



Robot Grasping by Exploiting Compliance and Environmental Constraints

VORGELEGT VON

DIPL.-INF.

CLEMENS EPPNER

ORCID: 0000-0002-5398-4037

VON DER FAKULTÄT IV — ELEKTROTECHNIK UND INFORMATIK
DER TECHNISCHEN UNIVERSITÄT BERLIN
ZUR ERLANGUNG DES AKADEMISCHEN GRADES

DOKTOR DER NATURWISSENSCHAFTEN
— DR. RER. NAT. —

GENEHMIGTE DISSERTATION

PROMOTIONS AUSSCHUSS

VORSITZENDER : PROF. DR. MARC ALEXA
GUTACHTER : PROF. DR. OLIVER BROCK
GUTACHTER : PROF. DR. ANTONIO BICCHI
GUTACHTER : PROF. DR. HELGE RITTER

TAG DER WISSENSCHAFTLICHEN AUSSPRACHE: 6. SEPTEMBER 2018

BERLIN, 2019

Declaration

Declarations according to §5, Sec. 1 of the Doctoral Regulations.

- re. (1): I hereby declare that I am acquainted with the current doctoral regulations of the TU Berlin as of 23 October 2006, last amended on 5 February 2014.
- re. (5): I hereby declare that all pre-publications of the dissertation or parts thereof and details of own contributions according to §2, subparagraph 4 doctoral regulations are listed in the attachment.
- re. (6): I hereby declare in lieu of an oath that I have independently completed the dissertation. All aids and sources have been listed and all details regarding own contributions according to (5) are correct.
- re. (7): I hereby declare that I have listed all applications (if any) for admission as a doctoral candidate or admission to doctoral procedure according in Sec. 4. of this form.

Berlin,

Clemens Eppner

Robot Grasping by Exploiting Compliance and Environmental Constraints

ABSTRACT

Grasping is a crucial skill for any autonomous system that needs to alter the physical world. The complexity of robot grasping stems from the fact that any solution comprises various components: Hand design, control, perception, and planning all affect the success of a grasp. Apart from picking solutions in well-defined industrial scenarios, general grasping in unstructured environment is still an open problem.

In this thesis, we exploit two general properties to devise grasp planning algorithms: the compliance of robot hands and the stiffness of the environment that surrounds an object. We view hand compliance as an enabler for local adaptability in the grasping process that does not require explicit reasoning or planning. As a result, we study compliance-aware algorithms to synthesize grasps. Exploiting hand compliance also simplifies perception, since precise geometric object models are not needed. Complementary to hand compliance is the idea of exploiting the stiffness of the environment. In real-world scenarios, objects never occur in isolation. They are situated in an environmental context: on a table, in a shelf, inside a drawer, etc. Robotic grasp strategies can benefit from contact with the environment by pulling objects to edges, pushing them against surfaces etc. We call this principle the *exploitation of environmental constraints*. We present grasp planning algorithms which detect and sequence environmental constraint exploitations.

We study the two ideas by focusing on the relationships between the three main constituents of the grasping problem: hand, object, and environment. We show that the interactions between adaptable hands and objects lend themselves to low-dimensional grasp actions. Based on this insight, we devise two grasp planning algorithms which map compliance modes to raw sensor signals using minimal prior knowledge. Next, we focus on the interactions between hand and environment. We show that contacting the environment can improve success in motion and grasping tasks. We conclude our investigations by considering interactions between all three factors: hand, object, and environment. We extend our grasping approach to

select the most appropriate environmental constraint exploitation based on the shape of an object. Finally, we consider simple manipulation tasks that require individual finger movements. Although compliant hands pose challenges due to the difficulty in modeling and limitations in sensing, we propose an approach to learn feedback control strategies that solve these tasks. We evaluate all algorithms presented in this thesis in extensive real-world experiments, compare their assumptions and discuss limitations. The investigations and planning algorithms show that exploiting compliance in hands and stiffness in the environment leads to improved grasp performance.

Robotisches Greifen unter Ausnutzung der Nachgiebigkeit von Händen und der Einbeziehung der Umgebung

ZUSAMMENFASSUNG

Greifen ist eine fundamentale Fähigkeit, die jedes autonome System beherrschen muss, welches die physikalische Welt verändern will. Die Komplexität des robotischen Greifens entspringt dem Umstand, dass jede Lösung eine Vielzahl an unterschiedlichen Komponenten enthält: den Handmechanismus, seine Regelung, die Wahrnehmung der Umwelt und Bewegungsplanung. Sie alle beeinflussen den Erfolg eines Griffes. Obwohl Greiflösungen in wohldefinierten Industrieanwendungen existieren, ist das generelle Greifproblem in unstrukturierten Umgebungen noch immer ungelöst.

Diese Dissertation stellt Greifplanungsalgorithmen vor, die zwei allgemeine Eigenschaften ausnutzen: die Nachgiebigkeit von Roboterhänden und die Einbeziehung der Umgebung eines Objekts. Wir betrachten Nachgiebigkeit als einen Wegbereiter für die lokale Anpassungsfähigkeit der Hand im Greifprozess, die keinerlei logisches Denken oder Planung benötigt. Darauf aufbauend präsentieren wir Greifplanungsalgorithmen, die die Nachgiebigkeit einer Hand explizit berücksichtigen. Dies erleichtert auch das Wahrnehmungsproblem, da nun keine akkurate geometrischen Modelle des Objekts notwendig sind. Komplementär zur Nachgiebigkeit der Hand steht die Idee, die Festigkeit der Umgebung für das Greifen auszunutzen. In menschlichen Lebenswelten treten Objekte nie isoliert auf. Stattdessen sind sie in einem Umgebungskontext eingebettet: Sie befinden sich auf Tischen, in Regalen, Schubladen, usw. Robotische Greifstrategien können davon profitieren, indem sie Objekte an Kanten ziehen, gegen Oberflächen drücken, usw. Wir nennen diese Prinzip die *Ausnutzung von Beschränkungen durch die Umgebung*. Wir stellen Greifplanungsalgorithmen vor, die auf der Sequenzierung von umgebungsausnutzenden Bewegungen basieren.

Wir studieren diese beiden Ideen, indem wir uns auf die Beziehungen der Hauptbestandteile des Greifproblems konzentrieren: Hand, Objekt und Umwelt. Wir zeigen, dass die Interaktionen zwischen nachgiebigen Händen und Objekten sich für eine niedrigdimensionale Greifbeschreibung eignen. Auf dieser Einsicht aufbauend, entwickeln wir zwei Algorithmen,

die mit minimalem Vorwissen passende Greifstrategien auf der Basis von visuellen Sensordaten auswählen. Anschließend widmen wir uns den Interaktionen zwischen Hand und Umwelt. Wir zeigen, dass Kontakt mit der Umgebung die Erfolgsrate bei Bewegungs- und Greifproblemen erhöhen kann. Unsere finalen Untersuchungen betreffen die Interaktionen zwischen allen drei Faktoren: Hand, Objekt und Umwelt. Wir erweitern unseren Ansatz, indem wir die beste Greifstrategie zur Ausnutzung der Umgebung aufgrund der Formeigenschaft eines Objekts auswählen. Im letzten Abschnitt betrachten wir einfache Manipulationsaufgaben, die die gezielte Bewegung einzelner Fingerglieder verlangt. In diesem Fall stellen nachgiebige Hände eine Herausforderung dar, da ihre Effekte schwer zu modellieren sind und ihre Sensorik eingeschränkt ist. Wir stellen eine Methode vor, die trotz dieser Hindernisse Regler lernt, die einfache Manipulationsaufgaben lösen können. Alle vorgestellten Algorithmen in dieser Dissertation werden ausführlich auf echten Robotersystemen evaluiert. Wir vergleichen ihre Annahmen und diskutieren Limitierungen. Unsere Untersuchungen und Planungsalgorithmen zeigen, dass die Ausnutzung von Nachgiebigkeit in Händen und Festigkeit in der Umgebung zu erfolgreicherem Greifen führt.

*And there ain't nothin' like a friend who can tell you
you're just pissin' in the wind.*

Neil Young

Acknowledgments

OFTEN THE ACKNOWLEDGMENTS of a doctoral dissertation start with the bold claim that “it takes a village to raise a Ph.D. student”*. I wholeheartedly agree that it usually requires a multitude of mentors to convey the diverse set of academic skills associated with a Ph.D. But my experience during the past years has also taught me about a roaring alternative to this village: Oliver Brock! My advisor’s insurmountable breadth of knowledge and wisdom was a source for constant excitement and learning: doing presentations, mopping floors, fighting bureaucracy, teaching students, managing time, projects and people, cleaning bibliographies, organizing anything, communicating, writing papers and proposals, thinking outside the box, selecting wine, arguing, contemplating research trends, ... I will always be influenced by the way Oliver handled these things and I am deeply grateful that I had the opportunity to be advised by him. Unfortunately, like any big kahuna, Oliver had to travel the world sometimes. Not only in those moments I was happy to share time and space with the wonderful population in and around the RBO village. Thank you Alexander, Angela, Armin, Arne, Can, Dubi, Előd, Eveline, Fabian H., Fabian S., Felix, Florian, Friedrich, Georg B., Georg H., George, Gudrun, Henrietta, Ines, Ingo, Janika, Jessica, Johannes K., Johannes W., José, Kolja, Lars, Lefteris, Lizzy, Lorenz, Mahmoud, Manuel B., Manuel W., Marianne,



O. Brock, black acrylic linocut print
on white paper.
Berlin, December 2013.

*Based on the proverb “It takes a village to raise a child”, found in various African societies. The Google search for “*it takes a village to raise a phd*” returns 4790 results.

Maximilian, Melinda, Michael, Nasir, Raphael, Rico, Robert, Roberto, Sabrina, Sebastian H., Sebastian K., Stanio, Stefan, Steffen, Thomas, Tim, Vincent, Wolf, Wolfgang, Wouter. The fact that this list of names was generated by a Python script does not downplay the formidable roles you have played in my daily research life. It only emphasizes my love for the art of programming.

Although I was never keen to leave the coziness of the RBO village, I am very thankful for the months I could spent in Berkeley. I would like to thank Prof. Pieter Abbeel and his group for the valuable experiences and insights I gained during my research stay. They strongly influenced my view on robotics. Thank you Abhishek, Igor, Nikhil, and Sergey.

Finally, I would like to thank the institutions that supported the RBO village and me throughout the years: Technische Universität Berlin, the Alexander-von-Humboldt Foundation, the European Commission and the DAAD.

Prepublication and Statement of Contribution

Parts of this thesis have been previously published in the following peer-reviewed articles or as a technical report:

- [O1] Clemens Eppner, Georg Bartels, and Oliver Brock. A compliance-centric view of grasping. Technical report, Technische Universität Berlin, 2012.
- [O2] Clemens Eppner and Oliver Brock. Grasping unknown objects by exploiting shape adaptability and environmental constraints. In *Proc. 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 4000–4006, 2013.
- [O3] Clemens Eppner, Raphael Deimel, José Álvarez-Ruiz, Marianne Maertens, and Oliver Brock. Exploitation of environmental constraints in human and robotic grasping. *The International Journal of Robotics Research*, 34(7):1021–1038, 2015.
- [O4] Clemens Eppner and Oliver Brock. Planning grasp strategies that exploit environmental constraints. In *Proc. 2015 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 4947–4952, 2015.
- [O5] Abhishek Gupta, Clemens Eppner, Sergey Levine, and Pieter Abbeel. Learning dexterous manipulation for a soft robotic hand from human demonstrations. In *Proc. 2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3786–3793, 2016.
- [O6] Arne Sieverling, Clemens Eppner, Felix Wolff, and Oliver Brock. Interleaving motion in contact and in free space for planning under uncertainty. In *Proc. 2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 4011–4017, 2017.
- [O7] Clemens Eppner and Oliver Brock. Visual detection of opportunities to exploit contact in grasping using contextual multi-armed bandits. In *Proc. 2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 273–278, 2017.

Own contributions to (Eppner et al., 2012, O1): OB and I conceived the project idea. GB and I contributed to the algorithmic implementations and experimental evaluations. All authors contributed to manuscript writing.

Own contributions to (Eppner and Brock, 2013, O2), (Eppner and Brock, 2015, O4), (Eppner and Brock, 2017, O7): I conceived, designed, implemented and evaluated the algorithms and experiments presented in the papers and made the main contribution to paper writing. OB gave scientific advice and contributed to paper writing.

Own contributions to (Eppner et al., 2015, O3): OB conceived the project idea. JAR and MM contributed to the part of human grasp experiments (Sec. 3). RD contributed to the part of hardware design (Sec. 5). I contributed to the part of implementing and evaluating grasp strategies (Sec. 4). All authors contributed to paper writing.

Own contributions to (Gupta et al., 2016, O5): AG is the sole first author. PA conceived the project idea. AG made main contributions to implementation, experimental evaluation, and paper writing. I contributed to implementation, preliminary evaluation and paper writing. PA and SL gave scientific advice and contributed to paper writing.

Own contributions to (Sieverling et al., 2017, O6): AS is the sole first author. AS and I conceived the project idea and conceptual underpinnings. AS made the main contributions to paper writing and algorithmic implementation. FW contributed to implementation and empirical evaluation. I contributed to implementation and evaluation of Sec. 3E and Sec. 4B. OB gave scientific advice and contributed to paper writing.

APPEARANCE OF PREVIOUS PUBLICATIONS IN THE THESIS

CHAPTER 0, 1 and 9 are original to this thesis.

CHAPTER 2 presents human grasping experiments which are original to this thesis. The Sun-Flower-Annulus hypothesis (Sec. 2.2) has been previously published in (Eppner et al., 2012, O1). The supporting simulation experiments in 2D are original.

CHAPTER 3 presents two algorithms for grasp planning. The algorithm in Sec. 3.1 has been previously published in (Eppner et al., 2012, O1). The algorithm in Sec. 3.2 has been previously published in (Eppner and Brock, 2013, O2).

CHAPTER 4 presents an algorithm to plan motions under uncertainty based on a geometric model of the environment. This work has been previously published in ([Sieverling et al., 2017](#), O6). This thesis includes additional experimental results for achieving a pre-grasp pose which are also used to compare this algorithm to the grasp planning method presented in Chapter 5.

CHAPTER 5 presents a method for planning grasp actions that exploit the environment. It is based on work that has been previously published in ([Eppner and Brock, 2015](#), O4). The thesis extends the original method by introducing a more rigorous symbolic reasoning (Sec. 5.4) at the discrete level and applying an optimization at the continuous level (Sec. 5.5). The presentation of environmental constrained grasp strategies extends the material that has been previously published in ([Eppner et al., 2015](#), O3).

CHAPTER 6 is original to this thesis, except for Sec. 6.1 which is based on the method that has been previously published in ([Eppner and Brock, 2013](#), O2).

CHAPTER 7 is based on the approach that has been previously published ([Eppner and Brock, 2017](#), O7). This thesis presents a set of additional experiments in simulation that test the shape classifiers (Sec. 7.1.3) and includes a more elaborate and concise explanation of the applied contextual multi-armed bandit algorithms (Sec. 7.2).

CHAPTER 8 has been previously published ([Gupta et al., 2016](#), O5). This theses adds an low-dimensional simulation experiment for illustrative purposes (Sec. 8.5).

Overview of Contents

o INTRODUCTION	I
i BACKGROUND AND RELATED WORK	10
I Interactions Between Hand and Object	41
2 ADAPTABILITY BETWEEN HAND AND OBJECT	43
3 PLANNING GRASPS THAT EXPLOIT HAND ADAPTABILITY	69
II Interactions Between Hand and Environment	87
4 PLANNING MOTIONS THAT EXPLOIT CONTACT WITH THE ENVIRONMENT	90
5 PLANNING GRASP STRATEGIES THAT EXPLOIT THE ENVIRONMENT	109
III Interactions Between Hand, Object and Environment	135
6 COMBINING PROPERTIES OF OBJECT AND ENVIRONMENT TO ADAPT GRASP STRATEGIES	140
7 SELECTING ENVIRONMENT-CONSTRAINED GRASP STRATEGIES BASED ON OBJECT PROPERTIES	157
8 BEYOND GRASPING: LEARNING DEXTEROUS MANIPULATION FOR SOFT HANDS	179
9 CONCLUSION	198
REFERENCES	218

Contents

o	INTRODUCTION	I
o.1	Goal of the Thesis	3
o.2	Exploiting the Compliance and Adaptability of Robot Hands	4
o.3	Exploiting the Stiffness of the Environment	5
o.4	Content and Contributions of the Thesis	7
i	BACKGROUND AND RELATED WORK	10
i.1	Robotic Hands for Grasping	10
i.2	Classical Grasp and Restraint Analysis	18
i.3	Algorithms for Synthesizing Grasps	30
i.4	Conclusion	40
I	Interactions Between Hand and Object	41
2	ADAPTABILITY BETWEEN HAND AND OBJECT	43
2.1	Exploratory Grasping Experiments	44
2.2	The Sun-Flower-Annulus Hypothesis	55
2.3	The Funnel View of Grasping	65
2.4	Conclusion	68
3	PLANNING GRASPS THAT EXPLOIT HAND ADAPTABILITY	69
3.1	Features Based on Active Vision	70
3.2	Features Based on 3D Shape Fitting	79
3.3	Related Work	84
3.4	Conclusion	86

II	Interactions Between Hand and Environment	87
4	PLANNING MOTIONS THAT EXPLOIT CONTACT WITH THE ENVIRONMENT	90
4.1	An Algorithm for Interleaving Motion in Contact and in Free Space: Contact-Exploiting RRT (CERRT)	91
4.2	Experiments	100
4.3	Related Work	105
4.4	Conclusion	108
5	PLANNING GRASP STRATEGIES THAT EXPLOIT THE ENVIRONMENT	109
5.1	Grasp Strategies That Exploit the Environment	110
5.2	Planning Environmental-Constrained Grasps	116
5.3	Identifying Environmental Constraints From RGB-D Measurements	118
5.4	Sequencing Environmental Constraint Exploitations	120
5.5	From ECE Sequences to Hybrid Automata	122
5.6	Experiments	124
5.7	Related Work	131
5.8	Conclusion	133
III	Interactions Between Hand, Object and Environment	135
6	COMBINING PROPERTIES OF OBJECT AND ENVIRONMENT TO ADAPT GRASP STRATEGIES	140
6.1	Model-Based Adaptation of Surface-Constrained Grasps	141
6.2	Adapting Strategies Through Learning	148
6.3	Related Work	153
6.4	Conclusion	155
7	SELECTING ENVIRONMENT-CONSTRAINED GRASP STRATEGIES BASED ON OBJECT PROPERTIES	157
7.1	Features for Selecting the Right EC Exploitation	158
7.2	Improving the Learning Process	169
7.3	Limitations	176
7.4	Related Work	177

7.5	Conclusion	178
8	BEYOND GRASPING: LEARNING DEXTEROUS MANIPULATION FOR SOFT HANDS	179
8.1	Overview of the Algorithm	180
8.2	Learning Controllers From Multiple Demonstrations	181
8.3	Controller Optimization With an LfD Objective	185
8.4	Supervised Learning Using Guided Policy Search	187
8.5	An Illustrative Example	187
8.6	Controlling the RBO Hand 2	189
8.7	Experiments	190
8.8	Related Work	195
8.9	Conclusions	197
9	CONCLUSION	198
9.1	Discussion of Grasping Challenges	200
9.2	Limitations and Directions for Future Research	201
REFERENCES		218

List of Figures

1	Number of robotic publications about grasping	2
1.1	Recently, more and more underactuated robotic hands are being developed.	13
1.2	Robotic hands used in this thesis	14
1.3	Examples of hand adaptability for the RBO Hand 2	15
1.4	Comparison of the capture regions of the Barrett Hand 262 and its compliant counterpart	17
1.5	Formalization of a grasp	18
1.6	Diagram of force and form-closure grasps	23
1.7	ε -metric of two example grasps	25
1.8	Diagram of grasp properties	27
2.1	Mitten thought experiment	43
2.2	Object sets for human-in-the-loop grasp experiments	46
2.3	Structure of the human teleoperation trials	49
2.4	Setup of the Meka teleoperation experiment	51
2.5	Cartoons illustrating the effect of complementarity of compliance mode and object shape on grasp success	56
2.6	Example grasps for different regions of the SFA cartoon	57
2.7	Simulated grasp in 2D	58
2.8	Results of grasp simulations in 2D	59
2.9	Effects on SFA for varying noise of the approach direction	60
2.10	Grasps and shapes which are sensitive to the pre-grasp pose of the hand	61
2.11	Grasp quality as a function of object size and pose error	63
2.12	The funnel view of grasping	66
3.1	Compliance modes of the Barrett Hand	70
3.2	Objects used in the grasping experiments	74
3.3	Correlation between shape resemblance and grasp success	75

3.4	Comparison between our approach and the Eigengrasp planner	77
3.5	Experimental object test set	81
3.6	Grasping results of different compliance modes	82
4.1	Contact can efficiently reduce the uncertainty about the robot's state.	91
4.2	Cartoons illustrating uncertainty reduction through contact	93
4.3	Exploration of free-space and contact space in a cube	93
4.4	Cartoons illustrating a free-space move, a guarded move, and sliding.	95
4.5	Average planning time of our approach	101
4.6	Solutions found by our approach for the grasping POMDP problem	102
5.1	Different planned options to grasp a banana	110
5.2	Surface-constrained grasp strategy	111
5.3	Edge-constrained grasp strategy	114
5.4	Wall-constrained grasp strategy	115
5.5	Illustration of planning environmental constraint exploitations for grasping	116
5.6	Detecting environmental constraints from RGB-D sensor data	119
5.7	Comparison of EC-agnostic with EC-based strategies	125
5.8	Success of wall-constrained grasp w.r.t. varying wall inclinations	127
5.9	Example plans in different human environments.	128
5.10	Experimental banana scene	129
5.11	Graph showing possible sequences of EC exploitations	129
5.12	Comparison between grasp solutions found by CERRT and ECE planning	134
5.13	Effect of interactions between objects and environment on object pose and geometry.	138
6.1	Four different pre-grasp manifolds	141
6.2	Formalization of the environmental adaption to a surface	142
6.3	Test objects and cluttered scene used in the experiments.	144
6.4	Experimental setup	145
6.5	Results of the surface-constrained grasp adaptation experiment	145
6.6	Example grasps planned with our approach	146
6.7	Grasps chosen to clear the cluttered scene	147
6.8	Results of the cross-entropy method in simulation	151
6.9	Cross-entropy method for a wall-constrained grasp	152

6.10	Real-world experiment with cross-entropy method	152
6.11	Exemplary failure and success cases for the force-compliant closing strategy	156
7.1	Deciding between grasp options	158
7.2	Simulation setup for learning experiment	162
7.3	Real-world experimental setup	162
7.4	Success rates of different grasping strategies in simulation	164
7.5	Success rates of different grasping strategies in the real world scenario	165
7.6	Superellipsoids used for testing classifiers	167
7.7	Results of applying the trained classifiers on simulated sensor data of the superellipsoids	168
7.8	Comparison of different bandit algorithms	174
7.9	Ablation study of bandit algorithms	175
8.1	Illustration of the learning algorithm with a point-mass system	188
8.2	Comparison of our approach with baselines for valve task	192
8.3	Our approach learning to grasp a bottle	194
9.1	The funnel view of grasping	200

List of Algorithms

4.1	CERRT	94
4.2	CERRT – NEW_STATE	96
4.3	CERRT – SLIDE	98
5.4	Grasp Planning based on EC exploitation	117
6.5	Cross-Entropy Method for EC-based grasping strategies	150
7.6	Contextual Multi-Armed Bandits for EC Grasping	170
8.7	Guided Policy Search with Demonstration Selection	181

List of Tables

1	Contributions of this thesis	9
1.1	Grasp planning principles based on hand adaptability	30
1.2	Examples of different shape approximations used for planning grasps	32
2.1	Experiments about the effect of adaptability in grasping	45
2.2	Effect of number of particles on final grasp posture of power grasps	52
4.1	Symbols used in this chapter	92
4.2	Effect of number of particles on final position error	101
7.1	Accuracy ($F_{0.5}$ -scores) for different grasp features	166
8.1	Comparison of our approach with baselines for the abacus task	193

List of Videos

1	Examples of human exploitation of environmental constraints during grasping	6
	https://youtu.be/5w9n5xVTFTo	
1.1	Comparison between the Barrett Hand 262 and its soft counterpart	16
	https://youtu.be/Clyh9mAzzRQ	
2.1	Teleoperation experiment with the Allegro Hand	48
	https://youtu.be/4gdAMPVCCsY	
3.1	Example grasps of our compliance-centric algorithm	76
	https://youtu.be/Pm1A5AMBH_0	
3.2	Example grasps of the Eigengrasp planner	76
	https://youtu.be/a9IIB87wZ9w	
3.3	Segmentation and primitive fitting for grasp planning	80
	https://youtu.be/XboHUM6Y5Ps	
4.1	Different solutions found by CERRT for pre-grasp scenario	103
	https://youtu.be/WCMOISZPZ5s	
4.2	Real-world experiment using our approach on a 7-DOF arm	104
	https://youtu.be/CXaN8ZWRMT0	
5.1	Sequences of executing three plans found in the banana scene	130
	https://youtu.be/Va000JS9bx0	
5.2	Simulation of falling mugs	137
	https://youtu.be/8VNIOqsyXj4	
6.1	Example of surface adaption	143
	https://youtu.be/9fDFCaQ8WYE	
6.2	CEM for wall grasp	150
	https://youtu.be/zmsS0Bx-6x8	

7.1	Selecting EC exploitation based on object features	159
	https://youtu.be/I_RXzIcx6M	
7.2	The surface-constrained grasp with the RBO Hand 2	163
	https://youtu.be/HsD2yBxxWpg	
7.3	The edge-constrained grasp with the RBO Hand 2	163
	https://youtu.be/I3f63Ve2b9U	
7.4	The wall-constrained grasp with the RBO Hand 2	163
	https://youtu.be/hz3My08I5P0	
8.1	Three manipulation tasks for the RBO Hand 2	191
	https://youtu.be/XyZFkJwu0Q0	

This thesis contains figures with links to external videos. This is meant to enhance understanding of the material. Grasping is a very dynamic and complex process and a video often conveys information more easily than written words or static images. However, nowhere in this thesis it is required to watch these videos in order to follow the flow of arguments.

All videos can also be found here:

https://www.youtube.com/playlist?list=PLzPHJWMXrQM7_-A0vx26nX1PDzGLkoYZb

Notation and Abbreviations

v	A vector
M	A matrix
S	A set
$SE(3)$	The special Euclidean group $SE(3)$ is the symmetry group of three-dimensional Euclidean space. An element of this group is used to describe the motion of a rigid body.
$SE(2)$	The special Euclidean group $SE(2)$ describes all translations and rotations in the plane. It is homeomorphic to $\mathbb{R}^2 \times S^1$.
DOF	Degree of freedom. The number of independent parameters that define the configuration of a mechanical system.
EC	Environmental constraint. A feature of the environment that enables replacing aspects of control and/or perception with interaction between hand and environment.
ECE	Environmental constraint exploitation. The act of using an environmental constraint.
SFA	Sun-Flower-Annulus. A hypothesis about grasp success as a function of object shape presented in Sec. 2.2.
CERRT	Contact-exploiting rapidly-exploring random tree. A motion planning algorithm presented in Chapter 4. It is based on the rapidly-exploring random tree (RRT) by LaValle (1998).

All titles in this thesis are capitalized according to the “The New York Times Manual of Style and Usage”, A.M. Siegal and W.G. Connolly, Three Rivers Press, 5th edition, 2015.

I will use the first-person plural narrative “we” throughout this thesis, since a large part of the presented work is collaborative and to keep the language consistent.

0

Introduction

MORAVEC'S PARADOX (Moravec, 1988) is the insight that recreating higher-level reasoning requires much less computational resources compared to low-level sensorimotor skills. Success stories in the field of artificial intelligence have confirmed Moravec's paradox: machines beat humans in chess (Campbell et al., 2002), jeopardy (Ferrucci et al., 2010) and go (Silver et al., 2016), yet in tasks such as walking and manipulation they are easily outperformed by a five-year old. To put it with the words of Minsky (1986): "we're more aware of simple processes that don't work well than of complex ones that work flawlessly." Grasping is one such unconscious process and synthesizing it is the central topic of this thesis.

Some roboticists might argue that "grasping is solved"^{*}, so why bother writing a thesis about it? The fact that general robot grasping might still be an open problem can be easily verified when looking at industry. Logistics companies actively work on solutions for robust grasping, yet no general solutions exist. Their warehouses are largely automated with conveyor systems moving objects from a to b. But when it comes to picking up different objects

*This controversial statement was made by roboticist Gil Pratt during a plenary talk at the International Conference on Intelligent Robots and Systems 2012. He later went into detail saying: "I don't regret saying that. [...] What I had meant is that the program that I ran [at DARPA] was on grasping and manipulation, where once you grasp the thing, so you can pick it up, you need to have skills to turn the key, or to operate the tool, or to do anything else. The latter, the manipulation at that point, was still very much not solved - all the different dynamic behaviors that you need to do with the grasper in order to do a complete task in time. But the fundamental problem of picking this thing up had become easier, and people had mostly figured out how to do it." (<http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/gill-pratt-on-toyota-robot-plans>)

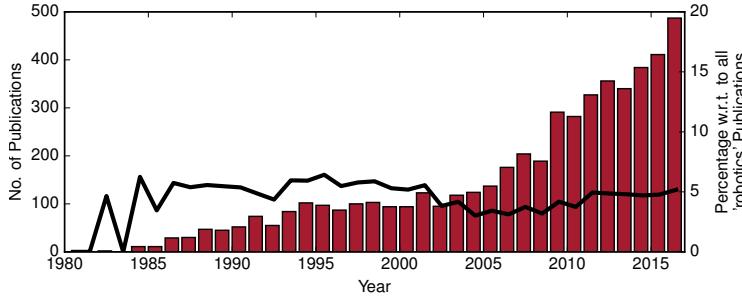


Figure 1: Number of new publications in robotics about grasping per year: The absolute amount of new publications constantly increases (red bars). A decline in scientific interest is also not visible when considering the relative percentage (black line). The numbers are based on a keyword search in the IEEE Xplore digital library.

and putting them into totes and parcels, human workers are still indispensable. Similar to industry, the scientific research community continues to actively work on the “solved problem” of grasping since Pratt’s remark in 2012. Fig. 1 shows that more and more researchers dive into the grasping problem as the field grows.

The continued interest in the grasping problem is also due to its complexity. This complexity stems from the fact that robot grasping is inherently a systemic problem: Any solution relies on an interplay between mechanism design, low-level control, perception and planning algorithms. Apart from the systemic nature of the grasping problem, we identify three main challenges:

GRASP DECISIONS ARE HIGH-DIMENSIONAL: The human hand has 23 degrees of freedom (Napier, 1956). Grasping an object requires a coordinated movement of all these individual DOF. A grasp decision also includes where to contact the object surface, which parts of the hand to use, and ultimately which forces to apply. Additionally, all of these decisions might vary on a temporal scale. Imagine picking up a coin for example: We first lift one end of the coin using the index finger’s nail while the thumb fixates the opposite end. Once the coin is vertical, the tip of the index slides down and forms a pinch grasp together with the thumb. One key question is how to formulate grasp representations that are expressive enough to cope with the large variety of grasping scenarios while at the same time enabling an effective search of this high-dimensional space.

CONTACT IS HARD TO MODEL: Grasping inevitably involves contact as hand and object touch each other. The impact itself is characterized by an abrupt change in the velocities of the contacting bodies, high forces, short duration, and rapid energy dissipation. Thus, the resulting dynamics contain non-linearities and discontinuities which are difficult to model and which do not lend themselves to gradient-based optimization methods. After the im-

pact the contact mechanics are characterized by interacting surfaces and the resulting friction phenomena. Friction is so complex that an entire research field is devoted to its study (tribology, e.g. Bhushan (2013)). The combination of continuous and discrete states in the physics of grasping led Ritter et al. (2007) to even call grasping the “Rosetta Stone” for designing cognitive robotic architectures.

INFORMATION IS ALWAYS INCOMPLETE: Apart from the fact that the grasping process has a complex structure due to contact phenomena, it also depends on a multitude of parameters that vary from object to object. These properties include: shape, mass, inertia, friction, stiffness, etc. They need to be known for the object, environment, and hand to make accurate predictions and plan robust grasps. Gaining information about these quantities with a robotic system is challenging, since measurements are noisy and observations are limited. Visual sensors e.g. are prone to occlusions and depend on the selected view point. Tactile or haptic sensing relies on contact which might be difficult to achieve in the first place. Furthermore, if we are interested in less structured environments than factory floors (e.g. households) the limitations due to missing a priori knowledge and on-board sensing are even more severe.

0.1 GOAL OF THE THESIS

In this thesis we tackle the problem of planning robust grasps for arbitrary objects. We assume knowledge about the kinematics and dynamics of the robot. We will focus our experimental evaluation on multi-fingered robot hands, although the introduced concepts and insights are often independent of the exact nature of the hand. The planned grasps do not satisfy any task constraints. Instead they are supposed to meet general pick-and-place demands. We focus on scenarios with unknown objects and little prior knowledge, grasp planning will be based on on-board sensor measurements.

We investigate two main ideas to solve the challenges associated with robot grasping:

1. How to exploit the compliance and adaptability of robotic hands?
2. How to exploit the stiffness and predictability of the environment for grasping?

In the following we will explain those two ideas in more detail.

0.2 EXPLOITING THE COMPLIANCE AND ADAPTABILITY OF ROBOT HANDS

Compliance can be defined in various ways. According to Merriam Webster, compliance is “the ability of an object to yield elastically when a force is applied”[†]. In physics, the mechanical compliance of a point is defined as the ratio of its displacement from an equilibrium state and the force that is causing it. If a small force causes a large displacement the point is highly compliant. The inverse of compliance is called stiffness. Compliance in robot hands can be implemented in various ways. Passive compliance is realized by using elastic materials for links or adding springs to the joint actuators. Active compliance is achieved through control. In this thesis we use the terms compliance and adaptability interchangeably and in the broadest possible sense: Compliance describes the ability of the grasper and the object to match each other’s shape in response to contact forces.

In robots as well as in humans, grasp success is greatly affected by the ability of the hand to adapt to the shape of the object. This adaptation increases the contact area and thereby the robustness of the grasp. The positive effect of shape adaptation on grasp success motivates the design of compliant, underactuated gripper devices: The tendon-driven SDM hand ([Dollar and Howe, 2010](#)) mechanically balances contact forces among its four flexible fingers. A gripper based on the jamming of granular material ([Brown et al., 2010](#)) can conform to a wide variety of object shapes. [Deimel and Brock \(2016\)](#) developed a soft pneumatic hand that adapts to the object’s shape through inflation. All of these examples show robust grasp performance through shape adaptation implemented in hardware, without explicit control or planning.

The first core idea of this thesis is to exploit the compliance of robot hands for grasping. The inherent low-level adaptation between hand and object can simplify perception and planning. Instead of planning precise contact locations based on high-fidelity models, it is sufficient to plan rough grasping poses. This in turn reduces the burden on perception, since only an approximate estimate of the object’s shape is needed. Ultimately, this approximation should exploit the specific capabilities of hand adaptation. Humans also exploit their adaptability during grasping as shown by [Christopoulos and Schrater \(2009\)](#). In this study, human graspers were confronted with noisy object positions. In response the subjects aligned their hand’s approach direction with the dimension of largest variance in object position.

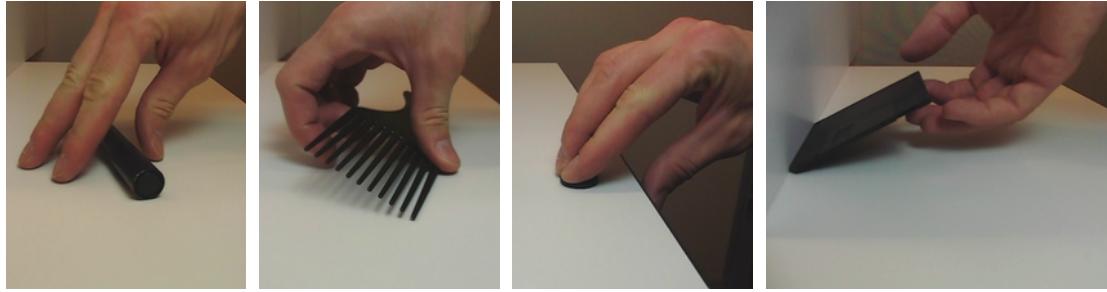
[†]<https://www.merriam-webster.com/dictionary/compliance>

0.3 EXPLOITING THE STIFFNESS OF THE ENVIRONMENT

The second core idea of this thesis focuses on the environment during grasping. We want to exploit the fact that the recurring structure and properties of the environment can help to attain a successful grasp. Objects in everyday life are situated in a context. They do not exist in free space — they are located on tables, in shelves, drawers, boxes, bowls, etc. These planar and curved surfaces, edges and corners are environmental structures that can be thought as an additional finger during grasping. Their high stiffness and low friction can be used to enable robust grasping strategies that involve pressing objects against a surface, sliding them towards an edge etc. Throughout this thesis we will refer to such an environmental structure as an environmental constraint (EC). In a more general sense an environmental constraint is a feature of the environment that enables replacing aspects of control and/or perception with interaction between hand and environment. We call the *exploitation of environmental constraints* the usage of any number of ECs to replace aspects of control and/or perception with interaction between hand and environment.

In robotic grasping the idea of exploiting ECs has not been extensively investigated, although a few examples exist. Kazemi et al. (2014) describe a grasping strategy for lifting thin and flat objects from a table. The strategy tries to keep contact between fingertips and the support surface during finger closing. In this case the environment guides the fingertips towards the object surface, which would be more difficult using pure position control in free space. Kappler et al. (2010) present another strategy which pulls a compact disk towards the edge of a table to pick it up. This kind of pre-manipulation allows the thumb to create contact forces underneath the object. For underactuated mechanisms contact with the environment can help attain pre-grasp configurations, as shown with the PISA/IIT Hand when grasping a book (Bonilla et al., 2014). In contrast to exploiting the environment, a much more common approach in grasping is to consider the environment as an obstacle and avoid contact with it (Miller and Allen, 2004).

The idea of exploiting contact with the environment to simplify and solve robotic motion tasks is not new. In the bracing strategy (Book et al., 1985) the lower part of a robot arm (which is usually heavy and less agile) first attaches to a rigid structure in the environment. Once attached, a lighter structure at the end of the manipulator can be actuated and fulfill fine motion requirements. Another common control strategy are so-called guarded moves (Will and Grossman, 1975). These linear motions are performed until an easily detectable contact event with the environment occurs. The combination of guarded moves can lead to complex



Video Figure 1: One core idea of this thesis is to simplify robot grasping by making extensive use of the environment. We call this the exploitation of environmental constraints. Human grasping often also benefits from this effect as can be seen in the examples above. [<https://youtu.be/5w9n5xVTfTo>]

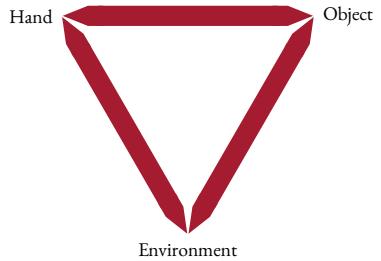
motion behaviors that succeed despite significant uncertainty of the system state (Lozano-Pérez et al., 1984). Finally, humanoid morphologies can also increase stability and robustness by contacting the environment. Borras and Asfour (2015) categorize body configurations in locomotion and manipulation tasks based on multiple contacts.

The exploitation of ECs to improve grasp success can also be observed in humans and animals. Deimel et al. (2013) analyzed humans grasping objects from a sensorized surface when subjects are visually impaired. They showed that humans with limited visual feedback deliberately increase contact with the support surface on which the object is positioned to compensate uncertainty and increase grasp success. This is reflected in an increase of applied forces, length and duration of fingertip contact traces, and time from first contact to object lift. In the animal kingdom exploitation of ECs is also present. Raccoons exhibit great dexterity despite the kinematic limitations of their hands. They do not have a real opposable thumb and as a consequence use a lot of “flat” grasps. But by exploiting environmental constraints they increase their dexterity and grasping repertoire: “For example, a long stick will be pushed or pulled along the ground, constrained against a fixed object (a stone, wall, or perhaps the raccoon’s body) before being scooped up in the fingers.” (Walker, 1995)

The idea of focusing on the environment for grasping complements our first theme in this thesis: the exploitation of adaptability in hands. Hands that are compliant can get into contact with the environment more easily since catastrophic outcomes are less likely. Nevertheless, the idea of environmental constraint exploitation is not limited to soft hands and we will prove this by showing experiments with stiff end-effectors (see e.g. Chapter 5).

0.4 CONTENT AND CONTRIBUTIONS OF THE THESIS

The two main ideas of this thesis can be expressed as the relationships between the different actors that are present in grasping: hand, object, and environment. The relationship between hand and object is governed by the hand's adaptability, while the relation between hand and environment is dominated by the environment's stiffness. After investigating these binary relationships, we look at the combined effects of hand, object, and environment and present a grasp planning method that takes all of them into account. We will treat the relationship between object and environment only briefly (introduction of Part III), since characterizing it is not a robotic problem per se. Accordingly, we organize the thesis into three separate parts:



Part I investigates the interactions between hand and object during grasping. We start with a series of small human grasping experiments, in which the role and importance of hand adaptability becomes visible. We complement these experiments with a large amount of simulations in which adaptability in the face of varying object shape is evaluated. These initial investigations show the low-dimensionality of grasping due to hand adaptability. Based on this insight we devise two grasp planning algorithms which extract the necessary information to execute robotic grasps from raw sensor input. These algorithms differ in their sensor requirements (color vs. depth images), but they both exploit compliance and work without strong assumptions about the availability of prior object models.

Part II is based on our second core idea of exploiting the stiffness of the environment. The investigation is focused on the interactions between hand and environment, ignoring any notion of object. We will present two methods for exploiting the environment: the first one targets motion planning problems, while the second is more tailored to grasping scenarios. We experimentally evaluate both algorithm in simulation and the real world and draw conclusions about their advantages and limitations.

Part III extends the methods and insights derived in the previous parts by focusing on the ternary relationship between hand, object and environment. We will present methods that adapt grasping strategies locally to concurrently comply with the shape of the object and the environment. Furthermore, we present a method that globally selects which environmental constraints to exploit in an unknown scene based on object properties. This approach improves grasp performance over time by incorporating past experiences. Finally, we are

looking at simple manipulations beyond the realm of grasping. The compliant hands we are using for grasping in this thesis exhibit significant challenges when it comes to dexterous skills due to their underactuation and sensing limitations. The part finishes with an approach that tackles these challenges through a sample-efficient trial-and-error learning procedure.

In the following table we list the different conceptual, technical, and empirical contributions of this thesis in detail:

Chapter	Sec.	Contributions
Hand – Object	1	1.1 An overview of hand design features that provide adaptability
	1.3	A taxonomy of seven principles used by grasp planning algorithms to increase adaptability
	2	2.2 A hypothesis about the relationship between object shape and grasp success, which we call the <i>Sun-Flower-Annulus</i> hypothesis
	2.3	A view on the problem of grasping as a sequence of uncertainty-reducing funnel operations
	3	3.1 A method to synthesize grasps based on the change of contour of an object when being viewed by an active camera
	3.2	A grasp planning approach that matches arbitrary 3D measurements with hand pre-shapes using geometric prototypes
	4	A planning method that finds motions from A to B. It assumes a noisy motion model, a geometric model of the environment and access to an uncertainty-free contact signal. As a result it finds motions that combine classical free-space paths with segments of uncertainty-reducing contact actions (Algorithm 4.1).
	5	5.1 A characterization of different grasping strategies that explicitly exploit the environment. It is based on the concept of environmental constraints (ECs).
	5.2	A planning method that finds grasping motions based on sequences of contact exploitations (Algorithm 5.4). The method does not assume an a priori model of the environment but builds a model from RGB-D data.

	Chapter	Sec.	Contributions
Hand – Object – Environment	6		Two methods to adapt grasping strategies to the environment, a model-based method for stiff hands (Sec. 6.1) and a learning-based method for soft end-effectors (Sec. 6.2)
	7	7.1	A predictive model to select the best ECEs for sensory inputs of novel objects. It includes an evaluation of different geometric features.
		7.2	A formalization of the ECE selection problem as a contextual multi-armed bandit problem and an extensive evaluation of multiple common exploration schemes
	8		A trial-and-error-based algorithm to learn more general manipulation policies for a soft hand using human demonstrations

Table 1: Contributions of this thesis

Before starting with Part I, we review hands, theory, and algorithms in Chapter 1. We will do this from the perspective of our proposed focus on hand adaptability and environmental stiffness.

1

Background and Related Work

ROBOTIC GRASPING is an extensively studied phenomenon that spans hardware, theory, and algorithms. While computer-controlled hands were invented as early as 1962 (Ernst, 1962), the analytical origin of grasping dates back to Reuleaux (1876) who studied contact conditions in bearings and mechanisms. Rather than repeating a condensed version of the existing literature we will base our review on the first core idea of this thesis: How can adaptability and compliance help in grasping?

1.1 ROBOTIC HANDS FOR GRASPING

Although this thesis mainly deals with the algorithmic aspects of grasp planning, we need to look at current hand designs if we want to exploit their ability to adapt and comply. We identified four different design features that impact adaptability and thereby grasp performance. These features relate to the size of hands, surface properties, geometry, and actuation method. After explaining them and giving examples, we give a quick overview of the hands that are used in the experiments of this thesis. Finally, we will show a case where compliance is detrimental to grasp success.

1.1.1 HAND SCALE

The opportunity of a hand to adapt is strongly constrained by the size difference between itself and the objects it is supposed to grasp. Enveloping grasps that create large contact ar-

eas with the object benefit from sufficiently big hands. Longer fingers also allow to capture objects despite significant pose uncertainty. At the same time increasing size generates problems: Larger and heavier hands limit dynamic bandwidth, reduce the maximum payload due to the arm's capabilities and create fewer possibilities to move the arm around obstacles.

To find the sweet spot robotic hand designs usually refer to the dimensions of the human hand, since most objects of interest are made by humans for humans. Assuming that anthropomorphism contributes to functional capabilities, how important is hand scale? [Biagiotti et al. \(2004\)](#) answer this question by proposing an anthropomorphism index for robotic hands in which size makes up one fifth of the final value. [Feix et al. \(2014\)](#) take a more empirical look at the average size of objects that humans manipulate in everyday life. They collected ~ 9000 grasps during 32 hours of manual labor from two housekeepers and two machinists. Most of the 306 analyzed objects fitted inside a bounding box of dimensions $15\text{ cm} \times 6\text{ cm} \times 5\text{ cm}$. Although the largest object dimension exceeded the maximum grip aperture of the human hand in half of the cases, the experiments also showed that the chosen grasps opened the hand less than 10 cm in 98% of the cases (and even less than 5 cm in 83% of the cases).

Instead of reverting to a single sized hand, [Odhner et al. \(2012\)](#) present the concept of a recursively scaled-down gripper. Similar to a fractal, a two-fingered gripper is mounted on the fingertip of another two-fingered gripper twice the size. This design is an example of increasing shape adaptability by leveraging the right hand scale for a given object. We will show in Sec. 2.2 the importance of hand size for adaptability and grasp success.

I.I.2 HAND SURFACE

Human evolution shows the importance of contact surfaces for manipulation: While the hands of our old ancestors have only few patches of specialized skin and pulp on the palmar surface these features are much more prominent in more advanced species such as chimpanzees. In robotic hands, only few works have systematically investigated the effect of different surface materials. [Shimoga and Goldenberg \(1992\)](#) show that soft materials are beneficial due to three reasons: they attenuate impact forces during grasping, conform to uneven object surfaces, and dissipate repetitive strain during manipulation. They compare six different materials and conclude that sponge and gel fingertips work best while plastic ones are the least suitable. [Cutkosky et al. \(1987\)](#) argue in favor of fingerprints for robotic skin by showing that a textured surface increases friction under moist conditions. Robotic hands are often equipped with gloves to supply contact compliance and grip, such as the working glove with

padded rubber surfaces used with the PISA/IIT SoftHand ([Catalano et al., 2014](#)).

1.1.3 HAND GEOMETRY

Apart from the surface properties concerning friction and elasticity of hands the geometry plays a fundamental role when designing adaptable hands. [Rodriguez \(2013\)](#) proposed an algorithm that calculates the shape of a rigid finger which maximizes the grasp probability for a given object with uncertain pose and size. The resulting finger deviates from the usual cylindrical shape – it is formed like a hook and was also empirically validated in a simple 3-DOF gripper ([Rodriguez and Mason, 2012](#)). [Morrow et al. \(2016\)](#) analyzed the effect of differently shaped fingertips for a pneumatic actuator. They compared six different shapes, showing that the geometry of the "fingernail" significantly affects grasp success (the best design could grasp nearly twice as many objects as a finger without a nail). The best performing geometries were a dustpan-inspired one and a wedge-shaped nail. They could get under objects and push them onto the actuator. Another purely geometric effect can be observed when looking at sensorless manipulation in the form of vibratory bowl feeders ([Boothroyd, 1991](#)) which are often used in industrial automation pipelines. Here, parts are reoriented by passing a sequence of geometric features along a track specifically designed for that part. The geometry of the "grasping mechanism" creates contacts with the object that either rotate it or eject it, outputting an object in a desired orientation.

1.1.4 HAND ACTUATION

Adaptability can be easily increased by introducing more degrees of freedom (DOF) into a hand design. Two early examples are the 6-DOF finger of the Soft Gripper ([Hirose and Umetani, 1978](#)) and the Omnidripper ([Scott, 1985](#)) with 72 prismatic joints. These mechanisms can match a large variety of object shapes. But additional motorized joints also come at the price of increased complexity and the burden of planning and control. An alternative solution is to use *underactuation*, i.e., mechanisms with fewer degrees of actuation than degrees of freedom. Fig. 1.1 shows that the amount of newly developed underactuated robot hands has significantly increased in recent years. Underactuation can be implemented in multiple non-exclusive ways. We follow the discrimination proposed by [Krut \(2005\)](#), who distinguishes underactuation in differential, triggered, and compliant mechanisms.

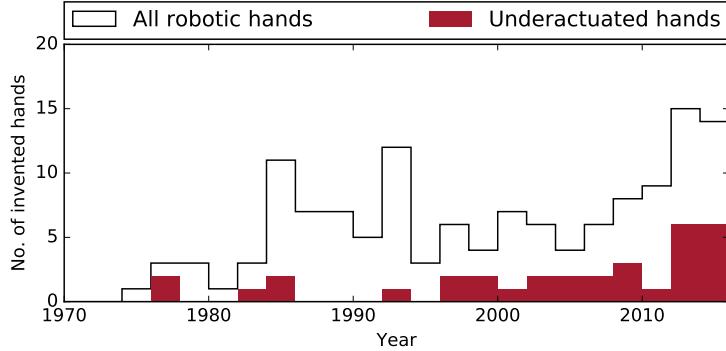


Figure 1.1: Recently, more and more underactuated robotic hands are being developed. Numbers are based on data collected by the author, also available at www.robotgrasping.org/hands.

UNDERACTUATION: DIFFERENTIAL MECHANISMS “A differential mechanism is a mechanism in which the amount of dynamical inputs from three ports acts in balance” (Hirose, 1985) These mechanisms usually rely on pulleys and cables or linkages. Examples are the Soft Gripper (Hirose and Umetani, 1978), the SDM Hand (Dollar and Howe, 2010) and the Pisa/IIT SoftHand (Catalano et al., 2014).

UNDERACTUATION: TRIGGERED MECHANISMS Some hands include mechanisms which lock a joint once a certain threshold is exceeded and switch the torque transmission to different joints. The Barrett Hand (Townsend, 2000) for example contains a breakaway mechanism, which is activated if an external force is applied to the finger’s proximal phalanx. Once active, the motor torque is only transmitted to the distal phalanx which continues to close while the proximal link of the finger is locked.

UNDERACTUATION: COMPLIANT MECHANISMS This class of mechanisms includes hands made from soft materials. They often also contain less common actuation types based on pneumatics, hydraulics, electro-adhesion, etc. The Positive Pressure Gripper (Amend et al., 2012) is an extremely compliant example. It is based on a balloon filled with granular material which adapts to any object shape. Once the balloon is decompressed, the material jams and keeps its current shape. In experiments the gripper tolerates position errors of up to 72% of the balloon radius.

I.I.5 HANDS USED IN THIS THESIS

Grasping experiments throughout this thesis are conducted with different types of hands. They differ in the amount of adaptability they expose and how adaptability is implemented according to the principles listed in the previous section.

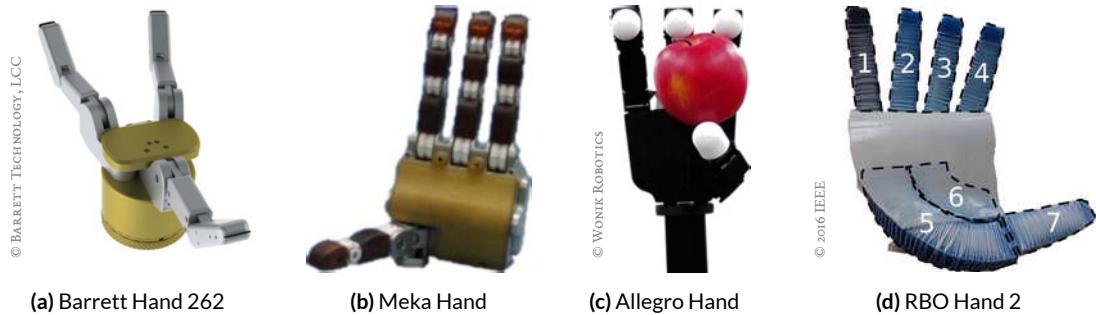


Figure 1.2: We use four different hands in the experimental evaluations of this thesis. They all expose some degree of adaptability, although implemented by various means. A common theme is underactuation: The Barrett Hand, Meka Hand, and RBO Hand 2 are all underactuated mechanisms. Another possible feature is scale. The Allegro Hand is 50% bigger than a common human hand.

BARRETT HAND 262 (TOWNSEND, 2000) The 4-DOF Barrett Hand 262 consists of three fingers and a palm, all made from aluminum. Each finger has two joints which are coupled and driven by a single motor via a tendon. This coupling can be mechanically removed if the closing direction of the proximal link is obstructed, in which case only the distal link continues to move. This type of underactuation gives the hand the ability to adapt to objects at the mechanical level (see Sec. 1.1.4). The fourth motor allows two fingers to rotate around the palm up to 180° and oppose the third finger.

MEKA H2 COMPLIANT HAND (MEKABOT, 2009) The Meka H2 Hand is tendon-driven, has three fingers and one opposable thumb. Each finger is driven by a single motor, the thumb by two, making it five DOF. It is roughly the size of a human hand. The fingers are made from flexible urethane. Adaptability is achieved through underactuation and compliant materials.

ALLEGRO HAND (BAE ET AL., 2012) The 16-DOF Allegro Hand consists of three fingers and an opposable thumb. Each finger contains four electrical motors. The hand is made from aluminum, the fingertips are coated with rubber. It is approximately 50 % bigger than a human hand. Adaptability is achieved through scale, surface friction, and a high number of DOF.

RBO HAND 2 (DEIMEL AND BROCK, 2014) The RBO Hand 2 is a pneumatically actuated anthropomorphic hand made out of silicon. It has four fingers and an opposable thumb. The hand consists of a polyamide scaffold to which multiple pneumatic actuators are attached.

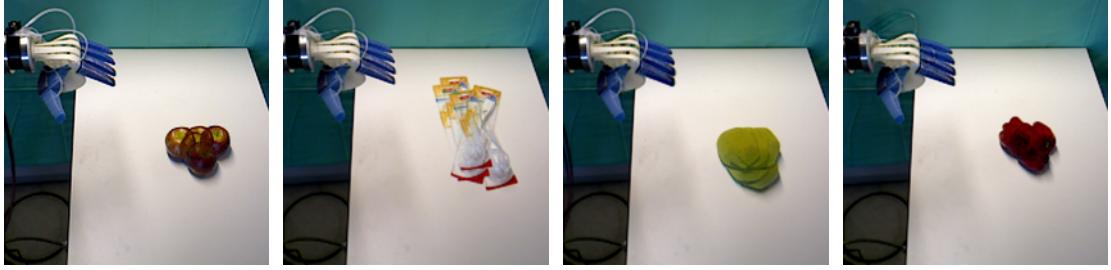


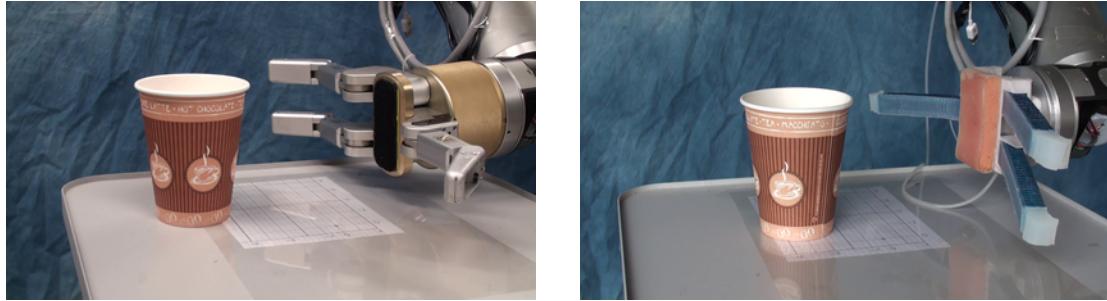
Figure 1.3: The images show the adaptability of the RBO Hand 2 during grasping. In this experiment we executed the exact same top-down grasp which simply inflates the hand multiple times. Throughout the trials we varied objects and their positions. In the panels we visualize the different object positions that could be grasped successfully by overlaying them (from left to right: apple, bottle brush, sponge, bell pepper).

Each of the four fingers is a single actuator, while the thumb consists of three independent pneumatic actuators (see Fig. I.2d). This makes the thumb the most dexterous part of the hand, achieving seven out of eight configurations of the Kapandji test (Kapandji, 1986). Each finger consists of a single inflatable air chamber the size of a human finger. The silicon envelope is reinforced with textile and an X-wounded thread, such that during inflation the finger bends in the axial direction while radial expansion is minimized. Fig. I.3 shows that the hand inherently adapts to different object shapes and poses during grasping.

I.I.6 NEGATIVE EFFECTS OF HAND COMPLIANCE ON GRASPING

Adding adaptability to robotic hands based on the four properties mentioned above not necessarily enhances grasp performance. There is a trade-off since parasitic effects such as compliance in unwanted directions or fingers with high friction sticking to the environment lead to unstable or no grasps at all.

