZUTANI, K. (2002). Investigation of the impedance characteristic of human arm for development of robots to cooperate with humans. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, **45**, 510–518. 14, 95

REULEAUX, F. (1875). *The kinematics of machinery*. Reuleaux, F. 1875. Kinematics of Machinery. First published in German. Reprinted 1963 by Dover, New York, NY. 7

RIZZOLATTI, G., SCANDOLARA, C., MATELLI, M. & GENTILUCCI, M. (1981). Afferent properties of periarcuate neurons in macaque monkeys. ii. visual responses. *Behavioural Brain Research*, **2**. 75

RODRÍGUEZ, C., ROJAS-DE SILVA, A. & SUÁREZ, R. (2016). Dual-arm framework for cooperative applications. In *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, 1–6, IEEE. 91, 97

ROHRDANZ, F. & WAHL, F.M. (1997). Generating and evaluating regrasp operations. In *Proceedings of international conference on robotics and automation*, vol. 3, 2013–2018, IEEE. 12

ROZO, L., CALINON, S., CALDWELL, D.G., JIMENEZ, P. & TORRAS, C. (2016). Learning physical collaborative robot behaviors from human demonstrations. *IEEE Transactions on Robotics*. 14, 15, 80

SAKITA, K., OGAWARA, K., MURAKAMI, S., KAWAMURA, K. & IKEUCHI, K. (2004). Flexible cooperation between human and robot by interpreting human intention from gaze information. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 1, 846–851, IEEE. 17

SALISBURY, J.K. & ROTH, B. (1983). Kinematic and force analysis of articulated mechanical hands. *Journal of Mechanisms, Transmissions, and Automation in Design*, **105**, 35–41. 9

SANTELLO, M., FLANDERS, M. & SOECHTING, J.F. (1998). Postural hand synergies for tool use. *Journal of Neuroscience*, **18**, 10105–10115. 96

SAUT, J.P., GHARBI, M., CORTÉS, J., SIDOBRE, D. & SIMÉON, T. (2010). Planning pick-and-place tasks with two-hand regrasping. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4528–4533, IEEE. 40

SCHIEBER, M.H. & SANTELLO, M. (2004). Hand function: peripheral and central constraints on performance. *Journal of applied physiology*, **96**, 2293–2300. 96

SCHMIDTS, A.M., LEE, D. & PEER, A. (2011). Imitation learning of human grasping skills from motion and force data. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1002–1007, IEEE. 96

SHINGAREY, D., KAUL, L. & ASFOUR, T. (2019). Torque-based velocity control for safe human-humanoid interaction. In *International Conference on Robotics in Alpe-Adria Danube Region*, 61–68, Springer. 16

SICILIANO, B. & KHATIB, O. (2016). *Springer handbook of robotics*. Springer. 9

SIERLA, S., KYRKI, V., AARNIO, P. & VYATKIN, V. (2018). Automatic assembly planning based on digital product descriptions. *Computers in Industry*, **97**, 34–46. 11

SIMÉON, T., LAUMOND, J.P., CORTÉS, J. & SAHBANI, A. (2004). Manipulation planning with probabilistic roadmaps. *The International Journal of Robotics Research*, **23**, 729–746. 11

SISBOT, E.A. & ALAMI, R. (2012). A human-aware manipulation planner. *IEEE Transactions on Robotics*, **28**, 1045–1057. 15, 16, 80, 91, 95, 97

SOLANES, J.E., GRACIA, L., MUÑOZ-BENAVENT, P., MIRO, J.V., CARMICHAEL, M.G. & TORNERO, J. (2018). Human-robot collaboration for safe object transportation using force feedback. *Robotics and Autonomous Systems*. 15, 16, 80

STOETER, S.A., VOSS, S., PAPANIKOLOPOULOS, N.P. & MOSEMANN, H. (1999). Planning of regrasp operations. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 1, 245–250, IEEE. 12

STRABALA, K., LEE, M.K., DRAGAN, A., FORLIZZI, J., SRINIVASA, S.S., CAKMAK, M. & MICELLI, V. (2013a). Toward seamless human-robot handovers. *Journal of Human-Robot Interaction*, **2**, 112–132. 15

STRABALA, K.W., LEE, M.K., DRAGAN, A.D., FORLIZZI, J.L., SRINIVASA, S., CAKMAK, M. & MICELLI, V. (2013b). Towards seamless human-robot handovers. *Journal of Human-Robot Interaction*, **2**. 15, 80

SUITA, K., YAMADA, Y., TSUCHIDA, N., IMAI, K., IKEDA, H. & SUGIMOTO, N. (1995). A failure-to-safety" kyozon" system with simple contact detection and stop capabilities for safe human-autonomous robot coexistence. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, vol. 3, 3089–3096, IEEE. 16

TAKASE, K. (1974). The design of an articulated manipulator with torque control ability. In *Proc. 4th Int. Symp. on Industrial Robots, Tokyo*. 10, 28

TANAKA, Y., NISHIKAWA, K., YAMADA, N. & TSUJI, T. (2014). Analysis of operational comfort in manual tasks using human force manipulability measure. *IEEE transactions on haptics*, **8**, 8–19. 79, 80