We conducted an experiment in which the negative effects of compliance in a hand design outweighed the positive ones. The grasping capabilities of the hand prototype shown in Video Fig. I.1 were compared to those of the Barrett Hand 262. We designed the prototype such that it resembles the Barrett Hand 262 in terms of scale, finger arrangement and geometry. The main difference are the silicon-based pneumatic actuators (Deimel and Brock, 2016) used as fingers instead of the much stiffer two-link aluminum fingers of the Barrett Hand 262. We grasped twelve different objects with each hand, using two strategies: a top-down grasp and a push grasp approaching the object side-ways (see Video Fig. I.1). While the grasps were executed open-loop, we changed the position of the objects on the table surface in each trial. This way we could characterize the capture region of the hands assuming that the exact object



Video Figure 1.1: We evaluated grasp performance of the Barrett Hand 262 (**left**) with a soft version made out of silicone and pneumatic actuators (**right**). This experiment showed that increasing compliance not necessarily leads to better grasp performance. [<https://youtu.be/CLYh9mAZzRQ>]

position is never known. A larger capture region would mean that the hand could deal with more uncertainty without increasing the computational burden for planning or feedback-based control.

The plots in Fig. 1.4 show the capture regions per object for both hands and both grasp strategies. The Barrett Hand 262 nearly always grasps more reliable than its more compliant counterpart. The soft prototype failed due to multiple reasons. In the case of the push grasp the torsional and lateral compliance of the fingers resulted in too small contact forces to lift the water bottle (500 g). The upright standing tube is a rather thin object (2 cm) which is often too small given the relatively large curvature of the maximally inflated finger. As a result only few areas of contact between hand and object are created. Furthermore, the soft fingers are far from being force-isotropic, i.e., the strongest contact forces are created at the tip while the intensity of contact forces near the base are rather small. Thus, when the fingertips are not involved in a grasp it is less stable. The top-down strategy nearly always fails because of the high surface friction of the silicone. During inflation the fingertips get stuck in the table and most of the energy is converted into the bending of the fingers without enveloping the object. Once the hand is lifted from the table surface, the fingers abruptly curl, hitting the object but not creating a stable grasp.

It is evident that some of the described problems of the soft prototype could be alleviated by introducing or altering certain design features such as fingernails. But more importantly, the experiment shows that adding compliance and adaptability – here by changing the actuation of a more traditional stiff hand – not necessarily leads to higher grasp success. Instead, beneficial compliance and adaptability result from careful, oftentimes non-intuitive hardware designs in combination with appropriate grasping strategies.

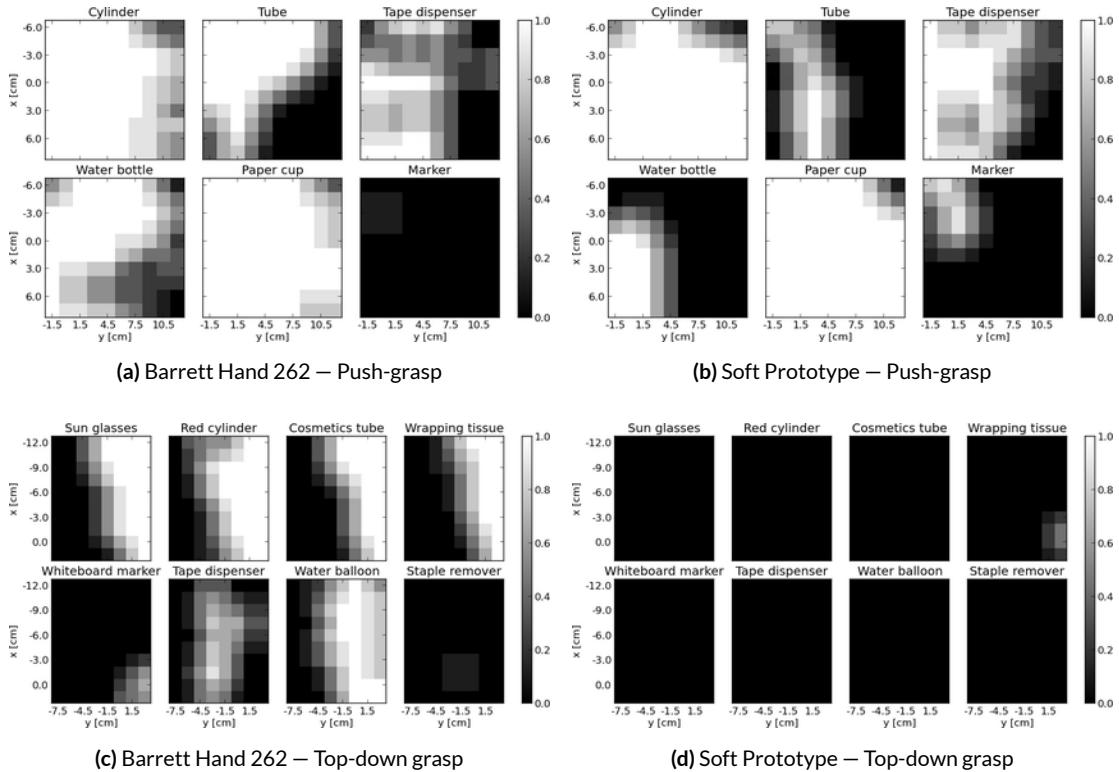


Figure 1.4: The plots show the capture regions of the Barrett Hand 262 (**left**) in comparison to the soft prototype (**right**). Each image describes the grasp success for a particular object under varying positions in the table plane. White indicates grasp success, black failure. The Barrett Hand 262 grasps more reliably, especially when using top-down grasps. In these cases the fingers of the soft prototype got stuck in the table due to their high surface friction.

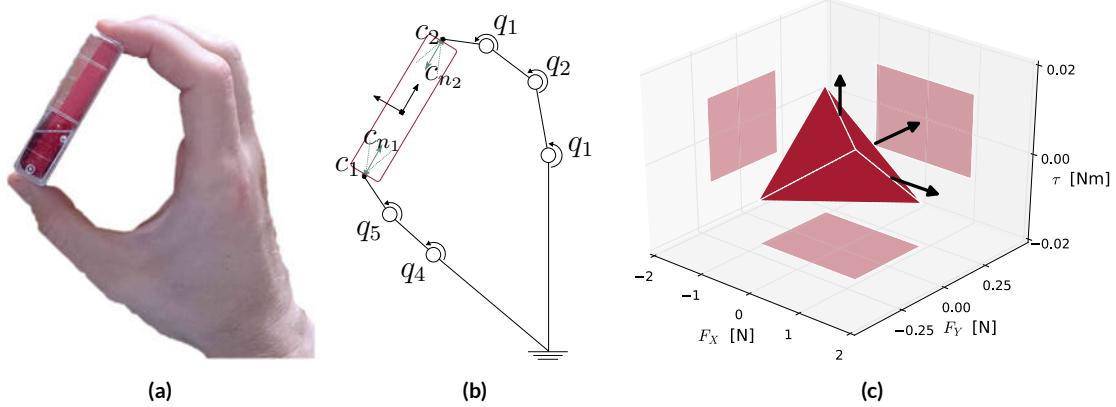


Figure 1.5: An example grasp of an audio tape (a), its formalization via two point contacts with friction (b) and the associated 3D polytope that represents the set of all applicable object wrenches (c). The plot includes the 2D projections along all three dimensions to improve visibility. Since the origin of the wrench space is inside the polytope this grasp is a force-closure grasp, i.e., it can balance external forces from any direction.

1.2 CLASSICAL GRASP AND RESTRAINT ANALYSIS

This section serves two purposes. First, we want to introduce the most common model of grasping, which provides a basis for many grasp planning algorithms and benchmarks. This formalization assumes rigid bodies, point contacts, and is quasistatic, i.e., inertial forces are ignored. At the heart of this model are the concepts of *force* and *form closure*, which we will use in the remainder of this thesis. We will focus on a brief and comprehensible explanation, more elaborate treatments can be found in [Murray et al. \(1994\)](#), [Mason \(2001\)](#), [Prattichizzo and Trinkle \(2008\)](#) and [Lynch and Park \(2017\)](#).

Second, we will highlight the limitations of this formalization when it comes to real-world grasping. This is closely related to the fact that hand adaptability and the potential role of the environment are not adequately reflected.

1.2.1 FORCE CLOSURE

In the following explanations we will assume a three-dimensional world and complement this with illustrations of a planar 2D example. The formal analysis of robotic grasps is based on the contact configuration between the grasping mechanism h and the object o . The contacts depend on the configuration $\mathbf{q}_o \in SE(3)$ of the object and the configuration of the hand $\mathbf{q}_h \in \mathbb{R}^n$ with n being the number of DOF. Each of the m resulting contacts is characterized by its location $\mathbf{c}_i \in \mathbb{R}^3$ and surface normal vector $\mathbf{c}_{n_i} \in \mathbb{R}^3$, $\|\mathbf{c}_{n_i}\|_2 = 1$. Fig. 1.5b

shows all the elements in a planar example grasp.

The force $\mathbf{f}_i \in \mathbb{R}^3$ and moment $\mathbf{m}_i \in \mathbb{R}^3$ that are imparted onto the object at each contact i can be described as a wrench $\mathbf{w}_i = (\mathbf{f}_i, \mathbf{m}_i)$. An arbitrary frame of reference can be chosen to express this wrench, but most commonly the center of mass or the geometric center of the object o are used. All wrenches that can be transmitted at the i -th contact form a wrench cone $\mathcal{WC}_i \subset \mathbb{R}^6$. The wrench cones depend on the assumed friction model. The most common friction models in grasping are:

- **FRICITIONLESS POINT CONTACT OR HARD FINGER.** This type of contact can only transmit forces along the contact normal:

$$\mathcal{WC}_i = \left\{ \begin{bmatrix} \alpha \mathbf{c}_{n_i} \\ \mathbf{c}_i \times \alpha \mathbf{c}_{n_i} \end{bmatrix} \in \mathbb{R}^6 \mid \alpha \geq 0 \right\},$$

where the term $\mathbf{c}_i \times \alpha \mathbf{c}_{n_i}$ represents the resulting moment of force, which depends on the contact location \mathbf{c}_i . The force intensity α needs to be positive since we do not allow pulling forces at a contact. Although frictionless contacts arguably never occur in practice they can be a useful conservative approximation in cases where the coefficient of friction is small or unknown.

- **POINT CONTACT WITH FRICTION.** This contact can transmit any force within the friction cone \mathcal{FC} defined by Coulomb's law. Aligning the axis of the friction cone with the contact normal \mathbf{c}_{n_i} results in the set of admissible forces

$$\mathcal{FC}_i = \left\{ \mathbf{f} \in \mathbb{R}^3 \mid \arccos \left(\frac{\mathbf{f}^\top \mathbf{c}_{n_i}}{\|\mathbf{f}\|} \right) \leq \arctan(\lambda), \mathbf{f}^\top \mathbf{c}_{n_i} > 0 \right\},$$

where the coefficient of friction $\lambda \in \mathbb{R}$ is an empirically derived constant based on the two interacting materials (usually between 0.1 and 1). The friction cone can be turned into a wrench cone at the object level, just as in the previous case of point contacts without friction:

$$\mathcal{WC}_i = \left\{ \begin{bmatrix} \mathbf{f} \\ \mathbf{c}_i \times \mathbf{f} \end{bmatrix} \in \mathbb{R}^6 \mid \mathbf{f} \in \mathcal{FC}_i \right\}.$$

- SOFT FINGER CONTACT. Besides transmitting tangential forces due to friction, the soft contact additionally allows to transmit a moment m around the contact normal due to torsional friction:

$$\mathcal{WC}_i = \left\{ \begin{bmatrix} \mathbf{f} \\ \mathbf{c}_i \times \mathbf{f} + m\mathbf{c}_{n_i} \end{bmatrix} \in \mathbb{R}^6 \mid \mathbf{f} \in \mathcal{FC}_i, |m| \leq \gamma \mathbf{f} \cdot \mathbf{c}_{n_i} \right\},$$

where γ is the torsional coefficient of friction. In the case of planar grasps the soft finger contact model and point contact with friction are the same.

The set of all wrenches that can be imparted on the object o through the contacts is the linear positive span of each contact's wrench cone:

$$\mathcal{WC} = pos(\{\mathcal{WC}_i\}) = \left\{ \sum_{i=0}^m k_i \mathbf{w}_i \mid \mathbf{w}_i \in \mathcal{WC}_i, k_i \in \mathbb{R}^{\geq 0} \right\}.$$

This composite wrench cone contains all possible wrenches that the contacts can exert to balance external disturbances. A special situation arises when a grasp can withstand object disturbances in any direction, i.e., the composite wrench cone contains the entire wrench space (\mathbb{R}^6 in spatial or \mathbb{R}^3 in planar grasps). Such a grasp is called a *force-closure* grasp. In the spatial case, at least seven wrenches are needed to positively span the six-dimensional wrench space, which can be achieved with three frictional point contacts (two are needed in the planar case). There are certain exceptional object shapes which require more contacts or cannot be force closed at all, such as surfaces of revolution.

The composite wrench cone assumes potentially infinitely large contact forces. But we can easily introduce limitations imposed by the hand. Most commonly, two types of limitations are considered: Either we assume that the total force is limited due to a single energy source that powers the hand or we assume that each contact force is limited independently. In both cases the composite wrench cone turns into a wrench polytope:

$$\begin{aligned} \mathcal{WP}^1 &= \left\{ \sum_{i=0}^m k_i \mathbf{w}_i \mid \mathbf{w}_i \in \mathcal{WC}_i, \|\mathbf{w}_i\|_2 = 1, k_i \in \mathbb{R}^{\geq 0}, \|\mathbf{k}\|_1 = k_{max} \right\}, \\ \mathcal{WP}^\infty &= \left\{ \sum_{i=0}^m k_i \mathbf{w}_i \mid \mathbf{w}_i \in \mathcal{WC}_i, \|\mathbf{w}_i\|_2 = 1, k_i \in \mathbb{R}^{\geq 0}, \|\mathbf{k}\|_\infty = k_{max} \right\}, \quad (1.1) \end{aligned}$$

where k_{max} is the limiting wrench magnitude. The composite wrench cone and wrench poly-

tope are often interchangeably called *grasp wrench space*. An example of the wrench polytope of a grasp is shown in Fig. 1.5c.

To algorithmically test the force-closure condition the set notation from above is not useful. Instead we introduce the grasp matrix $\mathbf{G} \in \mathbb{R}^{6 \times (m \cdot fc)}$, where fc is the number of faces of the polyhedral approximation of the friction cone. The grasp matrix is defined as

$$\mathbf{G} = \left[\underbrace{\mathbf{w}_{1_1} \cdots \mathbf{w}_{1_{fc}}}_{\text{contact 1}} \mid \underbrace{\mathbf{w}_{2_1} \cdots \mathbf{w}_{2_{fc}}}_{\text{contact 2}} \mid \cdots \mid \underbrace{\mathbf{w}_{m_1} \cdots \mathbf{w}_{m_{fc}}}_{\text{contact } m} \right]$$

and maps contact wrench intensities \mathbf{k} to the resulting object wrench: $\mathbf{w}_o = \mathbf{G}\mathbf{k}$. Given the grasp matrix, the following statements are identical:

- The grasp is force closure.
- $\mathcal{WC} = \mathbb{R}^6$.
- The origin of the wrench space is *inside* of \mathcal{WP} .
- \mathbf{G} is full rank and there is a solution to the equality

$$\mathbf{G}\mathbf{k} = \mathbf{0}, \text{ s.t. } \mathbf{k} > 0. \quad (1.2)$$

This also means that a force-closure grasp always generates internal forces.

We can see in Fig. 1.5c that the origin is inside of \mathcal{WP} , thus this grasp is a force-closure grasp. The force-closure property plays a dominant role in modeling robot grasping. Nevertheless, one should bear in mind its limitations and disadvantages (see Sec. 1.2.5).

1.2.2 FORM CLOSURE

In contrast to force closure, the concept of *form closure* is based on a purely kinematic and geometric account of grasp mechanics which ignores forces that occur at contacts. Roughly speaking, a grasp is in form closure if any instantaneous acceleration of the object o will create a collision with the grasp mechanism h , or as Mason (2001) defines it: “The object is at an isolated point in configuration space.”

We formalize form closure by introducing the gap function $g(\mathbf{q}_h, \mathbf{q}_o) : \mathbb{R}^n \times SE(3) \rightarrow \mathbb{R}^m$ which maps the configurations of hand and object onto the distances between them at

the m different contacts. The meaning of the i -th component of g is as follows:

$$g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o)_i \begin{cases} > 0, & \text{if hand and object are separated and free to move at contact } i. \\ = 0, & \text{if hand and object are touching each other at contact } i. \\ < 0, & \text{if hand and object are penetrating each other at contact } i. \end{cases}$$

For a general grasp $(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o) \in \mathbb{R}^n \times SE(3)$ the equation $g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o) = \mathbf{0}$ needs to hold, where equality refers to an element-wise comparison between the two vectors. Additionally, this grasp is in form closure if every infinitesimal change $d\mathbf{q}_o$ in the object configuration leads to a penetration with the hand:

$$d\mathbf{q}_o \neq \mathbf{0} \implies \exists i \in \{1, \dots, m\} : g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o + d\mathbf{q}_o)_i < 0.$$

This implication can be rewritten as its contrapositive:

$$\forall i \in \{1, \dots, m\} : g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o + d\mathbf{q}_o)_i \geq 0 \implies d\mathbf{q}_o = \mathbf{0}, \quad (1.3)$$

which means that any non-collision between object and hand must be due to a non-moving object. Similar to the concept of the wrench cone used in force closure we can use twist cones to express the condition for form closure. The twist cone \mathcal{TC}_i due to contact i is the set of all object twists that do not create a collision with contact i :

$$\mathcal{TC}_i = \{d\mathbf{q}_o \in se(3) \mid g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o + d\mathbf{q}_o) \geq 0\}$$

In a form-closure grasp all object pose changes result in at least one collision. The union of all feasible twist cones represents the possible twists that do not generate any collision. Thus, we can rewrite the condition for a form-closure grasp (Eq. 1.3) as:

$$\mathcal{TC} = \mathcal{TC}_1 \cap \mathcal{TC}_2 \cap \dots \cap \mathcal{TC}_m = \{\mathbf{0}\},$$

where the zero twist is the only collision-free twist allowed.

It is important to note that there are different *orders* of form closure depending on how we approximate the effect of the infinitesimal change in object pose. A Taylor series expansion

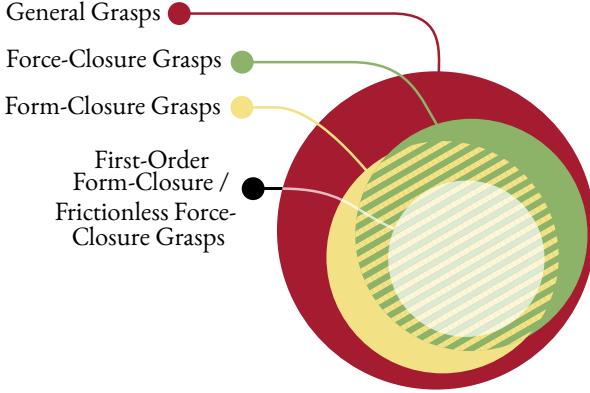


Figure 1.6: Diagram showing the relationship between form and force-closure grasps. Contrary to intuition, form closure is not a sufficient condition for force closure.

of the gap function g around $\bar{\mathbf{q}}_o$ gives:

$$\begin{aligned}
 g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o + \mathbf{d}\mathbf{q}_o)_i &= g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o)_i \quad \left. \right\} \text{0-th order approximation} \\
 &\quad + \frac{\partial g}{\partial \mathbf{q}_o}(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o)_i \cdot \mathbf{d}\mathbf{q}_o \quad \left. \right\} \text{1st order approximation} \\
 &\quad + \mathbf{d}\mathbf{q}_o^{2\top} \frac{\partial^2 g}{\partial \mathbf{q}_o^2}(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o)_i \cdot \mathbf{d}\mathbf{q}_o^2 + \frac{\partial g}{\partial \mathbf{q}_o}(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o)_i \cdot \mathbf{d}\mathbf{q}_o^2 \quad \left. \right\} \text{2nd order approx.} \\
 &\quad + \dots \quad \left. \right\} n\text{-th order approx.}
 \end{aligned}$$

By truncating the Taylor series after terms of order n , we can derive form closure tests for different orders. The most common form-closure test is the one which describes first-order form-closure grasps. The first-order approximation of g contains two terms. The first one ($g(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o)_i$) will be zero in case of a grasp. The second one contains the gradient vector $\frac{\partial g}{\partial \mathbf{q}_o}(\bar{\mathbf{q}}_h, \bar{\mathbf{q}}_o)_i$ which corresponds to the separation direction in \mathbf{q}_o space at the i -th contact, i.e., the contact normal \mathbf{c}_{n_i} . Since the contact normal points into the object and the dot product projects the change in object configuration onto the normal, the second term always becomes negative if hand and object collide due to the object configuration change $\mathbf{d}\mathbf{q}_o$. Thus, the first-order form-closure condition is equivalent to the force-closure condition with frictionless point contacts.

Higher orders of form-closure tests are especially relevant for concave contact surfaces. The general relationship between force-closure and form-closure grasps is depicted in the diagram

of Fig. 1.6. Our example grasp in Fig. 1.5 is not a form-closure grasp, since any velocity perpendicular to the contact normals will allow the object to “escape”, i.e., there are non-zero object pose changes for which the gap function is not negative (Eq. 1.3).

Closely related to the notion of form closure are cages. A grasp is a cage, if the object is at an isolated component in configuration space (in contrast to be at an isolated *point* for form-closure). This means that there exists no collision-free path from the object’s current configuration to a configuration outside the hand. Every form-closure grasp is a cage but not vice versa.

1.2.3 GRASP METRICS

Given two grasps we can test if they are in force or form closure. But if both fulfill e.g. the force-closure condition, which one should we prefer (see for example the force-closure grasps shown in Fig. 1.7a and Fig. 1.7b)? This is where grasp metrics come into play. Grasp metrics are functions that map grasps to scalar values, indicating some form of quality of a grasp (Roa and Suárez, 2015). They are a crucial tool to synthesize grasps, since many algorithms are based on maximizing/minimizing a particular grasp metric.

The most popular metric was introduced by Ferrari and Canny (1992). It is based on the characterization of the wrench polytope \mathcal{WP} , i.e., the set of all object wrenches a grasp can apply (Eq. 1.1). One way to describe the wrench polytope is by looking at its volume: A larger volume means that there are more disturbance wrenches that can be balanced. Another more popular choice is to quantify the largest disturbance that can be balanced from any direction. Geometrically, this corresponds to the radius of the largest sphere that is inscribed in \mathcal{WP} and centered at the origin. This quality measure is commonly called the ε -metric:

$$\varepsilon = \underset{r}{\operatorname{argmax}} \mathcal{B}_r(\mathbf{0}) \subset \mathcal{WP},$$

where $\mathcal{B}_r(\mathbf{p}) = \{\mathbf{x} \in \mathbb{R}^6 \mid \|\mathbf{x} - \mathbf{p}\|_2 < r\}$ is a six-dimensional ball. Note that ε is based on the Euclidean distance in wrench space which mixes units of forces and torques. Thus, it is common to scale torques with a factor that is inversely proportional to the size of the object. Fig. 1.7 shows the comparison of two grasps of a rectangular object according to the ε -metric. In this example grasp 1 might be intuitively more desirable and robust because the contacts are at the center of each edge and uncertainty in their location will only minimally affect the grasp. In contrast, the contacts of grasp 2 are close to the object’s corner. A slight deviation of their location will lead the grasp to become unstable. Nevertheless, the ε value of grasp 2 is

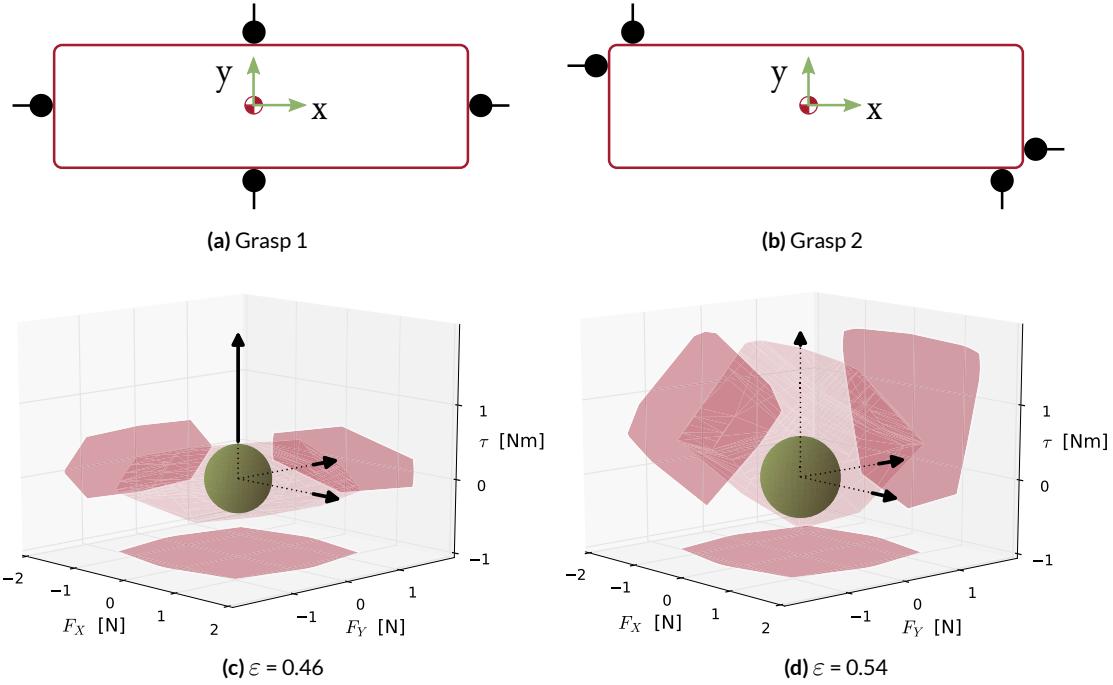


Figure 1.7: The plots show the quality of two grasps of a rectangle according to the ε -metric. The first grasp (left) contacts the four sides of the rectangle at their center points. Intuitively, this might be considered more robust than the second grasp (right) which contacts the rectangle at two opposite corners. Their respective wrench polytopes are shown in the lower row (light red, including projections along all three dimensions). The ε -metric is the radius of the largest inscribing ball (green) of the polytope, centered at the origin. The ε value of the second grasp is indeed higher because of the higher torques that can be balanced by grasping at the corners.

higher, mainly because it can withstand higher torque perturbations of the object. This can be seen in Fig. 1.7 by comparing the grasp wrench polytopes, whose sizes differ mostly along the τ dimension.

1.2.4 OTHER DESIRABLE GRASP PROPERTIES

Although force and form closure are the most popular conditions to characterize immobilized objects, there are more properties of grasps that are important and often desirable. The different classes and their relationships are shown as a Venn diagram in Fig. 1.8 which we adopted from Shimoga (1996). They are:

- **GENERAL GRASPS:** This class includes all possible contact configurations between hand and object, including non-prehensile grasps etc.

- **EQUILIBRIUM GRASPS:** This set contains all grasps whose contact wrenches sum up to zero. All force-closure grasps are in equilibrium, since they require a non-trivial solution to the equality $\mathbf{G}\mathbf{k} = \mathbf{0}$ (Eq. 1.2).
- **STABLE GRASPS:** Stability is a stronger condition than force-closure. A system is considered stable if it returns to its equilibrium state in finite time once it is deflected from it. What constitutes a stable grasp depends on the definition of the state. This can be the pose and velocity of the object, or the positions and velocities of the contacts (Montana, 1991). In contrast to force closure, stability depends on the controller, shape of object and fingers, and the contact impedances. The fact that force closure is too often equated with stability is also lamented by Howard and Kumar (1996):

“Finally, we close by pointing out that in most grasping situations, stability is a much more important criteria [*sic*] than force closure, despite the fact that force closure has received far greater attention in recent years. Indeed, often the only reason a researcher tries to obtain a force-closed grasp is that it is stable.”

- **GRASPS WITH DESIRED DYNAMIC BEHAVIOR:** This includes all stable grasps which in addition satisfy a desired dynamic behavior of the object.
- **DEXTEROUS GRASPS:** The set of grasps which are able to move the object according to the task, under the constraints given by the contacts between hand and object. If no task is given, a dexterous grasp needs to be capable to move the object in any direction.

In this thesis we are not concerned with the last two categories – dexterous grasps and grasps with a desired dynamic behavior. Instead we are interested in achieving stable grasps. To measure grasp success throughout this thesis, we will usually resort to empirical tests: shaking the hand arbitrarily and checking whether the object will fall down. Only for few experiments in simulation we report force-closure or ε -metric values.

*We added the set of form-closure grasps and corrected the set of equilibrium grasps to be a proper *superset* of force-closure grasps. Shimoga (1996) claims that equilibrium grasps are a proper *subset* of force-closure grasps. This cannot be true, since the equilibrium condition (all contact wrenches sum up to zero) is necessary for every force-closure grasp. Furthermore, there are grasps that are in equilibrium but not force-closure (e.g. two antipodal frictionless contacts).

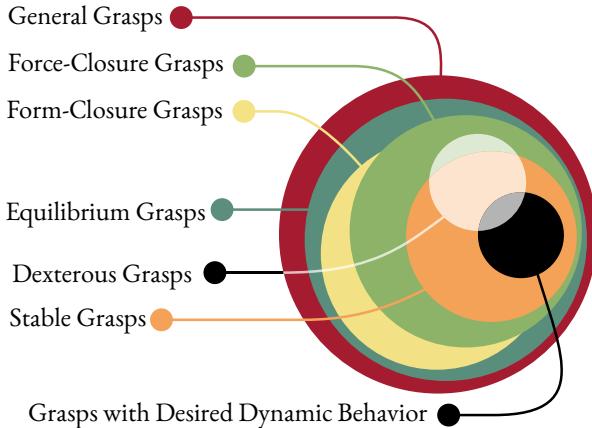


Figure 1.8: Diagram of desirable grasp properties and their relationships, adapted from Shimoga (1996).*

1.2.5 LIMITATIONS

The classical model of grasping used for force and form closure is based on point contacts. One might argue that this is an irrelevant approximation, since no real grasp exhibits only point contacts but rather large patches of contact, especially in robust grasps. But the model of point contacts is able to express also other contact geometries. A line contact is equivalent to two point contacts located at each end of the line segment. A contact region with uniform friction can be represented by a set of point contacts located at the vertices of its convex hull. While this is helpful for analyzing grasps, it does not help when trying to find good grasps in combination with the assumptions of rigid bodies and static objects. In this case, most contacts will end up being points.

While some simplifications such as frictionless grasps seem to serve as a reasonable worst-case approximation, form closure with its requirement to completely restrain an object and prohibit any motion seems unrealistic. No human grasp completely immobilizes an object. There is always a certain wiggle room left (due to finger pulp etc.) which would prevent form closure. This compliance at the contact level has been addressed by various grasp models: Either by using linear spring models (Howard and Kumar, 1994), Hertz contact models (Lin et al., 1997), or discretizing contact regions into finite elements (Ciocarlie et al., 2005). The significance of these models is hard to assess since none of them is evaluated w.r.t. the real world phenomena they are trying to describe. And because of their computational complexity and high demand for parameter identification they are not commonly used in robotic applications to find good grasps.

Hence, the simpler models assuming rigid bodies and evaluating force closure based on point contacts are dominant when it comes to synthesizing robotic grasps in the real world.

The problems arising in such approaches are twofold. First, a lot of knowledge is required to analyze grasps with the classical model. The object geometry needs to be known, which requires a segmentation of the scene and integration of multiple views due to occlusions. The coefficient of friction of different materials needs to be inferred, preferably only through vision. Gathering this information is usually bundled in a process that resembles *object recognition* – a field of research itself that is complex and still unsolved. Second, executing grasps that were planned using the classical model will often deviate from expectations. This is because of model inaccuracies and the uncertainty of the collected information.

We can take a look at two representatives of this approach ([Morales et al., 2006](#), [Goldfeder et al., 2009b](#)) to see that the classical model does not easily transfer to the real world. Both are based on databases which store the best grasps according to the ε -metric (see Sec. 1.2.3) for a variety of objects. After identifying and localizing objects in the scene, the looked-up grasps are executed. [Morales et al. \(2006\)](#) claim that such an execution module is outside the scope of their paper: “And finally, a module that executes the grasp using tactile and visual feedback has to be developed too.” We can assume that they came to this conclusion after not being able to reproduce the planned grasps successfully. Similarly, the Columbia grasp database ([Goldfeder et al., 2009b](#)) contains grasps for several thousand objects and multiple hands. Applying these grasps in practice often fails due to the gap between simulation and real world.[†]

In Sec. 3.1.3 we compare grasps planned by one of the algorithms in this thesis to planned grasps with known geometries based on the ε -metric. In this experiment, we found that executing these grasps often leads to failure instead of stable contact configurations.

A final hint on the weaknesses of the classical grasp model is given by [Balasubramanian et al. \(2010\)](#). They asked humans to guide a robotic hand for finding the most promising grasps. These grasps were evaluated repeatedly and compared to grasps planned with the

[†]Matei Ciocarlie (co-author of the Columbia grasp database): “I believe that a direction that has received less attention, and needs to catch up, is quantifying a grasp by robustness to execution errors. I have seen the need for this recently in two very different settings. First, at Columbia, we had a strong effort to create a very large database of labeled grasp information, in simulation. The goal was to use the database in order to grasp similar (but not identical) objects in the real world. We found that some of the grasps in our database had great quality metrics, but looked marginal to the human eye and translated poorly to other similar objects because they were relying on detailed features of an object that were unlikely to generalize well, and difficult to execute correctly. Second, at Willow Garage, we have been building a grasping pipeline that, on recognized objects, executes pre-stored grasps from a database. Even when recognition does its job, and the exact object is identified, the work is far from done. Due to calibration errors, a grasp can never be executed perfectly, which introduces errors that the database was not originally equipped to handle.” (from https://web.archive.org/web/20100801094740/http://www.grssp.org:80/blog/?no_cache=1)

ε -metric. It turned out that humans found much more robust grasps and that these grasps could not be explained by the ε -metric but rather by an alignment of the hand with the object's principal axis.

1.3 ALGORITHMS FOR SYNTHESIZING GRASPS

In this section, we look through the lens of adaptability to classify different existing grasp algorithms. We identify seven planning principles which implicitly or explicitly exploit the adaptability of the hand to guide the search for good grasps or compress the underlying search space (see Table 1.1).

Principle		
I	Exploiting Adaptability in Grasp Representations	(Sec. 1.3.1)
II	Exploiting Adaptability in Object Representations	(Sec. 1.3.2)
III	Assuming Object Similarity Implies Grasp Similarity	(Sec. 1.3.3)
IV	Exploiting Adaptability of Objects	(Sec. 1.3.4)
V	Exploiting the Environment to Increase Adaptability	(Sec. 1.3.5)
VI	Assuming Uncertainty in Grasp Execution and Sensing	(Sec. 1.3.6)
VII	Learning Hand-Specific Data-Driven Grasp Models	(Sec. 1.3.7)

Table 1.1: Grasp planning principles based on hand adaptability

This is a non-exclusive list. Algorithms can exploit multiple of these seven principles. In contrast to the first six principles, the last one also addresses the problem of transferring model-based grasps to the real world (one of the main limitations identified for classical grasp analysis in the previous Sec. 1.2).

1.3.1 PRINCIPLE I: EXPLOITING ADAPTABILITY IN GRASP REPRESENTATIONS

The core problem of planning grasps is the high number of parameters that is needed to describe grasping motions. Imagine we would only consider static grasp postures, i.e., each grasp is fully described by a single point in the configuration space of the hand. Let us assume that this configuration space is defined by the intrinsic DOF of the hand and its pose in three-dimensional space, i.e., a configuration is a point $\mathbf{q} \in \mathbb{R}^{\text{DOF}} \times SE(3)$. Given the human hand with 24 DOF this results in a space with 30 dimensions. Discretizing each revolute DOF by 1° and each prismatic DOF by 1 mm results in $\approx 5 \cdot 10^{76}$ different options to grasp. Note that in this view we have ignored the motion of the hand or its joint impedances, which would lead to even more dimensions. Even worse, this space does not lend itself to gradient-based search methods since contact creates strong non-linearities and discontinu-

ties. Consequently, there have been different attempts to either reduce this dimensionality or to devise entirely different grasp representations by using heuristics, engineering intuition, machine learning techniques or human studies.

Nearly all approaches to grasp planning – as we will see below – treat the extrinsic and intrinsic DOF of a hand differently. Their underlying assumption is that the range of motion of a DOF affects its ability to adapt and match object shape. The DOF of a finger have a much smaller influence on contact point locations than the extrinsic ones which depend on the entire arm configuration. As a result, search often focuses primarily on the extrinsic DOF while intrinsic DOF are ignored or significantly compressed. [Miller and Allen \(2004\)](#) represent all possible finger configurations by a few prototypical grasp pre-shapes. For the three-fingered Barrett Hand they identify the spherical, cylindrical, precision-tip, and hook pre-shape. Other sources often used for defining pre-shapes are human grasp taxonomies ([Feix et al., 2009](#)). The extrinsic DOF are often parametrized in more detail, e.g. using the *approach & squeeze* grasp representation ([Berenson et al., 2007](#)): Given a pre-shape and a roll angle the hand moves along an approach vector until contact, then the fingers close until they hit the object surface. The approach vector can be defined by cylindrical or spherical coordinates originating at the center of mass of the object ([Miller et al., 2003](#)), or by the surface normals of the object ([Diankov, 2010](#)).

Another common strategy for reducing intrinsic DOF is based on *synergies*. The concept of synergies originates from analyzing human movements ([Bernstein, 1966](#)) to answer the question of how motion redundancies are resolved. Since there are more DOF in a human limb than those required to solve a task, i.e. multiple solutions exist, the question is how the human neural system decides which one to choose and which muscles to recruit. A single synergy describes a coordinated motion pattern that can be observed at different levels (neural, biomechanical) and for various modalities (kinematics, kinetics). The linear combination of a small set of synergies can be used to explain a large variety of human grasp postures ([Santello et al., 1998](#)). The simplified view of synergetic control has also inspired robot grasp planning approaches. [Grioli et al. \(2012\)](#) applied different variants of synergies to grasping: geometric synergies, soft synergies, adaptive synergies. [Ciocarlie and Allen \(2009\)](#) use the low-dimensional subspace defined by postural synergies to sample candidate pre-shapes. Although most grasping synergies are extracted using principal component analysis, non-linear transformations are also used to find low-dimensional embeddings ([Romero et al., 2010](#)).

APPLICATIONS OF THIS PRINCIPLE IN THE THESIS: We use many of the existing ideas to

represent grasps. The planning algorithms in Chapter 3 search for the pre-grasp pose of the hand which describes the extrinsic DOF and considers only a limited set of pre-shapes of the hand which relates to the intrinsic DOF. The grasp parametrizations that exploit the environment (see Chapter 5) are novel, since they take advantage of contact-events to structure grasps.

1.3.2 PRINCIPLE II: EXPLOITING ADAPTABILITY IN OBJECT REPRESENTATIONS

Another recurring principle that implicitly exploits adaptability is the simplification of object geometry. It is based on the observation that small-scale and high-frequency shape features in general do not affect grasp success. Enveloping grasps will locally adapt; only the more general shape features such as a planar vs. curved surfaces matter. A variety of shape representations have been presented that approximate real object geometry, ranging from bounding boxes and superquadrics to topological concepts such as holes. Table 1.2 lists different approximations used for grasp planning algorithms.

Publication	Shape Approximation
Bard and Troccaz (1990), Sweeney and Grupen (2007)	Axis-aligned 2D bounding ellipses
Miller and Allen (2004), Ekvall and Kragic (2007), Nieuwenhuisen et al. (2012)	3D primitives (box, cylinder, cone, sphere)
Przybylski et al. (2010)	Medial-axis of maximum inscribing balls
Aleotti and Caselli (2012)	Reeb graph based on the integral geodesic function
Dune et al. (2008)	Ellipsoids
Goldfeder et al. (2007), Ückermann et al. (2012)	Superquadrics
Pas and Platt (2013)	Quadratics
Bone et al. (2008)	Planar parallel patches
Huebner and Kragic (2008)	Minimum volume bounding boxes
Kootstra et al. (2012)	Edges, surface patches, contours
Pokorný et al. (2013)	Holes

Table 1.2: Examples of different shape approximations used for planning grasps

Approximations are usually used to guide the grasp sampling process or mapped directly to pre-defined grasps. Instead of trying to approximate the shape of the object to grasp, a more goal-directed approach is to use the gripper geometry to represent shape. [Klingbeil et al. \(2011\)](#) search for complementary geometric structure of a parallel-jaw gripper in single-shot depth maps. Grasps are found wherever a part of the depth map matches the palmar gripper surface.

APPLICATIONS OF THIS PRINCIPLE IN THE THESIS: We categorize shapes into few classes and map them to the appropriate pre-shape configuration of the hand. Our grasp planning algorithm in Sec. 3.1 uses the eccentricity of the object contour in the camera image to distinguish pre-shapes. In the algorithm presented in Sec. 3.2 we use simple geometric primitives like spheres, boxes, cylinders, and disks to select the appropriate hand configuration. These simplifications are done on purpose and will show that they do not affect grasp success due to the hand’s adaptability.

1.3.3 PRINCIPLE III: ASSUMING OBJECT SIMILARITY IMPLIES GRASP SIMILARITY

This principle is based on the intuition that small changes in object shape have small effects on the change of a grasp. It is similar to the previous principle (Subsec. 1.3.2) in the sense that it is enabled by local hand-driven adaptability. An approach that exploits this principle is presented by [Hillenbrand and Roa \(2012\)](#). They use a geometric warping between two point sets of similar objects to transfer known contact points from a previously grasped object to a novel one. The search for a successful grasp is then initialized with the known grasp.

Another group of approaches uses shape similarity between the unknown object and a database of known objects and grasps. They differ in the way they measure similarity and what kind of grasp information is transferred from the known to the unknown geometry. E.g. [Goldfeder et al. \(2009a\)](#) match 3D sensor data describing partial views of unknown objects to find a suitable grasp. Their matching uses a codebook of SIFT features from depth images to retrieve the k-nearest neighbors in the Columbia grasp database ([Goldfeder et al., 2009b](#)). The associated pre-grasps that perform constantly good on all neighbors are supposed to be good generalizations and are applied to the unknown object.

APPLICATIONS OF THIS PRINCIPLE IN THE THESIS: We only implicitly exploit this principle in the algorithms presented in this thesis. We avoid the notion of object but rather try to extract shape features from 3D point measurements which match certain hand pre-

shapes (Sec. 3.2). The learning approach in Chapter 8 uses shape similarity to select between grasping strategies. In both cases, similarity in local shape will lead to similarities in grasping.

1.3.4 PRINCIPLE IV: EXPLOITING ADAPTABILITY OF OBJECTS

Objects can adapt to the hand in case they are deformable. But also rigid bodies can adapt if we consider their external DOF. Traditionally though, grasp planning tried to find the static grasp configuration between hand and object. This means unintentional contact with the object during reaching needs to be avoided. Few approaches have argued in favor of a more dynamic approach to grasping and exploit object motion to increase grasp success.

Mason (1982) generalized a hinge grasp strategy which first makes contact between one gripper jaw and a hinge plate while the gripper continues to move until the hinge plate rotates into the palm and can be grasped. He analyzed the conditions under which such grasping motions would completely constrain an object. The similar push-grasp strategy introduced by Dogar and Srinivasa (2010, 2012) is defined by its capture region, i.e., all poses in the plane from which an object can be grasped by first pushing towards it and then closing the fingers. The capture region can be used to plan push-grasps in cluttered environments and to include uncertainty (Koval et al., 2016).

Bin-picking scenarios which are common in industry also benefit from the adaptability of objects, especially when a bin contains objects of the same type. Rodriguez et al. (2014) argue to “*let the fingers fall where they may*” instead of “*put[ting] the fingers in the right place*”. Instead of planning a priori where to grasp they favor to learn a statistical model of grasp outcome, which they say is easier to do for simple hands with limited actuation and sensing capabilities. The usefulness of their approach is shown by picking single items from a box full of markers.

Apart from grasping, there are also methods that exploit object motion to re-orient them: Chavan-Dafle et al. (2014) introduce the concept of *extrinsic dexterity* which involves a set of manipulation strategies in which objects are pushed against surfaces or thrown into the air to reconfigure them. Other strategies include swing-up regrasping (Sintov and Shapiro, 2016). This repertoire substantially increases the dexterity of a simple gripper without including additional degrees of freedom.

APPLICATIONS OF THIS PRINCIPLE IN THE THESIS: The edge- and wall-constrained grasping strategies that we present in Chapter 5 are exploiting this principle. They reorient the object by pushing and sliding the objects before picking them up.

1.3.5 PRINCIPLE V: EXPLOITING THE ENVIRONMENT TO INCREASE ADAPTABILITY

Most classical approaches to grasping do not explicitly consider the environment. They either ignore it or include it as an obstacle that reduces the amount of possible grasping motions. Some grasp planning approaches implicitly exploit certain environmental features. The push grasping strategy ([Dogar and Srinivasa, 2010](#)) assumes planar support surface that can be used to potentially slide the object while capturing it.

Few grasp planners take explicitly advantage of the environment. [Kazemi et al. \(2014\)](#) present a grasp strategy for a three-fingered hand which grasps objects from the top. During the closing motion, the wrist pose is controlled such that the three fingertips constantly touch the support surface. This allows them to grasp also smaller objects which usually would require high precision. Other grasp approaches take pre-grasp manipulations with the environment into account, e.g. sliding flat objects across the table before picking them up at the edge ([Kappler et al., 2010](#)). But they only do this for a single object type (a CD) using manually coded strategies and ignoring perception. [King et al. \(2013\)](#) also tackle the problem of pushing objects towards an edge. Although their formulation is more general, they focus on planning a pushing motion by modeling the objects feasible motion in the plane as a Dubins car. In general, there have been very few works in robot grasp planning that focus on the beneficial effects of the environment.

APPLICATIONS OF THIS PRINCIPLE IN THE THESIS: Exploiting the environment during grasping is a central principle which we extensively apply in Part II and III of the thesis. We present a grasp planning algorithm in Chapter 5 (Algorithm 5.4) that searches for contacts with the environment to simplify grasping. Based on this algorithm Chapter 7 introduces a learning problem which selects the most successful of these strategies based on object properties.

1.3.6 PRINCIPLE VI: ASSUMING UNCERTAINTY IN GRASP EXECUTION AND SENSING

The exploitation of adaptability can also be implicitly maximized by assuming a noisy unreliable action execution for grasping. Modeling the uncertainty that occurs due to imprecise controllers can lead to choosing those grasps which are least affected by it. Similarly, if uncertainty is also assumed as an inevitable part of estimation and sensing (e.g. object localization, shape estimation) this can also be used to choose grasps that will adapt to a wider variety of possible conditions.

Weisz and Allen (2012) represent the effect of object pose uncertainty on grasp quality. They calculate the probability of force-closure of a given grasp by densely sampling the object pose in a grid around its expected pose. In comparison to the ε -metric (Sec. 1.2.3), this produces grasps that are much more robust when being executed in the real world. Inspired by human studies, Stulp et al. (2011) showed a method which adapts approaching trajectories for grasping according to the estimated object pose uncertainty. As a result, the maximum finger span is aligned with the dimension of largest uncertainty in object position.

Uncertainty in shape usually results from a measurement process which only partially reveals an object. Goldfeder et al. (2009a) tackle this problem by designing a shape descriptor which takes a specific viewing direction into account. Given depth data of an unknown object they can match it to similar and known objects and use pre-computed grasps. Another way to deal with partial views is to plan grasps based on the assumption of reflective symmetries (Bohg et al., 2011). A probabilistic shape representation (Dragiev et al., 2011) can be used to plan information-gathering poking actions given tactile measurements vs. executing a robust grasp once enough uncertainty is reduced.

Hsiao et al. (2007) plan in the space of all possible state distributions, formulating grasping as a partially observable Markov decision process (POMDP). They discretize the state space along contact-events and actions according to guarded motions. An approximate POMDP solver is then used to find policies for planar grasping using fingertip contact sensors. Platt et al. (2011) try to avoid the discretization of the belief space in their problem definition of SLAG (simultaneous localization and grasping). Instead they plan using the maximum-likelihood hypothesis and replan in case of invalidating observations.

Instead of explicitly representing uncertainties and planning in belief space, adaptability between hand and object can also be increased by devising contact-reactive control strategies based on heuristics. Felip and Morales (2009) present a grasp primitive for a three-fingered hand which aligns the hand based on force and tactile feedback and tries to detect two parallel faces. Similarly, Hsiao et al. (2010) plan a rough approach direction and adapt the grasp pose of a parallel-jaw gripper based on contact feedback.

APPLICATIONS OF THIS PRINCIPLE IN THE THESIS: The motion planning algorithm presented in Chapter 4 (CERRT) explicitly models uncertainty in actuation. In combination with the assumption of uncertainty-free contact sensing abilities, contact-seeking behavior emerges. The grasp planning algorithms presented in Chapter 3 and 5 do not model uncertainty but implicitly assume that hand adaptability and contact with the environment re-

duces it.

1.3.7 PRINCIPLE VII: LEARNING HAND-SPECIFIC DATA-DRIVEN GRASP MODELS

Relying on analytical models to capture the complex physics happening between hand and object during grasping is dangerous, especially when dealing with compliant, underactuated mechanisms. We have seen in Sec. 1.2.5 that those models often do not transfer to the real world. Instead of analytical models, another popular way is to use statistical models based on actual experience. These learning-based approaches can in principle achieve significantly improved performance when provided with enough data. But in robot grasping data acquisition is expensive: real-world experiments are tedious to set up, take time to run, and often need human supervision. There are four ways to deal with this appetite for data: Use low-dimensional hand-engineered features that require less data for training, set up a system that allows collecting a lot of data, use active learning to gather data more efficiently or use simulations to create vast amounts of data.

HAND-ENGINEERED FEATURES

An early, prominent instance of a learning-based approach for grasping was introduced by [Saxena et al. \(2008a\)](#). They built a grasp point detector for images based on supervised learning. Their 459-dimensional feature vector captures edges, texture, and color. A 3D grasping pose is triangulated using detected grasp points from multiple views. In an extension of their approach, [Saxena et al. \(2008b\)](#) extract more global features from point cloud data to characterize a grasp. These features include symmetry, center of mass and planarity of the point set enveloped by a possible grasp. This supervised learning approach is shown to work for parallel-jaw grippers, a three-fingered hand, and for cluttered scenes. [Bohg and Kragic \(2010\)](#) improved over the results achieved by [Saxena et al. \(2008a\)](#) by using shape-context features. Shape-contexts are histogram features based on the distances and directions of points w.r.t. a reference frame. They capture a larger neighborhood than the 10×10 image patches used by [Saxena et al. \(2008a\)](#). The hand-designed grasp descriptor by [Herzog et al. \(2014\)](#) consists of a normal vector, a height map, and a tile-type map (classifying tiles into “void”, “surface”, “occlusion”, “background”). The quality of a grasp is calculated based on their similarity to templates acquired through kinesthetic training. [Fischinger et al. \(2015\)](#) engineered a similar descriptor based on the discretization of 2D space along planar surfaces that describes the relation of heights of neighboring cells. It is well suited to describe protruding parts of ob-

jects or object piles which can be grasped using a top-down pinch grasp. They validate their approach in a scenario where a robot needs to empty a basket filled with unknown objects.

LARGE REAL-WORLD DATA SETS

Deep learning architectures have outperformed state-of-the-art methods in image, speech recognition and machine translation tasks. Consequently, they have also been applied to learn models for grasping with increasing success. Instead of first extracting hand-engineered features, these methods are based on the raw sensor input (usually RGB-D cameras). Lenz et al. (2013) were the first to adopt a deep learning architecture for predicting robot grasps, parametrized as oriented rectangular image regions. Their human-labeled dataset contains a few thousand grasps. The prediction pipeline is separated into two stages: A smaller network which extracts the most graspable rectangles, while a second larger network is trained to re-rank those grasp candidates. This separation lowers the computational effort and slightly increases prediction accuracy. A larger dataset of 50k robotic grasp attempts was collected by Pinto and Gupta (2015). They use the same grasp rectangle representation but learn an 18-way binary classifier, discretizing all possible orientations of a parallel-jaw gripper in the image plane. Their convolutional neural network benefits from weights that are pre-trained on the object recognition task defined by ImageNet. Their planar grasps achieve 79.5% accuracy (66% for novel objects) while using a heuristic approach (grasping the rectangle by its smallest side) achieves only 59.9%.

Even more data was collected by Levine et al. (2016) using multiple robots in parallel (800k grasp attempts from cluttered bins). In contrast to previous approaches their model predicts grasp success based on an external camera image, a displacement of the end-effector, and the closing of the fingers. This allows them to use the model in a feedback loop: They repeatedly query the most promising end-effector motion and finger closing command based on the current camera image. As a result, some learned grasping strategies include singulating an object before picking it up, or pinching into soft objects. Their approach (82.5% success rate) beats an open loop version (66.3%) and a hand-designed grasping pipeline based on bounding boxes (49.2%).

ACTIVE LEARNING

The problem of learning-based approaches being so data hungry is tackled by Montesano and Lopes (2012). They use Gaussian, Laplacian, and Sobel filters as features for detecting grasp

points in images. Formulating grasp attempts in an active learning setting, they show how to select successful ones as quickly as possible; trading exploration off against exploitation. Similarly, an active learning approach is often combined with systems that collect large-scale data sets in the real world (Oberlin and Tellex, 2015, Pinto and Gupta, 2015).

SIMULATED DATA

Another approach to fill the need of grasp data for learning is to use simulations. Simulations provide full control over the data acquisition process, circumventing real-world problems such as resetting the environment or labeling grasp success due to partial observability. Gualtieri et al. (2016) show how to exploit simulated data for both, sensor measurements and grasp outcomes. Their grasp representation is a $60 \times 60 \times 60$ voxel grid between two fingers projected onto 60×60 depth images from three directions. They simulate 215k anti-podal grasps and transfer these grasps to the real world. Kappler et al. (2015) present a database with 300k grasps for 700 objects. They do not transfer them to the real world but use crowdsourcing to annotate the grasps with success labels. They show that correlation with these labels is higher for a prediction based on physics simulations than using the classical ε -metric (Sec. 1.2.3). Dex-Net 2.0 (Mahler et al., 2017) is a convolutional neural network trained on 6.7m simulated depth images patches to predict the success of planar grasps with a parallel-jaw gripper. The labels are calculated from antipodal grasps. When applied in the real world on training objects, the grasps succeed in 93% of all cases (80% with novel objects). An even larger effort is presented by Zhou and Hauser (2017). They simulate 23.3m grasps for a three-fingered hand of which \approx 500k are successful and robust. For each grasp, multiple synthesized depth images of the object are generated.

APPLICATIONS OF THIS PRINCIPLE IN THE THESIS: We extensively exploit the principle of learning hand-specific models in Part III of the thesis. This is especially useful if the properties are hard to model analytically, which is often the case with very compliant hands such as the RBO Hand 2 (Sec. 1.1.5). Algorithm 7.6 is a formulation of the active learning problem, in which the robot decides for a grasp based on its certainty of success. Furthermore, Algorithm 8.7 learns models for simple manipulation tasks via reinforcement learning.

1.4 CONCLUSION

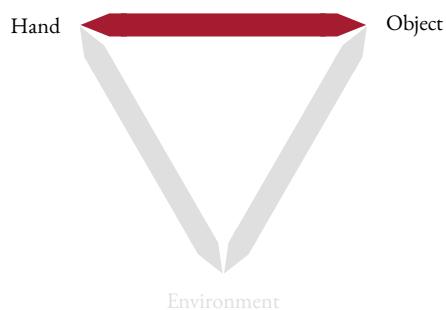
In this chapter we have looked at the state-of-the-art in robot grasping, focusing on three areas which impact robotic grasping solutions the most: hardware, theory, and algorithms.

The capabilities and features of robotic hands significantly impact the design of grasp planning algorithms. Our analysis has shown that low-level adaptability in grasping mechanisms – unlike pure mechanical compliance – can be achieved through various means. The size, kinematic structure, surface materials, and actuation methods all affect the adaptability of robot hands. Recently, there has been a surge of soft, passively adaptable, and underactuated robotic hands that are contrary to traditional stiff, fully actuated grippers. This is mainly due to the availability of novel manufacturing techniques such as 3D printing. The *mechanical intelligence* (Ulrich, 1989) or *morphological computation* (Pfeifer and Iida, 2005) that comes with these mechanisms should be acknowledged, characterized, and exploited by higher-level grasp planning processes. This is one goal of this thesis.

We have shown that the dominating theoretical contact model that underlies grasp analysis cannot adequately capture the adaptive nature of non-traditional hands during grasping. It is based on rigid bodies, quasi-statics, point contacts, and full observability. It describes a final grasp configuration without giving insights into the temporal evolution that is required to actually achieve a grasp. We showed that the assumptions of the classical model often do not match reality – making it hard to transfer grasps that are working according to the model onto real robotic systems. This thesis does not advance the formalism of the classical grasp model. Instead, we will focus on a more holistic approach, modeling the macroscopic consequences of exploiting adaptable hands and stiff environments.

Finally, we presented an analysis of the diverse landscape of current grasp planning algorithms. This analysis focused on the exploitation of hand adaptability to increase grasp success in planning. The identified principles are general and will also guide the design of the algorithms presented in the remainder of the thesis.

PART I



Interactions Between
Hand and Object

MOTIVATION

When reviewing the state-of-the-art in the last chapter, we realized that the dominating paradigm in robotic grasping is largely based on precise, uncertainty-free modeling of rigid point contacts. This view seems mostly suited for stiff, fully actuated hands and requires precise models of hand and object. In contrast, we rather want to exploit the inherent adaptability of hands for enveloping grasps given realistic sensing requirements.

In the first part of this thesis we will investigate a holistic approach to robotic grasping that centers around the shape of hand and object and the idea to find a match between the two. We explicitly avoid the factorization into perception and grasp planning, but instead tackle the problem as a whole.

CONTRIBUTIONS

The main contributions of this part are:

- A hypothesis about the relationship between object shape and grasp success, which we call the *Sun-Flower-Annulus* (Sec. 2.2).
- A view on the problem of grasping as a sequence of funnel operations (Sec. 2.3).
- A method to synthesize grasps based on the change of contour of an object when being viewed by an active camera (Sec. 3.1).
- A grasp planning approach that matches arbitrary 3D measurements with hand pre-shapes by using geometric prototypes (Sec. 3.2).

OUTLINE

We start Chapter 2 with a series of small real-world experiments that investigate the role of adaptation between hand and object during grasping. We complement this with a more elaborate experiment in simulation to show how adaptability helps when object geometry changes. These results are finally accumulated in what we call the *funnel view of grasping*.

Based on the insights gained in Chapter 2 we will present two algorithms that synthesize grasps in Chapter 3. These algorithms do not assume a priori knowledge about object models. They calculate grasps from sensor measurements – the first one using RGB images, the second based on depth images.

2

Adaptability Between Hand and Object

LET US START WITH A THOUGHT EXPERIMENT. Imagine a person wearing a blindfold and a thick mitten. The task of that person is to grasp arbitrary objects, despite her limitations. Since this is close to impossible, an experimenter is allowed to help. This person holds the object and can position it close to the opened hand of the blindfolded subject (see Fig. 2.1). She can also give the blindfolded subject a signal to close the fingers, whenever she thinks that this would result in a stable grasp. Once the experimenter triggers the closing she is not allowed to move. The assumption is that this protocol would lead to very successful grasp performance.



Figure 2.1: The mitten thought experiment includes a sensory-deprived subject grasping unknown objects.

But what does this thought experiment tell us about grasping? The experiment factorizes successful grasping into two complementary processes: the sensory-deprived subject and the experimenter providing the object. The subject does not have access to visual or haptic information (although the mitten only suppresses parts of the tactile feedback, let us assume –

for the sake of the thought experiment – that all tactile and kinesthetic feedback is absent). However, the subject is in control of a compliant closing strategy. The experimenter effectively positions the subject’s hand relative to the object (by doing the inverse). She will do this based on her internal model about the subject’s closing strategy. The communication channel between the two only requires a very low capacity, basically 1 bit per grasp to trigger the compliant closing strategy. The experiment illustrates that an appropriate perceptual strategy (the experimenter) in conjunction with a simple compliance-based control strategy (the subject wearing a mitten) can lead to outstanding grasping performance.

In the spirit of the mitten thought experiment we will start this chapter with a series of small exploratory grasping experiments. Through these experiments we try to gain insights about the structure of possible grasp representations. We will complement these real-world experiments with a vast amount of simulations to gain more understanding about the relationship between hand and object under the influence of adaptability. This leads us to the *Sun-Flower-Anulus* hypothesis (Sec. 2.2). Based on the collected insights, we will conclude this chapter by sketching the *funnel view of grasping* (Sec. 2.3). This view emphasizes the holistic and dynamic nature of grasping in contrast to the classical theory on grasp analysis (Sec. 1.2) that focuses on grasping as the compound of local contact effects.

2.1 EXPLORATORY GRASPING EXPERIMENTS – FROM THE MITTEN-THOUGHT-EXPERIMENT TO TELEOPERATING MULTIFINGERED HANDS

In this section we will examine the effect of different types of grasp adaptability in a series of small exploratory experiments. Table 2.1 summarizes the different experiments. They all use human subjects to control robotic hardware. But they differ in the control and feedback channels that are provided to the human user. By imposing structure on the applied motion behaviors we hope to gain insights into potential representations for grasp planning algorithms.

We exploit human grasping and manipulation knowledge in these experiments to gain understanding about the grasping process. Please note that all experiments are exploratory in nature. They usually only include a single subject and will not hold up to strict experimental rigor (especially given the involvement of humans). We still think that they show meaningful trends.

Experiment	Adaptation through...			
	Robot Hand	One-Shot Visual Information	Continuous Visual Feedback	Haptic Feedback
Operating the RBO Hand 2	×	×	×	⊗
Teleoperating the Allegro Hand	×	×	×	○
Mitten Thought w/ RBO Hand 2	×	×	○	○
Teleoperating the Meka Hand	×	×	○	○

Table 2.1: We conduct four experiments about the effect of adaptability in grasping. Adaptability is either achieved passively through the mechanism or actively by creating feedback loops based on different modalities (haptics vs. vision) and frequencies (one-shot vs. continuous). Crosses indicate the possibility of adaptation.

2.1.1 OPERATING THE RBO HAND 2

In a first small experiment, we want to assess whether an adaptable robot hand (with few degrees of actuation) is capable to grasp different objects and how complex the associated motions are. To do that we harvest human intelligence: Instead of using a robot arm the hand is operated by a human user. The human has time to familiarize herself with the grasping mechanism for a few trials. One object at a time is placed in front of the user on a table. Grasping succeeds if the user can pick up the object and place it into a box next to the table. There is no time limit for a single grasp but the user can ask to label the current attempt as failure at any moment.

We use the RBO Hand 2 mounted on a stick with two buttons to inflate or deflate it. The user can decide between two inflation modes, one including all digits and another excluding the thumb. Note that the user’s sensory feedback is nearly unrestricted. Only the richness of the haptic feedback is reduced by holding the robot hand. We use 20 objects from a wooden object set and 20 items from a set of common household products (see Fig. 2.2). Both object sets provide a variety of shapes, sizes and materials. Each object was supposed to be grasped five times, some of them from different initial orientations, totaling 240 trials.

Out of all trials 191 resulted in successful grasps (79%). A grasp was considered successful if the object did not fall when rotating the hand and moving it up and down. The user usually decided to inflate fingers and thumb, only 25% of all successful grasps excluded the thumb. This mode was used for flat objects like a book, file cards, a credit card, a package of markers, wooden disks and a board. Here, not inflating the thumb kept the palm flat and



Figure 2.2: We use two complementary object sets for our human-in-the-loop grasping experiments. **Left:** The first set contains simple geometric shapes such as spheres, cylinders, and prisms. **Right:** The second set consists of household items such as books, plastic fruits, and stationery. It contains many objects from the 2015 Amazon Picking Challenge ([Correll et al., 2016](#)).

provided a large contact area that the fingers could press the flat objects against. We also noticed that the user grasped these kinds of objects after re-positioning them, e.g. sliding them towards the table edge to be able to wrap the fingers around them. The objects that could not be grasped were bulky ones which surpassed the hand's maximum grip aperture (a cube with length 10 cm) or immediately started to slip due to their weight (a wooden cylinder and cuboid weighing 1 kg each).

The experiment shows that although the user had only limited control over the internal DOF of the hand, a wide variety of objects could be grasped. The hand's inherent adaptability played a beneficial role.

2.1.2 TELEOPERATING THE ALLEGRO HAND

In a second experiment, we wanted to restrict the haptic feedback the human user can utilize to make grasp decisions (similar to the Mitten thought experiment). At the same time we wanted to verify whether the chosen hand pre-shapes would be low-dimensional even if the user had access to a high number of controllable DOF in the hand. Finally, we wanted to make sure that the limitations of using a robot instead of a human arm will not significantly affect grasp success.

This lead us to a teleoperation experiment using the 16-DOF Allegro hand mounted on a 7-DOF Barrett WAM. A human operator was equipped with a Polhemus Patriot 6-DOF electromagnetic tracker at the wrist and a 22-DOF CyberGlove II data glove to measure finger

joint configurations. The 6-DOF pose x_{wrist} and velocity \dot{x}_{wrist} of the wrist tracker serve as an input to a velocity-based operational space control scheme (Nakanishi et al., 2008) to produce the joint torques τ_{arm} :

$$\tau_{\text{arm}} = \mathbf{M}(\mathbf{q}_r)\ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}_r, \dot{\mathbf{q}}_r) + \mathbf{g}(\mathbf{q}_r) + \mathbf{K}_{q,d}(\dot{\mathbf{q}}_r - \dot{\mathbf{q}}) + \mathbf{K}_{q,p}(\mathbf{q}_r - \mathbf{q})$$

$$\begin{aligned}\dot{\mathbf{q}}_r &= \mathbf{J}^+ \dot{\mathbf{x}}_r = \mathbf{J}^+(\dot{\mathbf{x}}_{\text{wrist}} + \mathbf{K}_p(x_{\text{wrist}} - \mathbf{x})) \\ \ddot{\mathbf{q}}_r &= \frac{d}{dt} \dot{\mathbf{q}}_r \\ \mathbf{q}_r &= \int_{t_0}^t \dot{\mathbf{q}}_r dt' \quad ,\end{aligned}$$

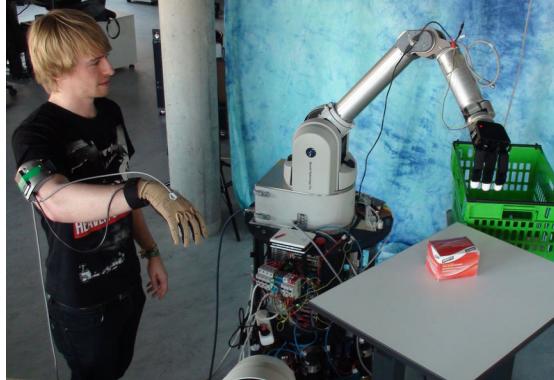
where \mathbf{M} is the mass matrix, \mathbf{C} describes Coriolis forces, \mathbf{g} the gravity torque, $\mathbf{K}_{q,d}$ and $\mathbf{K}_{q,p}$ are joint-space damping and stiffness matrices, \mathbf{q} is the current configuration of the robot, $\dot{\mathbf{q}}$ its velocity and \mathbf{x} the current end-effector pose. The reference velocity $\dot{\mathbf{q}}_r$ depends on the error between the pose of the teleoperator's wrist and the current end-effector pose of the robot, scaled by the stiffness matrix \mathbf{K}_p . The matrix \mathbf{J}^+ is the pseudo-inverse of the Jacobian of the robot end-effector. The reference velocities $\dot{\mathbf{q}}_r$ and accelerations $\ddot{\mathbf{q}}_r$ are computed through numerical differentiation and integration.

We map the finger joint measurements of the data glove $\mathbf{q}^{\text{glove}}$ linearly onto joint angles (via $\mathbf{A} \in \mathbb{R}^{16 \times 22}$) which are used as input for a joint position PD-controller to calculate the applied torques of the Allegro hand:

$$\tau_{\text{hand}} = \mathbf{K}_p^{\text{hand}}(\mathbf{A} \mathbf{q}^{\text{glove}} - \mathbf{q}_{\text{hand}}) - \mathbf{K}_d^{\text{hand}} \dot{\mathbf{q}}_{\text{hand}},$$

where $\mathbf{K}_p^{\text{hand}} \in \mathbb{R}^{16 \times 16}$ and $\mathbf{K}_d^{\text{hand}} \in \mathbb{R}^{16 \times 16}$ are diagonal gain matrices.

A randomly selected object (among 35 objects of the household set (Fig. 2.2)) was placed on a table in front of the robot. The task was to grasp this object and place it into a tote next to the table. Once the object was placed into the tote the experimenter placed a new randomly selected one onto the table. The human teleoperator was standing next to the robot having an unoccluded view onto the robot, object, table, and tote (see Video Fig. 2.1). During a single trial all 35 objects had to be grasped. In the beginning of each trial the operator's current wrist pose was mapped onto the robot's current wrist pose. There were no constraints on the maximum number of attempts or the maximum time needed to grasp an object. The grasp was counted as a success if the object ended up inside the tote. Objects that ended outside



Video Figure 2.1: In the teleoperation grasping experiment a human user controls the end-effector pose of a robot arm and the 16-DOF Allegro Hand. The operator wears a data glove and his wrist pose is located with an electromagnetic tracker. The task is to pick up objects from a table and put them into a box. The teleoperator can only rely on visual feedback. [<https://youtu.be/4gdAMPVCCsY>]

of the robot's workspace were counted as failures. The operator could also ask for the next object if she felt that no successful grasp was possible. We conducted five trials per object, totaling 175 trials.

RESULTS AND DISCUSSION

The results show a total success rate of 82%, which is similar to the previous human experiment (79%). If we only compare the subset of 20 objects used in both experiments (excluding the wooden objects), there is a small performance decrease of 82% vs. 88%. The errors in the hand-on-a-stick experiment were more systematic than in the case of teleoperation: The RBO Hand 2 could not grasp certain objects at all (due to its size; box with straws, pen basket), while the Allegro Hand could grasp every object at least once but had in general more failures due to slippage or instabilities (which could be caused by the delay and noise of the teleoperation setup). But in general, the results show that despite the limitations induced by using a robot arm and not having any haptic feedback, grasp success is not significantly decreased. Note that haptic feedback was most likely replaced by the operator's continuous visual feedback.

The experiments also show that there is a common temporal structure in the applied grasping motions. Fig. 2.3a shows three quantities plotted as a function of time: the linear and angular velocity of a reference frame rigidly attached to the hand and the change of the grip aperture. The grip aperture is the distance between the tip of the thumb and the index finger of the Allegro Hand, which we calculate using forward kinematics. The plot shows mean and standard deviation over all successful grasps. Note that we do not use any time warping technique to align the time series as it is usually done. Instead, we only align the trajectories according to a labeled timestamp of the grasp itself ($t = 0$). Still, a common temporal struc-

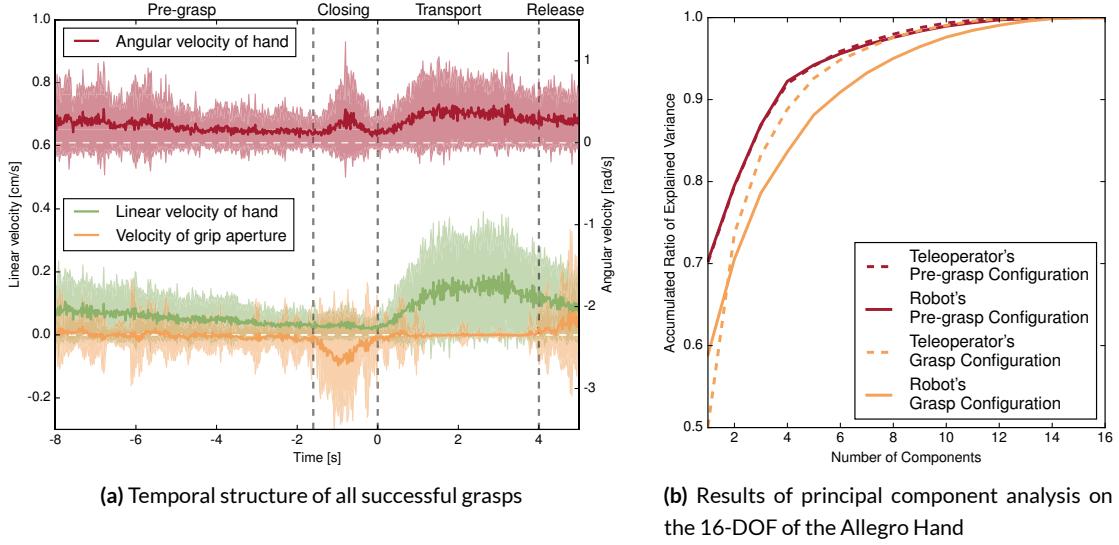


Figure 2.3: The plots show the structure of successfully teleoperated grasps with the Allegro Hand. (a) When aligning all time series a consistent temporal structure is visible. The velocity of the hand decreases until the object is reached. In this phase the finger joints hardly move, as can be seen by the constant grip aperture. Then the grip aperture significantly changes to make contact with the object. Interestingly, there is a peak in the angular hand velocity during that phase. This is due to the large size of the Allegro Hand. (b) The plot shows the accumulated ratio of explained variance using a principal component analysis on the 16 DOF of the Allegro Hand. We compare commanded values by the operator (dashed lines) with actual velocities in the robot hand (continuous lines). We also compare between pre-grasp ($t = -1.6$ s, red lines) and grasp ($t = 0$ s, yellow lines) configurations. The pre-grasp configurations are more easily explained than the grasp configurations. But it also shows that the operators commanded configurations can be explained with fewer Eigenvectors than the resulting Allegro Hand configuration. This difference shows the effect of adaptability in the hand.

ture is visible: The grasps start with an approaching phase in which the hand moves towards the object without changing any internal DOF (grip aperture does not change). During this phase the hand decelerates (angular and linear velocity constantly decreases). Once the hand is close to the object ($t = -1.6$ s) the fingers start to move, indicated by an decreasing grip aperture. Interestingly, the hand pose is not static during this closing phase, especially the changes in orientation are significant. This is most likely due to the large size of the Allegro Hand: the contact locations are influenced significantly by the orientation of the hand in conjunction with the configuration of the fingers. After the grasp is established, the hand lifts the object and the internal DOF are kept constant (very small standard deviation of the grip aperture velocity). When reaching the target location, the grip aperture increases again and the object is placed in the tote. The observation of this temporal grasp structure is also in par with old results by Jeannerod (1981). He distinguished a transport and a grip phase in human grasping, although on a much smaller time scale since our teleoperation setup (without any

haptic feedback) causes the human to act more cautiously.

Apart from the temporal structure, the experiment also confirms a low-dimensional structure of the used hand configurations. Although the teleoperator could control all 16 DOF of the Allegro Hand (the rank of the mapping matrix \mathbf{A} is 16), the successful grasping configurations can be represented with a linear combination of much fewer basis vectors. Fig. 2.3b shows the result of applying a principal component analysis to the successful grasping configurations. We can see that already four eigenvectors can be used to explain 84% of all the variance in the data. This is in line with older results by Santello et al. (1998). If we look at the hand's pre-grasp configurations, we can see that four eigenvectors describe even 92% of the variance. This hints to the fact that during the closing phase the interactions between object and hand add complexity. But interestingly, this added complexity is not fully reflected in what the teleoperator does. If we apply principal component analysis on $\mathbf{Aq}^{\text{cyberglove}}$, the plot shows that the grasp configuration lies on a lower-dimensional hyperplane (three eigenvectors of the teleoperators configurations explain the same variance as four of the Allegro Hand). Since this difference is not present when comparing the (contact-free) pre-grasp configurations, we can assume that the contact creates more varying configurations while the teleoperator's control input does not need to account for it. This result is in line with the concept of adaptive synergies (Grioli et al., 2012) and the notion of prescriptive (teleoperator) vs. descriptive (robot) synergies by Brock and Valero-Cuevas (2016).

2.1.3 MITTEN THOUGHT EXPERIMENT WITH THE RBO HAND 2

The previous experiment showed that there is significant structure in the grasping process. In the next experiment we wanted to enforce this structure and verify that it will not hamper grasp success. This leads us to a robotic version of the initially explained Mitten Thought experiment: A robotic hand is mounted on a fixed stand (resembling the glove-wearing participant in the Mitten Thought experiment) and a human user is required to position an object close to the hand, choose a pre-shape and give a closing command so as to maximize grasp success. In contrast to the previous experiment the human user has no continuous visual feedback that can be used to correct a grasp.

We used 40 different objects (Fig. 2.2) from the household item set and the wooden object set. Grasping was done with the RBO Hand 2. The human user initiated the grasping procedure by selecting a closing strategy and pressing a button. We used the same two closing strategies for the RBO Hand 2 as in the first experiment: a power grasp with thumb abduction and one involving thumb adduction. Each object was grasped five times, totaling 200

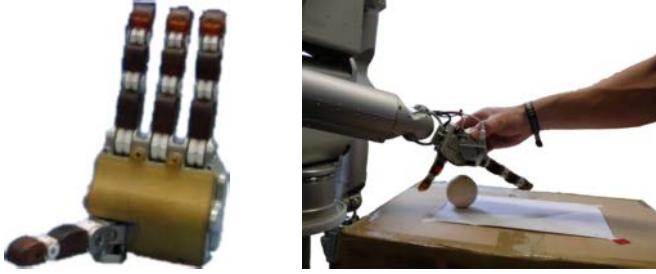


Figure 2.4: In the teleoperation experiment with the Meka M2 Hand, a human operator selects a pre-grasp pose and a compliance mode. We record this information for each object and replay the grasping motion.

trials.

Out of all trials 149 were successful, which resembles the results from the first experiment, in which a hand-on-a-stick was used (80% vs. 79%). It is also similar to the teleoperation results (80% vs. 82%). These results show that limiting the grasp representation by choosing a pre-grasp pose does not hamper grasp success. The hand’s adaptability is capable of compensating slight pose deviations. The failed grasps are due to heavy and large objects.

Note that in this experiment we are grasping objects out of ‘thin air’, there are no other things to take care of (like a table) as in the first two experiments. This might be an advantage. But on the downside the static pre-grasp pose limits us to strategies that do not manipulate the object before grasping, something we observed in the first experiment. To relax the assumption of grasping out of thin air, we conducted a final experiment.

2.1.4 TELEOPERATION EXPERIMENT WITH THE MEKA M2 HAND

In a final experiment we wanted to verify that the two limitations present in the previous experiment (not using a real robot arm and grasping objects in free space) do not significantly affect our insights. Therefore the robot hand is mounted on a 7-DOF Meka A2 compliant arm and objects are placed on a table surface. A human user is asked to move the hand in such a position that a subsequent closing motion will grasp the object. We record the position, replay the motion including the grasp and measure success. Additionally, we record the hand configuration during pre-grasp and after grasping. We use the Meka M2 Hand, a compliant tendon-driven underactuated hand. Note that the role of the human user is very similar to the one of the experimentator in the Mitten Thought experiment.

A total of 20 different objects were presented on a support surface in front of the robot. We presented eight of the objects in a horizontal and vertical orientation. The human user moves the hand while the arm is only applying the torques necessary to compensate gravity (see Fig. 2.4). Beside choosing the grasping pose, the user also chooses a grasp type. These are inspired by [Feix et al. \(2009\)](#) and include: a power grasp with thumb abduction, a power

grasp with thumb adduction, and a precision/pinch grasp using thumb and index finger. We use a joint space controller to replay the user's chosen grasping motion. A grasp is deemed successful if the object can be lifted up to 10 cm and rotated by 60° without falling. We conducted a total of 135 grasping trials, presenting each object up to five times.

RESULTS AND DISCUSSION

Out of all trials 117 grasps were successful (87%). This shows again that only choosing a pose and closing strategy already results in reliable grasping. Objects that failed included very small ones (a ball with 10 mm radius, a torus with a height of 13 mm) and objects that were too large to be enveloped by the hand (a disk with 100 mm radius). Another failure case involved a medium sized triangular prism. While there was no problem to grasp the upright standing prism, it was close to impossible to be lifted when it was laying flat on the table. This is even hard for a dexterous human grasper since only two of the three sides of the triangular shape can be accessed. Here, successful grasping requires either very high contact forces, high frictional forces, hand features such as fingernails, or more elaborate pre-grasp manipulations.

In general, the high success rate of grasping in this experiment is again due to the adaptability of the hand. The human user most often chose a power grasp with thumb abduction (85%) where the underactuated, rubber-made fingers match the shape of different objects. This compliant behavior is reflected in the hands pre-grasp and post-grasp configuration for the successful power grasps (see Table 2.2).

Joint (Finger)	1 (Thumb)	2 (Thumb)	3 (Index)	4 (Middle)	5 (Little)
Initial SD [°]	2.0	4.0	4.9	3.32	2.3
Final SD [°]	5.2	53.0	66.0	51.3	73.0

Table 2.2: The standard deviation of the initial and final grasp postures of successful power grasps.

The initial standard deviation per joint is low (it is not zero due to tendon slack) while the final configurations vary significantly. This variation happens despite applying the same closing strategy. It shows that the hand adapts to the actual object shape without the user having to explicitly control it. The standard deviation of the first joint is still low, since it controls the abduction/adduction of the thumb.

2.1.5 PRE-GRASPS, PRE-SHAPES, AND COMPLIANCE MODES

Our experiments revealed a certain reoccurring structure of grasping motions. In order to relate to this structure we now introduce some terms which describe pivotal states within the continuous grasping process. We will use them throughout the rest of the thesis.

PRE-SHAPE We abstract the hand geometry into a set of pre-shapes (or pre-grasp shapes) – configurations that define the internal DOF of the hand before contacting the object. This is a common concept, e.g. used by Miller et al. (2003) who distinguishes cylindrical and spherical hand pre-shapes. It is also commonly used to categorize human grasps (Napier, 1956, Feix et al., 2009).

PRE-GRASP A pre-grasp is the state of the hand before closing the fingers. It consists of the pre-shape (which defines internal DOF) and the pre-grasp pose which specifies the external DOF of the hand. We refer to *pre-grasp manipulation* when describing all non-prehensile actions that do not represent a grasping configuration but are crucial to obtain a grasp, such as sliding, toppling, reorienting or pushing an object.

CLOSING MOTION A closing strategy or motion defines how the hand behaves between pre-grasp and grasp. This can be as simple as closing the fingers until motor stall while keeping the hand pose stiff. But also more compliant strategies in which the wrist is controlled based on force-feedback to keep contact with a table surface while the fingers close are possible (see Sec. 5.1.1).

COMPLIANCE MODE Robotic hands may be configured to exhibit different modes of compliance. A compliance mode specifies a pre-grasp of the hand and a closing motion. A compliance mode thus captures the hand’s ability to conform to a particular object geometry in the absence of explicit sensing and control.

2.1.6 CONCLUSION

We showed a series of small experiments in which human knowledge was harvested to gain insights into the grasping process. With each new experiment we limited the human’s access to perception and control during grasping. The imposed limitations/structure did not significantly decrease grasp success while reducing the dimensionality of the grasping solutions.

Chapter 2 ▼ ADAPTABILITY BETWEEN HAND AND OBJECT

We conclude that successful grasping can be decomposed into a pre-grasp pose and pre-shape of the hand combined with a compliant closing strategy. Only a few pre-shapes and closing strategies are necessary.

In all experiments, the hand's adaptability in the last phase of contact interactions was the main enabler for grasp success. The specific implementation of adaptability did not play a role. In our experiments we showed that the adaptability can be realized by different hands in very different ways: the high number of DOF of the Allegro hand, the under-actuation of the Meka Hand, or the mechanical compliance of the silicone used in the RBO Hand 2.

2.2 THE SUN-FLOWER-ANNULUS HYPOTHESIS

The previous experiments revealed a simplified structure of successful grasping based on the exploitation of adaptability. But so far we gained only partial insights into the exact workings of shape matching between hand and object. Our object sets were limited and measuring all contact interactions between hand and object was not possible.

Additionally, the previous experiments involved human adults, who rarely make mistakes in grasping. This bias prohibits us to learn from those grasping strategies that do not work. To gain insights about the latent structure of the space of possible grasps, real-world experimentation is not suited. It is too slow and tedious to record large quantities of grasp attempts. Instead we will rely on simulations in this section to validate some of the findings about the relationship between object geometry and grasping strategies.

We introduce the *Sun-Flower-Annulus hypothesis* (SFA) as a characterization of the effect of shape complementarity between the hand's compliant grasping strategy and object shape on grasp success. It is best described as a marginal probability distribution of grasp success over the imaginary space of all possible object geometries. The specific grasp strategy is marginalized out. The level sets of this distribution are illustrated in a cartoon-like fashion in Fig. 2.5. There are three regions in the space of object shapes that are indicated with different colors and to which we attach specific semantics:

- **THE ANNULUS (RED)** corresponds to those objects for which caging effects dominate grasp success. This is true for medium sized objects (w.r.t. to the hand), independent of their shape. The particular configuration of the hand is less relevant as long as the grasp is enveloping the object. Finding grasps for these kinds of objects is easy.
- **THE SUN (YELLOW)** is characterized by its rays which represent objects of a specific shape that can still be grasped when being scaled up in size. It is much more difficult to find a successful grasping strategy than in the annulus, because the chosen strategy needs to match the object geometry precisely.
- **THE FLOWER (GREEN)** represents small objects which are hard to cage with few fingers. Thus, it is difficult to find successful grasps because they need to rely on high precision.

Fig. 2.6 visualizes example grasps from the different regions for the three-fingered Barrett

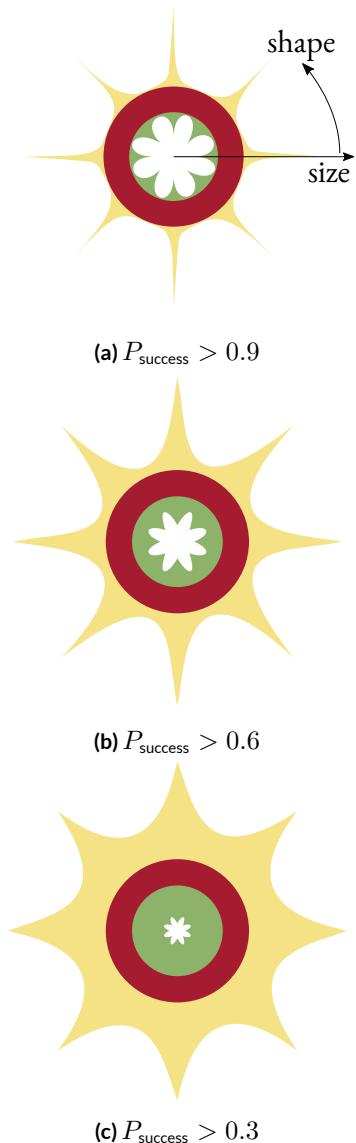


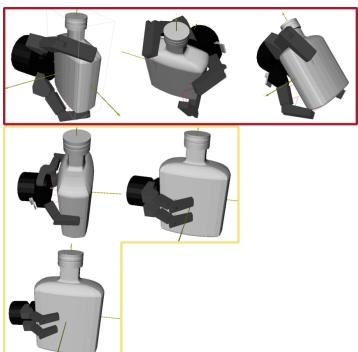
Figure 2.5: The cartoons illustrate the effect of complementarity of compliance mode and object shape on grasp success. Each of the three cartoons (a)–(c) represents a level set of the marginal probability distribution of grasp success. The polar coordinates (r, θ) correspond to object size and object shape. The yellow, red, and green regions indicate the parts of the plane where grasping success exceeds a given threshold. The red annulus corresponds to a region of the space in which caging effects dominate the probability of grasp success. The radially emanating rays of the yellow sun correspond to a particular compliance mode of the hand. If the shape of the object matches the compliance mode, grasp success is increased radially. As object size increases fewer pre-grasp poses yield successful grasps. The inverted, green flower around the origin characterizes a region in which precision grasps dominate.

Hand. An informal survey of the literature in robotic grasping imparts the impression that many of the reported grasping experiments lie in the annulus of the SFA where grasping strategies are less discriminative. This observation should impact future grasping benchmarks. One could argue that experiments in the annulus are appropriate as most objects in the real world lie in the annulus by design. Others may want to counter that general grasping can only be benchmarked outside the annulus.

In the following we would like to try to validate the SFA hypothesis. We will do this first in a large variety of 2D simulations (see Sec. 2.2.1) and afterwards on a more constrained set



(a) Annulus



(b) Sun ray: box



(c) Sun ray: cylinder

Figure 2.6: The three subfigures show example grasps for different regions of the SFA cartoon. The frame colors indicate SFA regions. Subfigure (a) shows example grasps for three objects (rows) within the annulus, each object corresponding to a different angle of the polar coordinate system. For objects in the annulus, grasp success is strongly influenced by caging effects and less by shape complementarity. As object size increases, the influence of shape complementarity and therefore compliance mode and hand pose on grasp success increases, as illustrated for two objects with distinct shapes in subfigure (b) and (c). In these subfigures, the first row corresponds to object sizes in the annulus. In each successive row, object size increases; fewer hand poses will lead to successful grasps, corresponding to the sun ray.

of 3D simulations (see Sec. 2.2.2).

2.2.1 VALIDATION OF SFA IN 2D

We first need to define a 2D shape space based on a polar coordinate system. Each shape (r, θ) is parametrized by the distance r and the angle θ w.r.t. the origin. According to the SFA hypothesis the distance r is the size of the shape. We simply use r as a uniform scaling factor.



Figure 2.7: To collect evidence for the SFA hypothesis, we simulate vast amounts of grasps with a two-fingered hand in 2D using the daVinci simulator (Berard et al., 2007).

The definition of the shape parameter θ is less concrete. We define θ to mark five canonical shapes (square, rectangle, circle, four-pointed star, triangle) at the coordinates $(0, \frac{2}{5}\pi, \frac{4}{5}\pi, \frac{6}{5}\pi, \frac{8}{5}\pi)$. If θ does not match any of these coordinates, the resulting shape is a linear interpolation between the two nearest neighbors weighted by their distance. We describe all shapes by discretized polygons. Thus, the interpolation is defined on pairs of vertices of the polygons that are closest in the Euclidean sense. We assume a uniform density, i.e., the center of mass coincides with the geometric center. This way we have defined a continuous 2D shape space in which we can evaluate grasp success.

To fully evaluate all possible grasps we choose an extremely low dimensional parametrization. Our grasp strategies are defined by a single scalar ρ that describes the approach direction of the hand in the plane. The hand always approaches the object along the line that crosses the object's geometric center. It will stop as soon as it contacts the object and initiate the closing of its fingers. Based on the shape complementarity between hand and object some approach directions will result in better grasps than others, only for a fully symmetric object like the circle the grasp success will not depend on ρ .

We simulate a common two-fingered gripper. Each finger consists of two joints similar to the design of the Barrett Hand. We use the 2D simulation framework daVinci (Berard et al., 2007) to evaluate grasping motions. This dynamic physics engine has proved to provide realistic outcomes, reproducing real-world manipulation results (Chakraborty et al., 2014). Fig. 2.7 shows snapshots of an example grasp from our experiment. We calculate the quality of each grasp by using the ε -metric (Sec. 1.2.3). A grasp is considered successful if its ε -metric exceeds a fixed threshold.

We generate 1000 different shapes by sampling uniformly in Cartesian coordinates before transforming the shape parameters to polar coordinates. We do not sample uniformly in polar coordinates to avoid a low density in the outer regions. For each shape we evaluate 50 grasping directions ρ equidistantly distributed on the interval $[0, 2\pi]$. This results in a total of 50 000 simulations.

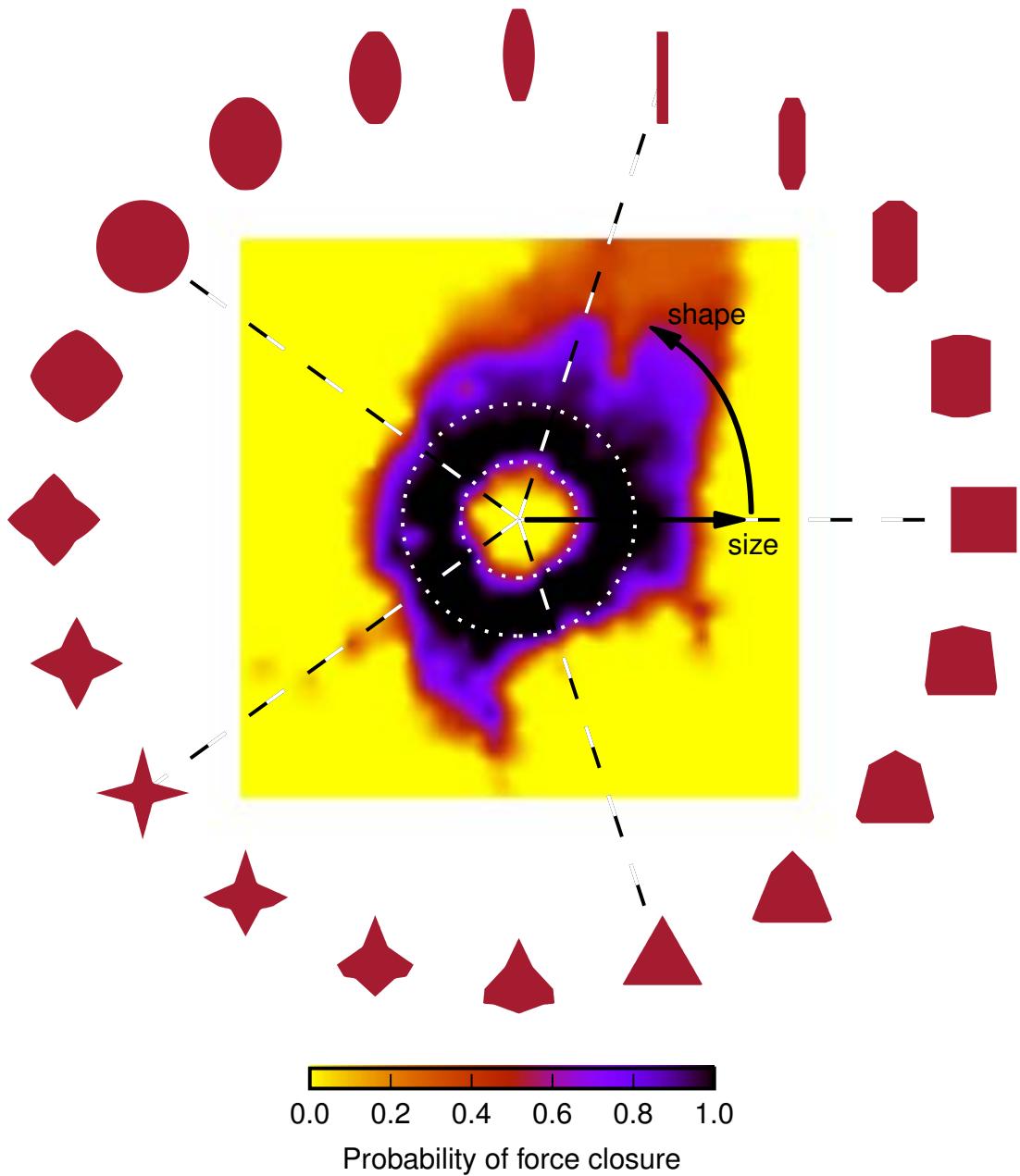


Figure 2.8: The plot shows the probability of the best grasp for a particular shape being in force closure. These are the results of 50k simulations in 2D. The distribution resembles the hypothetical cartoon version in Fig. 2.5. The annulus region is indicated by two dotted circles. In the region of the sun, a few rays can be identified where the compliance mode of the hand matches the object shape. This is especially true for elongated objects that are approached at their shorter side like the rectangle in the upper right.

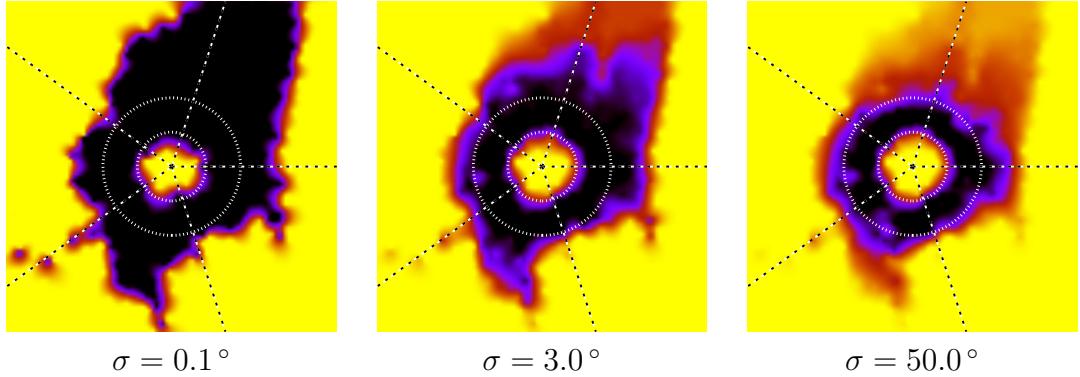


Figure 2.9: The plots show how increasing noise levels in the approach direction of the grasp affect success. The color codes are the same as in the previous plot (Fig. 2.8): yellow indicates zero probability of force closure and black a probability of 1. It can be seen that the annulus region (between the two dotted circles) is largely unaffected by noise. Outside the annulus, the match between the hand's compliance mode and the shape is much more sensitive to noise in the pre-grasp pose.

RESULTS AND DISCUSSION

Fig. 2.8 shows the probability of grasp success given the shape parameters (r, θ) . We calculate it by averaging the grasp success over all 50 grasp strategies ρ for each of the 1000 sampled shapes. The visualization uses a linear interpolation between those data points based on radial basis function kernels.

We can see the same main components as in our hypothesized structure (Fig. 2.5): the sun, flower, and annulus. For the medium-sized objects in the annulus the particular approach direction ρ does not matter. All grasping strategies produce equally good results.

In contrast, small objects in the flower cannot be grasped successfully regardless of the choice of ρ . If we would change our success criterion from the force-closure based ε -metric to also include caging, the flower will contain only successes. But assuming that this planar scenario represents a top-down view of a 3D tabletop setting, the caging condition would not yield robust grasps.

Grasp success of large objects in the sun usually depends on the approach direction. Exceptions are fully symmetric shapes, such as the circle. The most clearly visible rays are those pertaining to the rectangle ($\theta = \frac{2}{5}\pi$), the four-pointed star ($\theta = \frac{6}{5}\pi$), and the one closely related with the triangle ($\theta \approx \frac{7}{5}\pi$). In these cases some grasp directions resulted in a match between the shape of object and hand. For the rectangle, ρ needs to align with one of the two shorter sides to grasp it. This effect extends to nearby shapes that are interpolations between the rectangle and the square/circle. It is indicated by the width of the associated ray ($\theta \in [\frac{1}{5}\pi, \frac{3}{5}\pi]$).

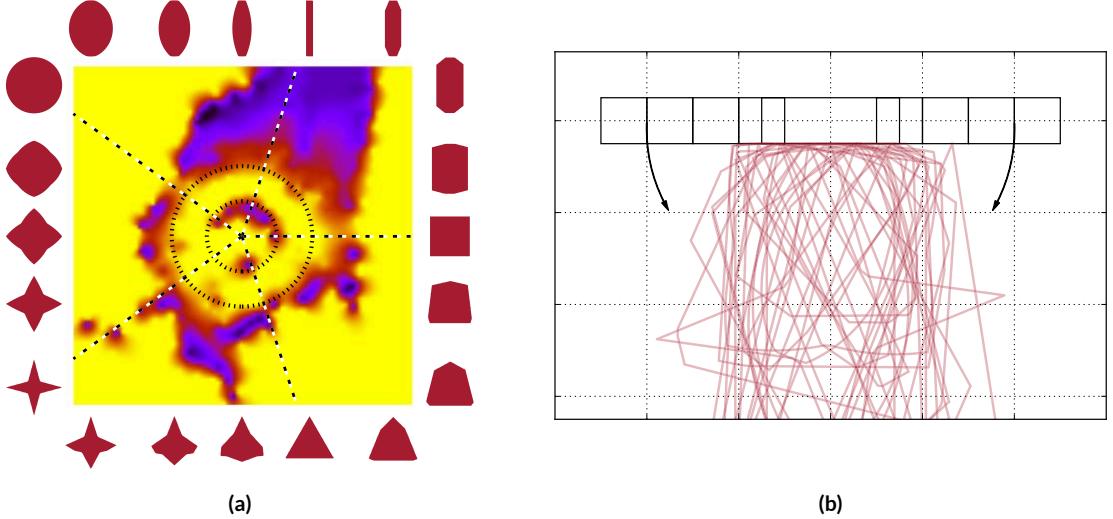


Figure 2.10: Plot (a) shows the force-closure probability of grasps that are force closure for low levels of noise ($\sigma = 0.1^\circ$) but not for high levels ($\sigma = 50^\circ$). As a result, we see a high probability for shapes for which a potential grasp planner needs to find a specific pre-grasp pose. Plot (b) shows an overlay of 30 samples (red shapes) and pre-grasp poses (black hand at the top, before the fingers close) from this distribution. We can see that there is a certain regularity: The hand tends to be positioned at the shorter side of the different objects to grasp them.

The four-pointed star exposes some kind of singularity: A scaling transformation will not change the local geometry of the convex corners. But since these parts are used to grasp the star, increasing its size will not change grasp success.

We also evaluated how sensitive grasps are w.r.t. noise. We injected different amounts of Gaussian distributed noise in the actual orientation of the objects. Fig. 2.9 shows how grasp success is affected by different levels of noise, by increasing the standard deviation of the Gaussian noise process. We can see that grasp success outside the annulus is affected the most; grasps in the annulus remain successful despite significant noise. This hints to another insight: When trying to find good grasps, objects outside the annulus are the hardest to deal with. Grasp parameters need to be chosen carefully to ensure a shape match between hand and object.

To get an idea about the successful grasps outside the annulus, i.e. those that we would like to be able to plan, we conducted another small experiment. Fig. 2.10a shows the probability distribution over shape parameters which have very few successful strategies. We sampled from this distribution to analyze successful grasps for difficult object geometries. Fig. 2.10b shows an overlay of 30 pre-grasp poses and objects of these samples. The hand is shown at the top, before initiating the closing. All of these pre-grasps result in successful grasps.

We can observe that the poses of the object shapes expose a regularity: surfaces are mostly aligned vertically to allow a shape match with the hand. As a consequence, finding grasps for these “difficult” objects, means finding local geometric regularities that allow the closing hand to match with the object. We will exploit this insight in the next chapter when presenting perceptual algorithms that plan grasps from sensor data.

2.2.2 VALIDATION OF SFA IN 3D

We complemented the 2D experiments with experiments in 3D to provide more quantitative support for the SFA characterization. We focus on validating our characterization of the annulus and of the grasp success variations along the rays of the sun.

Similar to the experiments in 2D, we simulate a hand with two links per finger – the Barrett Hand, a three-fingered 4-DOF hand. We employ two different pre-shapes: a spherical (the three fingers are 120° apart) and a cylindrical one (the three fingers are parallel to each other). The break-away mechanism of the Barrett Hand is simulated by keeping the pose of the proximal link of each finger constant as soon as collision is detected and only moving the distal link until collision or the joint limit. In contrast to the 2D experiments we do not simulate the full physics. Instead we rely on a large number of quasi-static grasps using OpenRAVE ([Diankov, 2010](#)). For each grasp we calculate the commonly used ε -grasp quality metric (Sec. 1.2.3), which indicates the minimum magnitude force required to break the grasp. Grasping experiments were conducted with three prototypical geometries: spheres, cubes, and cylinders. The results for sphere and cylinder match closely and we will only discuss results for cylinders.

Object shapes are deliberately kept simple to make a near-optimal grasping strategy obvious. Such a strategy is required to draw meaningful conclusions about the SFA, as it captures intuition about optimal strategies. For spheres, the robot employs the spherical hand pre-shape, approaches the center of the sphere with the center of the palm until contact is made, and then closes the fingers. For cubes (and cylinders), the robot employs the cylindrical hand pre-shape; during the approach the hand is aligned with the major axes of the object.

RESULTS AND DISCUSSION

The top graph in each of the four panels in Fig. 2.11 shows grasp quality as a function of object size and hand pose error along either the x or y direction. The graph below the image plot shows the mean grasp quality $\varepsilon_{\text{mean}}$ (red solid line) and the maximum grasp quality ε_{max} (green dotted line) across all pose errors as a function of object size. The blue dashed line

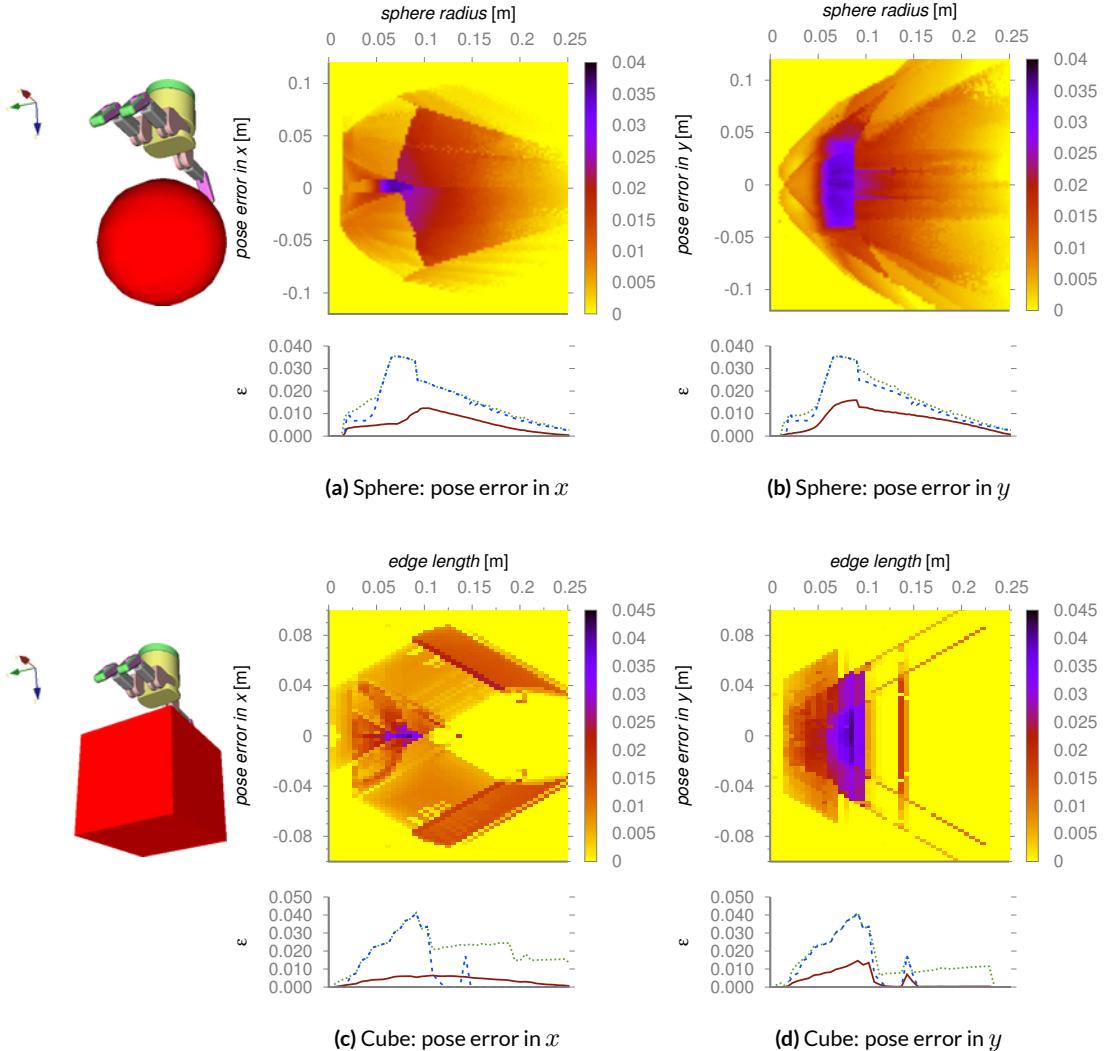


Figure 2.11: Grasp quality as a function of object size and pose error; the color scale indicates the grasp quality measure ε ; in the plots the red solid line corresponds to $\varepsilon_{\text{mean}}$, the green dotted line shows ε_{max} , and the blue dashed line ε_0 .

shows the grasp quality ε_0 in the absence of a pose error. For the cylindrical grasp experiments with cubes, the fingers close along the y -axis.

The close match between ε_{max} and ε_0 in all graphs provides a sanity check for the chosen grasp strategies. All graphs show peak grasp qualities for medium object size, providing evidence for the existence of the annulus, where grasp quality is high. The data also shows that grasp quality, indicative of the probability of grasp success, decreases as object size increases beyond the annulus. The sudden drop in grasp quality in Fig. 2.11d is the result of only two fingers making contact with the cube. Inside the flower, grasp quality is poor, indicating that

Chapter 2 ▼ ADAPTABILITY BETWEEN HAND AND OBJECT

different compliance modes might be required for robust grasping.

2.3 THE FUNNEL VIEW OF GRASPING

Our goal is to leverage hand adaptability and compliance in the development of novel grasping algorithms. The underlying hypothesis is as follows: compliance in the hand, irrespective of whether it is achieved passively or actively, can be viewed as a funnel that transforms configurations in a large region of the configuration space into a configuration in the smaller region of successful grasps (see Fig. 2.12, the bottom part of the funnel captures the compliance of the hand). Compliance therefore introduces robustness to uncertainty and reduces sensing requirements. Given this hypothesis, a compliance-centric grasping algorithm must

1. characterize the entrance to the bottom part of the funnel and
2. transfer initial configuration into those that lie at its entrance.

This is illustrated by the top part of the funnel in Fig. 2.12. We will show in the next chapter that compliance-centric grasping algorithms exhibit robust grasping performance, significantly reduce the requirements on perception, and eliminate the need for explicit planning of contact points.

We can view our grasp experiments from the previous two sections also through the lens of funnel transformations. The SFA hypothesis (Sec. 2.2) revealed some structure in the space of object geometries and grasping. In the case of the annulus, the entrance to the funnel induced by the hand’s adaptability is wide, uncertainty can be compensated easily. In contrast, the funnel entrance becomes more and more narrow as we consider objects in the sun. For large objects, a wide entrance only exists along the rays of the sun, where the compliance mode of the hand matches the object shape. Our initial experiments (Sec. 2.1) also explored the benefits of wide funnel entries through hand compliance. They showed that very few pre-shapes and closing strategies are necessary to grasp a wide variety of objects. In these experiments, humans (instead of algorithms) were used to find funnel entries.

Adaptability and compliance play a critical role in any real-world grasping experiment. Humans extensively rely on compliance to achieve robust grasping. Experiments performed by Santello et al. (1998) showed that humans use a small set of pre-grasp hand postures when grasping objects of widely varying shapes. Robust grasping then seems to be the result of “simply closing the hand”, leveraging the compliance of the skeletal hand structure, muscles, tendons, and skin to achieve complementarity of hand and object geometry. We showed that adaptability is implemented in many different ways in a lot of recent robotic hand designs (Sec. 1.1). The funnel view of grasping tries to emphasize the centrality of adaptabil-

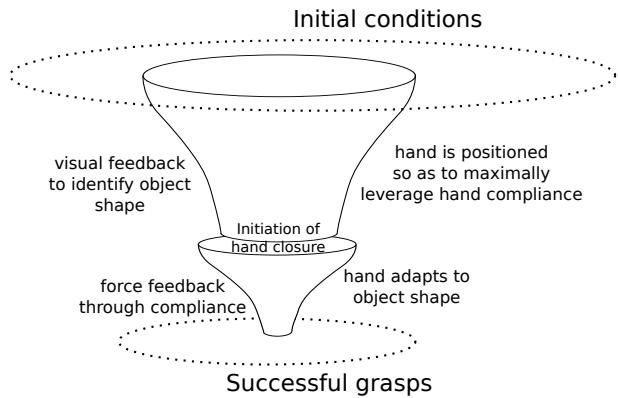


Figure 2.12: We visualize grasping algorithms as two consecutive funnels, transforming initial configurations into successful grasps through the use of feedback. The first funnel leverages visual feedback to transform the starting conformation into a configuration that lies inside the entrance of the second funnel. The second funnel relies on the force interaction between the hand and the object and the hand's compliance to obtain a successful grasp.

ity/compliance for grasping. Although this focus is not entirely new, we think it is an important step towards robust grasping.

The concept of relating motion to a sequential application of funnels has been explored by multiple authors in the past. [Mason \(1985\)](#) gives a humorous analogy to cleaning:

“Sweeping a floor is perhaps the most dramatic funnel operation. A sensor-based, pick-and-place approach to the dirty floor problem would require that the robot locate each dust mote visually, grasp it between the fingers, and place it in the dust-bin. This is rather tedious compared to sweeping the floor with a broom. It is simply more efficient to not worry about the locations of individual motes of dust.”

[Lozano-Pérez et al. \(1984\)](#) describe fine-motion plans as a sequence of funnel operations. Their notion of *pre-image* defines a set of states from which the same action will result in achieving a goal. These pre-images are what we refer to as “funnel entrance”. The calculation of pre-images requires to know whether the goal is reachable *and* recognizable. To relax this assumption, [Erdmann \(1986\)](#) introduced the notion of *back-projection*. The calculation of back-projections only relies on goal reachability and was used to compute plans for orienting parts via a tilting tray without the help of any sensors ([Erdmann and Mason, 1988](#)). [Burridge et al. \(1999\)](#) partition the obstacle-free state space using local funnels for batting maneuvers. [Tedrake et al. \(2010\)](#) computes Lyapunov functions to evaluate the basins of attraction of a randomized tree stabilized with LQR feedback that represent feedback motion plans. The tree can be viewed as combination of funnels.

The classical literature on grasp and restraint analysis predominantly considered static or quasi-static scenarios (Sec. 1.2). Although more complex models that include compliance exist, they are built on top of existing formulations, and are hardly used in practice due to their

complexity. Instead we want to shift the central focus towards adaptability and uncertainty, which is encompassed by the funnel view of grasping. Rather than taking a bottom-up approach in which local contact phenomena give rise to higher-level behavior, the funnel view is a top-down view which approximates global interaction behavior based on realistic assumptions about available sensor data. The underlying idea of this thesis is to see grasping as the problem of generating, sequencing, and characterizing such funnels; with a focus on considering compliance and adaptability as an enabler for more powerful funnels.

We think that classical grasp analysis does not oppose the funnel view. It rather emphasizes different aspects of the grasping problem. Ultimately, both views are compatible and should be unified.

2.4 CONCLUSION

In this chapter we developed a first understanding about the interactions between hand and object during grasping. Our exploratory experiments using humans showed that the seemingly high-dimensional grasping motions are based on low-dimensional building blocks that exploit the low-level adaptability of the hand. This underlying structure was shown through successful grasps based on very restrictive motion protocols with various compliant hands. Even when allowing the human operator to control finger motions, the successful grasping motions lie on a much lower dimensional manifold which can be represented by a few prototypes.

We provided a cartoon-ish characterization of the influence of shape complementarity between a hand’s adaptive grasp strategy and the object’s shape. Through experimental validation in simulation we showed that this characterization captures some interesting aspects of compliant grasping. In the subsequent chapter, we will validate this characterization on a real robotic platform.

The explicit consideration of hand compliance in robotic grasping improves performance and robustness. We argued that grasp algorithms should be compliance-centric, i.e. they should deliberately take advantage of hand compliance to improve performance. This lead us to the funnel view of grasping. A funnel represents an uncertainty-reducing action which collapses a large set of initial states into a specified region of desired states. During finger closing hand compliance acts as such a funnel and grasping can be viewed as finding and sequencing funnels with particularly large basins of attraction. Based on this insight, the next chapter will present grasp planning algorithms that try to maximally exploit adaptiveness and compliance.

3

Planning Grasps That Exploit Hand Adaptability

WE HAVE SHOWN in the previous chapter that the SFA characterization of compliance in grasping provides interesting insights that can guide the design of grasping algorithms. To fully take advantage of hand compliance in grasping, we must find the compliance mode of the hand that best matches the object shape. Doing so does not require an exact representation of the shape. Instead, we must obtain some estimate of how well the hand can accommodate an object’s shape in a particular compliance mode—we call this *shape resemblance*. We believe that the identification of a compliance mode is a much simpler perception problem than the one required for compliance-agnostic planning, namely, the acquisition of accurate three-dimensional geometric models. Furthermore, the need for the planning of contact states is completely eliminated.