TARIQ, U., MUTHUSAMY, R. & KYRKI, V. (2018). Grasp planning for load sharing in collaborative manipulation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 6847–6854, IEEE. 15

THOBBI, A., GU, Y. & SHENG, W. (2011). Using human motion estimation for human-robot cooperative manipulation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2873–2878, IEEE. 17

TOURNASSOUD, P., LOZANO-PÉREZ, T. & MAZER, E. (1987). Regrasping. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, vol. 4, 1924–1928, IEEE. 11, 13

TRINKLE, J.C. (1992). On the stability and instantaneous velocity of grasped frictionless objects. *IEEE Transactions on Robotics and Automation*, **8**, 560–572. 8

UCHIYAMA, M. & DAUCHEZ, P. (1988). A symmetric hybrid position/force control scheme for the coordination of two robots. In *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, 350–356, IEEE. 10, 28

UCHIYAMA, M. & DAUCHEZ, P. (1992). Symmetric kinematic formulation and non-master/slave coordinated control of two-arm robots. *Advanced Robotics*. 10, 28

UNSEREN, M.A. (1991). Rigid body dynamics and decoupled control architecture for two strongly interacting manipulators. *Robotica*, **9**, 421–430. 10

VAHRENKAMP, N. & ASFOUR, T. (2015). Representing the robots workspace through constrained manipulability analysis. *Autonomous Robots*, **38**, 17–30. 79

VAHRENKAMP, N., BERENSON, D., ASFOUR, T., KUFFNER, J. & DILLMANN, R. (2009). Humanoid motion planning for dual-arm manipulation and re-grasping tasks. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2464–2470, IEEE. 91, 97

VAHRENKAMP, N., PRZYBYLSKI, M., ASFOUR, T. & DILLMANN, R. (2011). Bimanual grasp planning. In *2011 11th IEEE-RAS International Conference on Humanoid Robots*, 493–499, IEEE. 97

VAHRENKAMP, N., ASFOUR, T. & DILLMANN, R. (2013). Robot placement based on reachability inversion. In *2013 IEEE International Conference on Robotics and Automation*, 1970–1975, IEEE. 91, 97

VOGT, D., STEPPUTTIS, S., WEINHOLD, R., JUNG, B. & AMOR, H.B. (2016). Learning human-robot interactions from human-human demonstrations (with applications in lego rocket assembly). In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 142–143, IEEE. 96

WALKER, I.D., MARCUS, S.I. & FREEMAN, R.A. (1989). Distribution of dynamic loads for multiple cooperating robot manipulators. *Journal of Robotic Systems*, **6**, 35–47. 10

WALKER, I.D., FREEMAN, R.A. & MARCUS, S.I. (1991). Analysis of motion and internal loading of objects grasped by multiple cooperating manipulators. *The International journal of robotics research*, **10**, 396–409. 10

WAN, W. & HARADA, K. (2016). Integrated assembly and motion planning using regrasp graphs. *Robotics and biomimetics*, **3**. 11, 12, 13

WAN, W. & HARADA, K. (2017). Regrasp planning using 10,000 s of grasps. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1929–1936, IEEE. 12, 13, 14

WAN, W., MASON, M.T., FUKUI, R. & KUNIYOSHI, Y. (2015). Improving regrasp algorithms to analyze the utility of work surfaces in a workcell. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 4326–4333, IEEE. 14

WAN, W., HARADA, K. & NAGATA, K. (2018). Assembly sequence planning for motion planning. *Assembly Automation*, **38**, 195–206. 11

WANG, M.Y. & PELINESCU, D.M. (2003). Contact force prediction and force closure analysis of a fixtured rigid workpiece with friction. *Journal of manufacturing science and engineering*, **125**. 31

WELKE, K., SCHIEBENER, D., ASFOUR, T. & DILLMANN, R. (2013). Gaze selection during manipulation tasks. In *2013 IEEE International Conference on Robotics and Automation*, 652–659, IEEE. 17

WORTHAM, R.H. & THEODOROU, A. (2017). Robot transparency, trust and utility. *Connection Science*, **29**, 242–248. 32, 96

XIAO, J. & JI, X. (2001). Automatic generation of high-level contact state space. *IJRR*, **20**. 64

YAMADA, Y., HIRASAWA, Y., HUANG, S., UMETANI, Y. & SUITA, K. (1997). Human-robot contact in the safeguarding space. *IEEE/ASME transactions on mechatronics*, **2**, 230–236. 16

YOSHIKAWA, T. (1985). Manipulability of robotic mechanisms. *The international journal of Robotics Research*, **4**, 3–9. 79

YOSHIKAWA, T. & ZHENG, X.Z. (1993). Coordinated dynamic hybrid position/force control for multiple robot manipulators handling one constrained object. *The International Journal of Robotics Research*, **12**, 219–230. 10

ZHENG, Y.F. & LUH, J. (1989). Optimal load distribution for two industrial robots handling a single object. *Journal of Dynamic Systems, Measurement, and Control*, **111**, 232–237. 9, 10