We consider the problem of robust grasping with hands in the absence of a priori object models. We focus on aspects of object capture and grasp stability under variations of object shape for a given robotic hand.

To evaluate the algorithms in this chapter we use the Barrett Hand 262, a three-fingered hand with four degrees of freedom. We define its compliance modes as shown in Fig. 3.1: a cylindrical and a spherical pre-shape in conjunction with a simple finger closing motion. Adaptability is ensured due to the break-away mechanism of the Barrett Hand (Sec. 1.1.5). The idea is that these compliance modes match a large variety of possibly occurring object



Figure 3.1: The motion of two different compliance modes of the Barrett Hand 262. To find suitable grasps we search for structure that is shaped complementarily.

shapes.

In the following sections we will present and analyze two types of visual features that try to match the hand's compliance modes to object shapes. The first one is a differential contour feature which requires a moving camera but is not affected by the drawbacks of using depth sensors. In contrast, the second type of feature represents the 3D structure of depth measurements by the similarity to a few canonical shapes. Each feature is explained, similar work is shown, and strengths and weaknesses are derived via experimental evaluation.

3.1 FEATURES BASED ON ACTIVE VISION

O'Regan and Noë (2001) state “that the visual quality of shape *is precisely* the set of all potential distortions that the shape undergoes when it is moved relative to us, or when we move relative to it.” They refer to these sets as *sensorimotor contingencies*. This active vision-based characterization of shape seems well suited for our purposes. We observe the changes of object silhouettes under object motion and represent different object shapes by different qualitative changes. How this can be done will be described in this section. Our hope, confirmed in the experimental evaluation in the next sections, is that the resulting visual primitives will be robust and yield good entrances to grasping funnels, as they only capture the overall shape of the object, ignoring less important details.

We will describe five visual primitives based on active vision sensorimotor contingencies. Three of these visual primitives determine shape resemblance for a specific object: *sphere*, *box*, and *cylinder*. The shape resemblance will be used to select the appropriate compliance mode. In addition, the visual primitives must acquire information needed for determining

the hand’s pre-grasp pose. Together, this information is sufficient to execute the most promising grasp strategy.

The simplicity and effectiveness of all active visual primitives is based on the concept of active vision: the motion of the camera is controlled to maximize the visual information obtained from the image stream. The advantage of active vision over dynamic vision (just knowing how the camera moves) when estimating the parameters of simple geometric objects was shown by Chaumette et al. (1996). Our visual primitives are similar in spirit to the ones presented there.

We evaluate our approach to perception in real-world experiments in Sec. 3.1.2. These experiments will show that the output of our visual primitives correlates with grasp success, which implies that they are able to identify the entrance to the funnel induced by hand compliance. We then show in Sec. 3.1.3 that compliance-centric grasping outperforms compliance-agnostic planning in real-world grasping experiments, even when we grant the latter access to a priori world models.

3.1.1 VISUAL PRIMITIVES FOR PLANNING GRASPS

The basic idea underlying the visual shape primitives is as follows: When a camera moves on an imagined sphere around the object of interest while pointing the optical axis towards the center of the object (sphere), the changes in the silhouette of the object reveal information about the object’s shape. We take advantage of this effect to discriminate between shapes. To extract the object’s contour in the camera image we use a simple blob detection (Suzuki et al., 1985).

To control the camera motion to remain on this imagined sphere around an object, we need to estimate spatial information. This is accomplished by the depth primitive and the principal axis primitive.

DEPTH PRIMITIVE

As we are using an eye-in-hand setting with a monocular camera, we are lacking instantaneous depth information. To retrieve depth, we designed an active vision controller that converges to a concentric trajectory around the object’s center (a geodesic on the imagined sphere). The controller moves the camera to keep the center of the object’s projection in the center of the image. In each time step, the controller commands the camera according to its current depth estimate. Motion of the object’s center in the image leads to a correction of the depth estimate.

When the controller converges, the camera moves on a sphere around the object center. The sphere's radius is equal to the estimated depth.

PRINCIPAL AXIS PRIMITIVE

Similar to the estimation of depth we actively estimate the main axis of an object. We derive the principal axis of the detected blob in the image plane via its second order central moments. A visual servoing loop keeps this axes aligned w.r.t. the image border. We are using the expected invariance of the axis' orientation during camera motion to update an estimate of the axis orientation in space. The camera motion describes again an arc around the object's principal axis. [Chaumette et al. \(1996\)](#) showed that this motion results in optimal information gain for the estimation of lines in space.

The visual primitives for estimating depth and the principal axis are executed in parallel, as their desired exploratory motions lie in the nullspace of each other. After their convergence, we invoke the shape resemblance primitives, all of which execute in parallel.

SHAPE RESEMBLANCE PRIMITIVES

To assess the shape resemblance of the object to a box, a sphere, or a cylinder, we determine two simple visual properties of the object's silhouette: its contour area and eccentricity. The resemblance with a particular shape class depends only on the variance of these two properties during the camera motion around the object. To calculate the contour area we count image pixels inside the contour. The eccentricity (or elongation) is the ratio of the lengths of the contour's shortest and longest chords.

A constant projected size and an eccentricity close to 1 indicate a sphere-like object. A constant contour area but an eccentricity $\gg 1$ suggest an object that is rotationally symmetric along its principal axis. We refer to this as the cylindrical shape resemblance primitive. A box shape resemblance is detected when the eccentricity $\gg 1$ and the projected size varies throughout the camera's motion around the object. Our experiments will show that the resulting shape information is sufficient for robust grasping of unmodeled objects.

The feature we present here to decide between different grasp strategies is very similar to spin-images [Johnson \(1997\)](#) which are used for 3D surface matching. Spin-images are histograms calculated by rotating a virtual image plane around the surface normal of a specific pivot point. By counting the 3D points that fall within the same pixel of the image, a characteristic signature of the surrounding surface of the pivot point is created. The area and

eccentricity that we calculate are statistics of these histograms. But while spin-images are generated virtually from point clouds or meshes we generate our features using an eye-in-hand manipulation system and taking images with an actual camera sensor. Thus, our features could also be termed “eye-in-hand spin-images”.

3.1.2 VALIDATION OF RELATIONSHIP BETWEEN SHAPE RESEMBLANCE AND GRASP SUCCESS

Our compliance-centric grasping algorithm rests on the assumption that the shape resemblance values determined by visual strategies are indicative of the width of the entrance to the bottom funnel in Fig. 2.12. The width of the funnel should lead to robustness in the grasping process. Hence, shape resemblance values should correspond to grasp success for the corresponding compliance-based grasp. We will now test this assumption in real-world experiments.

EXPERIMENTAL SETUP

For our real-world grasping experiments, we use a 7-DOF Barrett WAM in combination with a Barrett Hand BH-262 and a PointGrey Firefly camera mounted on the wrist. Fig. 3.2 shows the objects we used: a banana, apple, pepper, sponge, spectacle case, toy bridge, soccer ball, game box, and a cylindrical bottle case. To simplify the segmentation problem for the visual primitives, the game box and bottle case were wrapped in yellow paper.

We employed three grasp types: spherical grasp, cylindrical grasp, and box grasp, each corresponding to one of the visual strategies described in Sec. 3.1.1. The cylindrical grasp and the box grasp both share the cylindrical compliance mode of the hand but differ in the way they select the appropriate hand pose and approach direction.

Our experimental procedure is as follows: One object at a time is placed inside the robot’s workspace on a white table in a specified pose. In each grasping trial, the visual primitives are used to determine the shape resemblance. Subsequently, the corresponding grasp strategy is executed. For each of the nine objects and three grasp types we conducted 10 trials, for a total of 270 grasping trials. A grasp was deemed successful if after lifting the object 10 cm off the table no obvious slippage occurred within 10 s.



Figure 3.2: Objects used in the grasping experiments (from left to right): banana, apple, bell pepper, toy bridge, spectacle case, sponge, soccer ball, bottle case, and game box.

RESULTS AND DISCUSSION

The outcome of this experiment is shown as a scatter plot in Fig. 3.3. The color of the circles represents the visual primitive/compliant grasp (box is red, spherical is blue, cylindrical is yellow). Each circle's size represents the grasp success rate. The circle center's y -coordinates indicates the shape resemblance value determined by the corresponding visual strategy. The circle with the highest y -coordinate for each object represents the compliant grasp selected by our grasp algorithm; its diameter therefore represents the algorithm's success rate for that object. The averaged success rate over all objects is 95.6 % (86 out of 90 trials, two failures with the soccer ball, one with the spectacle case and the bell pepper).

We can interpret the results in light of the SFA characterization from the previous chapter (Sec. 2.2): Given our knowledge of the hand's grasping volume, the first six objects were chosen to lie in the annulus. Inside the annulus the three grasping strategies do not differ significantly in their grasp success. The minor variations in the case of apple, banana, and bell pepper are correctly detected by the visual strategies: they select the compliant grasp with the highest success rate. This indicates that even in the annulus there is a good match between shape resemblance value and predicted grasp success of the corresponding compliant grasp. For the spectacle case, however, we select the weakest strategy, even though it fails only once out of ten trials and we still achieve a success rate of 90 %.

The size of the remaining three objects (soccer ball, game box, bottle case) places them outside of the annulus. Each represents a different class of object shapes, thus representing a different ray of the sun. As expected, grasp success varies widely in the rays of the sun. The visual resemblance detected by the visual primitives strongly correlates with grasping success of the corresponding compliant grasp. For each of the objects, a different compliance mode

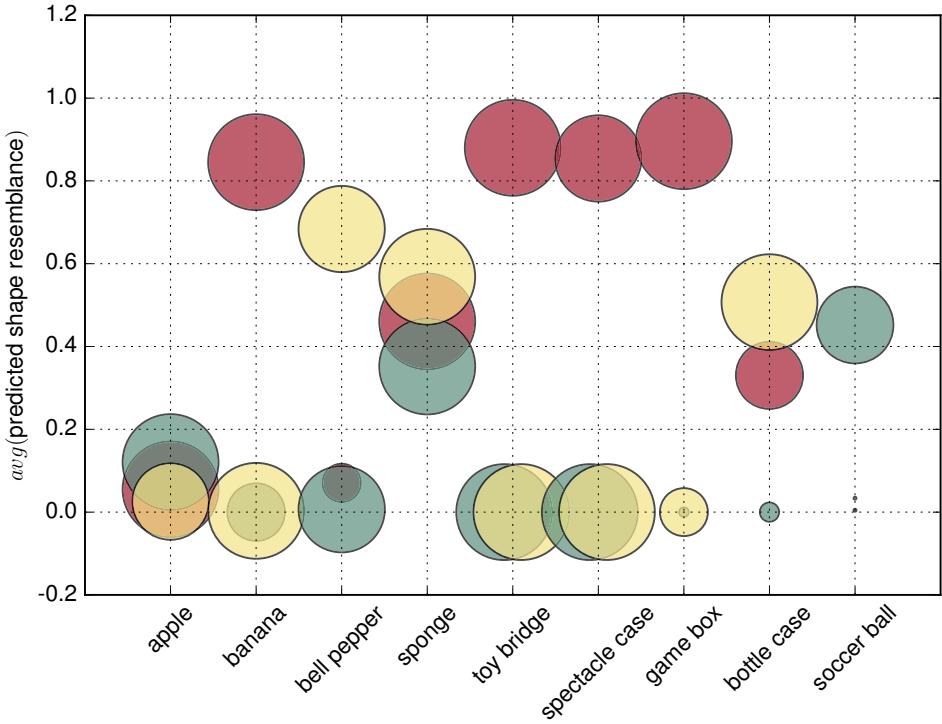


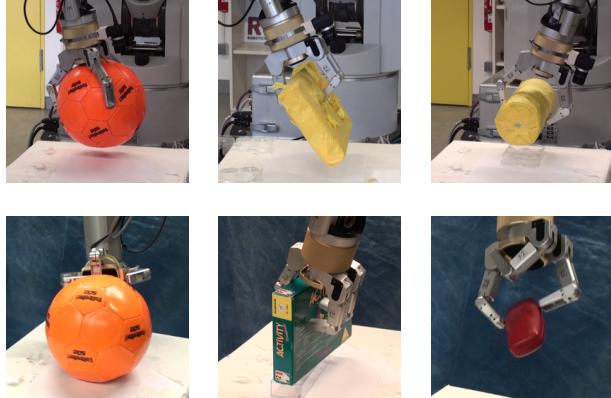
Figure 3.3: Shape resemblance determined by the visual primitives: box (red circles), spherical (blue), and cylindrical (yellow); the diameter of the spheres indicates the success of the corresponding compliant grasp.

is most appropriate, showing that the rays of the sun are separate, shape- and mode-specific regions of high grasp success probability. In the case of the soccer ball, only the spherical compliant grasp is successful at all.

The results show that visual resemblance is a good measure for selecting compliance modes and a good predictor of grasp success. The results also show that our spherical visual primitive is too selective, as it results in a visual resemblance value of less than 0.5 for a perfect sphere.

3.1.3 COMPARISON OF COMPLIANCE-CENTRIC AND COMPLIANCE-AGNOSTIC GRASPING

In our final set of experiments, we compare the proposed compliance-centric grasp algorithm to a specific compliance-agnostic grasp planner. We show that our algorithm outperforms the compliance-agnostic planner, even after we provide the latter with accurate a priori object models, which could be considered an important advantage over our method.



Video Figure 3.1: The images show example grasps of our compliance-centric algorithm. [https://youtu.be/Pm1A5AMBH_0]

Video Figure 3.2: The images show example grasps of the Eigengrasp planner. [<https://youtu.be/a9IIB87wZ9w>]

EXPERIMENTAL SETUP

For the experimental comparison we chose the Eigengrasp planner (Ciocarlie and Allen, 2009), implemented inside the *GraspIt!* framework (Miller and Allen, 2004). Our reason for choosing this planner were its relative recency, its popularity in terms of citations, and the availability of the source code.

We took great care in creating geometric models of six of the objects shown in Fig. 3.2 (all except the fruits and vegetable), performing several independent measurements for each. We included object-specific, conservative estimates of surface friction in the models.

To generate candidate grasps with the Eigengrasp planner, we used 70 000 iterations of simulated annealing, setting the energy formulation as “hand+object”. We eliminated candidate grasps for which no inverse kinematic solution for the arm existed (this only happened for two of the six objects and did not affect the two highest-quality grasps). We then selected the three best grasps, performing ten trials for each, for a total of 180 grasp attempts. The exact pose of the object was an input to the grasp attempt. Grasp success was measured as before.

For comparison, we calculated two measures of success for the Eigengrasp planner. The *average* success rate includes all 30 grasps per object, while the *best* success rate only represents the most successful of the three planned grasps. Note that the second metric selects the best grasp in hindsight, after all experiments are performed.

RESULTS AND DISCUSSION

Fig. 3.4 compares the grasp success obtained with the Eigengrasp planner to our compliance-centric algorithm. Among the objects inside the annulus, only the spectacle case shows sig-

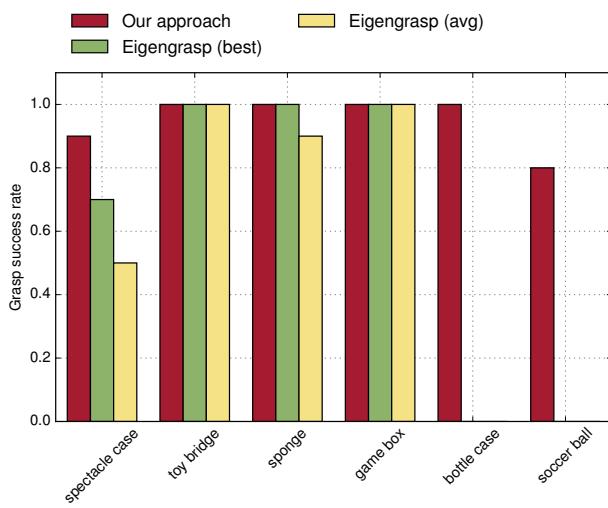


Figure 3.4: We compare our presented compliance-centric grasp planning algorithm with the Eigengrasp planner ([Ciocarlie and Allen, 2009](#)).

nificant difference in grasp success. Our algorithm outperforms the Eigengrasp planner in either metric. In the rays of the sun (bottle case and soccer ball), the Eigengrasp planner fails in all grasp attempts. This could be an indication that the advantage of compliance-centric grasping increase as we reach the boundary of the SFA. In the case of the box, however, both grasp methods achieve 100 % success.

Overall, our compliance-centric grasp algorithm always performs better or as good as the Eigengrasp planner. The grasp results demonstrate that the proposed visual primitives successfully identify configurations in the entrance to the bottom part of the funnel. These visual primitives are simple and effective. They eliminate the need for explicit contact planning, supporting the claim that hand compliance should become a central consideration in robot grasping.

3.1.4 LIMITATIONS

There are two main shortcomings of the presented algorithm:

- **MOTION REQUIREMENT:** It is not practical to move the camera around an object prior to each grasp. However, we view this algorithm as a proof of concept. In the future, the need for active camera motion can be significantly reduced in most cases by first deriving a shape hypothesis from 2D vision and then confirming this hypothesis through minimal camera motion. If the 2D hypothesis cannot be confirmed, the robot uses the procedure above to recover.

- **SEGMENTATION EFFORT:** We used a very simple color-based blob detection to segment the object from the background. This limits the application to uniformly colored objects. Although graph-based methods ([Felzenszwalb and Huttenlocher, 2004](#)) could solve more general scenarios, they will inevitably create more potential segments which need to be verified. This ties back to the motion requirement which will apply to every segment (given that some feature values require the camera to center the segment). The only alternative is to use a richer user input that specifies the appearance of the object to grasp.

The algorithm presented next will eliminate both of these shortcomings. It does not require excessive motions by using a depth sensor. It also generates multiple grasp hypothesis based on different segmentation algorithms.

3.2 FEATURES BASED ON 3D SHAPE FITTING

In this section, we present shape-adaptation-aware grasping strategies for unknown objects. These strategies exploit adaptation between hand and object and allows us to simplify perception. The shape adaptability of the hand adjusts to variations in object shape. Consequently, perception only needs to determine object shape to the level of detail not compensated by shape adaptation. The loss in geometric accuracy of the acquired object model is compensated by the shape adaptability of the hand.

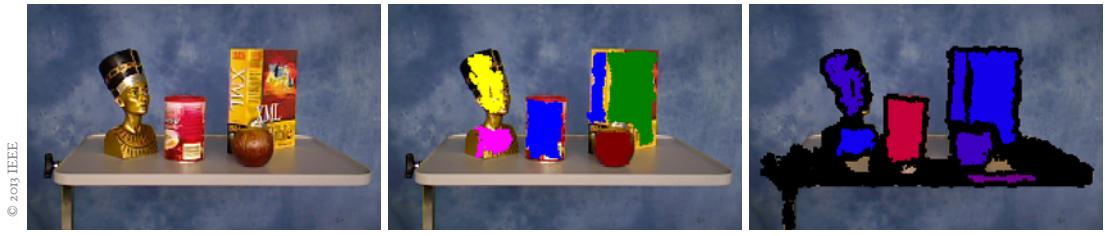
We introduce four different shape descriptors: a spherical, cylindrical, box, and disk one. Each one is supposed to recognize a different subset of all possible object shapes. We now explain how the four descriptors work, based on depth image measurements taken by any time-of-flight, stereo or structured-light sensor.

3.2.1 CALCULATING 3D SHAPE FEATURES

We avoid the highly non-convex parameter space that occurs when fitting geometric models to raw sensor data by first segmenting it. In general there is no single-best segmentation method for the variety of scenes we will face. Thus, we apply multiple segmentation methods ([Malisiewicz and Efros, 2007](#)). A flood fill algorithm similar to [Holz and Behnke \(2012\)](#) segments the depth image into coherent regions. This segmentation groups neighboring points o and p according to a boolean predicate, which we define as follows:

$$\begin{aligned} |o_{\text{depth}} - p_{\text{depth}}| &< t_{\text{depth}} \quad \wedge \\ \langle o_{\text{normal}}, p_{\text{normal}} \rangle &< t_{\text{angle}} \quad \wedge \\ o_{\text{curvature}} &< t_{\text{curvature}}, \end{aligned}$$

where o_{depth} is the depth of point o , o_{normal} its surface normal, $o_{\text{curvature}}$ its mean curvature, and $(t_{\text{depth}}, t_{\text{angle}}, t_{\text{curvature}})$ a set of thresholds. Small-sized segments are filtered out. We use segmentations with low $t_{\text{curvature}}$ and t_{angle} which favor edge boundaries and high values of t_{depth} that result in larger regions even in the presence of noise. The resulting segment soup builds the basis for our different shape descriptors. They all fit geometric models to the segments using the method of Random Sample Consensus (RANSAC by [Fischler and Bolles \(1981\)](#)). A large threshold in the inlier criterion allows for considerable shape variation which we assume can be compensated by the hand's closing motion. In the following we describe the goodness of fit for each geometric descriptor.



Video Figure 3.3: To improve clarity the depth values are plotted on top of the camera image, although our method only uses range data. The center image shows the result of one flood fill segmentation that separates the geometry at depth discontinuities and sharp edges. The segments are then described by different shape descriptors to match the hand geometry. The right image depicts the confidence of the cylindrical shape descriptor ranging from red (very cylindrical) to blue (hardly cylindrical). [<https://youtu.be/XboHUM6Y5Ps>]

SPHERICAL SHAPE DESCRIPTOR: A sphere is fit with a radius bounded to the range graspable by the Barrett Hand's spherical pre-shape. The goodness of fit is based on the ratio of inliers and segment size in combination with the visibility of the hypothesized sphere. The visibility criterion is the ratio of the segment size and the expected size of the sphere backprojected into the sensor frame:

$$\text{conf}_{\text{spherical}} = \frac{\text{inliers}}{\text{size(segment)}} \cdot \frac{\text{size(segment)}}{\text{size(expected projection)}}.$$

CYLINDRICAL SHAPE DESCRIPTOR: An infinitely tall cylinder is fitted which is bounded by the extreme inliers along the cylinder's axis. The height and radius are again constrained by the hand geometry. A goodness of fit value is given by the ratio of inliers and segment size in combination with the expected visibility analog to the spherical shape descriptor. Video Fig. 3.3 shows an example application of the cylindrical shape descriptor.

BOX SHAPE DESCRIPTOR: A plane fit is bounded by projecting its inliers orthogonal onto the plane and calculating the 2D minimum enclosing rectangle. If the rectangle size exceeds the graspable volume of the cylindrical pre-shape it is discarded. The goodness of fit is a combination of the inlier ratio and the rectangularity of the contour of the projected points. This rectangularity is defined as the ratio of the area of the convex hull and the fitted rectangle. The final confidence value is a weighted sum of both terms:

$$\text{conf}_{\text{box}} = w \cdot \frac{\text{inliers}}{\text{size(segment)}} + (1 - w) \cdot \frac{\text{area(convexHull(projection))}}{\text{area(minRect(projection))}}.$$



Figure 3.5: We used 21 different test objects to empirically evaluate our grasp planning algorithm.

DISK SHAPE DESCRIPTOR The disk descriptor works similar to the box descriptor with the exception that it uses the minimum enclosing circle of the projected points to define the goodness of fit instead of a rectangle:

$$\text{conf}_{\text{disk}} = w \cdot \frac{\text{inliers}}{\text{size}(\text{segment})} + (1 - w) \cdot \frac{\text{area}(\text{convexHull}(\text{projection}))}{\text{area}(\text{minEnclosingCircle}(\text{projection}))}.$$

3.2.2 EXPERIMENTAL EVALUATION

The goal of our experiments is to examine how well our perceptual grasping strategies can predict their success, based on the match between object and hand.

EXPERIMENTAL SETUP

We equipped a 6-DOF Unimation PUMA 560 with a Barrett Hand BH8-262 and an Asus Xtion Live depth sensor. The sensor was mounted on the wrist as can be seen in Fig. 3.1. During each grasping trial the robot was observing an object for 3 s from a single view point. During that time it chose the most promising pre-grasp strategy. If no pre-grasp confidence exceeded a pre-defined threshold, no grasp was executed and the next view point was considered. Otherwise, a force-based operational space control law was executed to approach the planned pre-grasp pose from an intermediate pose located 10 cm in the negative direction of the approach vector. After reaching the pre-grasp pose, the hand was pre-shaped and a closing motion executed. We considered a grasp to be successful if the robot could lift the

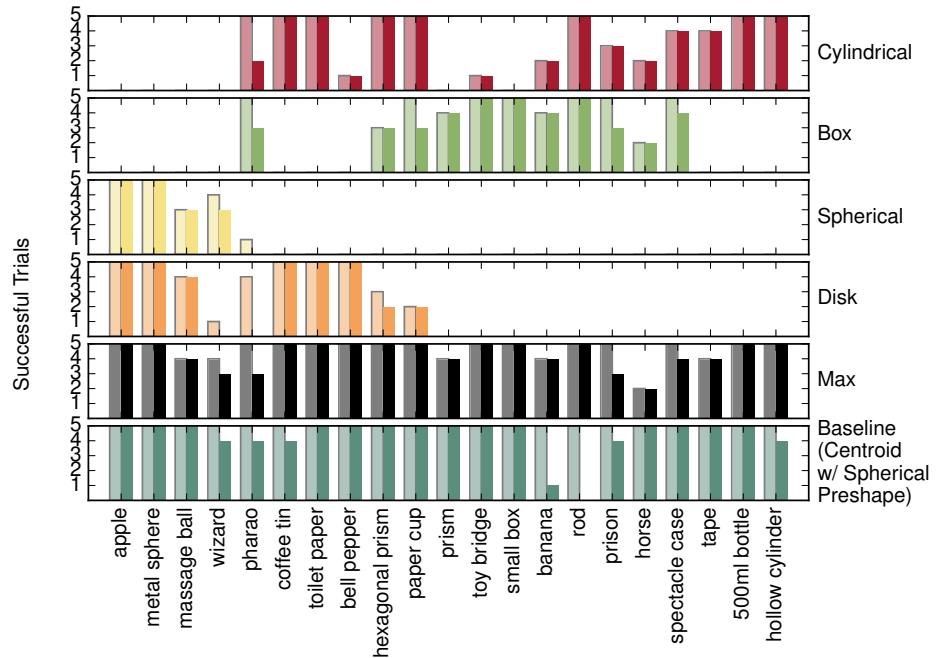


Figure 3.6: Grasping performance for each of the four compliance modes and a strategy which approached the objects towards the perceived centroid from the current view point with a spherical pre-shaped hand. The light-colored bar indicates a predicted grasp by the algorithm. The dark-colored one is the resulting grasp success.

object 30 cm and the object did not fall out of the hand within 10 s.

To measure the predictability of hand/object match of our pre-grasp strategies, we conducted a simplified experiment that excluded any environmental effects. The 21 objects shown in Fig. 3.5 were placed onto a sticky tripod. This resembled a quasi-static scenario free of any effects induced by interactions between hand and environment. Each object was placed five times in different orientations in front of the robot.

Additionally, we executed a baseline strategy that approached the object's centroid along the ray originating from the camera's view point. It always used the spherical pre-shape, ignoring any object shape information (apart from calculating the centroid of the point cloud segment).

RESULTS AND DISCUSSION

For each strategy we measured the rate of pre-grasp detection and grasp success as defined above. The results shown in Fig. 3.6 largely confirm our intuition: Whenever a promising

grasp was predicted by one of the descriptors, the likelihood that the corresponding grasp also succeeded was high. The worst mismatch between grasp detection and grasp success occurred with the pharao and the disk pre-shape. The top of the pharao sculpture is indeed disk shaped. But due to the low surface friction the heavy object (≈ 800 g) slipped out of the hand.

Among all pre-shapes the cylindrical one was the one most often applied. This matches the statistics collected by [Zheng et al. \(2011\)](#). Among the 1280 grasps done by a house maid they categorized 29% as being cylindrical. Note that although there are partial overlaps between strategies no strategy is dominated by another one and thereby obsolete.

All the analyzed objects were grasped by at least one of the strategies, showing that together they cover a significant amount of naturally occurring object shapes. This can be observed in the graph showing the *max strategy*, which selects the most promising of the four pre-shapes given their confidence values. Precision and recall of the max strategy are high with 94% and 92% respectively.

The results shown in the last row of Fig. 3.6 indicate that the simple baseline strategy already exploits enough information to grasp successfully, with a slightly lower success rate of 87% (vs. 94% of the max strategy). But notice that there are a lot of cases in which such a simple strategy (grasping the centroid) will eventually fail. We will look at more realistic scenarios that include the environment in Part III.

3.3 RELATED WORK

We presented two grasp planning algorithms which do not rely on a priori object knowledge. Instead they leverage the adaptability and compliance of the hand to match the shape of an object. We focus on related approaches to robotic grasping that are similar because they either exploit hand compliance (implicitly or explicitly) or they tightly integrate perception and action.

3.3.1 LEVERAGING ADAPTABILITY DURING SHAPE MATCH BETWEEN HAND AND OBJECT

Both presented grasp algorithms factorize the grasping motion into a pre-shaping of the hand followed by a closing motion. This is an application of Principle I: “Exploiting Adaptability in Grasp Representations” (Sec. 1.3.1). Similarly, Ciocarlie and Allen (2009) synthesize grasps based on the idea that the intrinsic DOF of a hand can be mapped into a lower dimensional sub-space, without losing much expressiveness. This way they can search the space of possible pre-grasps much faster. In executing grasps, they implicitly rely on compliance when closing the fingers. We compared our algorithm with this approach and showed that we find much more reliable grasps by taking the global shape properties into account. Miller et al. (2003) present a planner that uses heuristics which describe how to grasp basic shapes, such as boxes, cones, spheres and cylinders. Their pre-grasps and approach strategies resemble our grasp strategies. However, they rely on a known decomposition of the object, ignoring this non-trivial perception problem. Balasubramanian et al. (2010) showed that humans prefer orthogonal approach directions when controlling robotic hardware to produce robust grasps. Our algorithms also create orthogonal approach directions w.r.t. the estimated principal axis of the objects.

Both presented features simplify the shape of unknown objects: the active vision features describe shape as an expected visual change in response to observer motion while the 3D features compress any geometry to a set of a few canonical shapes. This kind of exploitation of hand compliance is an application of Principle II: “Exploiting Adaptability in Object Representations” (Sec. 1.3.2). Similar approaches in grasping are: Huebner and Kragic (2008) who use boxes to approximate shapes, Przybylski et al. (2010) who propose inscribing balls, Goldfeder et al. (2007) who decompose shapes into superquadrics, and Nieuwenhuisen et al. (2012) who recognize CAD models. But a clear distinction to us is that all of these approaches require complete object models. The problem of perception, which is an integral part of the

algorithms in this chapter, is ignored.

3.3.2 TIGHT INTEGRATION OF PERCEPTION AND GRASP PLANNING

This category of related approaches exploits aspects of shape adaptability between hand and object and also addresses the perception problem. However, each of these methods has limitations overcome by our approach. In one of the earliest approaches that exploit shape matching, the authors map bounding ellipses extracted from an image to three different hand pre-shapes of a two-fingered gripper ([Bard and Troccaz, 1990](#)). This method is restricted to top-down grasping. [Kootstra et al. \(2012\)](#) extract contour and surface features from visual input. The features are mapped to grasping actions: enveloping grasps for surfaces and pinch grasps for contours. A method proposed by [Klingbeil et al. \(2011\)](#) searches for protrusions in range scans as candidate locations for grasping with a parallel-jaw gripper, but not with multi-fingered hands. [Herzog et al. \(2012\)](#) learns graspable 3-D features in the environment from human demonstration. They side-step the perception problem with human input. [Maldonado et al. \(2010\)](#) assume that all objects are placed on top of a table. Each point cluster above the table surface is interpreted as an object. The pre-grasp pose is optimized to bring the center of the palm as close to the object while maximizing the distance between object and fingers.

Similar to our active vision features, [Calli et al. \(2011\)](#) use an eye-in-hand system and apply a visual servoing scheme that maximizes the curvature of the object silhouette, thus leading the hand to concave parts of the object. [Dune et al. \(2008\)](#) also use active vision to acquire information about object shape but then rely on shape approximation with quadrics to represent this information. Again, by relying on a shape approximation, the approach implicitly depends on hand compliance.

3.4 CONCLUSION

In this chapter, we presented two shape-adaptation-aware approaches that grasp unknown objects. There is much evidence in the human and robot grasping literature that shape adaptation significantly increases grasp success. We therefore explicitly account for the effects of shape adaptation in the design of grasping algorithms. The proposed methods consider shape adaptation between the hand and the grasped object to simplify perception. Rather than attempting to perceive the exact shape of the object to perform grasp planning, we assume that the shape must only be known to the level of detail necessary to decide which pre-grasp is most appropriate. The pre-grasp then invokes a particular mode of shape adaptation of the hand by closing the fingers, compensating for any infidelities in the perceived object model. Our experiments demonstrate that the explicit consideration of shape adaptability reduces the perceptual requirements of grasping and enables robust grasp performance without explicit planning of contact points.

PART II



Interactions Between
Hand and Environment

MOTIVATION

Robot motion planning ([LaValle, 2006](#)) tackles the problem of finding a collision-free path from A to B. This is closely related to the grasping problem, in which we also try to find a collision-free motion from a starting configuration to the configuration that establishes contact with the object. In both cases *collision-free* is referring to the obstacles in the environment. Intuitively this is a reasonable problem formulation, a colliding robot will most likely not achieve its goals.

But as soon as we consider uncertainty due to sensors, models, and actuators a different view emerges. Creating collisions and contact-exploiting motions such as sliding along a surface can be used to reduce uncertainty about the spatial state of a system. And indeed, it has been shown that this also happens in human behavior. [Deimel et al. \(2013\)](#) showed that visually impaired subjects contact the environment more intensively when grasping objects compared to their unimpaired counterparts. This brings us to the second core idea of this thesis: The environment is your friend, embrace and use it whenever possible!

CONTRIBUTIONS

The main contributions of the second part are the following:

- A planning method that finds motions from A to B. It assumes a noisy motion model, a geometric model of the environment and access to an uncertainty-free contact signal. As a result it finds motions that combine classical free-space paths with segments of uncertainty-reducing contact actions (Algorithm 4.1).
- A characterization of different grasping strategies that explicitly exploit the environment. To do this we introduce the concept of environmental constraints (ECs).
- A planning method that finds grasping motions based on sequences of contact exploitations (Algorithm 5.4). The method does not assume an a priori model of the environment but builds a representation from RGB-D data.

OUTLINE

This part presents two planning methods which exploit contact with the environment. Chapter 5 deals with motion planning problems. This leads to motions that explicitly exploit contact whenever position uncertainty of the end-effector is too high. We evaluate this algorithm

in different motion planning problems under uncertainty but do not have a particular focus on grasping.

In contrast, Chapter 6 introduces a planning algorithm with a focus on grasping actions. Here, we do not model position uncertainty as before but rather assume that contact with the environment is always beneficial for grasp success. As a result, grasping motions are generated that successively exploit different environmental features. Finally, we compare the two algorithms on a theoretical and practical level.

It is important to note that the main focus in this part is on the relationship between hand and environment; the effects of intrinsic object properties are ignored and will be resolved in Part III.

4

Planning Motions That Exploit Contact With the Environment

ROBOT MOTIONS ALWAYS SUFFER FROM UNCERTAINTIES. Uncertainties originate from various sources: sensor measurements are noisy, the models which are used to interpret these signals or plan ahead are uncertain and also actuators are prone to noise. As an example of the serious problems caused by uncertainty consider the Barrett WAM, which is used frequently in this thesis. When estimating the pose of its end-effector based on the motor encoder readings the error can be up to 8 cm (Krainin et al., 2011). This is enough to grasp thin air instead of the intended object. The error is due to the non-rigid cables that connect the motor shaft with the link. Modeling the actual length of such a cable is close to impossible. It depends on a variety of variables such as the temperature, payload, robot configuration, etc.

We propose to exploit contact to tackle the problem of uncertainty. Assuming that contact is a discrete event that can be reliably sensed, we argue for purposeful collisions with the environment in order to gain information, reaching motion targets more robustly. The advantages of exploiting motion in contact is not new. For example, strategies that interleave motion in contact and in free space have been the key to success in the DARPA ARM challenge 2011, where robots deliberately bumped into a door before pushing down the handle (see Fig. 4.1). But so far there has been little research on how to plan such strategies from a description of the scene geometry.

In this chapter we present a planner called *Contact-Exploiting RRT* (CERRT), based on

4.1 AN ALGORITHM FOR INTERLEAVING MOTION IN CONTACT AND IN FREE SPACE: CONTACT-EXPLOITING RRT (CERRT)



Figure 4.1: Contact can efficiently reduce the uncertainty about the robot’s state. **Left:** An example from a manipulation challenge, where the robot first touches the door to localize the handle before attempting a grasp (Righetti et al., 2014). **Right:** A similar strategy, generated by our planner CERRT. Shown in green are several executions of the motion under uncertainty, which all first contact the door and then slide down to the handle.

the rapidly-exploring random tree (RRT) by LaValle (1998). Our planner finds robust motion plans under uncertainty in robot position, actuation, and world model. The planner scales to high-dimensional configuration spaces. The resulting motions make and break contact with the environment, slide along surfaces, but also avoid collisions with links that have no contact sensing capability.

The main difficulty of planning under a partially observable robot state is the high dimensionality of the associated belief space. Our planner overcomes this problem by exploiting the insight that sensing contact is reliable and can be assumed to be fully observable. This factors the problem into a tractable reasoning over the robot’s position.

We evaluate the planner’s capability to reason efficiently about high uncertainty on a benchmark manipulation planning problem from the literature of partially observable Markov decision processes (POMDPs). We show how the planner generalizes to more complex problems by increasing the complexity and the dimensionality of the configuration space. We will validate our planning results with simulation and real world experiments for a motion task under significant uncertainty.

4.1 AN ALGORITHM FOR INTERLEAVING MOTION IN CONTACT AND IN FREE SPACE: CONTACT-EXPLOITING RRT (CERRT)

CERRT plans with a combined state of belief over configuration and fully-observable contact $x = (\mathcal{Q}, \mathcal{C})$. We represent the belief over the configuration with a set of particles $\mathcal{Q} =$

$\{q_1, \dots, q_N\}$, where each element q is an n -dimensional robot configuration. We will denote the sample mean and variance of \mathcal{Q} with μ_x and Σ_x . Each belief state is also associated with a fully observable set of contacts $\mathcal{C} = \{c_1, \dots, c_m\}$. Each contact c is a pair of surfaces in contact $(s_{\text{robot}}, s_{\text{env}})$. s_{robot} is a surface on the robot that has contact sensing capabilities and s_{env} a surface of the environment. The most important quantities are summarized in Table 4.1.

Symbol	Meaning
$G = (V, E)$	the search tree with nodes V and edges E
q	a configuration
x	a state
\mathcal{Q}	a set of configurations (particles)
$\mu_{\mathcal{Q}}$	the sample mean of \mathcal{Q}
$\Sigma_{\mathcal{Q}}$	the sample variance of \mathcal{Q}
σ_{start}	the variance of the initial error
$\delta(\dot{q})$	the motion error
ϵ	the resolution of the planner
γ	the contact exploration bias
$T = (p, R)$	a frame pose described by homogeneous transform or position and orientation
ξ	a velocity twist

Table 4.1: Symbols used in this chapter

The planner finds strategies that combine free-space and contact motion. We assume that free-space motion always increases uncertainty. This is represented by a noisy motion model $\delta(\dot{q})$ which is detailed in Sec. 4.1.3. Because free space motions increase uncertainty, the planner must sequence them with contact motions that reduce uncertainty. Fig. 4.2 shows an example of a decision the planner must take. The robot cannot directly enter the narrow passage but must first contact the wall to reduce uncertainty.

To find such strategies, we grow a tree in the combined space of contact state and belief over configuration. The key to the planners efficiency is a tailored exploration strategy of this space. To adjust the search behavior of the planner, we introduce the parameter $\gamma \in [0, 1]$ that describes the rate with which the planner attempts free-space or contact moves. If $\gamma = 0$, the planner only explores free space, and behaves like a vanilla RRT-Connect ([Kuffner and](#)

4.1 AN ALGORITHM FOR INTERLEAVING MOTION IN CONTACT AND IN FREE SPACE: CONTACT-EXPLOITING RRT (CERRT)

© 2017 IEEE

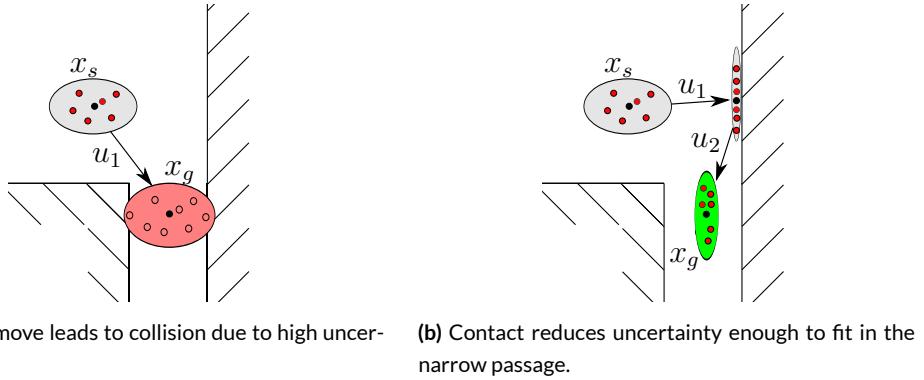


Figure 4.2: Left: To enter the narrow passage, the robot cannot directly take action u_1 because the resulting uncertainty would lead to collision. Right: By sequencing a contact move u_1 and a free space move u_2 , the robot reduces position uncertainty sufficiently to enter the narrow passage. CERRT finds such sequences of contact and free-space motions.

© 2017 IEEE

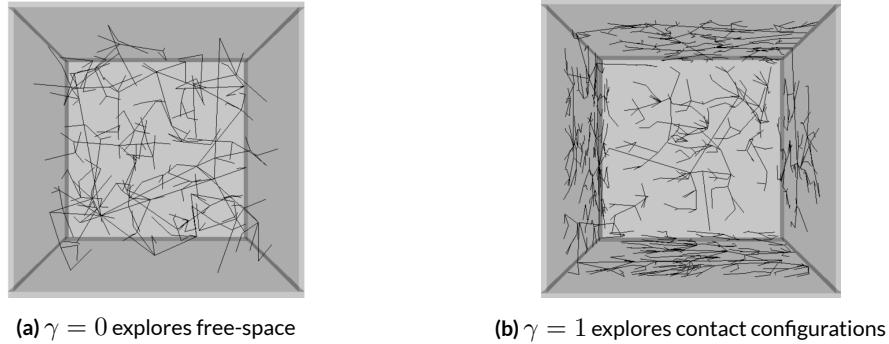


Figure 4.3: The search behavior of CERRT is governed by a free-space/contact exploration bias γ . We show two search trees of the CERRT planner exploring the inside of a cube for different values of γ . For $\gamma = 0$, the behavior matches that of a standard RRT (a). For $\gamma = 1$, the planner searches the space of configuration in contact with the walls of the cube (b). The CERRT planner interleaves both behaviors.

LaValle, 2000) with goal bias. If $\gamma = 1$, the planner's only objective is to reduce uncertainty. Thus it will favor moves that get the robot into contact. Values between 0 and 1 balance both objectives. Fig. 4.3 shows how the tree grows in a hollow cube based on different values of γ .

Our Contact-Exploiting RRT (CERRT) is closely related to the kinodynamic RRT developed by LaValle (1998). Its structure (Alg. 4.1) is identical to the RRT. However, CERRT differs substantially in the implementation of the subroutines which we will explain in detail in the rest of this section, following the order of the pseudocode in Algorithm 4.1.

Algorithm 4.1 CERRT

Input: $\mathbf{x}_{\text{start}}, \mathbf{x}_{\text{goal}}, \epsilon_{\text{goal}}, \gamma$
 Output: $G = (V, E)$

```

 $V \leftarrow \{\mathbf{x}_{\text{start}}\}$  ▷ initialize tree with start state
 $E \leftarrow \emptyset$ 
while true do ▷ search until goal reached
   $\mathbf{q}_{\text{rand}} \leftarrow \text{RANDOM\_CONFIG}()$  ▷ Sec. 4.I.1
   $\mathbf{x}_{\text{near}} \leftarrow \text{NEAREST\_NEIGHBOUR}(\mathbf{q}_{\text{rand}}, T, \gamma)$  ▷ Sec. 4.I.2
   $\mathbf{u} \leftarrow \text{SELECT\_INPUT}(\mathbf{q}_{\text{rand}}, \mathbf{x}_{\text{near}}, \gamma)$  ▷ Sec. 4.I.3
   $\mathbf{x}_{\text{new}} \leftarrow \text{NEW\_STATE}(\mathbf{x}_{\text{near}}, \mathbf{u}, \mathbf{q}_{\text{rand}})$  ▷ Sec. 4.I.4
  if IS_VALID( $\mathbf{x}_{\text{new}}$ ) then
     $V \leftarrow V \cup \{\mathbf{x}_{\text{new}}\}$ 
     $E \leftarrow E \cup \{(\mathbf{x}_{\text{near}}, \mathbf{x}_{\text{new}})\}$ 
     $\mathbf{x}_{\text{connect}} \leftarrow \text{NEW\_STATE}(\mathbf{x}_{\text{new}}, \text{connect}, \mu_{\mathbf{x}_{\text{goal}}})$ 
    if  $\|\mathbf{x}_{\text{connect}} - \mathbf{x}_{\text{goal}}\| < \epsilon_{\text{goal}}$  then
      return  $G$ 

```

4.I.1 NODE SELECTION: NEAREST_NEIGHBOUR

Like the RRT, our planner selects the next node to extend x_{near} with minimal distance to a randomly sampled configuration q_{rand} . Because the node is a belief state, we need to define a suitable metric for states x . The choice of metric strongly influences the planning performance (Littlefield et al., 2015). For CERRT, we use a metric that takes the parameter γ into account and can balance the search towards free space or contact motion.

For $\gamma = 0$ we want the tree to expand into free-space quickly, just like the RRT. We achieve this by choosing the node \mathbf{x}_n whose mean is closest to q_{rand} . To do so we compute the Euclidean distance $d_\mu(x_n) := \|\mu_{x_n} - q_{\text{rand}}\|$.

For $\gamma = 1$ we want to reduce uncertainty by exploring contact space. We achieve this by picking a node with low uncertainty. More specifically, we compute a norm of Σ_{p_n} , which is the covariance matrix of the robot's end-effector position p_n at configuration q_n . We then compute the trace norm, leading to: $d_\Sigma(x_n) := \sqrt{\text{tr}(\Sigma_{p_n})}$. We use this norm mainly because it does not become 0 if the distribution loses support in one dimension (which happens in contact), and also because it is inexpensive to compute.

For $0 < \gamma < 1$, we balance the two aforementioned metrics with a convex combination:

$$\mathbf{x}_{\text{near}} = \underset{\mathbf{x}_n}{\operatorname{argmin}} \left(\gamma \hat{d}_\Sigma(x_n) + (1 - \gamma) \hat{d}_\mu(x_n) \right).$$

Both distance terms are normalized to the interval $[0, 1]$ by dividing them by the maximum

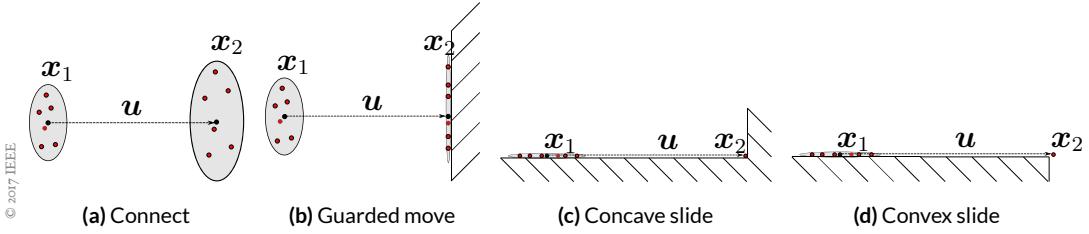


Figure 4.4: A free-space move (a), a move into contact (b), and two sliding actions (c),(d). x_1 and x_2 are the initial and final particle distributions before and after applying action u .

observed value over all samples.

4.1.2 ACTION SELECTION: SELECT_INPUT

After choosing a node for extension the planner needs to pick the next action. CERRT must have enough options to move in free space, along contact surfaces, or to switch from free space to contact and vice versa. We implement these options with three different action types. We will briefly introduce them here and give their implementation details later in Sec. 4.1.3.

- **CONNECT:** This action attempts to directly connect the sample q_{rand} via a straight line in configuration space. It explores the free space and usually increases position uncertainty (Fig. 4.4a).
- **GUARDED:** This action moves in the direction of q_{rand} until it establishes contact with the environment. The guarded move is required to switch from free space to contact and always reduces uncertainty in one dimension (Fig. 4.4b).
- **SLIDE:** This action slides along a surface until the contact state changes, either by moving into another contact (Fig. 4.4c) or by leaving the sliding surface (Fig. 4.4d). It explores the space of all contacts, always keeps uncertainty low in one dimension, and can reduce uncertainty in a second dimension.

Our planner selects one of the three actions randomly, biased by γ in the following way: if x_{near} is not in contact, it performs a connect move or a guarded move. If x_{near} is in contact, it slides or leaves the contact with a connect move. We choose actions based on these

Algorithm 4.2 CERRT – NEW_STATE

Input: $x_1, u, \mathbf{q}_{\text{rand}}$
 Output: x_2

```

for  $i \in N_{\text{particle}}$  do
     $\mathbf{q}_{\text{near}} \leftarrow \text{SAMPLE}(\mathcal{Q}_{x_1})$                                  $\triangleright$  sample particle from node
     $\boldsymbol{\alpha} \leftarrow \text{SAMPLE}(\mathcal{N}(0, \sigma_{\delta}))$                        $\triangleright$  sample motion error
     $\mathbf{q}_{\text{target}} \leftarrow \mathbf{q}_{\text{rand}} + (\mathbf{q}_{\text{near}} - \mu_{x_1})$            $\triangleright$  add the initial error
     $\mathbf{q}_{\text{sample}} \leftarrow \text{LOCAL\_PLANNER}(a, \mathbf{q}_{\text{near}}, \mathbf{q}_{\text{target}}, \delta_{\alpha})$        $\triangleright$  simulate action with one of
                                                the local planners (Sec. 4.1.3)
     $\mathcal{Q}_{x_2} \leftarrow \mathcal{Q}_{x_2} \cup \{\mathbf{q}_{\text{sample}}\}$ 
return  $x_2$ 
  
```

distributions:

$$\begin{aligned}
 p(\text{connect} | \mathcal{C}_x = \emptyset) &= 1 - \gamma \\
 p(\text{guarded} | \mathcal{C}_x = \emptyset) &= \gamma \\
 p(\text{connect} | \mathcal{C}_x \neq \emptyset) &= 1 - \gamma \\
 p(\text{slide} | \mathcal{C}_x \neq \emptyset) &= \gamma
 \end{aligned}$$

We chose these distributions so that the planner is an RRT-Connect for $\gamma = 0$.

4.1.3 FORWARD SIMULATION: NEW_STATE

For each of these actions, the planner must be able to reason about the change of uncertainty. We approximate this with a simulation of N noisy actions. The input to the simulation is a motion model $\delta_{\alpha}(\dot{q})$ with parameter vector $\boldsymbol{\alpha}$. Examples for the motion model δ are the classical angular and translational motion error for mobile robots or independent error for all joints of the robot.

To extend a node x_1 , CERRT samples a particle from \mathcal{Q}_{x_1} and also samples a vector $\boldsymbol{\alpha}$ of parameters of the motion model δ . The extension step then is an invocation of the local planner that executes action u with the motion error δ_{α} , which we will describe in detail in the next section. The target of the local planner is \mathbf{q}_{rand} with the initial error of the particle added. The extension step is repeated for all particles so that the outcome of the simulation is a new set of particles \mathcal{Q}_{x_2} which is added to the new state x_2 .

Each of the three action types invokes a different local planner. We implement them in the following way:

- **CONNECT:** A connect move is identical to the RRT version. A connect-particle moves on a straight line in configuration space towards the sample q_{rand} , checking for collisions with a fixed resolution of ϵ . If the particle reaches the sample or moves into contact the motion ends.
- **GUARDED:** A guarded motion is a connect move in the direction of q_{rand} . A guarded move always ends in contact so it might end before q_{rand} or move beyond q_{rand} .
- **SLIDE:** Sliding motions start with particles in contact and move them along the surface, always maintaining contact. We implement sliding motions as task-space force-feedback controllers with constant orientation. To simulate sliding actions we first choose a random sliding surface (because the node might be in contact with two surfaces at the same time) and then project the end-effector position of the robot in configuration q_{target} onto the sliding surface. The algorithm then alternates between 1) taking a step towards the projected goal, 2) applying the motion error for this step 3) projecting the configuration back on the surface (see Algorithm 4.3). In this way, the effect of the joint-space motion error can be projected onto the lower-dimensional manifold of configurations in contact with the environment. The slide ends if the robot reaches the projected goal, if there is another contact, or if the robot loses contact with the sliding surface (see Fig. 4.4). For all projections we use a damped pseudo-inverse. If the robot is close to a singularity at any step ($\sqrt{\det(JJ^T)} < 0.001$ (Yoshikawa, 1985)) the slide method returns failure.

4.1.4 NODE VALIDATION: IS_VALID

All nodes in CERRT must have a uniquely defined contact state. To ensure this, we only add those simulation outcomes to the tree that fulfill two requirements:

1. All $q \in \mathcal{Q}_{x_2}$ must either end up in free space or in contact with the same pair of surfaces.
2. If \mathcal{Q}_{x_2} contains configurations in contact, the contact must occur with a link that has a contact-sensor.

Algorithm 4.3 CERRT – SLIDE

Input: $\mathbf{q}_{\text{near}}, \mathbf{q}_{\text{sample}}$
 Output: \mathbf{q}_{real}

$$\begin{aligned}
 (\mathbf{p}_{\text{sample}}, R_{\text{sample}}) &\leftarrow T_{EE}(\mathbf{q}_{\text{sample}}) && \\
 (\mathbf{p}_{\text{surf}}, \mathbf{n}_{\text{surf}}) &\leftarrow \text{RANDCONTACT}(\mathbf{q}_{\text{near}}) && \\
 \mathbf{p}'_{\text{sample}} &\leftarrow \mathbf{p}_{\text{sample}} - ((\mathbf{p}_{\text{sample}} - \mathbf{p}_{\text{surf}}) \cdot \mathbf{n}_{\text{surf}}) \mathbf{n}_{\text{surf}} && \triangleright \text{sample random contact point} \\
 \boldsymbol{\xi} &\leftarrow \mathbf{T}_{\text{near}} - \mathbf{T}_{\text{sample}} && \triangleright \text{and surface normal of } \mathbf{q}_{\text{near}} \\
 \text{while } \|T_{EE}(\mathbf{q}_{\text{robot}}) - \mathbf{T}_{\text{sample}}\| > 0 \text{ do} && \triangleright \text{project } \mathbf{p}_{\text{sample}} \text{ on surface} \\
 \Delta\mathbf{q} &\leftarrow \mathbf{J}^\dagger(\mathbf{q}_{\text{robot}}) \boldsymbol{\xi} && \\
 \mathbf{q}_{\text{robot}} &\leftarrow \mathbf{q}_{\text{robot}} + \epsilon \cdot \Delta\hat{\mathbf{q}} && \triangleright \text{move along surface towards sample} \\
 \mathbf{q}_{\text{real}} &\leftarrow \mathbf{q}_{\text{robot}} + \delta(\epsilon \cdot \Delta\hat{\mathbf{q}}) && \triangleright \text{the particles most likely position} \\
 \text{while } \mathbf{q}_{\text{real}} \in \mathcal{C}_{\text{free}} \text{ do} && \triangleright \text{the particles actual position} \\
 \Delta\mathbf{q}_n &\leftarrow -\mathbf{J}^\dagger(\mathbf{q}_{\text{new}}) \mathbf{n}_{\text{surf}} && \triangleright \text{move towards surface} \\
 \mathbf{q}_{\text{real}} &\leftarrow \mathbf{q}_{\text{real}} + \epsilon \cdot \Delta\widehat{\mathbf{q}}_n &&
 \end{aligned}$$

The first condition is crucial for our planner’s performance because it ensures that the robots contact state is always fully observable. It prevents all actions that end in separate, indistinguishable contacts. Other Particle-RRT planners ([Melchior and Simmons, 2007](#), [Phillips-Grafflin and Berenson, 2016](#)) do not restrict these actions because they introduce a clustering method and insert multiple nodes for different action outcomes. The second condition allows to treat measurable contact separate from undesired non-observable contact.

After inserting a valid node, the planner attempts to reach the goal state from the newly inserted node, also using forward simulation. If the resulting distribution is close to the desired goal distribution, the planner returns success. Otherwise it moves to the next iteration and picks another sample.

4.1.5 POLICY GENERATION

Given a sequence of actions and nodes from start to goal $(u_1, x_1), \dots, (u_n, x_n)$, we need to generate a policy that can be executed on a robot. This policy is a sequence that alternates between controllers and contact-based jump conditions. We instantiate one controller followed by one jump condition for each tuple of action and node (u_t, x_t) . The type of controller depends on u_t : from *connect* and *guarded* we generate a joint-space velocity controller and from *slide* we generate a compliant operational-space controller. The type of jump condition depends on the contact state \mathcal{C}_{x_t} : If there is contact, the control switch is based on the magnitude of the measured force while for non-contact states, it is based on the covered

4.1 AN ALGORITHM FOR INTERLEAVING MOTION IN CONTACT AND IN FREE SPACE: CONTACT-EXPLOITING RRT (CERRT)

distance $\|\mu_{x_t} - \mu_{x_{t-1}}\|$. We execute all controllers with low gains to safely make and break contact. This leads to weak tracking performance on the real robot but, as the policy is inherently robust, does not critically affect the outcome.

4.2 EXPERIMENTS

In this section we will first show policies generated by CERRT for problems from the POMDP literature but also for a high-dimensional manipulation problem. Second, we will analyze the effect of the planner’s parameters quantitatively.

Our planner is implemented in the Robotics Library ([Rickert and Gaschler, 2017](#)) using the Bullet physics library ([Coumans, 2015](#)) for collision detection. We executed all experiments on an office PC with a 3.3 GHz Intel Core i5 CPU running the Linux operating system. In all experiments we use a constant number of particles $N = 20$ and a goal bias in the sampler of 10%. We always initialize the start belief state x_{start} by sampling N particles from the distribution $\mathcal{N}(q_{\text{start}}, \sigma_{\text{start}})$. All experiments use an independent linear motion error for all joints of $\delta_i(\dot{q}) = \mathcal{N}(0, \sigma_{\delta} \dot{q}_i)$.

4.2.1 QUANTITATIVE ANALYSIS OF PLANNER PARAMETERS

We will now present the results of quantitative experiments that suggest sensible values for the two parameters of the planner: the free-space/contact-space exploration bias γ , and the number of particles N .

THE INFLUENCE OF γ

We executed the planner on two different scenarios: 1) a 2D scenario with narrow passages 2) the 7D manipulation problem from Video Fig. 4.2. In our analysis we varied γ and the standard deviation of the motion uncertainty σ_{δ} . We set $\sigma_{\text{start}} = 0$. In both scenarios, we ran the planner ten times each for 66 different combinations of γ and σ_{δ} . We show the average planning time for each combination in Fig. 4.5. The results show a strong influence of γ on the planning time, depending on the uncertainty.

The border case $\gamma = 0$ corresponds to pure free space search or pure contact motion. For the 2D scenario this is only reliable for problems without uncertainty. The case $\gamma = 1$ corresponds to a pure contact-space exploration. This strategy succeeds in both scenarios because they can be solved by a sequence of sliding motions. For values between 0 and 1 in the 2D scenario the planner always solves the problem. In 2D, free-space exploration is effective as long as uncertainties are low. A value of $\gamma = 0.3$ has the best performance. For high uncertainties more contact must be made and a value of $\gamma = 0.7$ performs best. In the 7D scenario from Video Fig. 4.2, the planner starts failing for uncertainties higher than 0.02 (we stop the search after 180 s) and free-space exploration is far less effective. We achieved the

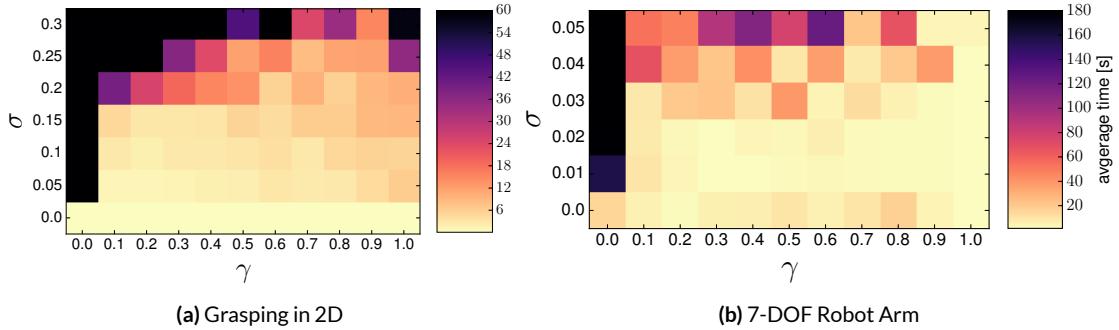


Figure 4.5: The plots show the average planning time for different combinations of γ and σ_δ . (a) For the 2D scenario from Fig. 4.6b the optimal value of γ depends on the uncertainty. (b) For the 7D manipulation scenario from Video Fig. 4.2, the planner performs best for high values of γ , which lead to a contact-seeking behavior.

best results with $\gamma = 0.95$. Our results show that for best planning performance, γ should be tuned to the problem at hand, as some problem require more free-space search and some require more contact.

THE NUMBER OF PARTICLES

The second important parameter is the number of particles to consider for planning. Too few particles will approximate the belief insufficiently which can lead to a policy with unexpected collisions. Too many particles will increase planning time since it depends on it in a linear fashion. To find a reasonable number, we run the manipulator experiment (Video Fig. 4.2) 21 times varying the numbers of particles. We execute the resulting plans in a dynamic simulation implemented in the RoboticsLab framework ([Wonik Robotics Co., 2010](#)) and execute each plan ten times with different motion error.

Number of Particles	1	2	4	8	16	32	64
SD [m]	0.3	0.27	0.275	0.24	0.3	0.1	0.06

Table 4.2: The standard deviation of the final position error drops significantly with 32 particles.

Fig. 4.2 shows the results of these experiments. While the average error of the robot’s final position is about constant for different runs the standard deviation of the error drops at 32 particles. This suggests that the generated plans are not reliable below 16 particles. A similar number of particles was reported in [Phillips-Grafflin and Berenson \(2016\)](#).

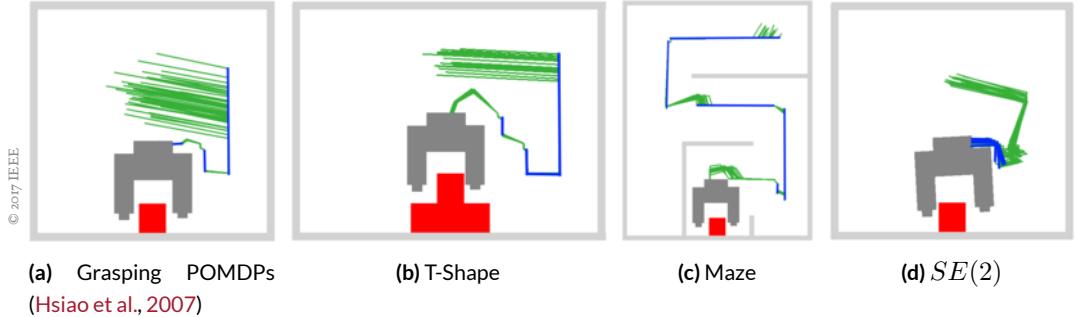


Figure 4.6: The subfigures show solutions of the CERRT planner for different grasping scenario. The gripper shows the final configuration of the path. The lines show 20 sampled trajectories, free-space motions are shown in green and slides in blue. The beginning of the paths is always in free space and the end is before grasping. CERRT outperforms POMDP planners on the benchmark (a) and scales to more complex problems.

4.2.2 PERFORMANCE ON MANIPULATION PROBLEMS

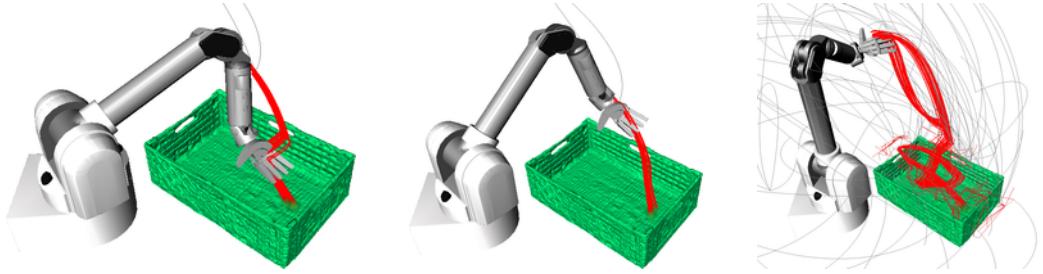
PRE-GRASPING IN 2D

This problem models a gripper picking up a square block at unknown location and is inspired by the POMDP literature (Hsiao et al., 2007, Bai et al., 2010). The gripper has contact sensors at each jaw and can translate in two dimensions. Because of a large initial uncertainty the gripper must contact the object or the walls first and then, after uncertainty is sufficiently reduced, attempt the grasp from the top.

Fig. 4.6a shows one of the solution paths CERRT found on the simple grasping scenario. All policies first establish contact with wall or object and then slide along the ground until contact with the object is perceived. The planning time for this problem averaged over ten runs is 6.8 s (± 5.1 s). A POMDP version of the problem with discrete state and actions required an average planning time of 8 s (Kurniawati et al., 2008) and 160 s with continuous state and discrete actions (Bai et al., 2010). Our approach easily scales to more complex scenarios. Fig. 4.6b shows the result for a multi-step piece (8.2 s \pm 6.9 s), Fig. 4.6c a version where the gripper must first navigate through a simple maze (23.4 s \pm 19.3 s), Fig. 4.6d a $SE(2)$ version of the problem with translation and rotation of the gripper.

PRE-GRASPING WITH A 7-DOF ROBOT ARM

CERRT is efficient enough to be directly applied to the seven-dimensional configuration space. We pose the problem of finding a path of a 7-DOF Barrett WAM to a pre-grasp position of a wall-constrained grasp. A wall-constrained grasp (5.1.3) is a grasping motion that



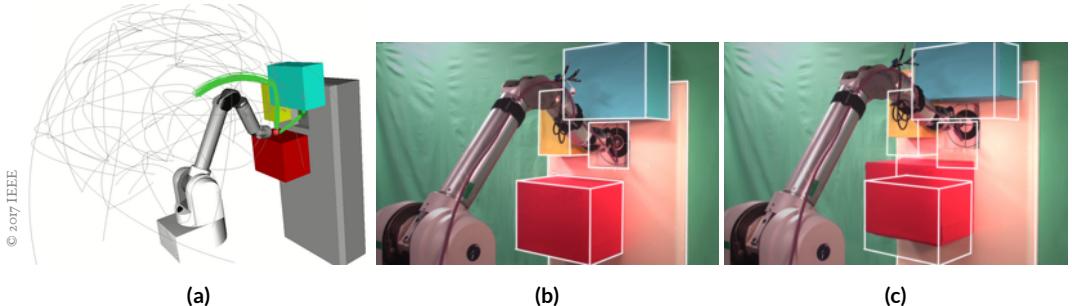
Video Figure 4.1: The images show different solutions found by CERRT for attaining a pre-grasp pose inside a container. Red lines indicate the end-effector path of five particles. The quality of the resulting plans vary, as some exhibit quite large motions (right). Note that the collision geometry for the green container is composed of five boxes and the hand is represented by a simple sphere. [<https://youtu.be/WCM0ISZPZ5s>]

pushes an object towards a vertical structure to scoop it. The scenario is depicted in Video Fig. 4.1. It contains a box in which the potential object is located. The robot can make contact with the four sides of the box and its bottom surface to reduce uncertainty. We model the end-effector as a simple sphere. The goal configuration is inside the box close to one end with the palm facing up. The start configuration is above the box.

Out of 20 attempts the algorithm fails two times to find a solution within 300 s. The average planning time for the successful runs is 50.4 s (STD 77.6 s) on a standard desktop computer at 2.2 GHz using a single-threaded implementation. Video Fig. 4.1 shows three different solution paths. The plans are either direct motions that only contact the box slightly at the edges and then at the bottom, or they take a long detour to reduce uncertainty as shown in the solution on the right in Video Fig. 4.1. Note that none of the paths exhibit the sliding motion at the bottom of the box that one might expect. This is due to the constrained interior of the box which makes it difficult to find sliding motions such that the upper links of the arm do not collide (sliding on the bottom of the box emerged when we removed two of the four sides of the collision model of the box). In the conclusion of this part (see Sec. 5.8), we will contrast the CERRT plans found for this problem with those computed with an alternative method presented in the next chapter.

REAL-WORLD MOTION WITH A 7-DOF ROBOT ARM

We conducted a final experiment, to show that the robust motion plans of CERRT also transfer to the real world. We place a 7-DOF Barrett WAM robot in front of the wall depicted in Video Fig. 4.2, similar to the scenario from Phillips-Grafflin and Berenson (2016). The goal configuration requires the arm to place its spherical end-effector into a cavity of the wall.



Video Figure 4.2: The manipulator must touch the target in the square opening of the wall. (a) The planner output. Gray lines show all explored motions. The green line indicates the found path. Our planner finds a strategy that moves to the cyan box, slides down until it loses contact, does a guarded move to the top of the red box, and moves to the target. (b) The outcome of executing the strategy on the real robot without uncertainty. The robot reaches the goal precisely. (c) We now raise the obstacles by 7 cm (the white overlay shows the wall position from (b)) and execute the policy from (b) again. The robot uses the contact to reduce uncertainty and reaches the target with an error of 2 cm. [<https://youtu.be/CXaN8ZWRMT0>]

The robot model has an initial uncertainty and a motion uncertainty of $\sigma_{\text{start}} = \sigma_\delta = 0.02$. Motion-dependent position error occurs in the real Barrett WAM robot due to stretch of the cables that move the joints. The robot uses a wrist-mounted ATI Gamma force-torque sensor to perceive contact with the end-effector but cannot perceive contact with any other part.

The outcome of the planner can be seen in Video Fig. 4.2. From ten attempts, the planner solved this problem six times within 180 s. The six successful searches required an average time of 23.8 ± 29.3 s. To validate the robustness of the plan, we introduce an unexpected disturbance. We raise the wall including all obstacles by 7 cm and execute the same policy as before. The contact with the cyan and red boxes reduces uncertainty and the robot reaches the target with an error of 2 cm which is an effective reduction of 5 cm.

4.3 RELATED WORK

Planning free-space motion and planning contact are two well-established research areas and we will briefly outline our planner’s connection to related work in both fields. Our planner balances free-space motion and contact by reasoning about uncertainty, for which we will review related work in the second half of this section.

4.3.1 FREE-SPACE MOTION

Sampling-based motion planners like the Probabilistic Roadmap (PRM) by [Kavraki et al. \(1996\)](#) or the RRT ([LaValle, 1998](#)) search the collision-free configuration space efficiently and without any fixed discretization. These planners assume no uncertainty and explicitly avoid contact. In this paper we modified RRTs to include contact and an explicit reasoning about uncertainty. In our planner, we exploit a handful of useful strategies from the motion planning literature: we utilize the Voronoi-bias ([LaValle, 1998](#)) to quickly explore configuration and contact spaces, we use the goal-connect strategy ([Kuffner and LaValle, 2000](#)) to balance exploration and exploitation ([Rickert et al., 2014](#)), and we use a projection strategy similar to task-constrained motion planning ([Stilman, 2007](#), [Berenson et al., 2011](#)) to implement sliding along surfaces.

4.3.2 CONTACT-SPACE MOTION

Classic work in manipulation planning showed how a sequence of compliant motions can be robust to uncertainty. So called pre-images ([Lozano-Pérez et al., 1984](#)) characterize the regions from which compliant actions reach a desired goal state. Chaining them gives uncertainty-tolerant plans. In certain cases robust manipulation can be achieved without any sensors ([Erdmann and Mason, 1988](#)). Sampling-based motion planning can explore the space of configurations in contact ([Ji and Xiao, 2001](#), [Siméon et al., 2004](#)) but does not reason about the uncertainty reducing capability. Our planner searches the space of all configurations in contact to exploit its uncertainty-reducing capability. Instead of backwards-chaining, it uses forward simulation to approximate pre-images.

4.3.3 REASONING ABOUT UNCERTAINTY

To decide whether to exploit contact or to move in free space, our planner reasons explicitly about uncertainty. This distinguishes it from all previously mentioned methods. We will

now review methods that reason about uncertainty explicitly.

PLANNING WITH UNCERTAIN ACTIONS

Markov Decision Processes (MDP) model actions with uncertain outcome. This framework allows robots to reason about the collision probability of actions and balance short and safe paths ([Alterovitz et al., 2007](#)). Very related to our method are particle-RRTs ([Melchior and Simmons, 2007](#), [Phillips-Grafflin and Berenson, 2016](#)) which represent the outcome of actions as a set of particles, just like our planner. Such a representation is suited to reason about the uncertainty-reducing capability of contact because the belief over configurations in contact is non-gaussian ([Phillips-Grafflin and Berenson, 2016](#)). There are three important differences of particle-RRTs to our work:

1. The particle-RRT assumes perfect knowledge about the robot state which CERRT does not. This allows our planner to solve a broader class of problems.
2. Our method explicitly seeks contact to reduce uncertainty, while the particle-RRT just achieves contact randomly. We believe this is the reason for our planner's efficiency.
3. CERRT only generates one sequence of free-space and contact-motions. In contrast, the particle-RRT has actions with multiple outcomes. This makes the particle-RRT's behavior more robust to failure.

PLANNING WITH UNCERTAIN ACTIONS AND OBSERVATIONS

Once uncertainty exists in action outcome and the robot cannot fully observe its own state, the planning problem can be modeled as a Partially-Observable MDP (POMDP). The solution to a POMDP is a global sensing-action strategy that balances uncertainty reduction optimally with goal achievement. Unfortunately POMDPs of realistic sizes are intractable to solve optimally and hard to approximate due to the combinatorial explosion of belief space. To tackle the high complexity, further assumptions must be made. Assuming Gaussian state uncertainty is effective ([Prentice and Roy, 2010](#), [Platt Jr et al., 2010](#), [Platt et al., 2011](#), [Van Den Berg et al., 2012](#), [Bry and Roy, 2011](#), [Agha-Mohammadi et al., 2014](#)) but does not adequately represent the belief state of configurations in contact. Sampling-based solvers ([Thrun, 1999](#), [Porta et al., 2006](#), [Kurniawati et al., 2008](#)) can approximate POMDP solutions in reasonable time but are limited to low-dimensional problems, often with discrete states and actions.

Touch-based localization of the robot relative to a known environment can be cast as an optimization of a submodular metric (Javdani et al., 2013, Vien and Toussaint, 2015), which is efficiently solved by a greedy algorithm. However, submodularity does not hold if motions in free-space increase uncertainty.

Our method tackles the high complexity by planning with a belief over the robot configuration but a fully observable contact-state. This moves our problem in the domain of Mixed-observability MDPs (Ong et al., 2010) which are easier to solve.

POMDPs FOR MANIPULATION

POMDP solvers were applied to low-dimensional versions of manipulation tasks such as in-hand manipulation to localize an object (Koval et al., 2016) or pre-grasp manipulation (Hsiao et al., 2007, Bai et al., 2010, Horowitz and Burdick, 2013). The latter application is relevant to our method and we showed the uncertainty-reducing capabilities of our planner on the same problem in Sec. 4.2, but with two important differences:

1. Unlike the POMDP-approaches (Hsiao et al., 2007, Bai et al., 2010, Horowitz and Burdick, 2013) our method does not assume any a priori discretization of state or action space. It can be directly applied just using the geometric model of world and robot as input.
2. We do not assume uncertain contact sensors while in the POMDP scenario sensors can return false measurements, which makes up a large part of the complexity. We think that our noise-free assumption is justified for undirected, binary contact-sensing.

Toussaint et al. (2014) optimizes trajectories for contact-exploiting motion by adding uncertainty reduction as an objective. This method does not do a global search like our method but might be used to post-process the paths found by our planner.

4.4 CONCLUSION

We presented a planner to generate robust motion strategies under significant uncertainty in robot state, action, and world model. The planner achieves robustness by interleaving motion in free space with motion in contact. The key to the planner’s efficiency is the assumption of a fully observable contact state and a search strategy tailored to the combined contact space and free space. Our experiments showed that the same planner can solve challenging benchmarks from the POMDP literature in continuous state and action spaces but also scales to realistic motion planning problems in configuration space.

We believe there is room for runtime improvement in our planner and most extensions from the motion planning literature such as bidirectional search (Kuffner and LaValle, 2000), guided sampling, and balancing of exploration and exploitation (Rickert et al., 2014) will be just as useful for planning interleaved free-space and contact motion.

5

Planning Grasp Strategies That Exploit the Environment

THE PREVIOUS CHAPTER SHOWED THE BENEFITS OF CONTACT when planning robust motions. It was based on the insight that making and breaking contact are easily measurable events that can be used to reduce uncertainty about the robot's configuration. As a result, contact-seeking motions emerged from a model of the belief about the system state and the desire to attain a goal with certainty. In this chapter we take a slightly different route. Rather than modeling uncertainty, we directly assume that contact motions should be favored over free-space motions. This allows us to focus on grasping strategies that make extensive use of contact.

Many recent advances in robot grasping and manipulation can be explained by a simple insight: contact with the environment can improve performance! For example, underactuated, soft hands greatly benefit from the interactions that occur naturally between hand, object, and environment (Dollar and Howe, 2010, Deimel and Brock, 2016, Catalano et al., 2014). Furthermore, the robustness of grasping can be increased through the use of contact with support surfaces (Kazemi et al., 2014). And the dexterity of simple, rigid hands is increased drastically through deliberate contact with the environment (Chavan-Dafle et al., 2014). In addition, human graspers routinely and amply employ environmental contact, especially in difficult grasping problems (Deimel et al., 2013). Given this broad support for the importance of contact in grasping and manipulation, it is surprising that recent grasp and manipulation

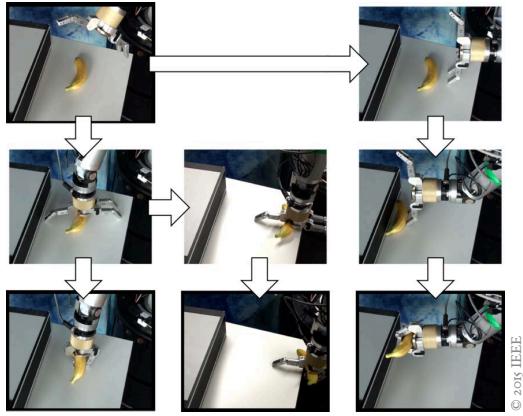


Figure 5.1: Our planner generates grasping strategies that make extensive use of contact with the environment. Thus, control and perception can be simplified. In this example, three different grasping strategies for a banana are shown. They exploit the table surface, edge, and the side panel of an object sitting on top of the table.

© 2015 IEEE

planners generally regard the environment as an obstacle, rather than as an opportunity.

In this chapter, we introduce a grasp planner that generates robust grasping strategies based on the exploitation of contact constraints available in the environment (see Fig. 5.1). We define an *environmental constraint* (EC) as a feature of the environment that enables replacing aspects of control and/or perception with interaction between hand and environment. To plan the exploitation of EC we must eliminate the existing separation between perception, planning, and control. Instead, we tightly integrate perception and action by realizing each to satisfy the others' requirements and to account for its limitations.

We will start by describing different types of grasping strategies that are based on the exploitation of ECs. This allows us to identify commonalities and substructures which we can use to describe a planning algorithm (Sec. 5.2). The details of this algorithm – extracting ECs from RGB-D measurements and sequencing exploitations – are described in the following sections (Sec. 5.3 to Sec. 5.5). Finally, we extensively evaluate different aspects of grasping strategies based on EC exploitation in simulation and on a real robotic platform.

5.1 GRASP STRATEGIES THAT EXPLOIT THE ENVIRONMENT

In the following we describe the structure of three different families of grasping strategies that rely on an explicit exploitation of environmental contact. For each strategy we will explain which ECs are exploited, what their constituting sensor events are, how they vary for different hand designs and what similar strategies are part of each family.

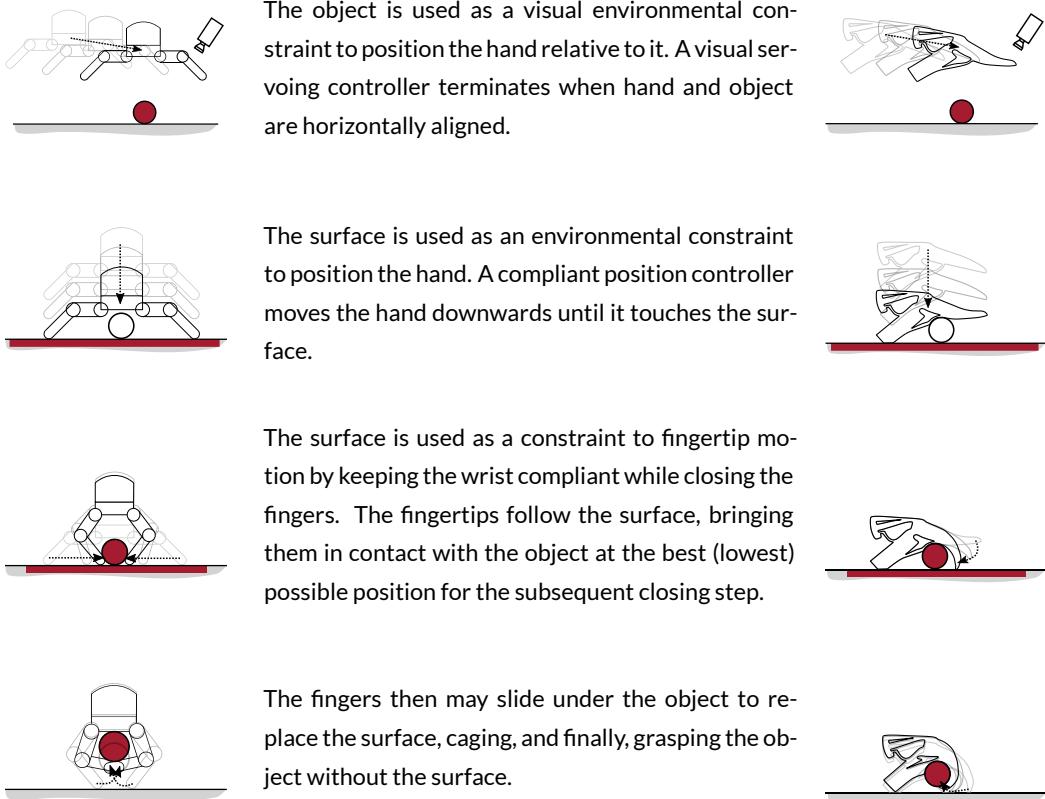


Figure 5.2: Temporal evolution of the surface-constrained grasp strategy (environmental constraints are shown in red)

5.1.1 SURFACE-CONSTRAINED GRASPS

The surface-constrained grasp assumes that the object is placed on a planar and stiff support surface. Due to the omnipresence of such surfaces it is probably the grasp strategy that is applied most often. The hand uses the surface to partly cage the object before closing its fingers. Fig. 5.2 describes the different phases of the motion in detail. Kazemi et al. (2012) present a similar strategy which they call “force-constrained closing strategy”. Their focus is on controlling the position and orientation of the wrist such that the fingertips maintain contact at all times.

HAND-SPECIFIC VARIATIONS: The strategy’s main challenge is to create the cage with the surface as early as possible and not to break it while the fingers close. The parts of the hand

that are used to create the cage can vary significantly, although the palm will nearly always be involved. To avoid breaking the cage the hand (usually the fingertips) needs to retain contact with the surface at all times. This contact-keeping behavior can be achieved through active compliance in the wrist (as done in the case of the Barrett Hand) or by passive compliance, e.g. through deformable material (as done in the case of the RBO Hand 2).

SENSOR EVENTS: The most important sensor events during the surface-constrained grasp are recognizing when the hand is aligned correctly above the object and when the hand contacts the object or surface. Aligning the hand in free-space can be achieved visually. Recognizing the state in which the finger closing can be initiated is more challenging. Depending on the object height the palm or the fingers might first touch the object or support surface respectively. Force-feedback in the wrist is most appropriate to detect this event. For using tactile feedback the hand needs to be equipped abundantly with sensor patches at all possible contact surfaces.

RELATED STRATEGIES: Odhner et al. (2013) present a two-fingered grasping strategy for picking up coins or other small, flat objects. Here, the surface constraint is used to press down on one side of the object such that the opposite end is lifted and the second finger can slide underneath.

5.1.2 EDGE-CONSTRAINED GRASPS

The edge-constrained grasping strategy exploits a surface and an edge feature in the environment. It contacts the object using the surface, slides it towards an edge, and wraps a finger around the protruding part of the object to establish a grasp. Edge constraints are often explicitly introduced in human environments to simplify grasping flat objects, e.g. coins from a coin tray. The different phases of the edge-constrained grasping strategy are illustrated in Fig. 5.3. This strategy can also be seen as a distinct pre-grasp interaction which reconfigures the object enabling contact with faces that were previously inaccessible. A similar strategy was presented by Kappler et al. (2012) and King et al. (2013), but with a strong focus on the planning of a feasible sliding motion towards a table edge.

HAND-SPECIFIC VARIATIONS The most common variation is the orientation of the wrist during the sliding motion. This affects whether the fingers, the thumb or even other DOF are used to wrap around the exposed part of the object as soon as it is located at the edge.

Another variation is whether to reorient the hand between sliding and finger closing. If this is done, the object's center of mass should not leave the support surface at the end of the sliding motion. Alternatively, the hand needs to constantly apply a force onto the object as it is done with the palm of the RBO Hand 2. In both scenarios the object is statically stable at all times.

SENSOR EVENTS The event that the object overhangs the edge and can be grasped can be detected through visual, haptic, tactile or even proprioceptive feedback. A combination of multiple modalities might be the most robust.

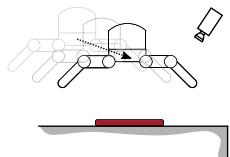
RELATED STRATEGIES Instead of exploiting an edge orthogonal to the direction of gravity, the same can be done with edges that are vertical, e.g. at shelves.

5.1.3 WALL-CONSTRAINED GRASPS

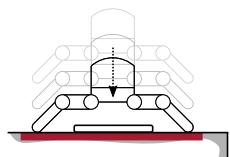
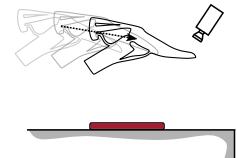
Walls are another common environmental feature that can be exploited, and can be found on bowls, drawers, shelves and boxes. A wall-constrained grasp pushes the object against a vertical structure (the “wall”) to either slide the fingers underneath or constrain the object between fingertips and wall. The grasp is completed by closing the fingers. Fig. 5.4 shows the phases in detail.

HAND-SPECIFIC VARIATIONS Pushing the object can be done in a variety of hand orientations. As a result the object is either moved through contact with the fingertips, finger pads, or even the palm. The number of fingers used also plays a role. Using fewer fingers or even only the thumb will allow an easier entrance between object and support surface. However, fewer fingers will also serve as weaker support during closing and object lift.

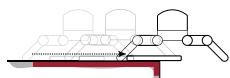
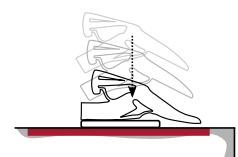
SENSOR EVENTS To push the object the same events need to be detected as with the edge-constrained grasp. The impact of the hand on the object can be recognized through haptic or tactile feedback or by optical flow. The main challenge of this grasping strategy is to reliably measure when to close the fingers. Since there are two modes (fingers slide underneath object or not) which might require different closing motions, haptic feedback might not be enough in some cases. It could be complemented by visual measurements.



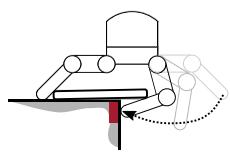
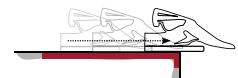
The object is used as a visual environmental constraint to position the hand relative to it. A visual servoing controller terminates when hand and object are horizontally aligned.



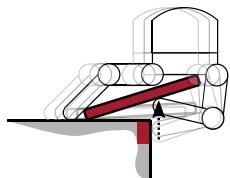
The surface is used as an environmental constraint to position the hand. A compliant position controller moves the hand downwards until it touches the surface. Upon termination, the fingertips are aligned vertically w.r.t. the object.



The surface in conjunction with the edge are used as environmental constraints to slide the object to the edge. A compliant position controller drags the fingertips across the surface. The motion is terminated when the object partially protrudes over the edge. This can be detected by using the edge either as a visual or as a haptic environmental constraint.



The vertical surface is used as an environmental constraint to position the right finger. A compliant position controller closes the finger until contact with the surface. Upon termination the object is caged.



By compliantly moving the end effector horizontally, the right finger slides along the vertical surface, contacting the object, and resulting in force closure without the table surface.



Figure 5.3: Temporal evolution of the edge-constrained grasping strategy (environmental constraints are shown in red)

RELATED STRATEGIES A similar strategy is used when grasping items from a fully stacked box. Pushing the fingers between the items resembles the wall-constrained motion. The only difference being that it occurs along the direction of gravity.

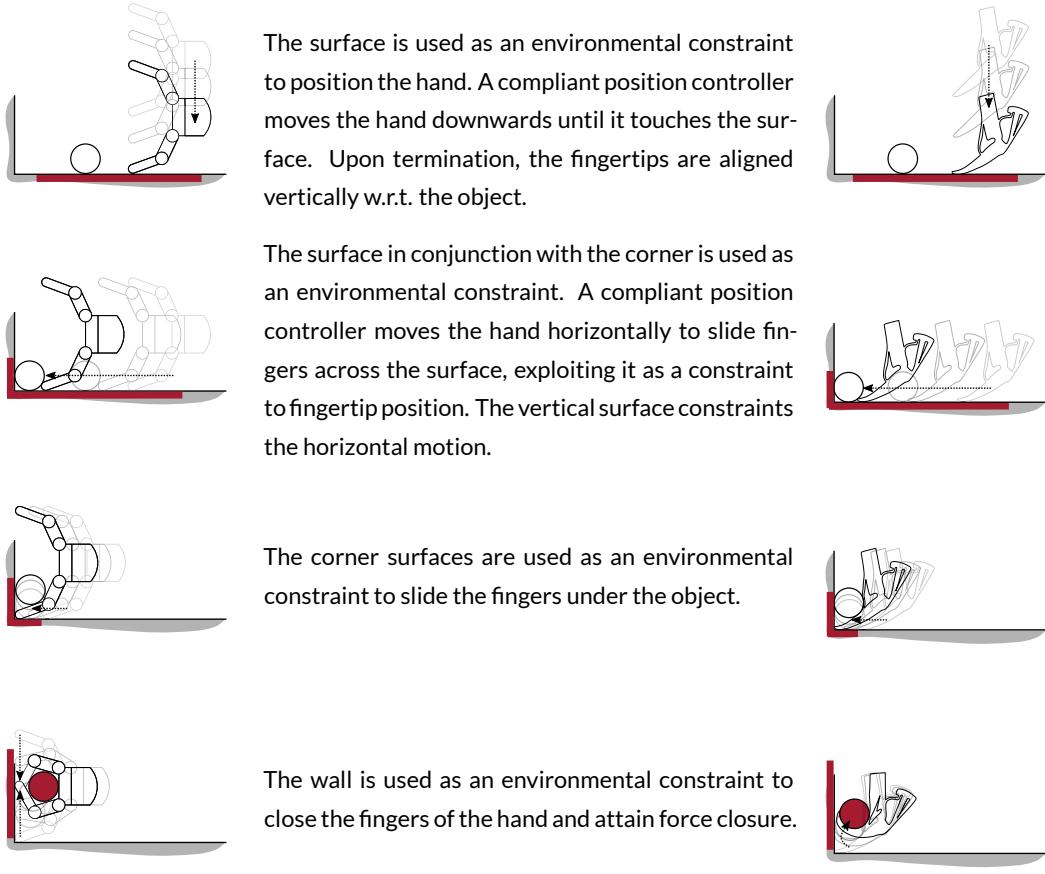


Figure 5.4: Temporal evolution of the wall-constrained grasp strategy (environmental constraints are shown in red)

Xu et al. (2009) introduce a picking device similar to a dustpan to pick up objects from the floor. The effect of the wall constraint is achieved by using an additional actuator which pushes the objects onto the dustpan.

Morales et al. (2007) present a controller for grasping a book from a row of books in a shelf. The strategy pivots the book around the lower end of the back of the book before wrapping the fingers around its exposed parts. The surrounding books act like walls that constrain the book during the first part of the motion. Alternatively, one could argue that this strategy is closer related to the edge-constrained grasp since it is based on incrementally exposing faces of an object that are otherwise inaccessible.

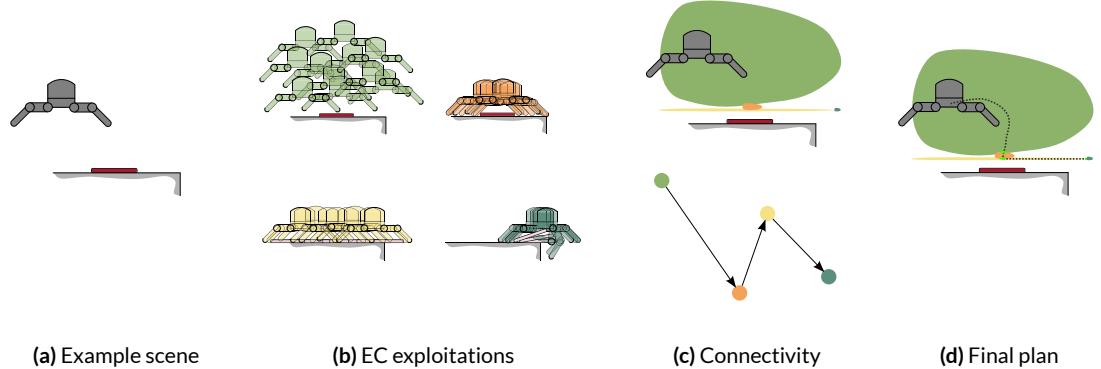


Figure 5.5: The panels illustrate the idea of planning environmental constraint exploitations for grasping. (a) The example scene contains a red object on flat support surface. (b) We divide the state space into regions, each representing the exploitation of a specific environmental constraint. (c) Intersections between those regions define a transition graph, in which a grasp plan can be determined. (d) The resulting plan is a sequence of environmental exploitations that end in a grasp.

5.2 PLANNING ENVIRONMENTAL-CONSTRAINED GRASPS

We have seen that the previously described strategies are composed of re-occurring motion primitives such as sliding across or bumping into surfaces. The key challenge is to search this vocabulary to find sequences of these primitives instead of exact planning in the combined state space of hand, object, and environment which is too difficult. To follow the rest of this section, please refer to Fig. 5.5.

Environmental constraints implicitly divide the state space into separate regions, i.e. regions that correspond to one particular type of EC exploitation. Samples from these regions projected into the workspace are shown in Fig. 5.5b. As we will see later, rather than computing these regions explicitly, which would be computationally challenging, we determine an approximate representation of these regions directly from sensor data. By determining intersections of these approximated state space regions in a graph (Fig. 5.5c), we obtain a representation of possible transitions between different EC. Given this information, we can search for a feasible sequence of EC exploitations that leads from the current state to a successful grasp (Fig. 5.5d).

We define a single EC exploitation X_{ECE} as a contiguous subset of all possible hand and object poses and the exerted forces onto the hand:

$$X_{ECE} \subset \mathcal{C}^{\text{hand}} \times \mathcal{C}^{\text{object}} \times \mathcal{W}^{\text{hand}},$$

Algorithm 5.4 Grasp Planning based on EC exploitation

Input: $P_{\text{RGB-D}}$, \mathbf{x}_{hand} , $\mathbf{x}_{\text{object}}$

Output: HA

$P \leftarrow \{\text{free, slide, surface, edge, wall}\}$	▷ return a hybrid automaton
	▷ all environmental constraint exploitation types
$ECEs \leftarrow \emptyset$	
for $p \in P$ do	▷ find all possible motions (Sec. 5.3)
$ECEs \leftarrow ECEs \cup \text{DETECT}_p(P_{\text{RGB-D}})$	
$S \leftarrow \text{SOLVE_PDDL}(\mathbf{x}_{\text{hand}}, \mathbf{x}_{\text{object}}, ECEs)$	▷ find sequence of ECEs (Sec. 5.4)
$HA \leftarrow \text{CONVERT}(S)$	▷ convert sequence to hybrid automaton (Sec. 5.5)

where $\mathcal{C}^{\text{hand}} = \mathcal{C}^{\text{object}} = SE(3)$ and $\mathcal{W}^{\text{hand}}$ is the 6D wrench space. To plan among EC exploitations we need to look at their connectivity. This is defined by their intersections, i.e. we can transit between two arbitrary EC exploitations X_{ECE_i} and X_{ECE_j} if $X_{ECE_i} \cap X_{ECE_j} \neq \emptyset$.

Due to its high complexity, we approximate X_{ECE} by using multiple assumptions. First, we assume that during a single exploitation the state of the object $\mathcal{C}^{\text{object}}$ will not be used as a continuous feedback signal, i.e., we can ignore it. Second, we assume that $\mathcal{C}^{\text{hand}}$ within an ECE is very structured and can be expressed by a low-dimensional manifold. Therefore, we represent the hand poses as an oriented bounding box and orientations by discretizing all possible rotations. Finally, we use only a single 6D vector to represent the contact wrench exerted onto the hand, since this contact property is constant within a single EC exploitation. The resulting parametrization used by the planner is:

$$\tilde{X}_{ECE_i}(obb^{\text{hand}}, R^{\text{hand}}, w^{\text{hand}}) = \left\{ \begin{pmatrix} c^{\text{hand}} \\ w^{\text{hand}} \end{pmatrix} \middle| \begin{array}{l} c_{\text{position}}^{\text{hand}} \in obb^{\text{hand}} \\ c_{\text{rotation}}^{\text{hand}} \in R^{\text{hand}} \\ w^{\text{hand}} \in \mathbb{R}^6 \end{array} \right\}, \quad (5.1)$$

where obb^{hand} is the 3D oriented bounding box and R^{hand} is the set of orientations. The wrench w^{hand} describes the forces and torques acting onto the hand within that particular EC exploitation.

The planning algorithm is summarized in Algorithm 5.4. Its main steps are the extraction of ECEs from raw sensor data and the sequencing of them to find a suitable grasping strategy. We will now explain the different steps in more detail.

5.3 IDENTIFYING ENVIRONMENTAL CONSTRAINTS FROM RGB-D MEASUREMENTS

In the following, we will explain how we perceive parts in the environment which afford three types of grasping strategies and three non-prehensile manipulations that make explicit use of ECs. For each of the six actions, we will devise a sensor model that is used to visually recognize the EC exploitation according to the parametrization given previously (see Eq. 5.1): $\tilde{X}_{ECE}(obb^{\text{hand}}, R^{\text{hand}}, w^{\text{hand}})$.

Sensory input is assumed to be in the form of a depth image of the scene. The object to be grasped is represented as a bounding box whose parameters are assumed to be known. Alternatively, the object can be given in a heuristic fashion such as: “the largest point cloud segment on the largest surface in the scene”. In this case the bounding box parameters are estimated.

We implement all functionality described below as a single computational graph ([Rubleev and Straszheim, 2011](#)). This allows us to re-use computation and quickly adopt changes.

SURFACE-CONSTRAINED GRASPS: DETECT_{surface}

We detect surface-constrained grasps by checking whether the immediate surroundings of the object are planar. To do this we crop the point cloud based on the object bounding box and size of the hand and fit a plane to the points. If the quality of the fit exceeds a given threshold we define the region obb^{hand} of the surface-constrained grasp by the object bounding box. The orientations R^{hand} are based on the orientation of the fitted plane. The wrench w^{hand} consists of a force component directed towards the palm of the hand. Fig. 5.6b shows an example of a detected surface-constrained grasp from sensor data.

WALL-CONSTRAINED GRASPS: DETECT_{wall}

The wall-constrained grasp exploits two surfaces that form a concavity. We detect such grasps by finding concave edges in the environment. We segment the point cloud using a variety of differently parametrized flood fill segmentations. From the resulting segmentation soup we extract planes using a least-square fitting method based on point-to-plane distances. For each plane we calculate a corresponding polygon by projecting the points onto the plane computing their convex hull ([Sklansky, 1982](#)). We find concave edges by testing all pairs of polygons for intersections if their normals differ by less than 180°. The region obb^{hand} is set along the edge and orientations R^{hand} are normal to the wall surface. The wrench w^{hand} consists only of a force component directed towards the palm of the hand.

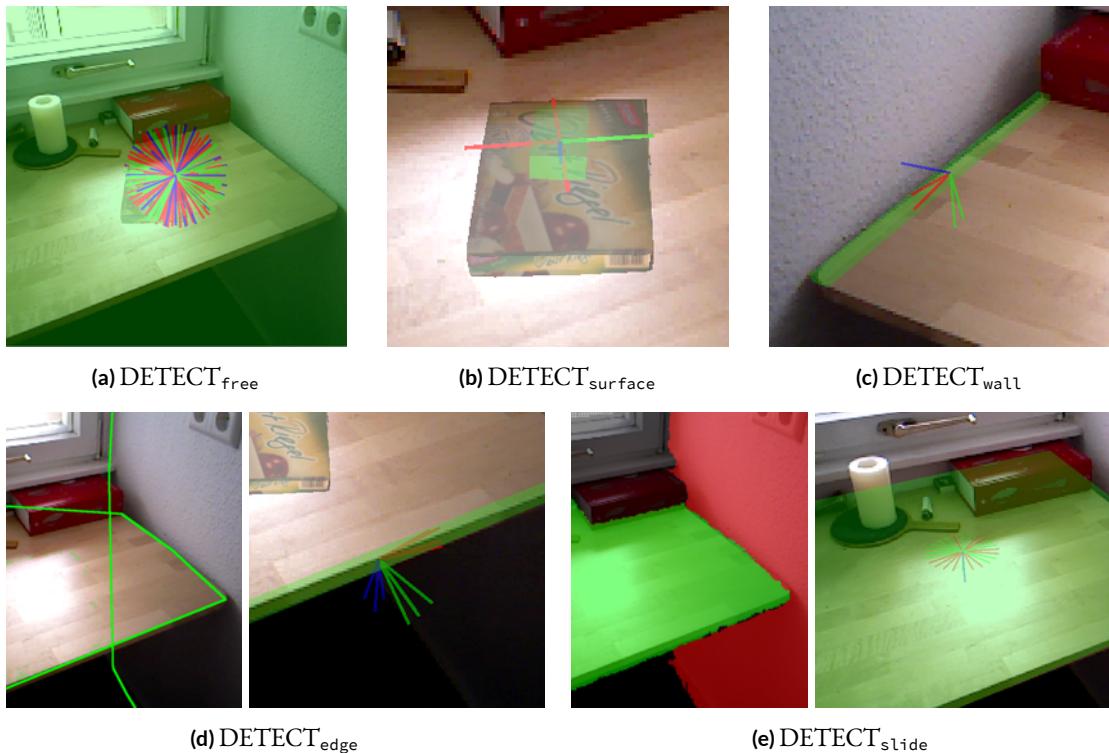


Figure 5.6: The subfigures visualize the results of detecting different environmental constraints in an RGB-D image. The green boxes represent the possible hand poses obb^{hand} , while orientations R^{hand} are shown as RGB axes. Subfigures (d) and (e) also show an intermediate segmentation result.

EDGE-CONSTRAINED GRASPS: DETECT_{edge}

The spatial parameters obb^{hand} and R^{hand} are computed based on the presence of convex edges in the scene. We extract edges by searching along the boundaries of planar segments in the depth image, see Fig. 5.6d. First, we segment, fit, and extract polygons as done in DETECT_{wall}. We iterate over the edges of each polygon and calculate their local curvature by selecting all points in the neighborhood of the edge and averaging their Gaussian curvatures (which is the product of the two principal curvatures at that point). Convexity is assumed if the average Gaussian curvature is positive. The wrench parameter w^{hand} represents the torque components that is exerted onto the hand due to the fact that some fingers still touch the surface while half of the hand already passed the edge.

VISUALLY-CONSTRAINED POSITIONING: DETECT_{free}

Inside the visible workspace, the hand can be constrained visually, i.e. by visual servoing. This strategy is helpful whenever positioning the hand in free space is hard due to missing external calibrations or poor sensor models (e.g. encoders that ignore cable stretch). Although we constrain the hand by model-based tracking in 3D using a depth sensor, more complex schemes are possible that use lower-level features like edges (Fox and Hutchinson, 1995). To compute the spatial extent obb^{hand} , we expand a box in visible free space starting from the current hand pose until collision with a depth measurement. The set R^{hand} includes all possible orientations. Since we do not expect any contact, the wrench parameter is set to zero ($w^{\text{hand}} = 0$). Fig. 5.6a shows an example of the detected free space.

SURFACE-CONSTRAINED SLIDING: DETECT_{slide}

During sliding the hand and object's motion are restricted by a support surface exposing only three DOF. We extract sliding constraints by segmenting the depth image with a flood-fill algorithm and multiple parametrizations of the segmentation criteria. The method clusters regions with low curvature and small changes of surface normals (see Fig. 5.6e). We fit planes to the points of each segment in a least-square sense. If the mean squared error of the fit is below a fixed threshold we assume that the surface can be used for sliding.

To generate the corresponding region obb^{hand} the points are projected onto the plane and the minimum enclosing rectangle is computed. The orientations R^{hand} include all rotations around the surface normal of the segment. The wrench w^{hand} is set with a force that points inside the hand's palm.

5.4 SEQUENCING ENVIRONMENTAL CONSTRAINT EXPLOITATIONS

To find sequential exploitations of environmental constraints we need to define the aforementioned connectivity check between two arbitrary EC exploitations. We use their spatial and contact properties to decide this:

$$\widetilde{\text{ECE}_i} \mapsto \widetilde{\text{ECE}_j} \iff \text{obb}_i \cap \text{obb}_j \neq \emptyset \wedge R_i \cap R_j \neq \emptyset$$

This means that the hand poses described by both EC exploitations need to overlap in position and orientation. We implement the overlap of the hand positions by intersecting the

two corresponding oriented boxes. Since the orientations are discretized, we can simply use a set intersection.

Based on the connectivity operator “ \rightarrow ” we can define our simple search problem using the planning domain definition language (PDDL, McDermott et al. (1998)). PDDL is a less restricted version of the STRIPS planning language (Fikes and Nilsson, 1971). A PDDL problem is defined by an initial and goal state, actions and the result of applying them. A single state is represented as a logical formula which is a conjunction of ground, functionless atoms. This factored representation is ideally suited for our problem, since our world can be described as a collection of (a few) entities – the ECEs.

We introduce the two simple predicates `Hand(·)` and `Object(·)` which are true if hand or object are currently exploiting a specific EC. We define the initial and goal state as follows:

$$\begin{aligned} \text{Init}(\text{Hand}(EC_{\text{hand}}) \wedge \text{Object}(EC_{\text{object}}) \wedge & \bigwedge_{\substack{EC_1, EC_2 \in ECs, \\ EC_1 \rightarrow EC_2}} \text{Connected}(EC_1, EC_2) \wedge \\ & \bigwedge_{\substack{EC \in ECs, \\ EC = \text{surface,wall,edge}}} \text{IsGraspingECE}(EC)) \\ \text{Goal}(\text{ObjectGrasped}) \end{aligned}$$

where EC_{hand} and EC_{object} are the initially exploited ECs given by the robot’s starting configuration \mathbf{x}_{hand} and the location of the object in the scene $\mathbf{x}_{\text{object}}$. The set ECs contains all detected environmental constraints from the point cloud. In PDDL the closed-world assumption is used which means that all atoms that are not mentioned in a state are supposed to be false (e.g. `ObjectGrasped` is not part of the initial state). We define three action schemas that affect the search state:

$$\begin{aligned} \text{Action}(\text{MoveHand}(ece_{\text{from}}, ece_{\text{to}}), \\ \text{PRECOND: } \text{Hand}(ece_{\text{from}}) \wedge \text{Connected}(ece_{\text{from}}, ece_{\text{to}}) \\ \text{EFFECT: } \neg \text{Hand}(ece_{\text{from}}) \wedge \text{Hand}(ece_{\text{to}})) \\ \text{Action}(\text{MoveObject}(ece_{\text{from}}, ece_{\text{to}}), \\ \text{PRECOND: } \text{Hand}(ece_{\text{from}}) \wedge \text{Object}(ece_{\text{from}}) \wedge \text{Connected}(ece_{\text{from}}, ece_{\text{to}}) \\ \text{EFFECT: } \neg \text{Hand}(ece_{\text{from}}) \wedge \neg \text{Object}(ece_{\text{from}}) \wedge \text{Hand}(ece_{\text{to}}) \wedge \\ \text{Object}(ece_{\text{to}})) \end{aligned}$$

```
Action(GraspObject,
  PRECOND: Hand(ece) ∧ Object(ece) ∧ IsGraspingECE(ece)
  EFFECT: ObjectGrasped)
```

Action schemas are universally quantified and apply to any state in which the precondition holds. The effect describes the result of applying a particular action schema. Our actions ensure that environmental constraint exploitations can only be sequenced if they are connected, and that any sequence needs to end in a surface-constrained, wall-constrained or edge-constrained grasp.

Although we use A^* to search for solutions we found that due to the small search tree size in practice the null heuristic can be applied, which results in Dijkstra's algorithm. The average planning time is ≈ 80 ms. A heuristic search procedure (e.g. Hoffmann and Nebel (2001)) would only marginally improve search times.

5.5 FROM ECE SEQUENCES TO HYBRID AUTOMATA

We are finally looking for controllers to execute the sequence on a real robotic platform. Similar to the conversion of the configuration-space path found by CERRT into a hybrid automaton (Sec. 4.1.5), we will turn the ECE sequence into a hybrid automaton (Egerstedt, 2000). Each EC exploitation can be seen as a controller with desired spatial and contact profiles and a termination predicate defining the switching condition.

Given a planned sequence of EC exploitations ($\widetilde{\text{ECE}_1} \mapsto \widetilde{\text{ECE}_2} \mapsto \dots \mapsto \widetilde{\text{ECE}_n}$), we construct a hybrid automaton by using hybrid position/force controllers. Their termination predicates are defined as the poses given by $(\text{obb}_i^{\text{hand}}, R_i^{\text{hand}}) \cap (\text{obb}_j^{\text{hand}}, R_j^{\text{hand}})$ and the wrenches given by $w_j^{\text{hand}} - w_i^{\text{hand}}$. Note that here we use simple linear interpolation of the end-effector pose to generate trajectories. This might create trajectories that are infeasible to execute due to the kinematic constraints imposed by the robot or obstacles in the environment. In the following we briefly explain how to generate trajectories which exploit the same ECs and satisfy kinematic and collision constraints.

We formulate this problem as constrained optimization problem. Our objective is to find the shortest path of robot configurations q_1, \dots, q_T that exploits a given sequence of ECs ($\widetilde{\text{ECE}_1} \mapsto \dots \mapsto \widetilde{\text{ECE}_n}$). At the same time we want to avoid collisions between the robot and the environment. We exclude the end-effector from the collision geometry since we only want to avoid collisions with non-compliant parts of the robot, e.g. its elbow or upper arm.

The resulting constrained optimization problem is given by:

$$\begin{aligned} \text{minimize}_{\boldsymbol{q}_{1:T}} \quad & \sum_{t=1}^{T-1} \|\boldsymbol{q}_t - \boldsymbol{q}_{t+1}\|^2 \\ \text{subject to} \quad & sd(\text{convexhull}(\boldsymbol{q}_t, \boldsymbol{q}_{t+1})) - d_{\text{safe}} \geq 0, \quad t = 1, \dots, T-1 \\ & d(\widetilde{\text{ECE}}_i, \boldsymbol{q}_{t(i)}) = 0, \quad i = 1, \dots, n \text{ and } t(i) = 1 + (i-1)\frac{T}{n}, \dots, i\frac{T}{n}. \end{aligned}$$

The T equality constraints $d(\widetilde{\text{ECE}}_i, \boldsymbol{q}_t)$ are defined as the shortest distance between the hand pose in configuration \boldsymbol{q}_t (given by forward kinematics) and all the possible hand poses of $\widetilde{\text{ECE}}_i$, which are specified by the bounding box obb_i^{hand} and the orientation set R_i^{hand} . We partition the trajectory $\boldsymbol{q}_{1:T}$ into n equal parts and assign $\frac{T}{n}$ configurations to each ECE. The $(T-1)$ inequality constraints sd ensure collision avoidance with the environment. They are defined through the signed distance function which is negative in case of a penetration between rigid bodies and positive otherwise. The convex hull of the volume swept by the robot between successive configurations is used to check for collisions with static objects. We set the minimal possible distance d_{safe} to be 3 cm.

We solve for $\boldsymbol{q}_{1:T}$ using the sequential convex optimization scheme presented by [Schulman et al. \(2013\)](#). The method optimizes a merit function which is the sum of the original objective and constraints scaled by a coefficient. This coefficient is increased until a feasible solution is found. During each iteration the merit function is turned into a convex approximation around the current iterate and optimized with a quadratic program solver.

The resulting path will satisfy the robot's kinematic capabilities and avoid obstacles. If no solution can be found, this might be due to the a local minimum or the non-existence of a solution path. To reduce the likelihood of a local minimum, we restart the sequential optimization with randomized initial values $\boldsymbol{q}_{1:T}$. If no solution can be found after 10 attempts, we return to the ECE graph and choose the next sequence.

5.6 EXPERIMENTS

Our experiments need to answer multiple questions:

1. Are grasping strategies that exploit the environment in fact more robust than those that do not? (Sec. 5.6.1)
2. Are strategies needed that exploit different environmental constraints? (Sec. 5.6.2)
3. Are the strategies sensitive to variations in the properties of a particular environmental constraint? (Sec. 5.6.3)
4. Does the planner find exploitable constraints in everyday scenes or are these specific cases? (Sec. 5.6.4)
5. Can the presented planner generate motion that can be executed successfully on a real robotic platform? (Sec. 5.6.5)

The following five experiments try to answer each of these questions. We will show the benefits of using environmental contact during grasping and the capabilities of the presented planner to find these types of strategies.

5.6.1 COMPARISON OF ECE STRATEGY WITH EC-AGNOSTIC STRATEGY

We compare an EC-agnostic with an ECE grasping strategy. The EC in this experiment is provided by the supporting table surface. As the height of objects decreases, grasping becomes more difficult. We expect grasp success to be higher if the constraint provided by the table surface to guide finger placement on the object is exploited.

The EC-agnostic strategy treats the table surface as an obstacle and avoids any collisions with it. Grasp poses are generated by fitting geometric primitives like cylinders, spheres, and boxes to depth measurements of the scene. For this strategy, the palm of the hand is aligned with the support surface. The hand is then positioned as low as possible above the support surface so that the fingers do not contact the surface during closing. During execution, the wrist pose is position controlled and does not change while the fingers are closed.

The second strategy is a surface-constrained grasp (Sec. 5.1.1). It uses force control to establish contact of the fingertips with the support surface and proceeds to slide the fingers along the surface during closing, maintaining constant contact force by compliantly repositioning the wrist.

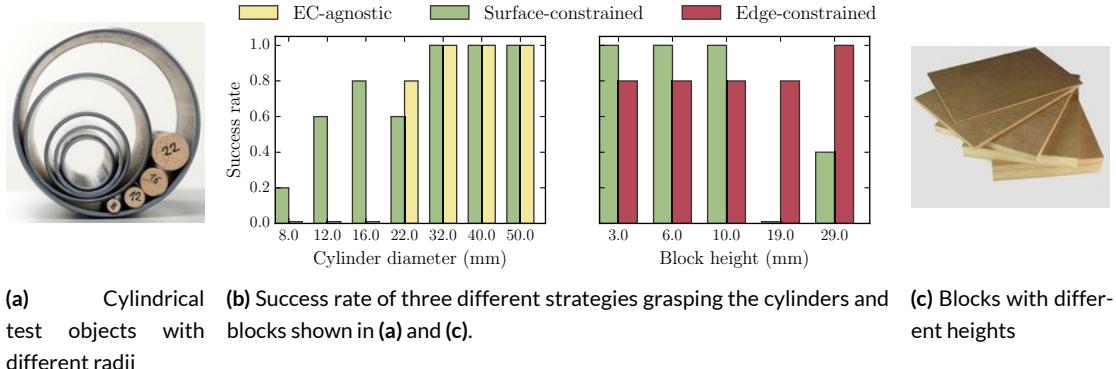


Figure 5.7: The plots compare a surface-constrained, edge-constrained, and EC-agnostic grasp strategy. The left plot shows that exploiting an environmental constraint improves success rate for grasping cylindrical objects in comparison to an EC-agnostic strategy. The right plot demonstrates that having a repertoire of different EC strategies is beneficial.

The main difference between the two compared strategies is that the first only attempts to come as close as possible to the surface using RGB-D information about the scene, whereas the second maintains physical contact with the surface throughout the whole grasp.

To evaluate the strategies we placed seven different sized cylinders (Fig. 5.7a) on a table in front of a 7-DOF WAM equipped with a force-torque sensor and a Barrett Hand BH-262. For each strategy-object pair we conducted five trials, resulting in a total of 70 grasps.

The left plot in Fig. 5.7b shows grasp success as a function of cylinder diameter. While big cylinders could be grasped reliably with both strategies, the grasp of smaller cylinders only succeeded with the surface-constrained strategy. The constant-wrist-pose strategy causes the finger tips to hover slightly above the surface when contact with the object is made, due to their circular trajectory during hand closure. This insufficient exploitation of the surface constraint leads to a reduced success rate for small-sized objects. In contrast, the surface-constrained grasp uses the surface at all times to position fingertips as close to the table as possible. Grasp success is not perfect though, as the cylinders can easily roll off the fingertips. This experiment shows that EC exploitation can lead to more robust grasping than EC-agnostic behaviors.

5.6.2 COMPLEMENTARITY OF DIFFERENT ECE STRATEGIES

We want to show that there are multiple environmental constraints that can be exploited. To achieve good grasping performance in a variety of settings and for diverse objects, it is necessary to employ the most appropriate strategy. The multitude of available constraints

also necessitates perceptual capabilities to distinguish situations in which one strategy should be preferred over the other. To demonstrate this point, we compared the surface-constrained grasp to the edge-constrained grasping strategy.

We evaluated both strategies for five different sized blocks (see Fig. 5.7c) placed on a table in front of the robot. For all blocks, the edge-constrained grasp achieves reliable performance (see Fig. 5.7b), whereas the surface-constrained strategy is only successful for flat blocks.

The edge-constrained strategy is less sensitive to variations in the size and weight of the blocks. The flat and wide shape of the blocks enables the robot to move parts of them over the edge, creating the opportunity to perform a more reliable grasp on the shorter side of the block. Failure cases for the edge-constrained strategy included wrong tracking during the visual servoing, missing object contact during sliding, and premature thumb closing.

The surface-constrained grasp succeeds when the fingernails jam against one of the block's sharp edges. This is achieved consistently for the smaller blocks. For taller blocks, the fingernails do not contact the object, leading to slip and grasp failure. In a few cases, however, the nails caught the object just before slipping out of the hand. While these cases are counted as grasp success in our experiments, one should note that the intended grasp was not achieved. Success must be attributed to coincidence and the design of the finger nails.

The experiment demonstrates that different ways of exploiting environmental constraints succeed under different conditions. It also shows that the success of exploiting environmental constraints depends on object characteristics in non-trivial ways. It is therefore desirable to employ a variety of grasp strategies for which the conditions of success have been characterized. Perceptual skills then must classify environments according to which of the strategies' conditions of success are met best.

5.6.3 ROBUSTNESS W.R.T. VARIATIONS OF ENVIRONMENTAL CONSTRAINT

To be able to robustly execute a constraint exploiting action, it is also necessary for the action to tolerate uncertainties in the placement of constraints. In the third experiment, we varied the angle between the two surfaces exploited as environmental constraints during the slide-to-wall grasp. While most walls are vertical, some are not, such as the walls of a bowl. The grasp sequence was constructed using a wall at 60° angle. During the experiment, the wall angle was varied between $\alpha = 40^\circ$ and $\alpha = 90^\circ$, in 10 degree increments. After that, the two interesting border regions were identified and two additional angles were tested to increase resolution. The results are shown in Fig. 5.8.

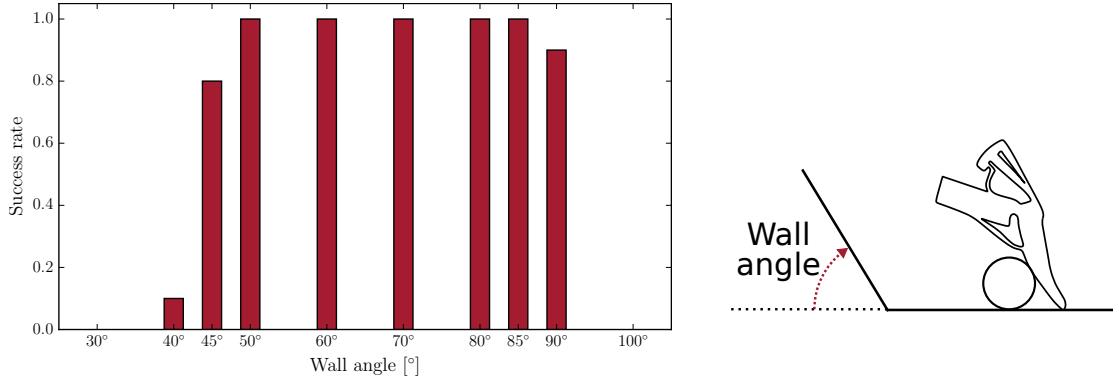


Figure 5.8: The plot shows the success rate of a wall-constrained grasp under varying wall angles relative to the horizontal table surface. Success rate is constantly high between 50° and 85° , showing the generality of the wall-constrained grasp strategy.

The grasp can be successfully executed without any adaptation of the actuation, in a large range of wall orientations, from approx. 45° to 90° . Larger angles could not be tested, because the wrist collided with the wall constraint during the slide motion. Therefore, angles larger than $\alpha = 90^\circ$ can be considered completely unsuccessful.

5.6.4 EVALUATION OF VARIOUS REAL-WORLD SCENES

We use 30 indoor scenes from a clutter data set (Karpathy et al., 2013) to evaluate the applicability of our planner. They depict office desks, book shelves, and kitchen environments and contain a significant amount of clutter. The scenes are encoded as polygonal meshes but we feed our algorithm with a ray-traced depth image of a single viewpoint in which most of the mesh is visible. For each scene, we position a box-like object at a random location that we assume to be statically stable. In total, our planner generated 218 grasping sequences, with at least one sequence per scene (average of ~ 3.6 per scene). Among the planned strategies most ended in an edge-constrained grasp (64%), followed by surface-constrained (27%), and wall-constrained (9%) ones. The most prominent problems the planner encountered were wrongly recognized edge-constraint grasps. To assess the quality of the generated plans, they were visually inspected and categorized according to their feasibility. In total, 62% of the plans were deemed feasible. The majority of the infeasible plans ended in edge-constrained grasps (92%). This had two reasons: Because of the nature of the data set, point clouds were often incomplete, increasing the occurrence of shadow edges (see Fig. 5.9). The data set also contains a significant amount of clutter making support surfaces unnavigable. Example plans showing the different failure cases and successes can be seen in Fig. 5.9. The results indicate

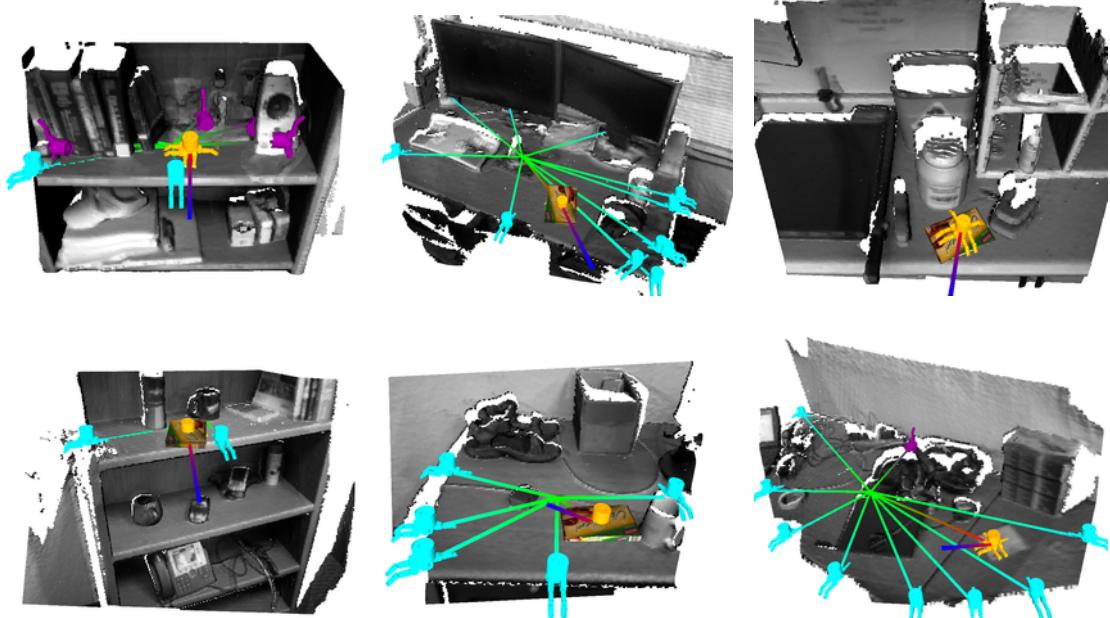


Figure 5.9: We evaluated the ECE planning algorithm in human environments from a clutter dataset (Karpathy et al., 2013). The resulting grasp plans are overlaid in color. Lines connect consecutive exploitations. The final prehensile exploitation for each sequence is plotted with a hand model (orange: surface-constrained, cyan: edge-constrained, purple: wall-constrained grasp). The data set revealed a lot of shadow edges which produced false positives among the edge-constrained grasps.

that the environmental constraints are general features that can be exploited for grasping in a wide variety of human environments.

5.6.5 EXECUTION OF PLANS ON A REAL ROBOT

To further evaluate the feasibility of the plans, we executed some of them on a real robotic platform. We used a Barrett WAM, an ATI Industrial Automation multi-axis force-torque sensor, and a Barrett Hand BH-262 with four DOF. We chose a scenario in which multiple grasping strategies would be possible. It contained a banana placed onto a table surface with a big electronic amplifier next to it (Fig. 5.10a). The scene was measured with an Asus Xtion Live depth sensor at QVGA resolution from a single static viewpoint. Color information is not used at any stage of the algorithm. Location and dimensions of a bounding box describing the banana were given to the algorithm.

Fig. 5.10b-5.10d show the extracted EC exploitations for the non-prehensile actions. Slidable surfaces were found on the table, at the side panels of the electronic amplifier, at the curtain and on top of the robot base. Fig. 5.10e-5.10g display the detected EC exploitations

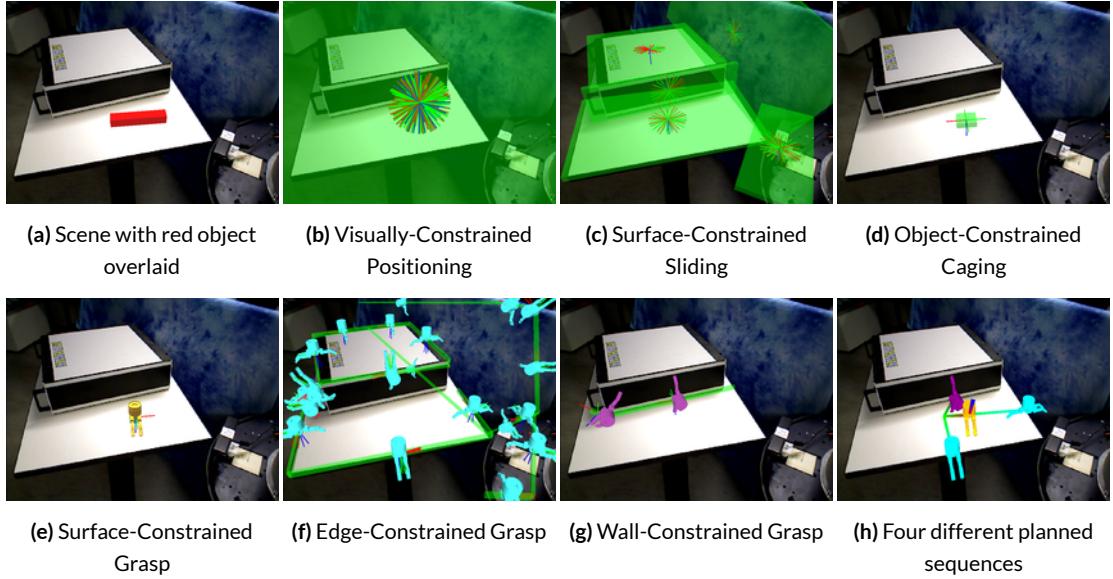


Figure 5.10: Experimental banana scene: For each of the six EC exploitations, the extracted spatial properties obb^{hand} and R^{hand} are shown. The final planned grasping sequences are shown in panel (h).

that refer to grasping actions. Two possible wall-constrained grasps were found between the table and the electronic amplifier 5.10g. Much more false positives were among the recognized edge-constrained grasps 5.10f: E.g. the lower part of the curtain was shadowed by the table and not a real edge due to depth discontinuities. In total, 28 EC exploitations were found. Their connectivity is depicted in the graph in Fig. 5.11. The graph also shows the four paths the algorithm finally found from the current unconstrained hand pose to one of the three types of prehensile EC exploitations. Fig. 5.10h visualizes the four sequences in the scene.

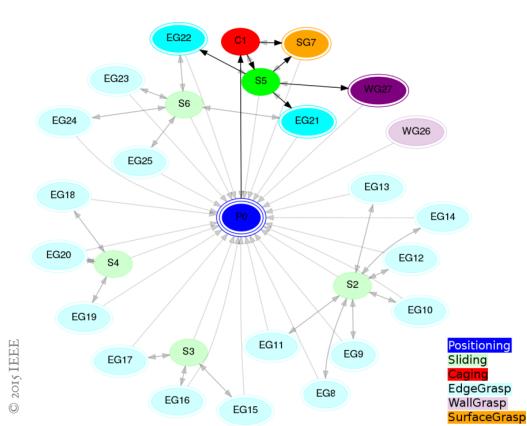
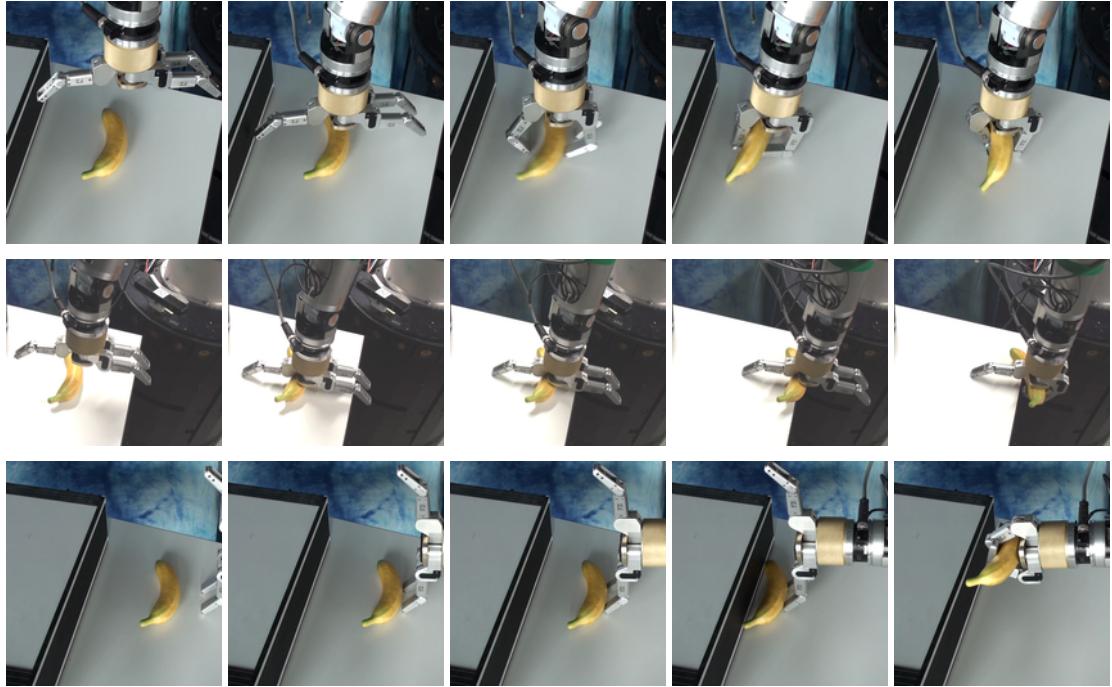


Figure 5.11: Graph showing the extracted EC exploitations and connectivity in the example scene 5.10. Nodes are color-coded according to EC exploitation (see Fig. 5.10). Framed nodes are goal nodes, start node is double-framed.



Video Figure 5.1: The three rows show sequences of executing three plans that were found in the banana scene shown in Fig. 5.10. First to last row: surface-constrained, edge-constrained, and wall-constrained grasping strategies executed with the Barrett Hand 262. [<https://youtu.be/Va000JS9bx0>]

For execution, the sequences are converted to multiple hybrid position/force controllers. Switching between them is governed by contact with a surface. Instances of the executed sequences are shown in Video Fig. 5.1.

5.6.6 LIMITATIONS

Though our method proves to be a powerful way of generating robust grasping behavior, there are limitations that require future work. As mentioned earlier, a planned sequence does never contain multiple contact-making/breaking events between hand and object contact establishing phases. For most grasping strategies this is a reasonable assumption. Additionally, the shape of the object is only represented by a bounding box, more complex kinematic relations between object and environment such as rolling contacts are missing. Presently, the algorithm does not use any intrinsic object properties during planning (e.g. friction, mass).

5.7 RELATED WORK

The idea of exploiting the environment and using contact via compliance can at least be traced back to the influential approach by Lozano-Pérez et al. (1984). This work introduced the central concept of the *pre-image*—a description of all positions that reach the goal given the same action—to enable the generation of fine-motion plans. Later, this idea was extended to the concept of *backprojections*, representing a weaker form of a pre-image, by separating goal reachability from goal recognizability (Erdmann, 1986). By restricting plans to temporal termination predicates of motion primitives, manipulation tasks can even be solved in an open-loop, sensorless fashion (Erdmann and Mason, 1988). Subsequently, the binary representation of a backprojection was given a probabilistic upgrade (Brost and Christiansen, 1996). These approaches depend on exact geometric models of the environment.

The geometric reasoning we apply is similar to classical work in assembly planning (Wilson and Latombe, 1994). The way we define pose constraints is similar to the task space regions (Berenson et al., 2011) used for motion planning. And the concept of contact-state graphs (Ji and Xiao, 1999) is reflected in our representation of environmental constraint transitions. However, in addition to contact states, our algorithm requires spatial information derived from sensor data. And again, in contrast to the mentioned approaches, we do so entirely based on sensor information.

Closely related to environmental constraint exploitations are *guarded moves*. A guarded move is “a move until some expected sensory event occurs” (Will and Grossman, 1975). Plans of guarded moves can include branches based on sensor events to compensate for uncertainty in world modeling (Finkemeyer et al., 2005). Our work includes this concept but in addition is concerned with the automated generation of such plans from sensor data.

Pre-grasp manipulation refers to the contact-driven modification of the environment to facilitate a subsequent manipulation action. These actions involve, for example, rotations due to payload limits (Chang et al., 2010), or sliding flat objects on table surfaces (King et al., 2013, Kappler et al., 2010). All these works realized specific, pre-programmed actions. In addition, pushing or sweeping can be considered as pre-grasp action (Dogar and Srinivasa, 2012). There, the environment is designed to present challenges to the planner, rather than opportunities.

Toussaint et al. provide an interesting approach to formalize the advantages of contact exploitation (Toussaint et al., 2014). They optimize plans so that they minimize uncertainty by contact with the environment. In our approach, the assumption that contact during manip-

ulation is beneficial is directly encoded by focusing on contact-based actions.

5.8 CONCLUSION

We presented a grasp planning algorithm to synthesize and execute grasping strategies that exploit environmental constraints. Recent results from the grasping literature lead to the conclusion that such exploitation plays an important role in achieving robust grasping performance. Our grasp planner leverages this insight and sequences constraint exploitations into versatile and robust grasp plans.

The algorithm tightly couples planning, perception, and control, thereby enabling grasp planning from real-world sensor data in the absence of prior information about the world. We demonstrated the effectiveness of the planner in experiments on a real robot platform and illustrated the generality of the planner by generating grasp plans in a great variety of environments. We believe that the exploitation of environmental constraints is a promising route leading towards robust grasping and manipulation with weak a priori object and world models.*

CONTACT-EXPLOITING RRT vs. ECE PLANNING

Both algorithms presented in Chapter 4 and 5 generate behaviors that make explicit use of the environment. But they differ in multiple regards.

In CERRT contact-rich motions *implicitly* emerge as a result of assuming that contact sensing is much more reliable than position sensing. Choosing actions such as bumping into things will reduce uncertainty about the state of the system. In contrast, ECE planning does not represent uncertainty. Instead it assumes that contact-based motions are in general beneficial and sequences them *explicitly*. Although the uncertainty-driven exploitation of contact done by CERRT seems favorable (contact only needs to happen when *necessary*), can all cases of environmental contact exploitation easily be modeled by a reduction of uncertainty?

This seems especially tricky when looking at object manipulation and grasping. The edge-constrained grasp for example allows the hand to generate contact forces underneath the object, as soon as it protrudes the edge. These contact locations were inaccessible before. Modeling this behavior through reduction of object pose or hand pose uncertainty is not possible. Another example is the wall-constrained grasp. The reaction force created by the contact between wall and object is used to push the object onto the palm. Again, the goal of generating forces to attain force-closure cannot be modeled as a reduction of uncertainty in the configu-

*The code of the planner is available at https://github.com/SoMa-Project/ec_grasp_planner.

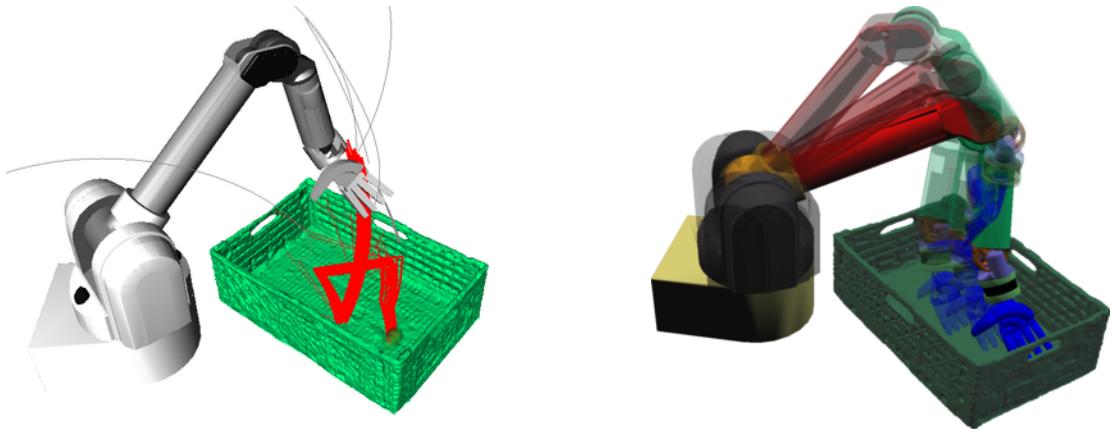


Figure 5.12: The images show the solutions found by CERRT (left) and the ECE planning algorithm (right) for reaching into the green box. ECE planning times are shorter and less variable compared to CERRT.

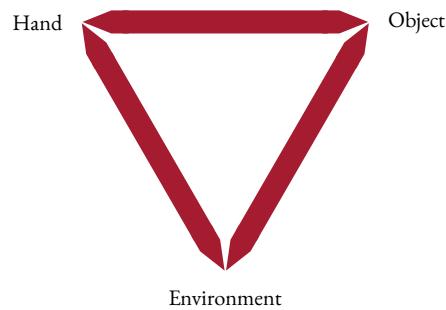
ration of object or robot. It is therefore not trivial to extend CERRT to the scenarios which ECE planning focuses on.

The ECE planning algorithm discretizes the sensor input to find contact surfaces, edges, etc. which constitute the state and action space. Depending on the parameters that any discretization depends on, it might fail to identify certain contact opportunities or overestimates them. CERRT does not rely on any a priori discretization of the state space and therefore does not suffer from any wrong discretization.

The search space that CERRT needs to explore is continuous and grows with the dimensionality of the system state. Due to the discretization into contact opportunities, ECE planning has to deal with a much smaller search space. It scales quadratically in the number of contact exploitations extracted from sensor input. The difference in planning time can also be observed when applying both algorithms to the problem presented in Sec. 4.2.2. In this problem, a path to a goal configuration inside a box needs to be found for a 7-DOF robot arm (see Fig. 5.12). Planning on a standard desktop computer (2.2 GHz) requires on average 50.4 s with CERRT and 2.75 s using the ECE algorithm (20 trials). More importantly, the standard deviation of the planning time is 77.6 s in case of CERRT and 0.3 s for ECE planning. Note that the shorter planning times for the ECE algorithm are paid with a lack of probabilistic completeness.

Finally, the algorithms differ in the assumed inputs. CERRT uses a geometric world model without making any statements about the feasibility of acquiring such a model. On the other hand, ECE planning is based on depth sensor measurements and focuses on the entire pipeline from sensing to acting.

PART III



Interactions Between
Hand, Object and Environment

MOTIVATION

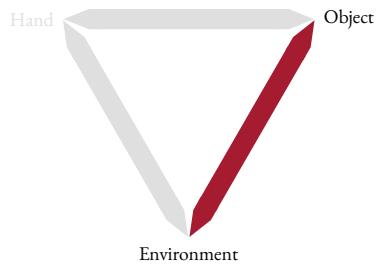
In the first two parts of this thesis we investigated how hand adaptability and the stiffness of the environment can be exploited for grasping. These investigations focused exclusively either on the relationship between hand and object or between hand and environment. But ultimately we would like to know how the triad – hand, object, and environment – can be taken into account to create robust grasping. What kinds of objects require which type of EC exploiting strategy?

We get a hint at what type of objects benefit from exploiting the environment when reconsidering the teleoperation experiment with the Allegro Hand from Sec. 2.1.2. In this experiment 35 different objects were grasped from a table. Due to the large and chunky fingertips of the Allegro Hand the operator had often difficulties picking up small objects. We introduced a second condition in which the robot’s workspace also included the edge of the table which was unreachable in the first condition. Again, we conducted 175 grasping trials (five per object). This time the operator used the table edge to apply an edge-constrained grasps. As a result grasp success increased w.r.t. to the first condition. This was true for flat objects such as the credit card ($0/5$ successes $\rightarrow 2/5$), a small book ($1/5 \rightarrow 2/5$) and a CD ($0/5 \rightarrow 2/5$).

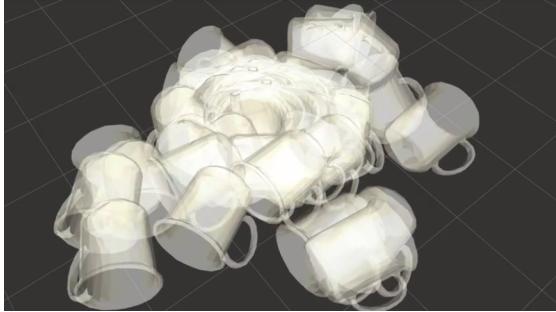
This shows that object height might already be a powerful indicator of selecting a particular ECE strategy. We will confirm this insight with a learning method presented in Chapter 7.

INTERMEZZO: INTERACTIONS BETWEEN OBJECT AND ENVIRONMENT

Before we get into detail about the relationship between hand, object, and environment we would like to briefly study some aspects about the interactions that only relate to object and environment. Since this is only a short investigation, we do not devote an entire Part of the thesis to this topic. However, we think it further strengthens the case for ECE because it shows how geometric variety is reduced through contact between object and environment.



Objects in the real-world do not occur in arbitrary places and configurations. They are embedded into a task-specific and physical context. A mug is usually standing upright to keep the liquid it is containing. The mug also has to obey the law of gravity, so it will usually be located on a support surface instead of hovering in free space. These kinds of biases reduce



Video Figure 5.2: Simulation of the stable poses of a mug w.r.t. a surface constraint. [<https://youtu.be/8VNI0qsyXj4>]

the variance of problem scenarios we have to face and be capable to solve. In the same spirit, our planner of EC-based grasping strategies exploits and benefits from such regularities.

It has been shown that humans exploit visual biases constantly. Kaiser et al. (2014) showed that identifying an object in a cluttered display is simplified if the distracting items are arranged in commonly experienced configurations, e.g. a lamp above a table vs. below the table. A study by Stansbury et al. (2013) hints to the possibility that natural scene representations in the human brain are guided by the co-occurrence statistics of objects in the world.

We conduct a small simulation experiment to quantify the effect of environmental constraints on the configuration and geometry of objects. We use 132 object models from the KIT dataset (Kasper et al., 2012), sample uniform random orientations and simulate the effect of gravity using a physics engine (Coumans, 2015). The objects are initialized slightly above a static plane. Video Fig. 5.2 shows an example of 100 simulated mugs. We collect the equilibrium poses in 13200 trials (100 per object). To evaluate the potential reduction in pose diversity, we first compare the entropy between the distribution of initial and final/stable orientations. We calculate the discrete entropy by first discretizing $SO(3)$ into a uniform grid with 576 bins. Therefore, the entropy of the uniformly distributed initial orientations is $\log_2 576 = 9.12$ bit. In contrast, the entropy of the distribution of stable orientations is between 4.3 bit (for cylindrical objects like a can of sauerkraut or a wine glass) and 3.1 bit (for objects with fewer stable orientations like a tube of toothpaste). To compare the effect over all different object geometries, we need a coordinate-free measure. Fig. 5.13a shows the distribution of angles that describe the shortest path between two random samples from each distribution of orientations. We can see that the histograms of surface-constrained and wall-constrained object poses exhibit stronger modes which reflect the fact that object orientations cluster in both conditions.

For grasping it is reasonable not to look only at the orientations but also at the actual geometry of the objects, since this affects the resulting contacts. The equilibrium orientations

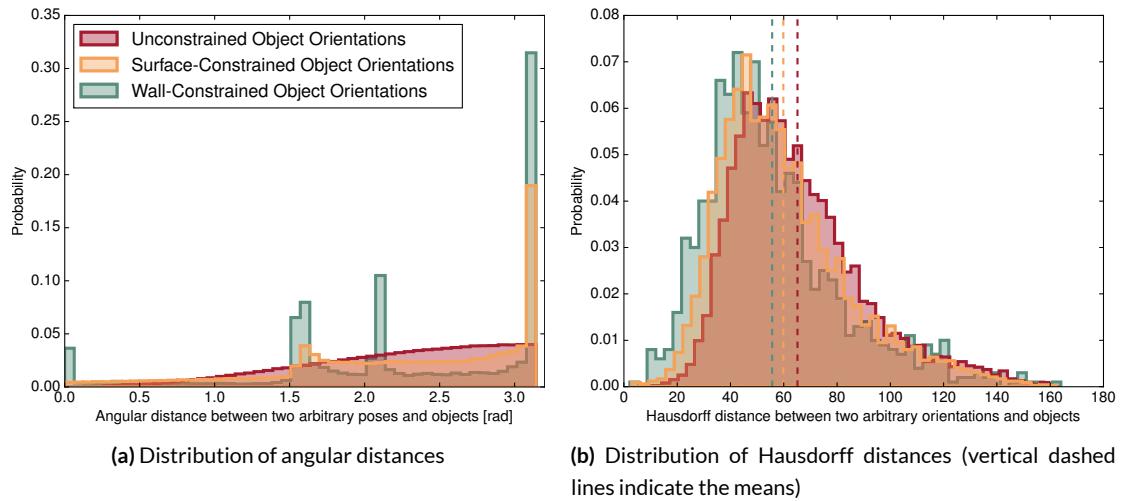


Figure 5.13: Effect of interactions between objects and environment on object pose and geometry.

of a sphere are not reduced due to contact with surfaces. However, since the sphere is fully symmetric grasping performance is not affected by the sphere’s orientation. To include the geometry, we compare the object orientations based on the Hausdorff distance between the oriented triangular meshes (Cignoni et al., 1998). The Hausdorff distance is a common tool to measure the difference between two shapes and depends on the largest distance of all shortest distances between the two sets of points that represent the shapes. Fig. 5.13b shows the distribution of Hausdorff distances in each condition: unconstrained, surface-constrained, and wall-constrained object orientations. Again, we observe a reduction in diversity: shapes become more similar and thus grasping becomes easier.

Note that our study only focuses on the interactions due to the geometry of object and environment. A further reduction in the variability of problem scenarios can be expected when considering the object’s function (e.g. a mug will most often stand upright to contain a liquid).

CONTRIBUTIONS

In this part we examine how the benefits of hand adaptability and environmental stiffness can be tied together to improve grasping. Based on the planning methods of the previous chapters we will devise new algorithms to grasp arbitrary objects. The main contributions in this part are:

- Two methods to adapt grasping strategies to the environment, a model-based method

for stiff hands (Sec. 6.1) and a learning-based method for soft end-effectors (Sec. 6.2).

- A predictive model to select the best ECEs for sensory inputs of novel objects. This includes an evaluation of different geometric features and more than 600 collected ECEs in the real world (Sec. 7.1).
- A formalization of the ECE selection problem as a contextual multi-armed bandit problem and an evaluation of multiple common exploration schemes (Sec. 7.2).
- A trial-and-error-based algorithm to learn more general manipulation policies for a soft hand using human demonstrations (Chapter 8).

OUTLINE

We will start by first looking into ways of adapting our previously presented grasp planners to different environments and objects (Chapter 6). Rather than locally adapting strategies, we pursue the idea of globally selecting between different EC-based grasping strategies in Chapter 7. This is done by learning a classifier of grasp success given object features.

Finally, we will go beyond grasping and take a look at manipulation tasks for soft hands in Chapter 8. This requires less structured motion descriptions and poses challenges for modeling the soft hand's adaptability.

6

Combining Properties of Object and Environment to Adapt Grasp Strategies

THINK ABOUT A BOTTLE lying flat on a table. Even if you see it from the front, your palm will most likely approach it from above, parallel to the table surface, with the fingers touching the table while they bend around the object. This grasping strategy clearly takes the specific geometry of both object and environment into account.

In this chapter we will examine the complex interactions between hand, object and environment by making only *local* changes to the previously introduced grasp strategies. These local changes are derived to improve grasp success. In terms of our funnel view (Sec. 2.3) it means that we are searching for grasp parameters such as the pre-grasp pose which create a wide entrance of the final funnel represented by the hand's compliance mode.

We will focus on the surface-constrained and wall-constrained grasp. But as we will show, the proposed algorithms can be extended to other EC-based grasp types. We will present two algorithms that accomplish the adaptation by following two very different paradigms:

1. A model-based approach which balances perceived information and exploitation of the environment (based on the algorithm presented in chapter 4).
2. A learning-based approach using black-box optimization and trial-and-error.

We will evaluate both algorithms in multiple real-world experiments and assess their strengths and weaknesses.

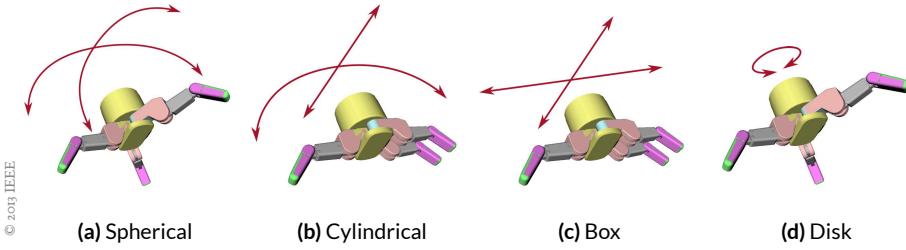


Figure 6.1: The four different pre-shapes and their associated curvilinear coordinates (red arrows) that make up the pre-grasp manifolds. Cylindrical and spherical manifolds are represented by cylindrical and spherical coordinates; the box manifold exhibits one translational degree of freedom while the disk manifold is defined about one axis of rotation. These manifolds are used to further refine the pose of a pre-grasp.

6.1 MODEL-BASED ADAPTATION OF SURFACE-CONSTRAINED GRASPS

The surface-constrained grasp is probably the one that is most often applicable in the real world, since objects are usually placed on a planar support surface. In this section we present an algorithm that combines object shape properties with environmental properties to plan surface-constrained grasps. The environment constraints the grasp strategy in a few dimensions but not all, while the object geometry does the same. The key idea is to find a grasp that satisfies as many constraints imposed by the geometry of environment and object.

We will base our method on the shape features that exploit hand adaptability (see Sec. 3.2). The importance of satisfying constraints depends on the visibility of the object shape. If large parts are occluded our method relies more on satisfying the geometric constraints of the object than the environment. We demonstrate the beneficial effect of considering object/environment shape match in 460 real-world grasping trials with 23 objects.

6.1.1 FROM SINGLE PRE-GRASPS TO PRE-GRASP MANIFOLDS

In Chapter 3 we introduced the notion of a pre-grasp which defines the hand pose before closing the fingers. We calculated the pre-grasp based on a simplification of object geometry. The simplification was based on a fitting scheme that was trying to match the object shape with cylinders, spheres, boxes and disks (see Sec. 3.2). The cylinder gives rise to two pre-grasps, both parallel to the cylinder's axis with the thumb pointing left or right. The box results in four possible poses, two along each pair of parallel edges. Disk and sphere create pre-grasps whose approach vector points towards their centers.

Instead of single poses we define bounded manifolds by exploiting the symmetry of the shape descriptors. A pre-grasp manifold is defined by a curvilinear coordinate system depend-

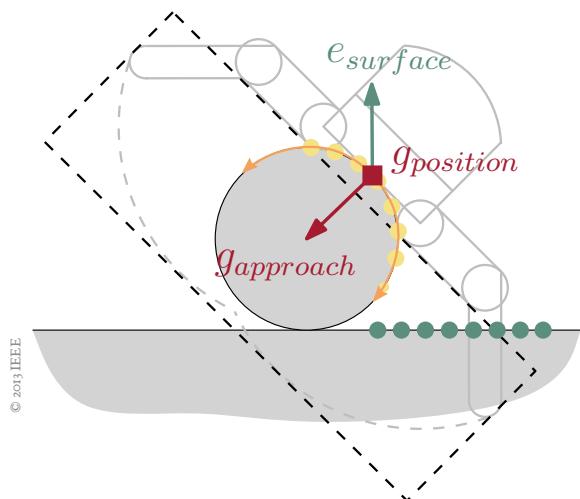


Figure 6.2: A 2D explanation of the environmental adaptation scheme is shown. Assume we have given range sensor readings (bold dots), a segmentation (dot's colors), a pre-grasp (grey two-fingered hand model with approach vector g_{approach} and position g_{position}) and its associated 1D manifold (orange half circle). The mean surface normal e_{surface} of the environment is the average over all measured surface normals that fall within the expected closing region (black dashed rectangle) and are not part of the segment to be grasped. We choose the pre-grasp from the region that minimizes a cost which includes the difference between $-g_{\text{approach}}$ and e_{surface} and the distance of g_{position} to the region's origin. See text for details.

ing on the grasp's pre-shape. In the cylindrical case the pre-grasp region is given by cylindrical coordinates with a fixed radius. The analogs for all other strategies are depicted in Fig. 6.1. Going from single pre-grasp poses to whole manifolds allows us to satisfy environmental constraints as we will show next.

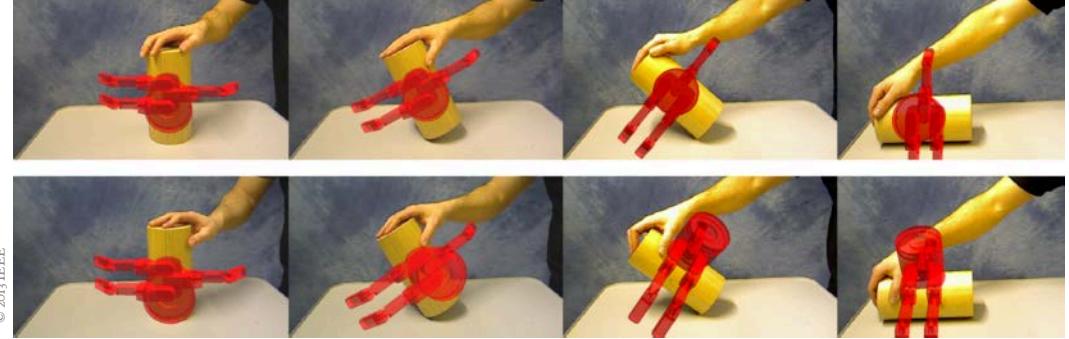
6.1.2 A SIMPLE GRASP MODEL FOR THE ADAPTATION BETWEEN HAND AND ENVIRONMENT

We anticipate the interactions between the hand's closing motion and its immediate environment by refining the pre-grasp pose within its respective pre-grasp manifold. The surface constraint is extracted by analyzing the depth measurements that fall within the closing volume of the hand: The number of points intersecting this volume g_{points} , the maximum possible number $g_{\text{maxpoints}}$, and their mean surface normal e_{surface} are computed.

We refine the pre-grasp pose by casting it as an bounded optimization problem. The objective function is given as the weighted sum of the orientation error (between grasp approach and surface normal) and the distance to the origin of the pre-grasp region. We are trying to find a minimizer of this function:

$$\underset{(g_{\text{position}}, g_{\text{approach}})}{\operatorname{argmin}} \left(w \frac{g_{\text{points}}}{g_{\text{maxpoints}}} \langle -g_{\text{approach}}, e_{\text{surface}} \rangle + (1-w)d(g_{\text{position}}, 0) \right),$$

where $d(\cdot, \cdot)$ is a distance measure depending on the curvilinear coordinates used, and w is a weight that balances the amount of constraint satisfaction with the environment versus preferring known zones inside the pre-grasp manifold. If $w = 0$ the surface constraint (if



Video Figure 6.1: These image sequences show how our grasp adaptation includes sensor data about the environment. **Upper row:** Shape match with the environment is ignored. The cylinder is always grasped from the front. **Lower row:** The closer the cylinder is to the table surface the more the pre-grasp pose tries to match with it. [<https://youtu.be/9fDFCaQ8WYE>]

present) is ignored and grasps only depend on the object shape. In contrast, if $w = 1$ the pre-grasp will be aligned with the surface, ignoring the shape of the object. Note that the value of w does not only reflect our intuition about whether the grasp dynamics are more strongly influenced by contact with the environment or the object. It also reflects how much we believe in the symmetry of the object, since we usually do not see the back part of an object.

To model the accessibility of a pre-grasp pose we add an additional term which depends on the free space along the approach vector. Therefore we sweep the hand volume through the depth image and count the colliding points. Apart from penalizing hard to reach grasps, it also rejects false positives due to concave shapes. Fig. 6.2 depicts a graphical explanation of the optimization problem. Video Fig. 6.1 shows an example for a cylindrical pre-grasp optimization on real sensor data. Note that optimization is done in real-time.

Finally, we track grasp hypotheses over time by associating the most similar ones in successive time steps. Similarity is based on pre-grasp configuration and pose. During tracking we filter out hypothesis that do not appear with a frequency of at least ~ 3 Hz. This eliminates unstable hypotheses caused by sensor noise.

All pre-grasp descriptors can be computed in parallel since they do not influence each other. Because we are not interested in subtle geometric features we can rely on a rather coarse depth image resolution of 320×240 . This additionally speeds up our processing pipeline, resulting in ~ 5.3 Hz on a standard desktop computer at 2.2 GHz using a single-threaded implementation. Roughly 70% of the load is produced by fitting the primitives and normal estimation.



Figure 6.3: The test objects (left) and the small cluttered scene (right) used in the experiments.

6.1.3 EXPERIMENTAL EVALUATION

The goal of our experiments is to examine how well our adaptation scheme works to find surface-constrained grasps for a variety of different objects. Although it might sound reasonable to compare our adaptation scheme against the same approach without adapting the pre-grasp pose, we found that the results differ significantly and the baseline is unfair. A baseline that completely ignores the surroundings will often collide with the table, failing to grasp the object.

Instead we test our approach in two other experiments: In the first one objects are presented in isolation on a table surface and we focus on precision and recall performance. The second experiment contains a small cluttered scene, where environmental constraints are more complex than a simple surface due to multiple objects. We would like to test how well the surface assumption generalizes to less structured surroundings.

In both experiments we used a 6-DOF Unimation PUMA 560 with a 4-DOF Barrett Hand BH8-262. The robot was equipped with an Asus Xtion Live depth sensor, mounted on the wrist of the manipulator.

GRASPING OBJECTS UNDER SURFACE CONSTRAINTS

We placed 23 different objects (Fig. 6.3) on a table in front of the robot. A grasping trial consisted of one object being observed from (maximally) five different pre-recorded viewpoints as shown in Fig. 6.4. The robot observed the object for 3 s from a single view point. During that time it chose the most promising surface-constrained grasp strategy. If no pre-grasp confidence exceeded a pre-defined threshold, no grasp was executed and the next view point was considered. For each object the four different surface-constrained strategies were evaluated independently, resulting in a total of 460 trials.

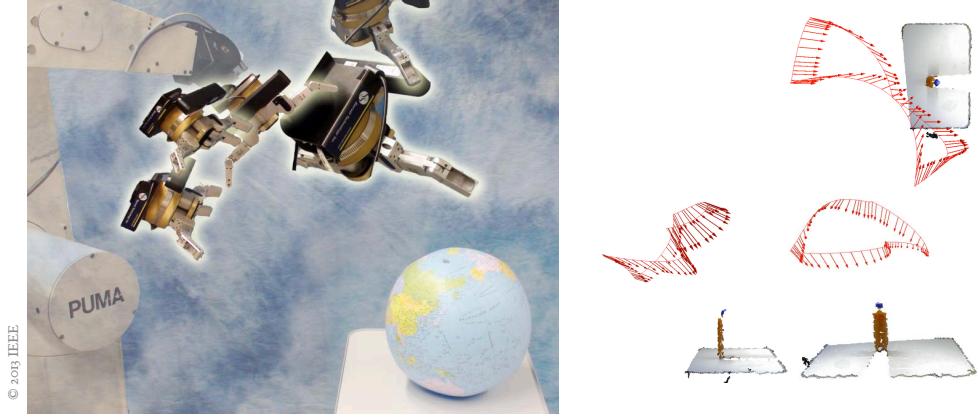


Figure 6.4: The experimental setup for the simple environmental constraints: The sensor observes the object from the five different perspectives shown.

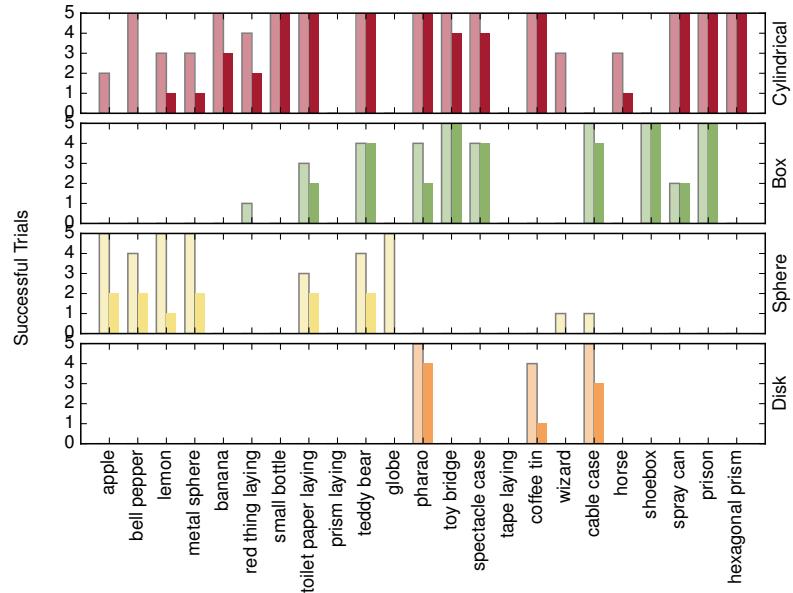


Figure 6.5: Surface-constrained grasping performance for each of the four pre-grasp strategies. The light-colored bar indicates a predicted grasp by the algorithm. The dark-colored one is the resulting grasp success.

The resulting detection and success rates are depicted in 6.5. Looking at the max strategy, our method detects a grasp in 88% of all cases. If we look at the recall performances for each pre-shape strategy separately, it matches our intuition that most objects are cylindrical while the disk strategy is detected seldom (recall of cylindrical strategy: 69%, box: 36%, sphere: 28%,

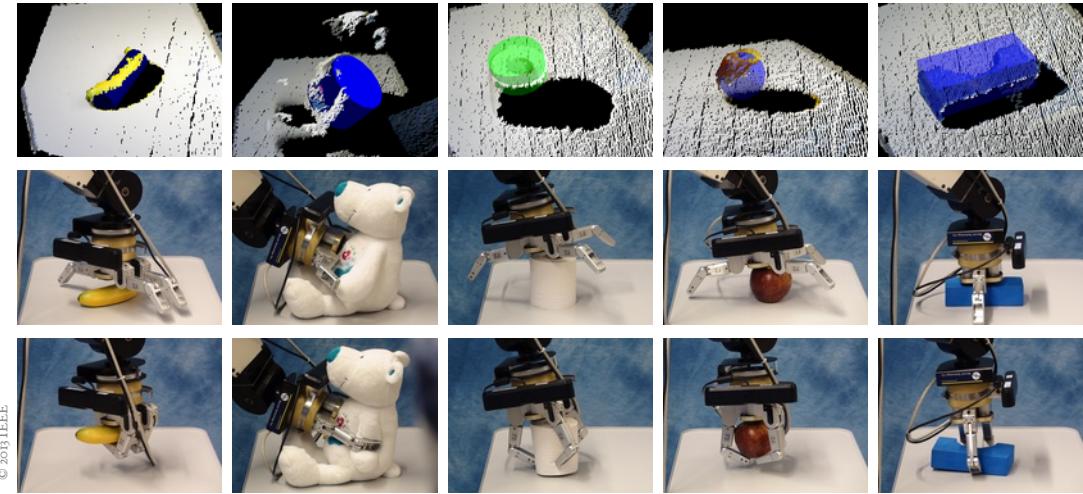


Figure 6.6: Some examples of planned surface-constrained grasps: The first row shows the fitted geometric models (from left to right: two cylinders, disk, sphere, box) within the point cloud perceived by the robot. The second and third row show the resulting pre-grasps and final grasps executed by the robot.

disk: 12%). More importantly, the success of the chosen strategy can be predicted most of the time. The precision of the max strategy is 72%. Among the individual strategies the box pre-grasp is the most reliable (precision: 88%), while the spherical one performs worst (33%).

The reason for the bad performance of the spherical pre-grasp is the following: The globe and wizard are close to prototypical spheres, but they exhibit low frictional surfaces which in combination with the aluminum cover of the Barrett Hand require very high contact forces to be grasped successfully. Another problem – especially with the spherical pre-shapes – was a pre-mature activation of the breakaway mechanism. Fingers stopped even when not in contact with object. Note though that only few objects cannot be detected by any of the proposed strategies: One such case is the flat laying tape which exhibits a disk-shaped top face which was not detected because of the low resolution of the sensor and the minimum distance it needs to maintain during sensing.

Some example grasps can be seen in Fig. 6.6. They show that the proposed strategies can successfully balance the exploitation of the environmental surface constraint with the shape match between hand and object.

We can also compare the results of this experiment directly with those obtained in Sec. 3.2.2: while recall is similar (88% vs. 92%), the precision of the grasps is now worse (72% vs. 94%). But remember that in this previous experiment we applied the same strategies to the scenario of objects in free space, without any environmental obstruction. Given the more realistic

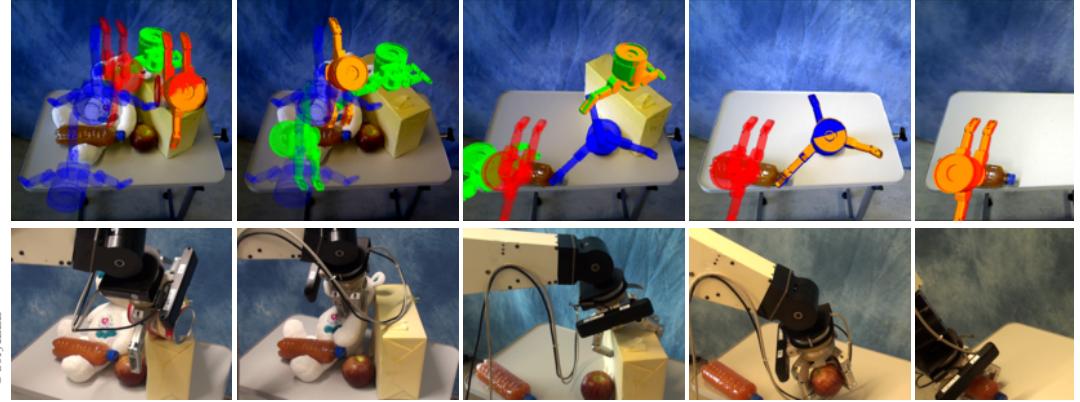


Figure 6.7: The five different decisions to empty the table are shown from left to right. The upper row shows the view of the sensor with the possible pre-grasp poses (green = box, blue = sphere, red = cylinder). Each time the robot chose the most promising pre-grasp (displayed in golden color) based on shape match with the object and environment. The lower row shows the grasps executed by the robot. All except the last were successful.

setting we are testing, this decrease in overall grasping performance could be expected.

SHAPE MATCH UNDER COMPLEX ENVIRONMENTAL CONSTRAINTS

In a final experiment, we wanted to test how well our methods scales with more complex scenes, in which multiple objects can also constrain each other. The promising result of an example with five objects is shown in Fig. 6.7. The robot observed the scene five times from the same view point, each time selecting the most promising grasp.

Note that our method has no notion of gravity or whether two grasps belong to the same object. Still, the accessibility criterion implicitly favors most of the time shape matches that are on top of each other. In this qualitative experiment the robot only failed grasping the last object, a bottle which was located close to the edge of the table. Overall the results for the cluttered scene are pointing in a promising direction.

6.2 ADAPTING STRATEGIES THROUGH LEARNING

The approach presented in the previous section hinges on a hand-crafted model. It requires knowledge about the physics and detailed features of the used end-effector. If this knowledge is unavailable or the kinematics and dynamics are less intuitive and a model cannot easily be hand-crafted (as in the case of a soft hand like the RBO Hand 2) we need a more general approach.

In this section we present an alternative method to adapt the parameters of a grasp strategy to increase grasp success. This method does not require any knowledge about the physics that happen due to interactions between hand, object and environment. It is based on the cross-entropy method (CEM by [Rubinstein \(1999\)](#)), a black box optimization approach. It might sound like an exaggeration to ignore all the insights we gained about the contact events that occur (even with a less intuitive/soft mechanism) and treat grasps as a completely unknown function. But remember that we will apply this optimization on a constrained parametrization, namely the grasp strategies provided by the ECE planner (Algorithm 5.4). Thus, we will effectively search the space of EC grasps only locally. We will show the impact of our method in experiments in simulation and in the real-world.

6.2.1 CROSS-ENTROPY METHOD FOR EC GRASPING

CEM is an evolutionary algorithm that is used to find the maximizer of an unknown function. It does not need access to the function's gradient, which allows it to be used for functions that are not differentiable or even continuous. We would like to maximize grasp success as a function of the parameters of a grasp strategy. Since grasp success is binary, our unknown function contains a lot of discontinuities. Hence, CEM is well suited for our problem.

We parameterize our EC-based grasping strategies the following way:

- **SURFACE-CONSTRAINED GRASP:** The parameter vector $\theta_{\text{surface}} \in \mathbb{R}^5$ contains: the initial finger inflation, the force that is used to press against the surface, the angle of attack of the hand relative to the surface, and the offsets in x and y on the surface relative to the object's centroid.
- **EDGE-CONSTRAINED GRASP:** The parameter vector $\theta_{\text{edge}} \in \mathbb{R}^6$ contains: the initial finger inflation, the force that is used during sliding to press against the surface, the

angle of attack of the hand relative to the surface, the offsets in x and y relative to the object's centroid before starting to slide, and the distance offset from the edge to start closing the fingers.

- **WALL-CONSTRAINED GRASP:** The parameter vector $\boldsymbol{\theta}_{\text{wall}} \in \mathbb{R}^6$ contains: the initial finger inflation, the force that is used during sliding to press against the surface, the angle of attack of the hand relative to the surface, the distance offset parallel to the wall and relative to the object's centroid, the velocity during sliding, and the force threshold that is used to start finger closing when reaching the wall.

Remember that all strategies are represented as hybrid automata (see Sec. 5.5). The parameters listed above influence not only the control modes but also the switch conditions connecting them.

CEM is an iterative process. In each iteration new potential maximizers are generated and evaluated. Only the best performing parameters survive and are used in the subsequent iteration to generate new candidates. This process is repeated until convergence or for a pre-defined number of iterations. We represent the grasp parameters $\boldsymbol{\theta}$ with a Gaussian distribution: $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}_\mu, \text{diag}(\boldsymbol{\theta}_{\sigma^2}))$, where $\text{diag}(\boldsymbol{\theta}_{\sigma^2})$ is a diagonal matrix representing the variance. Initially, we set the variances to high values indicating that we are not certain about those values. CEM draws N samples from this distribution and evaluates the associated grasp. The evaluation is either done in simulation or on a real robotic platform, labeling a grasp as success or failure. We keep the best $\lfloor 0.5N \rfloor$ parameters and fit a new Gaussian before initiating the next iteration. Algorithm 6.5 summarizes our approach.

Szita and Lörincz (2006) have shown that a noisy version of CEM outperforms the vanilla one. They inject noise into the sampling process but decrease it with every iteration. This way the exploratory behavior of the algorithm is intensified, avoiding pre-mature convergence to a local minimum. We do the same by adding the term $\max(5 - \frac{t}{10}, 0)$ to the variance when fitting the Gaussian in iteration t .

6.2.2 EXPERIMENTS

In our experiments we want to show that our approach converges and grasping performance improves. We will show this first in simulation, and later in a real world scenario. The experimental results also suggest that the parameters learned in simulation can be transferred to

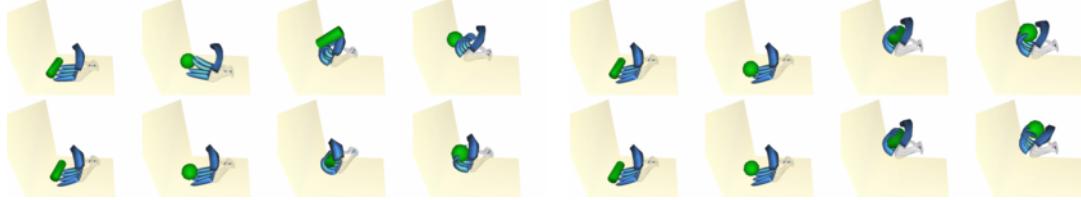
Algorithm 6.5 Cross-Entropy Method for EC-based grasping strategies

Input: $\mathbf{x}_{\text{hand}}, \mathbf{x}_{\text{object}}$
 Output: $\boldsymbol{\theta}$

$$\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}_\mu, \text{diag}(\boldsymbol{\theta}_{\sigma^2})) \quad \triangleright \text{Initialize parameters}$$

for $t = 1, 2, \dots$ do

- Take point cloud measurement $P_{\text{RGB-D}}$
- $S_\theta \leftarrow \text{ECE_PLANNING}(P_{\text{RGB-D}}, \mathbf{x}_{\text{hand}}, \mathbf{x}_{\text{object}}) \quad \triangleright \text{Algorithm 5.4}$
- Draw N samples $\boldsymbol{\theta}_i \sim \mathcal{N}(\boldsymbol{\theta}_\mu, \text{diag}(\boldsymbol{\theta}_{\sigma^2}))$
- Evaluate corresponding grasps and record success $r_i \in \{0, 1\}$
- $\mathcal{B} \leftarrow \text{Select the } \lfloor 0.5N \rfloor \text{ top samples of } \boldsymbol{\theta}_i$
- $\boldsymbol{\theta}_\mu \leftarrow \frac{1}{|\mathcal{B}|} \sum_{\mathbf{b} \in \mathcal{B}} \mathbf{b}$
- $\boldsymbol{\theta}_{\sigma^2} \leftarrow \frac{1}{|\mathcal{B}|} \sum_{\mathbf{b} \in \mathcal{B}} (\mathbf{b} - \boldsymbol{\theta}_\mu)^T (\mathbf{b} - \boldsymbol{\theta}_\mu) + \max(5 - \frac{t}{10}, 0)$



(a) Iteration 0: Initial and final visualizations of four exemplary wall grasps. Not all wrist orientations turn out to be successful when grasping using a wall constraint.
 (b) Iteration 6: After a few iterations the wrist orientation converges and $\sim 96\%$ of all grasps are successful.

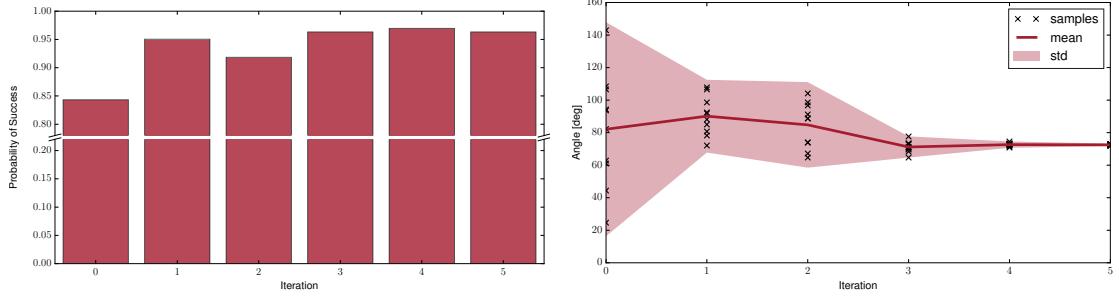
Video Figure 6.2: Cross-entropy method for optimizing the angle of attack of a wall-constrained grasp in simulation.
[\[https://youtu.be/zmsS0Bx-6x8\]](https://youtu.be/zmsS0Bx-6x8)

a real robotic system. We will focus on the wall-constrained grasping strategy, although our method is not limited to a specific type of grasp.

SIMULATION

We optimized the wall-constrained grasp strategy for the RBO Hand 2 (Deimel and Brock, 2016) in the SOFA simulation framework (Allard et al., 2007). Each finger is modeled as a Cosserat beam with empirically identified stiffnesses. We use dynamic skinning to calculate the collision geometry of the hand. The resulting collisions are resolved with a compliance-based constraint solver (Tournier et al., 2015). The simulation environment contained a vertical and horizontal surface. Instead of optimizing the entire six-dimensional parameter vector we started using only the angle of attack of the hand while sliding towards the wall.

During each iteration $N = 10$ wrist angles were sampled. Grasp success was defined by moving the hand on a pre-defined trajectory in simulation and checking whether afterwards



(a) Average success over 160 grasps per iteration during execution of the cross-entropy method. The success rate increases steadily up to 96%.

(b) The angle of attack of the ten sampled motions per iteration. The optimal parameter value converges to approximately 72° .

Figure 6.8: Results of the cross-entropy method in simulation.

the object was still positioned close to the hand’s palm. We evaluated every parameter sample in scenarios with a sphere and a cylinder each located at eight different initial poses. This ensures that the strategy will be invariant to small uncertainties in object position and shape. In this case the success r_i was not binary but averaged over all 16 simulations. This resulted in a total of 160 simulations per iteration. We continued the process for 7 iterations.

Fig. 6.8a shows that the success rate steadily increased to $\sim 96\%$. It is also visible that the success rate was already considerably high in the very first iteration ($\sim 84\%$). This is due to the reasonable initial parameter values that were set manually from prior experiences. Apart from the increasing success rate, Fig. 6.8b shows the convergence of the wrist angle over time. Video Fig. 6.2 shows sample executions during the first and last iteration.

REAL WORLD

Our simulation experiment showed that the proposed method increases grasp success and the learning process converges. We complimented this experiment with a real-world grasping experiment. This time we optimized all parameters of the wall-constrained strategy. We used a 7-DOF Barrett WAM equipped with a Asus Xtion Live camera sensor at the elbow and the RBO Hand 2 as end-effector. The robot was placed in front of a table with an apple and a vertical structure. We executed ten samples per iteration, with a total of nine iterations (see Fig. 6.9).

Fig. 6.10a shows that grasp success increases over time although it is much more noisy than in simulation and already plateaus at $\sim 60\%$. This is probably due to small number of samples per evaluation in combination with an increased parameter space. While in simulation

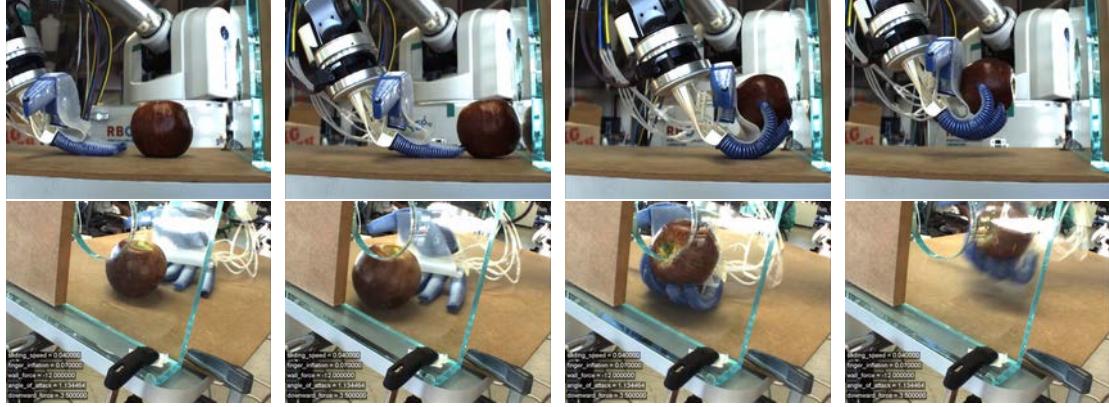
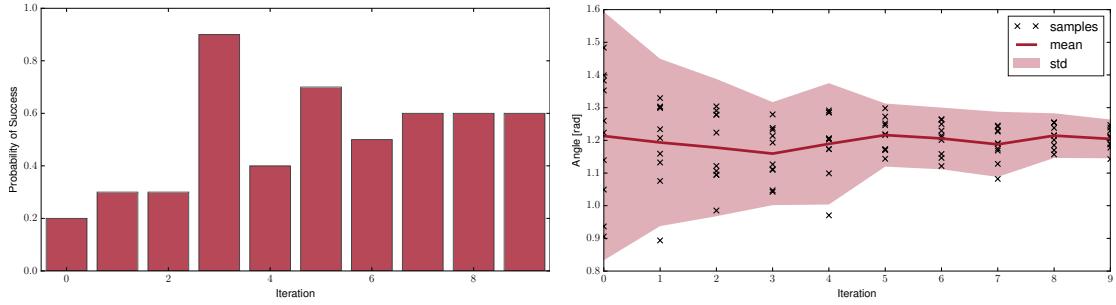


Figure 6.9: Setup of the cross-entropy method for optimizing multiple parameters of the wall grasp. The task is to grasp an apple whose initial position and orientation varies. Parameters included in the optimization are: initial finger pressure, force downwards, sliding speed, angle of attack, and force threshold in wall direction.



(a) Average success over 10 grasps per iteration during execution of the cross-entropy method on the real robot. The success rate increases up to $\sim 60\%$.

(b) The angle of attack of the ten sampled motions per iteration. The optimal parameter value converges to ~ 69 , a similar value compared to the simulation ($\sim 72^\circ$).

Figure 6.10: Results of the cross-entropy method in the real-world experiment.

we used 16 evaluations per parameter value $\theta \in \mathbb{R}$, here we rely on a single one per parameter value $\theta \in \mathbb{R}^5$. Still, Fig. 6.10b shows that the wrist angle converges over time. Interestingly, the final angle is similar to the one optimized in simulation. This hints at the possibility to transfer policies learned in simulation to the real world.

6.3 RELATED WORK

We presented two methods that locally adapt parameters of grasping strategies based on the object and environment. For each of the two algorithms we present similar approaches from the literature. Since this is a very specific topic, our related work is quite compact.

6.3.1 MODEL-BASED ADAPTATION OF SURFACE-CONSTRAINED GRASPS

Our first method (Sec. 6.1) used a hand-crafted model to adapt the approach vector of a surface-constrained grasp. [Maldonado et al. \(2010\)](#) also refine the pre-grasp pose based on sensor data. They assume that all objects are placed on top of a table and each point cluster above the table surface is interpreted as an object. In contrast to our approach, the pre-grasp pose is optimized to bring the center of the palm as close to the object while maximizing the distance between object and fingers. Similarly to our approach, [Ekvall and Kragic \(2007\)](#) find grasping approach vectors based on geometric simplifications of object shape and hand pre-shape. But instead of using a hand-crafted model they use human demonstrations to derive good approach directions.

[Berenson et al. \(2011\)](#) introduce the concept of task space regions which are manifolds that describe pose constraints. Task space regions are similar in spirit to the bounded manifolds we use in pre-grasp optimization. But while they are used in more general motion planning problems (not only as goals), the idea of balancing the influence of object shape and environment geometry is unique to our method. Another way to represent regions of successful pre-grasp poses are grasp densities ([Detry et al., 2011](#)). They can be generated from visual input or human demonstrations and can represent also disconnected and less structured topological spaces. But again, the model used by [Detry et al. \(2011\)](#) does not distinguish between effects induced by contact with the environment or object.

Finally, our small clutter experiment (Sec. 6.1.3) resembles the problem tackled by [Kitaev et al. \(2015\)](#). In contrast to our method, they optimize an entire approach trajectory and also include terms in the objective that penalize toppling over objects that are not supposed to be grasped but need to be pushed to the side to reach the target. Since their model is based on a physics simulation, it is computationally much more expensive than ours and cannot be evaluated in real-time.

6.3.2 ADAPTING STRATEGIES THROUGH LEARNING

Our second approach (Sec. 6.2) improves a planned grasping strategy by adapting its parameters through trial-and-error. We benefit from a structured action representation that is already tied to the task by requiring rather few data. In contrast, more general reinforcement learning approaches to grasping are characterized by a high demand in data from thousands to millions grasp examples (Pinto and Gupta, 2015, Levine et al., 2016). In terms of structured action representations, dynamic movement primitives (Schaal, 2006) are equally beneficial. Stulp et al. (2011) used them to learn the approach direction for grasping upright standing objects based on a model of pose uncertainty. Interactions with the environment were not part of the grasps.

For some manipulation tasks action representations are inherently low-dimensional. Thus, they lend themselves to learning methods. Antonova et al. (2017) learn a policy for pivoting a stick grasped by a parallel jaw-gripper into a desired configuration. They use a task-specific simulation model for learning with fewer real-world samples.

Finally, a common theme is to exploit human demonstrations to adapt grasp parameters (Ek-vall and Kragic, 2004, Sweeney and Grupen, 2007). We do this implicitly, since the structure of our EC-based grasping strategies are heavily influenced by human experience.

6.4 CONCLUSION

In this chapter we presented two methods which can be used to plan grasps that incorporate properties of objects and the environment. The first approach was creating a small optimization problem based on a simple model of surface-constrained grasps and the geometric simplification of object shape. It was a continuation of the planning algorithm based on shape fitting from the first part of the thesis. In contrast, the second method was based on a black-box optimization and parametrizations of the EC-based strategies that the planner from the second part produced.

We showed in various experiments the helpful effect of either incorporating the environment into an approach that focuses on the relation between hand and object, or by incorporating the object into an approach that focuses on the relation between hand and environment.

COMPARISON BETWEEN BOTH ADAPTATION STRATEGIES

Both methods locally adapt grasping strategies. In the following we point out their fundamental differences and implications for grasping. We will refer to our first approach (Sec. 6.1) as method A and to the second one (Sec. 6.2) as method B.

HAND-ENGINEERED VS. DATA-DRIVEN: Method A uses a hand-crafted model while method B uses actual grasp experiences. A hand-engineered model might be easier to derive for classical stiff grasping mechanisms such as the Barrett Hand, since hard contacts with low friction match physical intuition. The complex and less intuitive contact interactions of less conventional hands such as the RBO Hand 2 are harder to describe manually. In these cases the data-driven method B is certainly more suited. Method A is also less general than method B. For every new type of mechanism and grasp, new models need to be created. In contrast, it is much easier to use different mechanisms and grasps with method B but it requires a tedious data collection effort. Both methods can suffer from bias. While method A depends on the (potentially biased) engineer's knowledge, the bias of method B depends on the objects selected during the learning process.

CLOSED VS. OPEN-LOOP: On the planning-level method A can be considered closed-loop: the depth sensor measurement affects the planned approach direction. In contrast, method B finds good parameters during training time, which are then fixed for every grasp and independent of the actual sensor measurement. At first sight this seems reasonable: After all

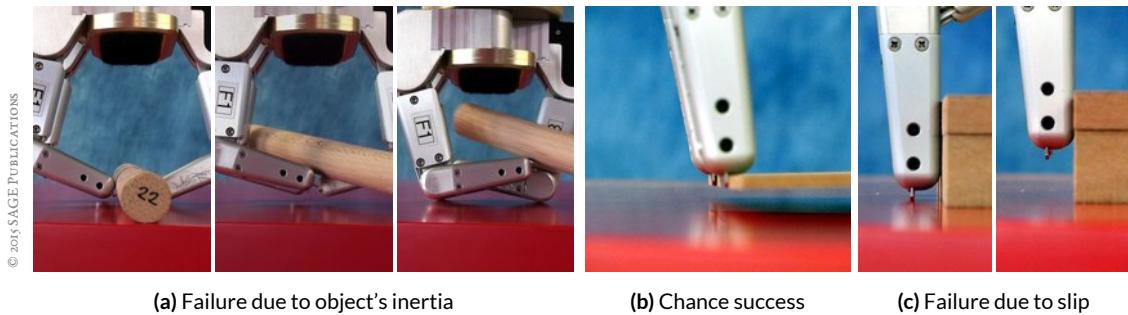


Figure 6.11: Exemplary failure cases for the surface-constrained grasping strategy

EC-based grasping strategies are dominated by the dynamics between hand and environment. But as we will see in the next section, there are important cases which cannot be captured in this case and even method A would fail.

LIMITATIONS

Although both methods adapt grasps by considering object knowledge (method A through shape fitting, method B through the selection of training objects), there are shortcomings. Certain object properties that are not captured by the methods can lead to grasp failure as shown in Fig. 6.ii. These include geometric details or non-visible properties such as mass, friction or inertia. Consider a surface-constrained grasp of differently sized blocks with the Barrett Hand. For smaller blocks the grasp succeeds when the fingernails jam against one of the block’s sharp edges, as can be seen in Fig. 6.iib. For taller blocks, the fingernails do not contact the object, leading to slip and grasp failure, as seen in Fig. 6.iic. In a few cases, however, the nails catch the object just before slipping out of the hand, leading to grasp success.

Here, local grasp adaptation (such as the approach direction) does not help and it is impossible to model those effects with simplified geometries. But instead of making local adaptations of the same grasp strategy it might be beneficial to take more global decisions of which ECE strategy to apply, based on object geometry. This will be the topic of the next chapter.

7

Selecting Environment-Constrained Grasp Strategies Based on Object Properties

WE HAVE SHOWN IN THE PREVIOUS CHAPTER how environmental and object properties can be combined to create more successful grasp strategies. In one case the environment influenced the planning of object-centric grasps, in another case an environment-centric strategy was adapted to apply to certain object features. But this kind of adaptation is limited. Imagine a credit card lying in the middle of a table. We can try to adapt the parameters of a surface-constrained grasp as much as we want, for a lot of hands it will be impossible to pick up such a flat object. In such a case an edge-constrained grasping strategy is much more likely to succeed. To tackle this problem it is not enough to locally adapt grasps. Instead we need to answer a more general question: Which environmental constraints should be exploited when being confronted with a particular object (see Fig. 7.1)?

In this chapter we are trying to answer this decision-making problem. It will be based on the ECE planner presented in Chapter 5 (Algorithm 5.4) that produces a set of feasible ECE-based grasps. We will predict the success of each grasp using a supervised learning setting based on object features and select the most promising strategy.

We are choosing a data-driven approach since the interactions between hand, object and environment and the resulting effects are complex to model. The ECEs are actions over long time horizons involving a lot of contact. Since we are also interested in exploiting the advantages of soft mechanisms more modeling challenges arise: unobservable deformations and

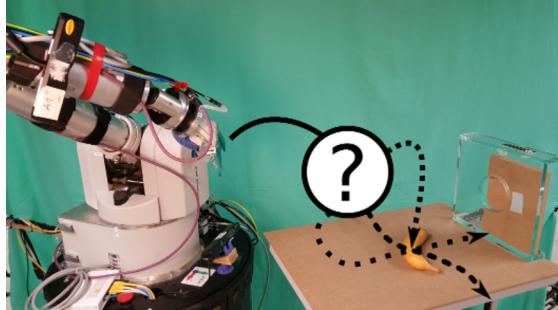


Figure 7.1: The main focus of this chapter is the decision of the most promising EC-based grasping strategy given a depth sensor measurement of the scene including an object. To solve this decision problem in an data-efficient way, we use a multi-armed bandit formulation that trades off exploration and exploitation.

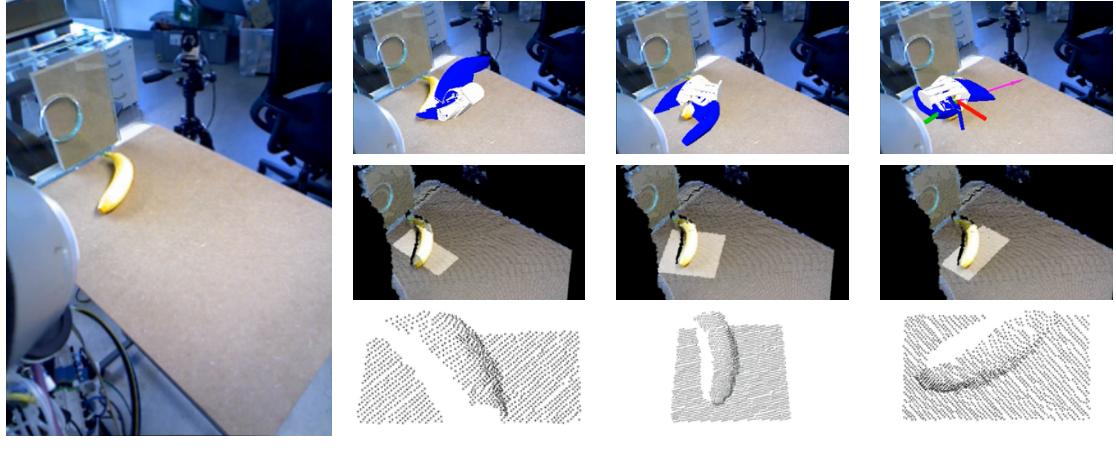
surface stiction and friction phenomena. Hand-engineering a model that captures all of this is infeasible.

We start our data-driven modeling investigation by looking at different representations of object geometry. Those representations should allow us to capture properties such as size and curvature, which we know will affect the decision of which ECE to chose. We will evaluate multiple possible representations in Sec. 7.1.

Additionally, we formulate grasp strategy selection as a multi-armed bandit problem (Robbins, 1985). This allows us to learn outcome models for each strategy from scratch while also finding the most promising strategy for each novel problem scenario as quickly as possible based on prior experience. We show how the model continuously improves throughout the robot's life time without an explicit training and test phase (Sec. 7.2).

7.1 FEATURES FOR SELECTING THE RIGHT EC EXPLOITATION

We consider three different grasping strategies: surface-constrained, edge-constrained and wall-constrained grasps (Sec. 5.1). The candidate grasp strategies are planned based on the geometry of the environment and the rough location of the object to be grasped (Algorithm 5.4); object knowledge apart from the approximate location of the target object is ignored. To include object knowledge we extract a feature vector from a depth sensor measurement, given the approximate location of the object and the environmental constraints that the grasp will exploit. We learn one binary classifier for each type of grasping strategy that maps the feature vector to success or failure. We prefer this learning setting to a single multiclass classifier which would predict the best grasp strategy for a given object. Using multiple binary classifiers allows us to easily extend our framework with new types of grasping strategies without any retraining. Additionally, it might be the case that more than a single strategy can be executed to grasp a particular object.



(a) Raw sensor input

(b) Three ECEs (wall-constrained, surface-constrained, and edge-constrained grasp) and the extracted 3D point cloud data to calculate features

Video Figure 7.1: We formulate the selection of ECE as a supervised learning problem. For each of the planned strategies we calculate features (b) and predict grasp success using a binary classifier. [https://youtu.be/I_RXzIcx6M]

Whether our supervised learning problem will work, depends mostly on the chosen feature representation. We focus on shape features and ignore color information. Color only implicitly hints at material properties that influence grasp success. But since we will deal with a rather limited set of real-world grasping data, including color will eventually lead the learning algorithm to exploit spurious correlations which lead to poor generalization.

We use different local grasp frames to compute the feature vectors. The grasp frame for the wall-constrained strategy is aligned with the normals of the wall and support surface. The surface-constrained grasp's frame is aligned with the normal of the support surface and the orientation of the hand. Finally, the frame for the edge-constrained strategy is aligned with the direction of the edge and the normal of the adjacent plane.

These choices of grasp frames reflect the insight that the success of the strategies is mostly invariant in the direction of the contact normal of the environmental contact that is being exploited. We crop the 3D points in the local neighborhood of the candidate grasp frame and analyze three geometric descriptors to characterize the associated grasp strategy which we will present next. Video Fig. 7.1 summarizes the computational pipeline of our approach.

7.1.1 GRASP DESCRIPTORS

We present three different types of geometric features. Most of them have shown to be efficient at object recognition or shape retrieval problems ([Tangelder and Veltkamp, 2004](#)). But

since object identity is often not informative for grasp success (e.g. it is not affected by rotation, in contrast to grasp success), it is hard to anticipate their usefulness for predicting the outcome of EC grasps. Thus, we will compare these features by applying them to a grasp data set in Sec. 7.1.3.

SHAPE DISTRIBUTIONS

Shape distributions ([Osada et al., 2002](#)) are signatures that are based on samples from a shape function, such as the distance between two random surface points or the angle between three points. We use the distance measure and sample it 2000 times within the grasp region. Distances are discretized into a histogram of 128 bins, each representing a space of 3 mm.

SHAPE HISTOGRAMS

Shape histograms ([Ankerst et al., 1999](#)) decompose the 3D space into bins, which can have the shape of shells, sectors or combinations of them. We use 128 slices of 2 mm length along the height axis with the intention of capturing geometric properties that influence grasp success. A wall grasp succeeds only if the fingers can slip underneath the object which depends on its flushness with the support surface. Since a depth sensor will not measure points in case of these cavities, our shape histogram should capture this property.

POINT FEATURE HISTOGRAMS

Point feature histograms ([Rusu et al., 2009](#)) are a popular feature descriptor for 3D object detection and recognition. They calculate a signature based on the relation of 3D points and their normal information. We use these histograms to characterize the local neighborhood of the grasp frame.

7.1.2 TRAINING

We assume that similarity in these feature spaces also translates to similarity in the resulting physics of the grasps. This is an oversimplification, since effects of mass and friction, for example, cannot be estimated from those features. Still, when casting the problem of predicting grasp success as a binary classification problem based on the geometric descriptors, we can show that a substantial amount of grasp outcomes can be predicted correctly (Sec. 7.1.3).

We compared the different grasp features presented in Sec. 7.1.1 based on the accuracy and $F_{0.5}$ measure of the trained classifiers. We include the $F_{0.5}$ measure because it weighs recall

lower than precision. In grasp detection it is favorable to find at least one robust grasp rather than finding all possible grasps.

The data sets for simulation contain 1340 samples for each grasp strategy while the real-world sets contain 220 samples each. They were split into training and test set 4 : 1 using stratified sampling. The ensemble classifiers are trained via AUTO-SKLEARN ([Feurer et al., 2015](#)) which searches a structured hypothesis space including multiple types of classifiers and preprocessing methods.

7.1.3 EXPERIMENTS

We perform grasping experiments in simulation and on a real robot to show that...

1. ...the limited set of the three presented grasping strategies captures a wide variety of objects.
2. ...selecting the best strategy can be learned from data (additionally we will evaluate a variety of feature representations to find the most suitable one).
3. ...the learned classifiers give reasonable and consistent results on novel data.

REAL-WORLD SETUP

All strategies extensively exploit the environment during grasping, which soft manipulators are especially suited for. We use the RBO Hand 2 ([Deimel and Brock, 2016](#)), a pneumatically actuated anthropomorphic hand made out of silicone that inherently adapts to the shape of a variety of objects when being inflated. In contrast to traditional, stiff hands, unplanned contacts do not necessarily lead to catastrophic outcomes.

The real-world experiments are conducted on a 7-DOF Barrett WAM platform, including an Asus XTion Pro RGB-D camera on the robot's forearm, a six-axis ATI Gamma force-torque sensor on the wrist, and the RBO Hand 2 as an end-effector. The experimental setup is shown in Fig. 7.3. It contains a table with a vertical structure that can be used as a wall constraint. In each trial the robot first moves to a pre-defined viewing configuration and uses the RGB-D input to plan a grasp. The object set, shown in Fig. 7.3, contains 22 items that differ widely in shape, rigidity, surface friction and mass. For each of the three strategies every object was randomly placed on the table ten times, totaling a number of 630 grasp attempts.



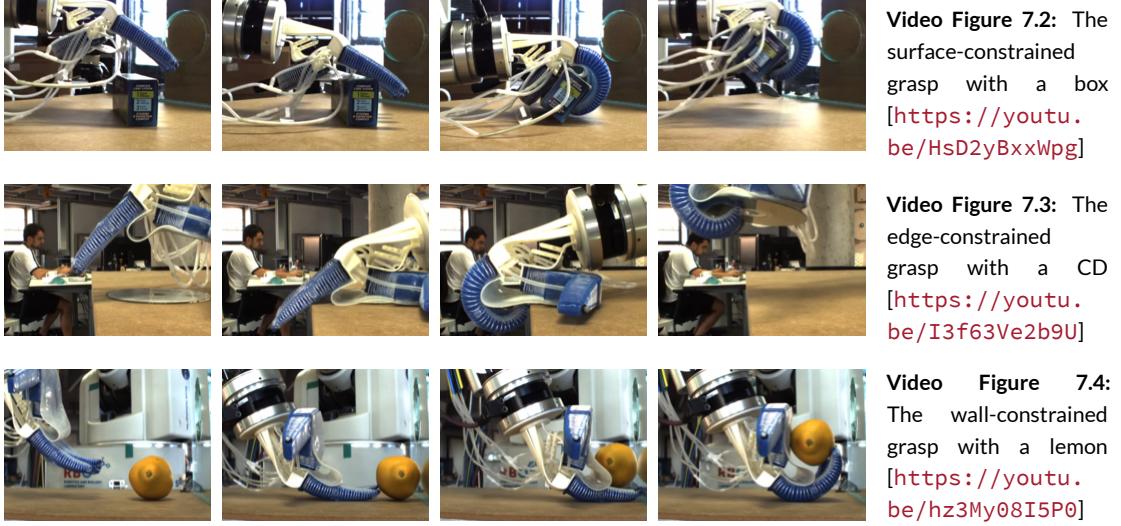
Figure 7.2: Simulation setup for the grasping experiment. **Left:** Simulated RBO Hand 2 in the SOFA simulation framework. **Right:** 45 objects from the KIT models database (Kasper et al., 2012). The data set contains a total of 134 objects. The majority of the objects are supermarket products. Since those objects are usually optimized for efficient storage, their geometries do not vary much. Most shapes are cuboids or cylinders.



Figure 7.3: Real-world setup for the grasping experiment. **Left:** Barrett WAM with RGB-D sensor and RBO Hand 2 in front of a table and a vertical structure (transparent). The setup is chosen such that it offers opportunities to do all three types of EC-based grasp strategies. **Right:** We use 22 different objects in the real-world experiment.

SIMULATION SETUP

Our simulation experiments are based on SOFA (Allard et al., 2007). To simulate the RBO Hand 2 (Deimel and Brock, 2016), we use a compliance-based constraint solver (Tournier et al., 2015). Each finger is modeled as a Cosserat beam with empirically identified stiffnesses. Skinning is used to determine the collision geometry. Fig. 7.2 shows the simulation setup. It includes the hand, an object, and environmental constraints such as a table surface, wall, or edge. Object meshes are used from the KIT object models database (Kasper et al., 2012) which contains mainly supermarket products. To find random initial object poses that are in static equilibrium we run a separate simulation in which the randomly oriented objects fall on a pla-



nar surface until they come to rest. A depth sensor is simulated using the intrinsic calibration parameters derived from an Asus XTion Pro with added Gaussian noise whose standard deviation scales quadratically with measured depth (Khoshelham and Elberink, 2012). We run all three grasping strategies for ten poses of each object, resulting in 4020 simulated grasps.

COVERAGE OF DETECTED GRASPING STRATEGIES

Before focusing on the high-level decision between different grasping strategies, we need to show that those options actually solve a significant amount of problem settings. This is done by looking at the performance of the most successful grasping strategy for each problem scenario. We would like to ensure that there are only very few objects/poses that cannot be grasped by any of the three strategies.

The real-world results (see Fig. 7.5) show that for each object there is at least one strategy that is able to grasp it. The most problematic objects are the gamepad and the marker. The geometry of the gamepad makes it most suitable for a surface-constrained grasp. But since this grasp type can only exert moderate forces that counteract gravity, the relatively heavy gamepad is lost 6/10 times. The thin long-shaped marker is most suited for a wall grasp, but half the times the fingertips fail to slip underneath it.

The simulated data shows a different picture, here the strategies only cover 41 % of all problem scenarios (see Fig. 7.4). This is due to multiple reasons. The KIT object set contains a lot of cuboid-shaped supermarket objects which are relatively large w.r.t. the hand. Twenty large objects could not be grasped by any strategy. The contribution of the edge-constrained

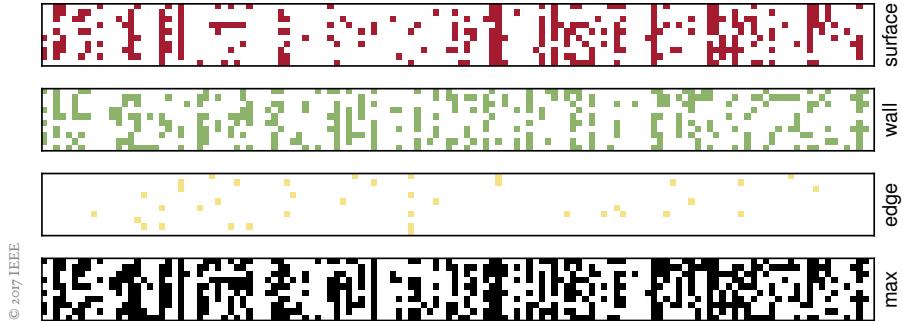


Figure 7.4: Each dot represents a successful grasp in simulation. Along the x-axis are the 134 objects and the y-axis depicts ten different poses. The last row named 'max' is the maximum over all strategies per object (and pose). It is an upper performance bound given the robot would choose the optimal grasp in each situation.

grasp strategy is extremely low (39/1340 successes) because of the lack of thin, easy to slide objects that do not topple over (the most successful one was a can of fish). Another problem was that the simulation framework did not allow to differentiate between friction coefficients of table, hand and object, which affected mostly the sliding phase of the edge grasp (objects would stick to the table surface).

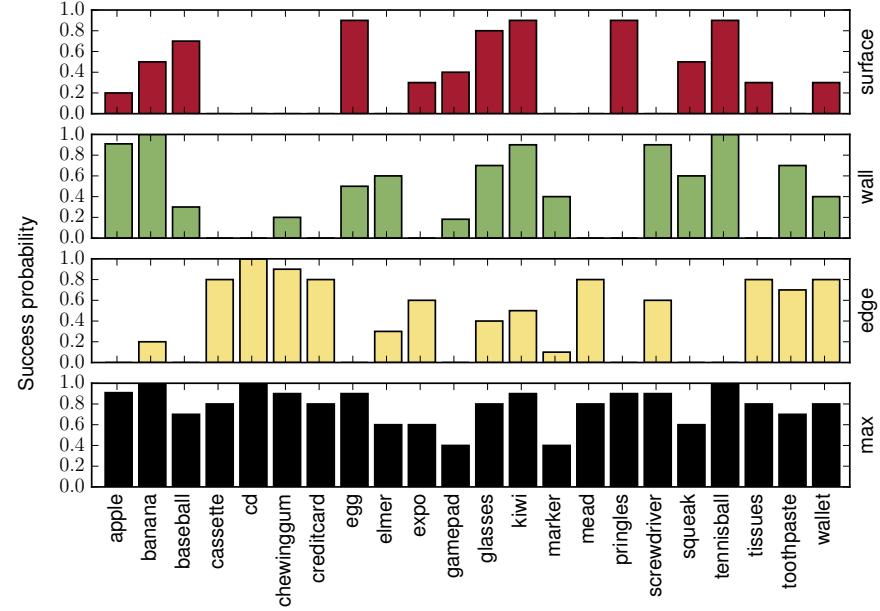
We have shown that on a real-world object set all objects can be grasped by at least one strategy. But the data also shows that the best strategy differs from object to object. Being able to predict the match between strategy and object is the focus of the next experiment.

GRASP FEATURES TO PREDICT GRASP OUTCOME

In this experiment we evaluated the different feature representations (Sec. 7.1.1) for predicting successful grasps. The results of the performance on the test set are listed in Table 7.1. In general, the point feature histograms are the most suitable feature descriptor to predict grasp success for all strategies in simulation and in the real-world. The high scores for the edge strategy in simulation are not representative, since it is a highly unbalanced data set with very few successful grasps.

TRANSFER AND GENERALIZATION OF CLASSIFIERS

Finally, we want to analyze in more detail what the grasp classifiers learned. To visualize their decision boundaries we will chose a continuous 2D shape space. We will sample shapes densely, simulate point cloud measurements, plan grasps, and classify their success according



© 2007 IEEE

Figure 7.5: The plot shows the success rate of the three different grasping strategies for 22 different objects on a real robotic platform. The success rate is based on ten grasp attempts per object and strategy. The last row named 'max' is the maximum over all strategies per object (and pose). It is an upper performance bound given the robot would choose the optimal grasp in each situation. It also shows that the real-world data set is much better covered than the KIT object set used in simulation (see Fig. 7.4).

to the three grasp types. Since these simulated shapes are different to the real ones seen by the classifiers during training, we will also test their ability to generalize and transfer knowledge between the real and simulated world.

As a continuous shape parametrization we chose superellipsoids (Barr, 1992). Superellipsoids are simple to define but still include a lot of different basic geometries. They can be defined implicitly by having all 3D surface points (x, y, z) satisfy the equation

$$\left(\left| \frac{x}{s_x} \right|^{\frac{2}{e}} + \left| \frac{y}{s_y} \right|^{\frac{2}{e}} \right)^{\frac{e}{n}} + \left| \frac{z}{s_z} \right|^{\frac{2}{n}} = 1,$$

where $e > 0$ and $n > 0$ are shape parameters that specify the shape along the horizontal and vertical sections and s_x , s_y , and s_z are size parameters in all three dimensions. The family of superellipsoids contains basic shapes such as the cube ($e \rightarrow 0$, $n \rightarrow 0$), cylinder ($e = 1$, $n \rightarrow 0$), or sphere ($e = 1$, $n = 1$) as special cases.

To generate superellipsoids we make use of a parametric equation rather than the implicit

Feature Data Set \	Shape Distributions	Shape Histograms	Point Feature Histograms
Simulation (surface)	0.78 (0.56)	0.85 (0.70)	0.88 (0.76)
Simulation (edge)	0.97 (0.94)	0.97 (0.94)	0.97 (0.94)
Simulation (wall)	0.77 (0.53)	0.79 (0.58)	0.86 (0.64)
Real-World (surface)	0.72 (0.44)	0.81 (0.62)	0.86 (0.72)
Real-World (edge)	0.71 (0.42)	0.78 (0.57)	0.78 (0.57)
Real-World (wall)	0.62 (0.25)	0.65 (0.30)	0.79 (0.58)

Table 7.1: Accuracy ($F_{0.5}$ -scores) for different grasp features. The bold numbers represent the best results for each data set.

one above. It describes each surface point (x, y, z) as a function of longitude $\theta \in [-\pi, \pi[$ and latitude $\phi \in [-\frac{\pi}{2}, \frac{\pi}{2}[$:

$$\begin{aligned} x(\theta, \phi) &= s_x \operatorname{sgn}(\cos \phi) |\cos \phi|^n \operatorname{sgn}(\cos \theta) |\cos \theta|^e \\ y(\theta, \phi) &= s_y \operatorname{sgn}(\cos \phi) |\cos \phi|^n \operatorname{sgn}(\sin \theta) |\sin \theta|^e \\ z(\theta, \phi) &= s_z \operatorname{sgn}(\sin \phi) |\sin \phi|^n \end{aligned}$$

We are interested in varying only two shape parameters to visualize the results in 2D. From our experiments it seems that the flatness of objects plays an important role when deciding for a surface-constrained or an edge-constrained grasp. The flushness with the support surface is another important feature. Objects that are flush with the table cannot be grasped by a wall-constrained strategy. Thus, we focus on the *roundness* r and *height* h of an object. In the context of superellipsoids, we define roundness as $r = n = e$ and height as $h = s_z$. The sizes $s_x = s_y = 0.05$ are kept constant. The resulting implicit equation for this shape family is

$$\left| \frac{x}{0.05} \right|^{\frac{2}{r}} + \left| \frac{y}{0.05} \right|^{\frac{2}{r}} + \left| \frac{z}{h} \right|^{\frac{2}{r}} = 1.$$

Thus, we have a 2D shape space which ranges from cubes to spheres, from flattened to raised objects. Fig. 7.6 shows equidistant samples in this 2D space.

For our experiment we generate $34 \times 24 = 816$ shape instances (roundness \times height). Each shape is simulated on a planar surface, next to vertical wall and close to an edge. We create point cloud sensor readings using the same method as before (Khoshelham and Elberink,

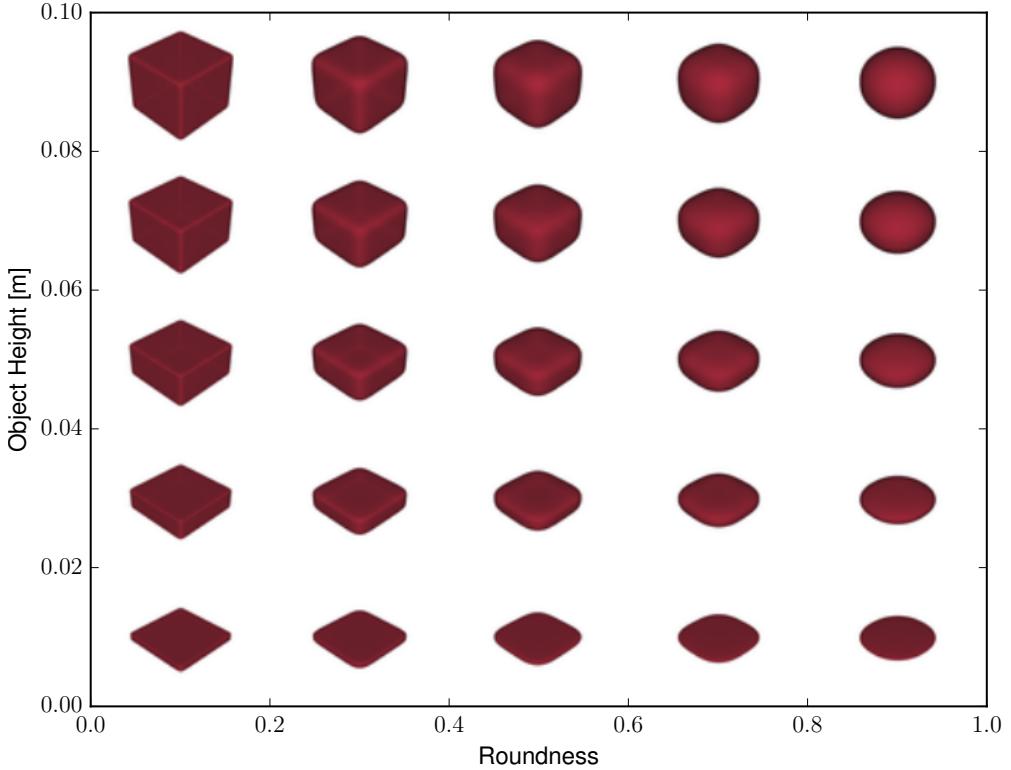


Figure 7.6: We use a 2D space of superellipsoids to test our grasp classifiers for surface-constrained, wall-constrained, and edge-constrained grasps. Our shapes are parametrized by their height and roundness, including spheres (upper right), boxes (upper left), and disks(lower right). See text for details.

2012). Based on the planned grasps we apply the binary classifier for all three types of grasps. Note that we use the classifiers that were trained on the real-world data set (Sec. 7.1.3).

The results are shown in Fig. 7.7. Each colored box denotes a positive classifier result, while a white one means a negative prediction. Although all three classifiers produce independent results, we plot the wall-constrained and edge-constrained classifier results in the same graph to emphasize how they complement each other and to facilitate comparison.

Overall, the results are promising, given the fact that the training and test data originate from two very different sources – real-world observations and simulated ones. The edge-constrained classifier predicts success for objects up to 2 cm in height independent of how the contour is shaped. This matches our experience for the CD and the audio tape. At an increased height the fingers have problems wrapping around the object at the edge.

In contrast, the wall-contrast grasp classifiers predicts failure for flat objects. We have seen that this is true even for non-flush objects, such as a marker lying on a table. Our assumption

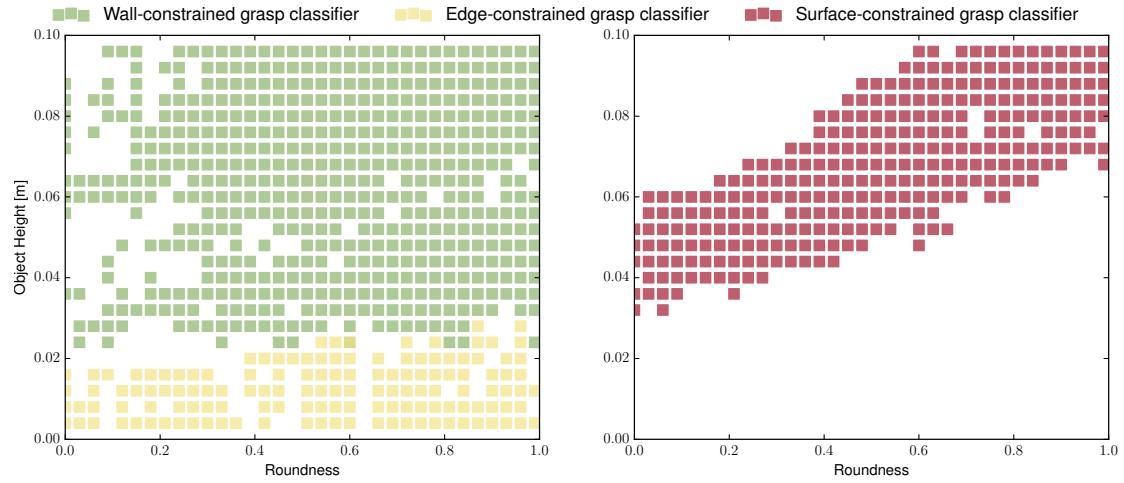


Figure 7.7: The plots show the results of applying the trained classifiers on simulated sensor data of the superellipsoids. A colored square indicates a positive classification, while a white one is a negative result. We plot the results of the wall-constrained and edge-constrained classifier into the same graph to show that they complement each other. See text for details.

that flushness affects wall-grasp success was partly learned by the classifier. We can see that shapes with low roundness values generally produce less positive classifications. But it is also visible that there is no clear decision boundary; the results seem noisy.

The surface-constrained grasp classifier produces a clear band of positive predictions. It matches our experience that flat objects (<3 cm) cannot be grasped with a surface-constrained grasp. The negative classification for low roundness and high height values also seems plausible: The RBO Hand 2 has problems enveloping large objects, while more spherical shapes with the same dimensions facilitate wrapping fingers around.

7.2 IMPROVING THE LEARNING PROCESS

Instead of using the classifiers trained in the previous section to decide between the three grasping strategies, we would like to continuously learn about their value and make decisions that maximize the expected long-term grasp success. This would allow us to apply the algorithm in a life-long-learning setting, where it can improve beyond a pre-determined training phase.

We can formulate such a learning process as a multi-armed bandit (MAB) problem ([Robbins, 1985](#)). A MAB problem is a repeated game in which an agent chooses one of multiple arms in each round and the environment returns a scalar reward (or payoff) based on this action. The goal of the agent is to maximize the expected total reward. Since the underlying reward structure is unknown a MAB algorithm needs to balance exploration (selecting an action whose payoff is uncertain) and exploitation (choosing the action that performed best so far). Algorithms that solve the MAB problem find the highest rewarding arm as quickly as possible.

In our case, each arm corresponds to one grasping strategy and the reward signal is binary, indicating grasp success. But in contrast to the vanilla MAB problem, there is no single best grasping strategy. Instead the best performing grasp depends on the object features. Thus, a more suitable formalization is the *contextual* multi-armed bandit problem ([Langford and Zhang, 2008](#)).

In the contextual MAB setting, the agent receives a context vector before taking an action. This information can be used to guide the decision. We use the features calculated from the RGB-D input as context and estimate the expected grasp success for each strategy. Algorithm 7.6 shows the iterative learning process. In each round, the robot is confronted with a random object. It uses the planning algorithm presented in Chapter 5 (Algorithm 5.4) to generate all potential grasping strategies. For each candidate grasp, we calculate a grasp feature (as shown in the previous section) which is used to select one of them. Finally, the outcome is used to update the robot's internal model of the expected grasp success and the robot is confronted with another random object.

A variety of algorithms have been presented for tackling contextual MAB problems. They differ in the way they model the estimated reward, how they are updated when facing new information (UPDATE_MODEL), and how actions are chosen based on the estimated reward (SELECT_ACTION). Exploration strategies range from simple ϵ -greedy algorithms to Thompson sampling or using confidence bounds ([Zhou, 2015](#)).

Algorithm 7.6 Contextual Multi-Armed Bandits for EC Grasping

Input: $\mathbf{x}_{\text{hand}}, \mathbf{x}_{\text{object}}$
 Output: R ▷ total reward (number of successful grasps)

```

for  $t = 1, 2, \dots, T$  do
    Take point cloud measurement  $P_{\text{RGB-D}}$ 
     $A \leftarrow \text{ECE\_PLANNING}(P_{\text{RGB-D}}, \mathbf{x}_{\text{hand}}, \mathbf{x}_{\text{object}})$  ▷ Plan all candidate grasps (Alg. 5.4)
     $C \leftarrow \emptyset$  ▷ Set of context vectors
    for  $a \in A$  do
         $C \leftarrow C \cup \text{GRASP\_FEATURE}(a)$  ▷ Calculate context vectors (Sec. 7.1)
    end
     $a_t \leftarrow \text{SELECT\_ACTION}(A, C)$  ▷ Select action (Sec. 7.2.1)
    Execute  $a_t$  and receive reward  $r_t \in \{0, 1\}$ 
     $\text{UPDATE\_MODEL}(a_t, r_t)$  ▷ Update the internal model (Sec. 7.2.1)
     $R \leftarrow R + r_t$ 

```

In the following we present five such methods for selecting grasps in more detail. We will evaluate them in the grasping scenario based on EC exploitation and also analyze their sensitivity to disturbed reward signals.

7.2.1 ACTION SELECTION STRATEGIES

For each exploration-exploitation strategy we will explain how it models the expected reward and what it does in the subroutines `SELECT_ACTION` and `UPDATE_MODEL` according to Algorithm 7.6. We will refer to the context vector for grasp strategy g in round t as c_t^g , the chosen action/grasp as a_t , and the obtained reward as r_t . In case a strategy models the expected reward/payoff, this is done separately for all three types of grasp strategies. A model parameter x in round t for grasp strategy g is written as x_t^g .

THOMPSON SAMPLING WITHOUT CONTEXT (TS)

Thompson sampling ([Thompson, 1933](#)) is a heuristic that chooses actions by sampling a belief and acting optimally with respect to it. We represent the belief of the payoff with a beta distribution for each action. The beta distribution is parametrized by two scalars $\alpha \in \mathbb{R}^{>0}$ and $\beta \in \mathbb{R}^{>0}$. To start with a uniform belief over the expected payoff, we initially set $\alpha_0^g = 1$ and $\beta_0^g = 1$. After drawing samples for each action type we choose the action with the highest expected payoff:

$$a_t = \underset{g}{\operatorname{argmax}} z_t^g, \quad \text{with } z_t^g \sim \text{Beta}(\alpha_t^g, \beta_t^g).$$

At the end of each iteration the belief is updated according to:

$$\begin{aligned}\alpha_{t+1}^g &= \alpha_t^g + r_t && \text{and} \\ \beta_{t+1}^g &= \beta_t^g + (1 - r_t).\end{aligned}$$

Note that this decision strategy does not take the context into account. Thus, it cannot distinguish between objects and will make decisions only based on the average reward per action. We still include it in our evaluation as indication for a lower performance bound – to see how much the selection process ultimately benefits from taking object features into account.

THOMPSON SAMPLING WITH LINEAR PAYOFFS (LINTS)

Thompson sampling with linear payoffs ([Agrawal and Goyal, 2013](#)) is similar to the sampling scheme described above, with the exception that it models the expected payoff as a linear function of the context vector c_t . The prior distribution over this linear function is given by the Gaussian $\mathcal{N}(\mathbf{b}_t, v^2 \mathbf{A}_t^{-1})$, where $\mathbf{b}_t \in \mathbb{R}^{\dim(c)}$ is the mean vector, $\mathbf{A}_t \in \mathbb{R}^{\dim(c) \times \dim(c)}$ the covariance matrix, and $v = 0.5$ the standard deviation of the likelihood of observing reward r_t at time t . The linear model is initialized with the zero vector $\mathbf{b}_0 = \mathbf{0}$ and the identity matrix $\mathbf{A}_0 = \mathbf{I}$. We draw samples for each action from this distribution and choose the maximum:

$$a_t = \underset{g}{\operatorname{argmax}} z_t^g, \quad \text{with } z_t^g \sim \mathcal{N}(\mathbf{b}_t^g, v^2 (\mathbf{A}_t^g)^{-1}).$$

The model is updated according to:

$$\begin{aligned}\mathbf{A}_{t+1}^g &= \mathbf{A}_t^g + \mathbf{c}_t^g (\mathbf{c}_t^g)^\top && \text{and} \\ \mathbf{b}_{t+1}^g &= (\mathbf{A}_t^g)^{-1} \sum_{s=0}^t r_s \mathbf{c}_s^g.\end{aligned}$$

UPPER-CONFIDENCE BOUNDS WITH LINEAR PAYOFFS (LINUCB)

Just as with the previous method, we use a linear model to estimate the expected rewards, parametrized by \mathbf{A} and \mathbf{b} and initialized with $\mathbf{A}_0 = \mathbf{I}$ and $\mathbf{b}_0 = \mathbf{0}$. But now we use upper

confidence bounds (Chu et al., 2011) of our estimates to select an action:

$$a_t = \operatorname{argmax}_g \left(((\mathbf{A}_t^g)^{-1} \mathbf{b}_t^g)^\top \mathbf{c}_t^g + \alpha \sqrt{(\mathbf{c}_t^g)^\top (\mathbf{A}_t^g)^{-1} \mathbf{c}_t^g} \right),$$

where $\alpha = 1.8$ is an empirically chosen constant. If there are multiple best actions, we choose randomly among them.

UPPER-CONFIDENCE BOUNDS WITH GAUSSIAN PROCESSES (GP-UCB)

The Gaussian process upper confidence bound rule (Srinivas et al., 2009) uses Gaussian process regression to model the reward as a function of the context. A Gaussian process (GP) is fully defined by its mean function $\mu(\mathbf{c})$ and kernel function $k(\mathbf{c}, \mathbf{c}')$. We choose a radial basis function kernel: $k(c, c') = \exp(-\frac{1}{2} \|c - c'\|_2^2)$. We initialize with zero mean. Given N past experiences of contexts $\{\mathbf{c}_0, \dots, \mathbf{c}_{N-1}\}$, corresponding rewards $\{r_0, \dots, r_{N-1}\}$ and a new grasp experience (\mathbf{c}_t^g, r_t) , the posterior is a GP with the following mean and variance:

$$\begin{aligned} \mu_{t+1}^g &= \mathbf{k}(\mathbf{c}_t^g)^\top (\mathbf{K}_t^g + \sigma^2 \mathbf{I})^{-1} r_t && \text{and} \\ (\sigma_{t+1}^g)^2 &= k(\mathbf{c}_t^g, \mathbf{c}_t^g) - \mathbf{k}_t^g(\mathbf{c}_t^g)^\top (\mathbf{K}_t^g + \sigma^2 \mathbf{I})^{-1} \mathbf{k}_t^g(\mathbf{c}_t^g), \end{aligned}$$

where

$$\begin{aligned} \mathbf{K}_t^g &= \begin{bmatrix} k(\mathbf{c}_0, \mathbf{c}_0) & k(\mathbf{c}_0, \mathbf{c}_1) & \dots & k(\mathbf{c}_0, \mathbf{c}_{N-1}) \\ k(\mathbf{c}_1, \mathbf{c}_0) & k(\mathbf{c}_1, \mathbf{c}_1) & \dots & k(\mathbf{c}_1, \mathbf{c}_{N-1}) \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{c}_{N-1}, \mathbf{c}_0) & k(\mathbf{c}_{N-1}, \mathbf{c}_1) & \dots & k(\mathbf{c}_{N-1}, \mathbf{c}_{N-1}) \end{bmatrix} && \text{and} \\ \mathbf{k}_t^g(\mathbf{c}) &= \begin{bmatrix} k(\mathbf{c}_0, \mathbf{c}) \\ k(\mathbf{c}_1, \mathbf{c}) \\ \vdots \\ k(\mathbf{c}_{N-1}, \mathbf{c}) \end{bmatrix}. \end{aligned}$$

We chose the action that maximizes the upper confidence bound of the expected payoff:

$$a_t = \operatorname{argmax}_g (\mu_t^g(\mathbf{c}_t^g) + \alpha \sigma_t(\mathbf{c}_t^g)),$$

where $\alpha \in \mathbb{R}$ is a problem-specific constant and we choose it to be equivalent to the LinUCB method.

ϵ -GREEDY WITH k -NEAREST-NEIGHBORS (ϵ -GREEDY k -NN)

We use a k -nearest-neighbor classifier to model the rewards (Yang et al., 2002). Let

$$R_t^g(\mathbf{c}) = (r_i \mid \forall i : \|\mathbf{c} - \mathbf{c}_i^g\|_2 \leq \|\mathbf{c} - \mathbf{c}_{i+1}^g\|_2, a_i = g, 0 < i \leq t)$$

be the sequence of rewards which are ordered increasingly according to the distance of their context vectors. The estimated expected reward for an action a and context \mathbf{c} at time step t is given by the k nearest experienced rewards:

$$\hat{r}_t^g(\mathbf{c}) = \frac{1}{k} \sum_{i=0, r \in R_t^g(\mathbf{c})}^{i < k} r.$$

We choose $k = 3$ based on an empirical validation. To select an action we chose the maximum in an ϵ -greedy fashion:

$$a_t = \begin{cases} \text{random}(A) & , \text{if } x < \epsilon \\ \underset{g}{\text{argmax}} \hat{r}_t^g(\mathbf{c}_t^g) & , \text{otherwise} \end{cases},$$

where $x \in [0, 1]$ is a uniformly distributed random variable. We set $\epsilon = 0.1$. During UPDATE_MODEL we simply store the chosen action and observed reward.

ORACLE

The Oracle strategy is able to look into future and will always choose the grasp with the best outcome:

$$a_t = \underset{g}{\text{argmax}}(r_t(g))$$

Although this decision strategy is not applicable in any practical scenario, we use it as a comparison. Its performance constitutes an upper bound to our grasping problem. Since there are situations in which none of the candidate grasps succeed, the accumulated long term reward will be below T .

Note that the *regret* of a decision strategy – a performance measure usually used in bandit

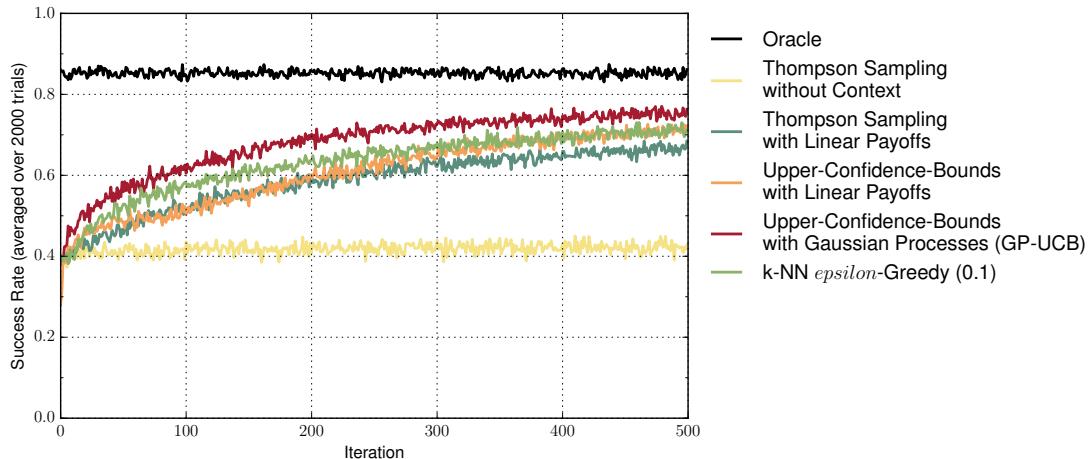


Figure 7.8: The plot shows the success rate of five different exploration strategies for grasping the 22 objects of the real-world data set, averaged over 2000 trials.

problems – is the accumulated difference between the reward secured by the Oracle and its own one: $\text{regret} = \sum_t (\max_g(r_t(g)) - r_{at})$.

7.2.2 EXPERIMENTS

The goal of our experiments is to evaluate which of the action selection strategies perform best and what are the trade-offs that need to be taken into account. We will apply Algorithm 7.6 to the data collected in the previous section. To compare the different selection schemes we define the subroutines `SELECT_ACTION` and `UPDATE_MODEL` accordingly. Since the experiment in the last section also showed that point feature histograms are the most suited features to describe our grasp strategies, we will use them going forward.

In each turn the robot faces a random object in a random pose. The ECE planning algorithm outputs three types of candidate grasps and their feature descriptors are calculated. One of the presented exploration schemes is used to choose one of the three actions and gets a reward of 1 if it succeeds, or 0 otherwise. The goal is to find a working grasp for any object and pose as quickly as possible, i.e., maximizing the accumulated reward. Note that the robot does not know which object it is facing, only a point cloud measurement is available.

EVALUATION OF ACTION SELECTION STRATEGIES

We evaluated the six strategies from Sec. 7.2.1 for $T = 500$ iterations. Since the order of objects and poses are random and some sequences might be easier than others, we evaluate

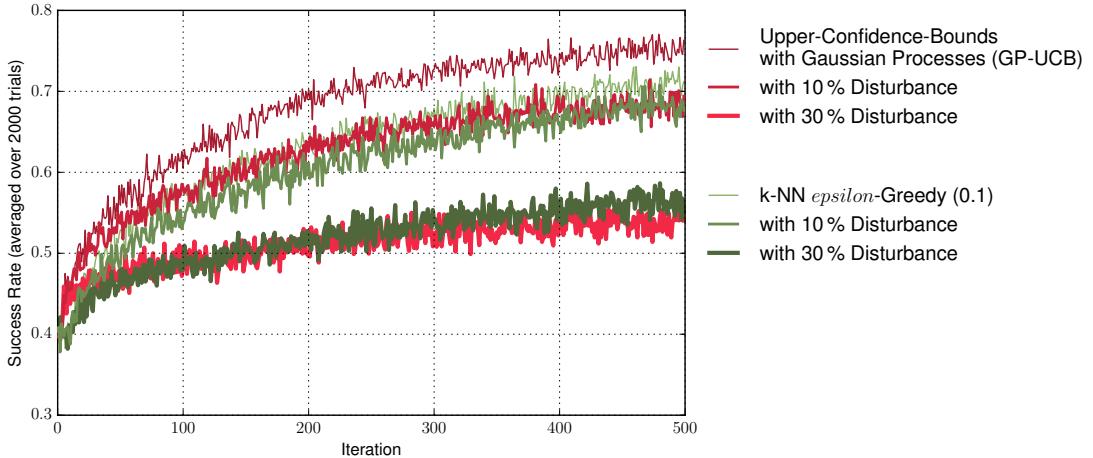


Figure 7.9: The plot compares the success rate of GP-UCB and ϵ -greedy k-NN with varying degrees of noise in the observed rewards, averaged over 2000 trials. The advantage of GP-UCB already vanishes with only 10 % noise.

each strategy 2000 times and report mean success values for each iteration.

The results are shown in Fig. 7.8. As expected the omniscient Oracle performs best, averaging 85 % success rate. The Thompson sampling scheme without context performs worst. This was also expected since ignoring object features will lead to choosing always the grasp that is performing best on average over all objects and poses. This scheme nearly does not learn anything, averaging a success rate of 42 %.

All other selection strategies show an increase in grasp performance over all iterations. The convergence rate of the Thompson sampling with linear payoffs (LinTS) seems to be the lowest. Although initially LinUCB converges similarly to LinTS, it significantly outperforms LinTS from the 200th iteration onward. Since the data provides 210 different object/pose scenarios, this hints at a better exploitation of past experiences. On average the 211th iteration is the first iteration when an object/pose combination appears that has been tried before.

In contrast to the linear models, the strategies based on approximations using k-nearest neighbors and Gaussian Processes perform very well. Although very simple, the ϵ -greedy k-NN is the second best and performs as good as LinUCB. The GP-UCB outperforms every other strategy, achieving 76 % success rate. This is due to the extrapolating qualities of the GP whose predictions of grasp success in the point feature histogram space seem to be the most reliable.

SENSITIVITY TO DISTURBANCES

In a second experiment, we wanted to test how sensitive the action selection schemes are w.r.t. false reward signals. It is a common problem in grasping to robustly determine whether a grasp on a real physical platform was indeed successful. We circumvented this problem by using human labeled data. For a fully autonomous system this is not an option. Commonly, estimators for grasp success are based on force-torque readings at the wrist (assuming the objects have a significant mass) or image differencing of the workspace below the hand before and after the grasp (assuming that objects fall down if they are not in force-closure). Nevertheless, such estimators will inevitably report false positives and false negatives.

To analyze how the selection strategies behave in these cases, we disturbed the reward signal that is observed by the robot. We randomly used $\text{UPDATE_MODEL}(a_t, 1 - r_t)$ in Algorithm 7.6 instead of $\text{UPDATE_MODEL}(a_t, r_t)$ in a pre-defined ratio of iterations, while still counting the actual achieved reward. We tested the two best performing methods from the previous experiment: ϵ -greedy k-NN and GP-UCB. Both methods were confronted with 0 %, 10 % and 30 % disturbed reward signals; again running for 2000 trials.

The comparison in Fig. 7.9 shows that the ϵ -greedy k-NN is better at dealing with noise. Their performance is already on par with only 10 % injected noise. With the highest level of noise the ϵ -greedy k-NN method even performs slightly better than GP-UCB. This is due to the relative high amount of constant exploration that the ϵ -greedy strategy conducts.

7.3 LIMITATIONS

The simulation data set has shown that, especially for larger objects, the proposed set of strategies might be too limited. Another strong assumption is the presence of environmental opportunities such as edges and walls to detect the proposed strategies.

Both drawbacks can be mitigated by expanding the set of grasp types and exploitable contact structures. Additionally, the algorithm only modifies a few grasp parameters such as the approach direction based on its extracted representation of the environment; the geometric description of the grasp itself has no influence (apart from rejecting it). Allowing modifications of grasp parameters would increase the strategies' applicability. Finally, the proposed geometric features only depend on depth measurements. They do not capture all relevant physical properties needed to predict grasp success, e.g. the baseball was rolling away during a wall grasp while the tennis ball succeeded (see Fig. 7.5). Color features could correlate with such properties (Levine et al., 2016) and allow more accurate predictions.

7.4 RELATED WORK

According to our contributions, we decompose the relevant related work into two categories: prior work that represents grasps based on sensor input to deal with novel scenarios, and approaches that look at how to improve grasp performance based on accumulated experience. For each category we emphasize similarities and differences concerning the problem at hand and the proposed solution.

7.4.1 DESCRIPTORS FOR MODELING GRASP SUCCESS

Since general models based on contact forces require a lot of prior knowledge, a variety of grasp representations have been proposed that describe the object’s local geometry in a grasp-variant frame (Bohg et al., 2014). The advantage of the majority of these representations is that they can be used with raw sensor data, thus, allowing grasp planning for unknown objects. Additionally, they can be learned from different data sources including real-world trials, which circumvents the problems that occur when relying on realistic physics-based simulations (Kappler et al., 2015). Examples are matching depth templates (Herzog et al., 2014), histogram-like features (Fischinger et al., 2015), or image-based descriptors derived with convolutional architectures (Pinto and Gupta, 2015), (Levine et al., 2016). Our method is similar, but differs w.r.t. the fact that our features exploit information of the relationship between object and environment by aligning the depth data to the main axis of the exploited EC. Another difference is that our proposed solution must predict the outcome of a time-extended contact interaction.

7.4.2 GRASP SELECTION AS EXPLORATION VS. EXPLOITATION

The data-driven techniques cited above usually consider a training phase, requiring a large set of grasp examples, followed by a test phase in which the learned grasp model is applied. We focus on the problem of acquiring grasp experiences to make more informed decisions while concurrently trying to grasp as successful as possible. Active learning approaches to grasping favor grasps that have a high probability of success or large uncertainty. Salganicoff et al. (1996) learn approach directions based on object dimensions, while Montesano and Lopes (2012) use expected improvement as exploration criteria to find the most promising grasping point based on image features. Kroemer et al. (2010) were the first to use a contextual bandit setting to find grasps. They estimate the reward and its uncertainty of grasps in $SE(3)$ via Gaussian process regression and apply a variant of UCB (Auer, 2002) to choose a grasp. Given

a grasp model that is expensive to evaluate, the bandit setting can also be used to find good grasps with as few samples as possible. [Laskey et al. \(2015\)](#) present a 2D grasping model based on uncertainty in shape, pose, friction coefficient, and gripper approach and explore grasps using a Bayesian MAB algorithm. A similar model was extended to 3D ([Mahler et al., 2016](#)). Here, grasps are represented as local depth maps along the approach direction. In both of these models the gap between simulation and reality was not considered. Bandit formulations are also used for large-scale grasp acquisition on real robotic systems ([Oberlin and Tellex, 2015](#), [Pinto and Gupta, 2015](#)).

7.5 CONCLUSION

We presented a grasp planner for unknown environments capable of exploiting contact with the environment. To develop our method, we relied on three contact-exploiting grasping strategies. We showed that it is possible to learn perceptual models that predict the success of these strategies based on a single depth image alone. We formulated the problem of deciding among this small number of grasping strategies while improving the perceptual models' predictive power from experience as a contextual multi-armed bandit problem. The quality of the decision improves by incorporating context in the form of perceptual information.

8

Beyond Grasping: Learning Dexterous Manipulation for Soft Hands

WE HAVE SHOWN in this thesis that compliant hands can be effectively used for grasping. But what about more dexterous manipulation tasks? In the context of grasping, adaptability between hand and object – passive or active – is a beneficial effect. It allows us to derive more contact-rich strategies that take advantage of the environment. But for dexterous manipulation, i.e., the deliberate control of the DOF of an object, passive adaptability might be an insurmountable obstacle. Additionally, soft grippers usually lack sensing and precise actuation. In this final chapter we want to tackle the challenges posed by soft mechanisms in the context of manipulation. We will present an approach based on learning from demonstrations (LfD) and reinforcement learning (RL) to solve simple manipulation tasks with the RBO Hand 2.

In LfD, the robot observes a human teacher solving a task and learns how to perform the demonstrated task and apply it to new situations. Many manipulation tasks can be fully defined by demonstrating the motion of one or multiple objects. These kinds of *object-centric* demonstrations are intuitive and easy to provide. But because the robot does not directly control the DOF of objects in the world, they cannot be imitated directly. One crucial challenge that we must address is to account for the mismatch between the morphology of the human expert and the robot. We propose a reinforcement learning method to reproduce object-centric demonstrations. Our algorithm automatically selects and blends demonstra-

tions that the robot can follow most closely, while ignoring infeasible ones.

Our goal is to find a unified control policy that can generalize to a variety of initial states. To achieve generalization, we train a single nonlinear neural network policy to reproduce the behavior of multiple object-centric demonstrations. This approach follows the framework of guided policy search (GPS) by Levine and Abbeel (2014) which is effective in learning high dimensional control policies for robots using a very small number of samples. However, unlike standard GPS, our approach requires only a set of object-centric demonstrations from a human expert to learn a new skill, rather than hand-specified cost functions. We demonstrate our approach on the RBO Hand 2, with learned motor skills for turning a valve and manipulating an abacus.

8.1 OVERVIEW OF THE ALGORITHM

We take advantage of two general concepts to find manipulation strategies for soft robotic hands: imitating human demonstrations and reinforcement learning. In order to learn from human demonstrations, we exploit task-specific information offered by human demonstrators using object-centric demonstrations, i.e., we only capture the motion of the object being manipulated, not hand-specific information. We use reinforcement learning to learn a policy which imitates these object-centric demonstrations. However, due to kinematic and dynamic differences between the human hand and the RBO Hand 2, it might be impossible to follow some of these demonstrations. Hence, trying to imitate them closely is undesirable. We describe a novel demonstration selection algorithm that selects *which* demonstration to imitate, and uses a reinforcement learning method to solve the problem of *how* to imitate.

We define our learning problem as optimizing a policy π_θ to perform the demonstrated task by learning from demonstrations. The policy maps observations to actions and is fully parametrized by θ .

In order to learn this policy, we first train multiple different local controllers to imitate the most closely achievable demonstration from their respective initial states. This involves solving the joint problem of selecting the appropriate demonstration for each controller, and using reinforcement learning to train each controller to actually follow its chosen demonstration. By modeling the objective as a minimization of KL divergence between a distribution of controllers and a mixture of demonstrations modeled as Gaussians (see Sec. 8.2), this joint problem reduces to an alternating optimization between computing correspondence weights that assign a demonstration to each controller, and optimizing each controller using an opti-

Algorithm 8.7 Guided Policy Search with Demonstration Selection

```

for iteration  $k = 1$  to  $K$  do
    Generate samples  $\{\bar{\tau}_j\}$  from each controller  $p_j(\bar{\tau})$  by running it on the soft hand.
    Compute soft correspondence weights  $a_{ij}$ 
    Estimate system dynamics  $p(x_{t+1}|x_t, u_t)$  from  $\{\tau_j\}$ 
    for iteration  $inner = 1$  to  $n$  do
        Perform optimal control to optimize objective defined in Section 8.2
        Perform supervised learning to match  $\pi_\theta$  with the samples  $\{\bar{\tau}_j\}$ 
    return  $\theta$                                  $\triangleright$  the optimized policy parameters

```

mal control algorithm. This algorithm can be used within the BADMM-based guided policy search framework (Levine et al., 2015), to train a neural network policy π_θ to generalize over the learned controllers. Thus, the three phases of our algorithm are (Alg. 8.7):

1. Perform a weight assignment which computes soft correspondences between demonstrations and individual controllers (Sec. 8.2).
2. With the soft correspondences fixed, solve an optimal control problem based on the correspondences and deviations from individual demonstrations (Sec. 8.3).
3. Perform supervised learning over the trajectory distributions from the optimal control phase, using the framework of BADMM-based GPS (Sec. 8.4).

8.2 LEARNING CONTROLLERS FROM MULTIPLE DEMONSTRATIONS

As the first step to generalizing dexterous manipulation skills, we learn a collection of controllers starting from different initial conditions, such that each controller imitates the demonstration which is most closely achievable from its initial condition. This problem can be cast as minimizing the divergence between two distributions: one corresponding to the demonstrated trajectories, and the other related to the controllers.

For our given dynamical system, we define the states to be x_t , and the actions to be u_t at every time step t . The system dynamics are specified by the model $p(x_{t+1}|x_t, u_t)$. Each controller j is defined in terms of a conditional distribution $p_j(u_t|x_t)$. Together with the dynamics model $p(x_{t+1}|x_t, u_t)$ the controller induces a distribution

$$p_j(\tau) = p_j(x_0) \prod p(x_{t+1}|x_t, u_t) p_j(u_t|x_t)$$

over trajectories $\tau = x_1, u_1, \dots, x_T, u_T$, where T is the length of an episode. We define

$$p(\tau) = \sum_{j=1}^C \frac{1}{C} p_j(\tau)$$

to be the uniform mixture of C individual controllers $p_j(\tau)$. Our state x_t can be expressed as $x_t = [\bar{x}_t, x'_t]$, where \bar{x}_t denotes the *object-centric* parts tracking the manipulated objects and x'_t is the rest of the state. In our experiments, \bar{x}_t consists of positions and velocities of motion capture markers placed on manipulated objects. Our objective is to match the controllers with the demonstrations but only over the object-centric elements (\bar{x}) of the state. We marginalize $p(\tau)$ to obtain $p(\bar{\tau})$, which is a uniform mixture of C distributions $p_j(\bar{\tau})$, such that

$$p(\bar{\tau}) = \sum_{j=1}^C w_j p_j(\bar{\tau}),$$

where $w_j = \frac{1}{C}$ and $\bar{\tau} = \{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_T\}$. This distribution is over just the object-centric trajectories $\bar{\tau}$.

The distribution of D demonstrations over the trajectories $\bar{\tau}$ is modeled as a mixture of multivariate Gaussians. It is given by

$$d(\bar{\tau}) = \sum_i^D v_i d_i(\bar{\tau}) = \sum_i^D v_i \mathcal{N}(\mu_i, \Sigma_i),$$

where $\mu_i = \{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_T\}$ is the trajectory of the objects recorded in each demonstration, and the covariance Σ_i is a parameter that decides how closely the demonstration needs to be tracked by the controller. The number of demonstrations and controllers do not have to be the same.

Our goal is to match the distribution of demonstrations with the distribution of controllers, which we formalize as a KL divergence objective:

$$\min_{p(\tau)} D_{KL}(p(\bar{\tau}) || d(\bar{\tau})).$$

Although the objective is defined with respect to the object-centric distributions $p(\bar{\tau})$, the optimization is done with respect to the entire controller mixture $p(\tau)$ which includes other parts of the state, and actions.

Due to the mode seeking behavior of the KL divergence, this objective encourages each $p_j(\bar{\tau})$ to match the closest achievable demonstration. However, the KL divergence between mixtures cannot be evaluated analytically. Methods such as MCMC sampling can be used to estimate it, but we find a variational upper bound (Hershey and Olsen, 2007) to be most suitable for our formulation. In order to simplify our objective, we decompose each mixture weight w_j and v_i into individual variational parameters a_{ij} and b_{ij} , such that:

$$\sum_i a_{ij} = w_j \quad \text{and} \quad \sum_j b_{ij} = v_i .$$

We can rewrite

$$\begin{aligned} D_{KL}(p(\bar{\tau})||d(\bar{\tau})) &= \int p(\bar{\tau}) \log \frac{p(\bar{\tau})}{d(\bar{\tau})} \\ &= \int -p(\bar{\tau}) \log \frac{\sum_{i,j} b_{ij} d_i(\bar{\tau})}{p(\bar{\tau})} \\ &= - \int p(\bar{\tau}) \log \sum_{i,j} \frac{b_{ij} d_i(\bar{\tau})}{a_{ij} p_j(\bar{\tau})} \frac{a_{ij} p_j(\bar{\tau})}{p(\bar{\tau})}. \end{aligned}$$

From Jensen's inequality we get an upper bound as follows:

$$\begin{aligned} D_{KL}(p(\bar{\tau})||d(\bar{\tau})) &\leq - \int p(\bar{\tau}) \sum_{i,j} \frac{a_{ij} p_j(\bar{\tau})}{p(\bar{\tau})} \log \frac{b_{ij} d_i(\bar{\tau})}{a_{ij} p_j(\bar{\tau})} \\ &= - \sum_{i,j} \int p_j(\bar{\tau}) a_{ij} \log \frac{b_{ij} d_i(\bar{\tau})}{a_{ij} p_j(\bar{\tau})} \\ &= \left[\sum_{i,j} a_{ij} \int p_j(\bar{\tau}) \log \frac{p_j(\bar{\tau})}{d_i(\bar{\tau})} \right] - \left[\sum_{i,j} a_{ij} \log \frac{b_{ij}}{a_{ij}} \right] \\ &= \sum_{i,j} a_{ij} D_{KL}(p_j(\bar{\tau})||d_i(\bar{\tau})) + D_{KL}(a||b). \end{aligned}$$

Thus, our optimization problem becomes

$$\min_{p(\tau), a, b} \sum_{i,j} a_{ij} D_{KL}(p_j(\bar{\tau})||d_i(\bar{\tau})) + D_{KL}(a||b). \quad (8.1)$$

While the first term $\sum_{i,j} a_{ij} D_{KL}(p_j(\bar{\tau})||d_i(\bar{\tau}))$ depends on the distribution $p(\tau)$, the second term $D_{KL}(a||b)$ depends on the mixing components a_{ij} and b_{ij} but is independent

of the distribution $p(\tau)$. Hence, we can perform the optimization in two alternating steps. We first optimize $D_{KL}(p(\bar{\tau})||d(\bar{\tau}))$ with respect to a and b , followed by an optimization of $D_{KL}(p(\bar{\tau})||d(\bar{\tau}))$ with respect to $p(\tau)$. This gives us a block coordinate descent algorithm in $\{a, b\}$ and p . The convergence of the algorithm is guaranteed by the convergence of the block coordinate descent method on a quasiconvex function and the fact that KL divergence is quasiconvex. The convergence properties of the weight assignment phase are shown by [Hershey and Olsen \(2007\)](#).

Intuitively, the first optimization with respect to a, b is a weight assignment with the correspondence weight a_{ij} representing the probability of assigning demonstration i to controller j . The second optimization with respect to $p(\tau)$, keeps the correspondence parameters a, b fixed, and finds optimal controllers using an optimal control algorithm to minimize a weighted objective specified in Eq. 8.2.

WEIGHT ASSIGNMENT PHASE

The objective function $D_{KL}(p(\bar{\tau})||d(\bar{\tau}))$ is convex in both a and b . Thus, we can optimize it by keeping one variable fixed, and vice versa. This yields the following closed form solutions:

$$b_{ij} = \frac{v_i a_{ij}}{\sum_{j'} a_{ij'}} \quad \text{and} \quad a_{ij} = \frac{w_j b_{ij} e^{-D_{KL}(p_j(\bar{\tau})||d_i(\bar{\tau}))}}{\sum_{i'} b_{i'j} e^{-D_{KL}(p_j(\bar{\tau})||d_{i'}(\bar{\tau}))}}.$$

In order to compute the optimal a and b , we alternate between these updates for a and b until convergence.

CONTROLLER OPTIMIZATION PHASE

Once the optimal values for a and b have been computed, we fix these as correspondences between demonstrations and controllers and optimize Eq. 8.1 to recover the optimal $p(\tau)$. As a and b are fixed, $D_{KL}(a||b)$ is independent of p . Hence, our optimization becomes:

$$\begin{aligned} \min_{p(\tau)} \sum_{i,j} a_{ij} D_{KL}(p_j(\bar{\tau})||d_i(\bar{\tau})) &= \sum_{i,j} a_{ij} (-E_{p_j(\bar{\tau})} [\log d_i(\bar{\tau})] - H(p_j(\bar{\tau}))) \\ &= \sum_j -w_j H(p_j(\bar{\tau})) - \sum_{i,j} a_{ij} E_{p_j(\bar{\tau})} [\log d_i(\bar{\tau})]. \end{aligned}$$

Factorizing the optimization independently over each of the controller distributions, we optimize the following objective for each controller $p_j(\tau)$:

$$-w_j H(p_j(\bar{\tau})) - \sum_i a_{ij} E_{p_j(\bar{\tau})}[\log d_i(\bar{\tau})] = -w_j \left(H(p_j(\bar{\tau})) + \sum_i \frac{a_{ij}}{w_j} E_{p_j(\bar{\tau})}[\log d_i(\bar{\tau})] \right). \quad (8.2)$$

In practice, the weight assignment is performed independently per time step, as the controllers we consider are time varying.

8.3 CONTROLLER OPTIMIZATION WITH AN LFD OBJECTIVE

While the controller optimization phase could be performed with a variety of optimal control and reinforcement learning methods, we choose a simple trajectory-centric reinforcement learning algorithm that allows us to control systems with unknown dynamics, such as soft hands. Building on prior work, we learn time-varying linear Gaussian controllers by using iteratively refitted time-varying local linear models (Levine and Abbeel, 2014). This assumes Gaussian system noise and has been shown to work well in practice for robotic manipulation systems. Action-conditionals for the time-varying linear-Gaussian controllers are given by

$$p_j(u_t|x_t) = \mathcal{N}(K_{jt}x_t + k_{jt}, C_{jt}),$$

where K_{jt} is a feedback term and k_{jt} is an open loop term. Given this form, the maximum entropy objective (Eq. 8.2) can be optimized using differential dynamic programming (Jacobson and Mayne, 1970, Levine and Koltun, 2013). Since $d_i(\bar{\tau})$ is a multivariate Gaussian $\mathcal{N}(\mu_i, \Sigma_i)$, we can rewrite the optimization problem in Eq. 8.2 as

$$\min_{p_j(\tau)} \sum_{t,i} \frac{a_{ijt}}{\sum_{i'} a_{i'jt}} E_{\bar{x}_t \sim p_j(\bar{\tau})} \left[\frac{1}{2} (\bar{x}_t - \mu_{it})^T \Sigma_i^{-1} (\bar{x}_t - \mu_{it}) \right] - H(p_j(\bar{\tau})),$$

where we express the objective as a sum over individual time steps. In this maximum entropy objective, the cost function is defined as the expectation of a sum of l_2 distances of trajectory samples to each demonstration, weighted by the normalized correspondence weights $\frac{a_{ij}}{\sum_{i'} a_{i'j}}$. The trajectory samples denote the trajectories of the objects which we recover through object markers, and we compute the l_2 distance of these samples to the object-centric demonstrations.

Under linearized dynamics, this objective can be locally optimized using iLQG (Li and Todorov, 2004). However, for robots like the RBO Hand 2, the dynamics are complex and difficult to model analytically. Instead, we can fit a time-varying locally linear model of the dynamics to samples obtained by running the physical system in the real world:

$$p(x_{t+1}|x_t, u_t) = \mathcal{N}(F_{xt}x_t + F_{ut}u_t | C_d).$$

The dynamics matrices F_{xt} and F_{ut} can then be used in place of the system linearization to optimize the controller objective using iLQG. It is important to note here that the iLQG optimization learns K_{jt} , k_{jt} and C_{jt} , while the term $\frac{a_{ij}}{\sum_i a_{i'j}}$ is learned in the weight assignment phase and kept fixed for the iLQG optimization.

One issue with optimizing a controller using fitted local dynamics models is that the model is only valid close to the previous controller. The new controller might visit very different states where the fitted dynamics are no longer valid, potentially causing the algorithm to diverge. To avoid this, we bound the maximum amount the controller changes between iterations. This can be expressed as an additional constraint on the optimization:

$$D_{KL}(p_j(\tau) || \hat{p}_j(\tau)) < \epsilon, \quad (8.3)$$

where $\hat{p}_j(\tau)$ is the previous trajectory-controller and $p_j(\tau)$ the new one. As shown by Levine et al. (2015), this constrained optimization problem can be formulated in the same maximum entropy form as Eq. 8.2 using Lagrange multipliers. It is solved via dual gradient descent (for details and a full derivation see Levine and Abbeel (2014) and Levine et al. (2015)). In practice, each iteration of this controller optimization algorithm involves

1. generating N samples on the real physical system by running the previous controller,
2. fitting a time-varying linear dynamics model to these samples,
3. and optimizing a new controller $p_j(\tau)$ by solving the constrained optimization using dual gradient descent, with iLQG used to optimize with respect to the primal variables K_{jt} , k_{jt} , and C_{jt} .

This can be viewed as an instance of model-based reinforcement learning.

8.4 SUPERVISED LEARNING USING GUIDED POLICY SEARCH

The multiple controllers defined in the previous section learn to imitate the most closely imitable demonstration from their individual starting positions. However, given an unseen initial state, it is not clear which controller $p_j(\tau)$ should be used. For effective generalization, we need to obtain a single policy $\pi_\theta(u_t \mid x_t)$ that will succeed under various conditions. To do this, we extend the framework of guided policy search (Levine et al., 2015) to combine controllers into a single nonlinear neural network policy.

We learn the parameters θ of a neural network π_θ to match the behavior shown by the individual controllers by regressing from the state x_t to the actions u_t taken by the controllers at each of the N samples generated on the physical system. In general, a simple supervised learning setting is not guaranteed to produce a policy with good long-horizon performance. In fact, supervised learning is effective only when the state distribution of π_θ matches that of the controllers $p_j(\tau)$.

To ensure this, we use the BADMM-based variant of GPS (Levine et al., 2015), which modifies the cost function for the controllers to include a KL-divergence term to penalize deviation of the controllers from the latest policy π_θ at each iteration. This is illustrated in Algorithm (8.7), by first assigning correspondences between demonstrations and controllers, then alternating between trajectory optimization and supervised learning at every iteration, eventually leading to a good neural network policy π_θ .

8.5 AN ILLUSTRATIVE EXAMPLE

In order to illustrate our method, we will apply it to a simple low-dimensional control problem: moving a unit mass particle on a line to a target position. This example allows us to gain a better understanding of the iterative learning process by visualizing the resulting policies, sampled trajectories, and weight assignments.

The state of the one-dimensional system is described by the particle's position $x \in \mathbb{R}$ and velocity \dot{x} . The action $u \in \mathbb{R}$ is a force that can be applied (or acceleration since the particle has a unit mass). We provide $D = 2$ demonstrations of our positioning task: $x_{d_1}(t) = -10$ and $x_{d_2}(t) = 10$. All demonstrations are constant over all time steps of an episode. We learn $C = 2$ controllers, parametrized over 50 time steps. The initial state of the particle is always zero. We initialize the correspondence weights that associate each controller with each demonstration uniformly. The network policy π_θ is a fully connected neural net with two inputs (position x and velocity \dot{x}), two hidden layers and a single output (the force to apply).

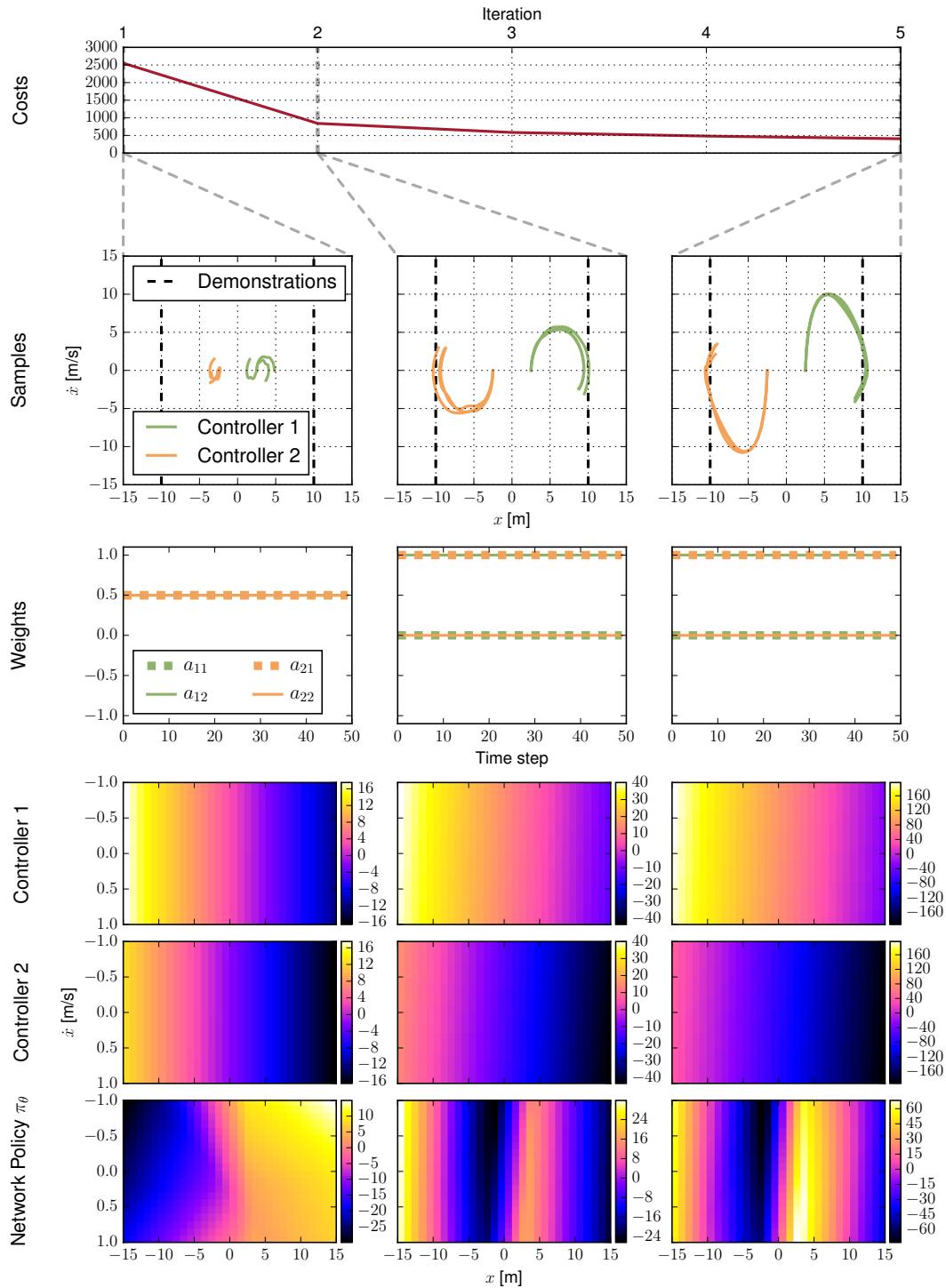


Figure 8.1: The plots illustrate the learning progress. We depict the point-mass system at three different iterations. For each iteration we show the phase portraits of the sampled roll-outs using the two local controllers. See text for details.

We run our learning algorithm for five iterations, each controller collects three samples per iteration.

Fig. 8.1 visualizes the learning progress by emphasizing the state of the network policy and the local Gaussian feedback controllers after one, two, and nine iterations (displayed in three columns). We can see that the cost is reduced rapidly. Remember that the cost is the distance of the controller samples to the demonstrations, weighted by the learned correspondence weights. The samples which are roll-outs of the two controllers are shown in the second row as phase portraits. Notice that they are quickly getting closer to the desired demonstrations marked by dashed vertical lines. Since we do not penalize the control magnitude with an additional cost term, we can see that the sample runs increase in velocity with each iteration to arrive faster at the goal locations. The correspondence weights (shown in the third row) also converge quickly to reasonable values. The weight a_{ij} indicates how much controller i imitates demonstration j . Initially, these weights are all the same (0.5). Already after the first learning iteration, they converge to $a_{11} = a_{22} = 0$ and $a_{12} = a_{21} = 1$. This means that controller 1 decides to always imitate demonstration 2, while controller 2 imitates demonstration 1.

The last three rows visualize the two controllers which are optimized with iLQG and the final neural network policy. Since they are time-variant, we depict them for the time step $t = 25$. The color of the phase portraits indicate the action u that would be taken in the state (x, \dot{x}) . We can observe that the actions of controller 1 are positive ($u > 0$) to reach the target position $x_{d_2}(t) = 10$. Controller 2 shows the opposite behavior to reach target $x_{d_1}(t) = -10$. The network policy π_θ generalizes the behavior of both controllers. If the position $x > 0$ it drives the particle towards x_{d_2} and vice versa. Overall, the fast convergence is due to the simple nature of the problem.

8.6 CONTROLLING THE RBO HAND 2

The RBO Hand 2 (see Sec. 1.1.5) is an inexpensive, compliant, underactuated robotic hand which has been shown to be effective for a variety of grasping tasks. It consists of seven actuators which are controlled via external air valves and a separate air supply. Control is challenging since the air valves can only be either fully closed or fully open and have a switching time of ~ 0.02 s. Each actuator has a pressure sensor located close to the air valve.

The hand is controlled by specifying valve opening durations to either inflate or deflate a single actuator. We turn the discrete valve actions into a continuous control signal using pulse

width modulation. Given a constant frequency of 5 Hz, the control signal is interpreted as the duration the inflation (if it is positive) or deflation (if it is negative) valve is opened during a single time step. To ensure that the control signal does not exceed the duration of a single time step we apply a sigmoid function to the commands from the learning algorithm.

The positions and velocities of the manipulated objects are captured in real time with a PhaseSpace Impulse system, which relies on active LED markers. The state x_t of our system is the concatenation of the seven pressure readings of the hand, their time derivatives, the 3D positions and velocities of markers attached to the object, and joint angles of the robot arm (depending on the task). We placed no LED markers on the hand itself, only the object was equipped to record object-centric demonstrations, and the positions and velocities of these markers constitute the object-centric state \bar{x}_t .

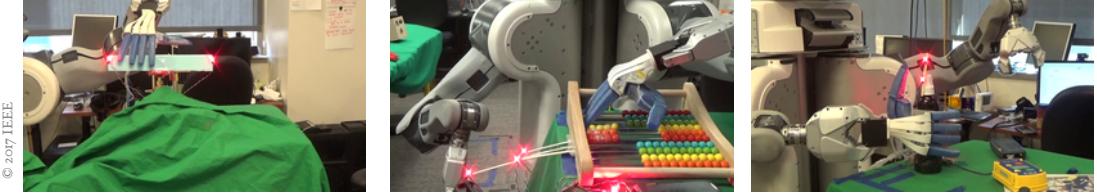
8.7 EXPERIMENTS

We evaluated our algorithm on a variety of manipulation and grasping tasks. Our experiments aim to show that:

1. It is possible to perform fine manipulation with the RBO Hand 2.
2. Our algorithm can learn feedback policies from demonstrations that perform nearly as well as an oracle with the correct demonstrations (depending on the context) manually assigned to controllers.
3. A single neural network policy learned from demonstrations with our method is able to generalize to different initial states.

We will evaluate our learning approach on three different tasks: turning a valve, pushing beads on an abacus, and grasping a bottle (see Video Fig. 8.1). For the first two tasks, we compare our method to the following baselines:

- **HAND DESIGNED BASELINE:** A controller with a hand-designed open loop policy. In the case of the abacus task, we evaluate the performance of two different strategies for simple hand-designed baselines.



Video Figure 8.1: The three manipulation tasks used in our experiments: Turning a valve, pushing beads on an abacus, and grasping a bottle from a table. [<https://youtu.be/XyZFkJWu0Q0>]

- **SINGLE DEMO BASELINE:** A single controller trained to imitate a single demonstration. We use two separate baselines which are trained to follow different demonstrations.
- **ORACLE:** Depending on the context we manually assign the correct achievable demonstration to controllers. This comparison is useful to test whether the correspondence assignments are accurate.

8.7.1 TURNING A VALVE

Rotating a gas valve is a challenging task since it involves coordinating multiple fingers. Our valve consists of a blue horizontal lever that increases its range of motion (Video Fig. 8.1). Varying wrist positions along the lever require different finger motions to rotate it.

We mount the RBO Hand 2 on a PR2 robot arm, with the objective to rotate the valve away from the initial center position in either direction, using just its fingers. The arm is kept stationary for each episode, but changes positions between the training of different controllers. The joint angles are part of the state to determine the relative position of the hand with respect to the valve.

A human demonstrated three different valve rotations with their own hand, while two LED markers tracked the motion of the lever. Two demonstrations were of the valve rotating clockwise and anti-clockwise at the same position, and a third demonstration with the valve placed at a different position and rotated anticlockwise. All three demonstrations are valid for the task, but not all of them are achievable from every training position. Our algorithm trained three individual controllers and a single neural network policy to generalize over them.

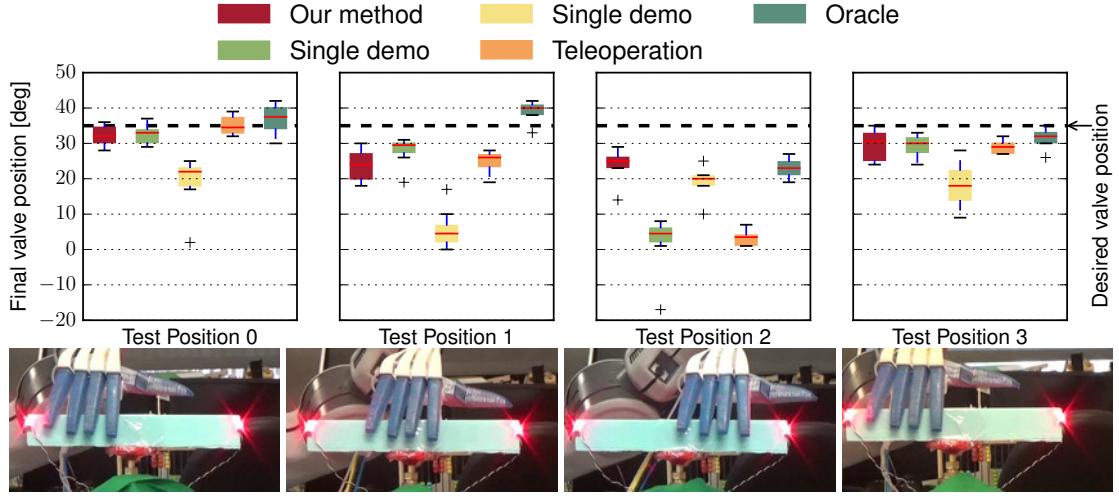


Figure 8.2: Comparison of different policies for the valve task: the dashed black line indicates the demonstrated rotation of the valve by approximately 35° . On average our method learns the most general feedback strategy. The box plot compares five methods for four different test positions. Although the baselines do well in some positions, the only methods which do consistently well across all positions are our method and the oracle.

RESULTS AND DISCUSSION

During evaluation, the policy learned by each method was sampled ten times at four positions of the hand relative to the valve. The results in Fig. 8.2 show that our method generates the most robust policy compared to the baselines, which each fail to turn the valve significantly in at least one position. Our method does nearly as well as the oracle, for which demonstrations are assigned to controllers manually. While learning the correspondence weights and the individual controller policies, our method determines which of the demonstration it can actually perform from its initial positions, and disregards distant unachievable demonstrations.

Our method is able to learn distinctly different behavior at various test positions. At position 1, the policy pushes the lever using its last two fingers, with support given by the thumb. At position 2, the policy uses the thumb to rotate the valve by pushing the lever as the fingers are blocked. At position 3, our policy extends the thumb out of the way and pushes strongly with the index finger to rotate the valve. Simple open loop hand-designed strategies and the baselines learned from a single demonstration fail to learn this distinctly different behavior needed to generalize to different positions along the valve lever. By learning that different joint angles of the arm require different behaviors to be performed, our method is able to perform the task in various initial conditions.



Bead	Target	Ours	SingleDemo1	SingleDemo2	Oracle	HandDesign1	HandDesign2
1	8.4	7.49 ± 0.47	7.02 ± 0.50	6.33 ± 2.15	7.66 ± 0.23	8.38 ± 0.04	0 ± 0
2	0	0.14 ± 0.18	0.60 ± 0.69	7.08 ± 1.04	0.27 ± 0.42	0 ± 0	6.5 ± 0
3	0	0.89 ± 1.00	0.28 ± 0.18	1.23 ± 2.20	1.08 ± 0.72	0 ± 0	8.43 ± 0.29



Bead	Target	Ours	SingleDemo1	SingleDemo2	Oracle	HandDesign1	HandDesign2
1	8.4	7.95 ± 0.19	1.04 ± 2.15	7.27 ± 0.65	7.52 ± 0.66	0.00 ± 0.00	8.38 ± 0.08
2	0	0.10 ± 0.10	0.85 ± 1.21	0.19 ± 0.14	0.09 ± 0.11	0.00 ± 0.00	8.40 ± 0.00
3	0	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00



Bead	Target	Ours	SingleDemo1	SingleDemo2	Oracle	HandDesign1	HandDesign2
1	8.4	7.21 ± 0.69	2.47 ± 2.22	3.39 ± 1.98	7.74 ± 0.23	0.00 ± 0.00	8.38 ± 0.05
2	0	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
3	0	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00

Table 8.1: Comparison of the distance moved by the various beads in cm using different policies for the abacus task, at 3 different positions, namely Positions 1, 2 and 3 going downwards. The target column in each table indicates the demonstrated movement of the three beads, and the other columns indicate the mean and standard deviation of other methods. On average our method learns the most general feedback strategy besides the oracle

8.7.2 PUSHING THE BEADS OF AN ABACUS

The RBO Hand 2 is required to push particular beads on an abacus while leaving other beads stationary. This task is challenging due to the precise individual finger motions needed to move only the desired beads. The hand is mounted on a stationary PR2 arm (Video Fig. 8.1), while the abacus is moved to several positions. The beads of relevance here are the central yellow, orange and red ones. Markers were attached to each of the three beads to capture their motion. As the position of the abacus with respect to the hand changes, different fingers need to be used. During demonstrations a human pushed only the yellow beads along their spindle at each of the three positions shown in Fig. 8.1.

RESULTS AND DISCUSSION

We evaluated ten samples of each policy at each of the three positions and recorded the distances that each bead moved. The results are shown in Fig. 8.1. Our method moves the bead closer to the target position than the single demonstration and hand designed baselines for all the test positions. Only the oracle policy produces equally good performance. By interleaving selection of the right demonstration to imitate, with optimal control and supervised learning, our algorithm is able to learn a policy which uses discretely different fingers depending on the positions of the abacus relative to the hand. On the other hand, the hand-designed baselines being open loop can never learn different behaviors for different fingers. The controller trained at a single position fails because it has no notion of generalization.

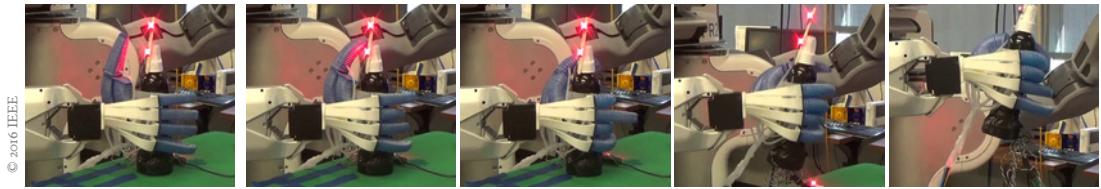


Figure 8.3: Execution of a learned policy to grasp a bottle. Our method learns to grasp the bottle tightly and performs as well as the hand designed baseline.

8.7.3 GRASPING A BOTTLE

This task involves using the soft hand mounted on a moving arm, to grasp a deodorant bottle placed on a table. The arm has a scripted motion of moving up after 8 seconds, and we use reinforcement learning to learn the behavior of the fingers to go with this arm motion. The objective of the task is to grasp the bottle before the arm starts moving and keep it grasped until the end of the episode at the final arm location.

As grasping tasks for several objects largely succeed in open loop, our aim is to demonstrate that we can match the performance of a hand-designed baseline with a controller learned from a human demonstration through optimal control. This experiment is challenging for reinforcement learning algorithms due to the delayed nature of the reward signal in grasping.

We provide a demonstration of the bottle being lifted by a human, and use it to define the cost function for trajectory optimization as the l_2 distance of trajectory samples from the provided demonstration. We also apply a Gaussian filter to the noise generated in the controllers to be more temporally coherent, allowing tight grasping. The resulting learned control policy is then tested on 10 sample trajectories in order to evaluate whether a successful grasp has occurred where the object is lifted and kept at the maximum arm height.

RESULTS AND DISCUSSION

We find that on the grasping task, the control policy learned through optimal control does just as well as a hand-designed policy on ten samples of grasping the bottle. Both the hand-designed policy and the learned policy were able to grasp the bottle for all 10 test samples. This indicates that the learning has comparable results to a hand-designed baseline, despite not having prior information besides a human-provided demonstration.

8.7.4 LIMITATIONS

Although the LfD algorithm shows good performance on several tasks using the RBO Hand 2, there are many directions for future work. Instead of using a motion capture system, we hope to use better computer vision techniques such as deep convolutional nets to track trajectories of relevant feature points in future work. Extending the neural network policy to learn policies dependent on just the pressure sensors in the fingers and/or additional tactile sensors, would be an exciting future direction.

8.8 RELATED WORK

8.8.1 DEXTEROUS MANIPULATION USING PLANNING

A variety of methods for generating manipulation behaviors with multi-fingered hands are based on planning. These approaches assume that a detailed model of the hand and object is available *a priori*. They generate open-loop trajectories that can be executed on real hardware. There exist planners that integrate contact kinematics, non-holonomic motion constraints, and grasp stability to come up with manipulation plans based on finger gaits (Han and Trinkle, 1998), rolling and sliding fingertip motions (Cherif and Gupta, 1999), or nonprehensile actions involving only the palm (Bai and Liu, 2014). Optimization-based techniques (Mordatch et al., 2012) have also been used for in-hand manipulation tasks. All of these approaches rely on detailed models, or make simplifying assumptions about the system. Modeling and simulating the behavior of a soft hand like the RBO Hand 2 is computationally expensive, since it requires finite-element method models (Polygerinos et al., 2015) to achieve accuracy. Moreover, it is extremely hard to do accurate system identification on such systems. In order to tackle this problem, our approach does not rely on detailed *a priori* models but learns the task-specific consequences of actions from interactions of the real hardware with the environment, during a task. Li et al. (2013) present a reactive control strategy to locally manipulate objects in a bi-manual setup. The method does not require rich object models but it is based on rich sensor feedback. In contrast, our method needs to deal with noisy, ambiguous pressure readings.

8.8.2 REINFORCEMENT LEARNING FOR MANIPULATION

In order to avoid planning with fixed handcrafted models, control policies that solve continuous manipulation problems can be found using reinforcement learning. A widely used

approach is to learn the parameters of a dynamic motor primitive (Ijspeert et al., 2003) (DMP) with relative entropy policy search (Peters et al., 2010) or PI² (Theodorou et al., 2010). This has been used to learn striking movements (Mülling et al., 2013) and bi-manual transportation tasks (Kroemer et al., 2015). Although DMPs are often used to describe the kinematic state of a system, they can be used to generate compliant behavior for picking up small objects or opening doors (Kalakrishnan et al., 2011). However, DMP's typically require either a model of the system or the ability to control kinematic state, neither of which is straightforward on a soft hand that lacks position sensing.

Controllers for reaching and grasping have been learned by approximating the Q-function with a multilayer perceptron (Lampe and Riedmiller, 2013). Policy search methods have succeeded in training neural network controllers to solve contact-rich peg-in-hole-like tasks (Levine and Abbeel, 2014) based on positional or visual feedback (Levine et al., 2015).

Some RL methods for manipulation have been applied to in-hand manipulation. van Hoof et al. (2015) learn a policy based on tactile feedback which lets an underactuated hand slide cylindrical objects horizontally while being rolled between two fingers. Similar to our work is the learning method for an in-hand rotation tasks by Kumar et al. (2016). In contrast, we learn global policies that aim to generalize local solutions.

8.8.3 EXPLOITING HUMAN DEMONSTRATIONS FOR LEARNING MANIPULATION SKILLS

Learning from demonstrations has been effective in teaching robots to perform manipulation tasks with a limited amount of human supervision. By building statistical models of human demonstrations, gestures (Calinon and Billard, 2007) and dual-arm manipulations (Asfour et al., 2008) have been reproduced on robotic systems. Pure LfD can lead to suboptimal behavior when demonstrator and imitator do not share the same embodiment. To circumvent this problem the learning objective is often extended with additional feedback. This can be provided by a human, e.g. in the case of iteratively improving grasp adaptation (Sauser et al., 2012). Alternatively, demonstrations can provide the coarse structure of a solution, while the details are iteratively refined and learned by the imitator itself. This has been shown for dexterous manipulation (Prieur et al., 2012) where an in-hand manipulation is broken down into a sequence of canonical grasps.

In combination with reinforcement learning, demonstrations often serve as an initial policy rollout or they constrain the search space by providing building blocks. This has been applied to reaching motions (Guenther et al., 2007) and dynamic manipulation tasks.

8.9 CONCLUSIONS

We presented an algorithm for learning dexterous manipulation skills with a soft hand from object-centric demonstrations. Unlike standard LfD methods, our approach only requires the human expert to demonstrate the desired behaviors with their own hand. Our method automatically determines the most relevant demonstrations to track, using reinforcement learning to optimize a collection of controllers together with controller to demonstration correspondences. To generalize the demonstrations to new initial conditions, we utilize the GPS framework to train nonlinear neural network policies that combine the capabilities of all of the controllers.

We evaluated our method on the RBO Hand 2 and showed that it is capable of learning a variety of dexterous manipulation skills, including valve turning and moving beads of an abacus. Although these tasks are rather simple, the results are promising. They show that despite the challenges associated with adaptable hands (lack of sensing, hard to model, noisy actuation) more elaborate manipulation skills beyond the common *grope and hope* (Mason, 2015) paradigm are possible.

9

Conclusion

IN THIS THESIS, we have tackled the problem of planning robotic grasps by focusing on two ideas: how can we effectively use the adaptability of hands and how can we exploit the environmental features surrounding the target object. We now summarize our main insights and revisit the core challenges of robot grasping from the introduction. We will finish this chapter by reflecting on the limitations of our approach and fruitful future research directions revolving around our view on grasping.

We restricted our investigations in Part I towards the interactions between hand and object. We initially observed and analyzed human grasp performance using robotic hardware. By increasingly limiting the feedback and grasp decisions the human could take, we could show that grasp success is not severely hampered. Thus, the underlying temporal and spatial structure of the grasps must be much lower-dimensional than the observed behavior. This effect is amplified when considering compliance and adaptability in the grasping mechanism. Even when initial conditions such as object geometry and pose are varied, the same commanded actions lead to a successful grasp, because fingers passively comply and wrap around different shapes or they pull the object towards the palm. The human operators often used only two different pre-shapes to configure the hand's internal DOF and selected a specific pre-grasp pose of the hand to grasp many different objects successfully. We called the closing motion of the fingers in conjunction with a certain pre-shape a compliance mode. We further investigated the relation between a compliance mode and the pre-grasp pose, resulting in the Sun-Flower-Annulus hypothesis (SFA, Sec. 2.2). SFA showed that the size of an object (rel-

ative to the hand) significantly affects the sensitivity of grasp success w.r.t. to a specific pre-grasp pose. The larger an object the more important it is to correctly match the compliance mode with the object’s shape. We could also show that e.g. for a compliance mode with a cylindrical pre-shape the pre-grasp pose should be orthogonal to the largest principal axis of elongated objects (something that was also shown in human reaching actions by [Wentworth et al. \(2000\)](#)). Based on this grasp representation we derived two grasp planning algorithms. The first one used active vision to extract the information for the appropriate compliance mode and pre-grasp pose of unknown objects. The second method used a depth image and did not depend on excessive motions to extract this information. The experiments showed that the algorithms choose the most appropriate compliance mode for different objects and achieved significant grasp success.

In Part II, we ignored objects, but rather focused on the effects between hand and environment. We derived two methods that exploit contact with the environment to increase task success. The first one is based on searching through a continuous state and action space and explicitly represents uncertainty due to noisy motion execution. Since we encoded the fact that contact observations are uncertainty-free, the resulting motion policies are concatenations of free-space movements and guarded moves or slides across surfaces. We showed that the resulting search space is still feasible for motion problems with a 7-DOF arm and that the policies are also robust when artificially adding noise. Our second method did not represent uncertainty but was based on a discretization of the sensed environment. We matched environmental features with contact exploiting actions such as sliding, wall-constrained, surface-constrained, or edge-constrained grasping. Due to this discretization the resulting search space is much smaller compared to the first method. We called these types of actions environmental constraint exploitations (ECE). Sequences of ECEs can be used to grasp objects in a variety of environments. We also showed that these environmental-constrained grasps are more powerful than those that ignore or avoid the environment.

Part III finally included all components: the hand, object, and environment. We showed that we can extend our grasp planner to include object properties. This was done locally by adapting grasp parameters to account for object properties. But we also showed how to do this globally, by selecting the most promising EC grasp. Our presented approach, learned from experience that flat objects should preferably be grasped by first pulling them to an edge. It was also learned that non-flush objects are successfully picked up by using a wall-constrained grasp. We then investigated a scenario in which learning is an incremental, never-ending process. The robot repeatedly has to decide whether to chose a known grasp or ex-

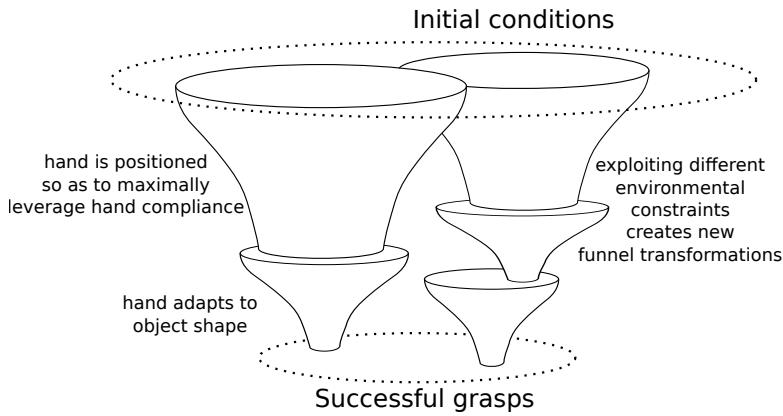


Figure 9.1: We visualize grasping algorithms as consecutive funnels, transforming initial configurations into successful grasps through the use of hand compliance and environmental constraints.

plore a new grasp which might work better. Our results showed that it is best to take the uncertainty of the expected result into account. But it also showed that a simpler model is less prone to disturbances of the detected grasp success. We finally investigated how a soft hand such as the RBO Hand 2 could be used in manipulation problems that do not exhibit the low-dimensional structure like grasping. Although models for such soft mechanisms are hard to derive manually and sensing capabilities are limited, we presented an approach that could learn simple manipulations based on trial-and-error.

9.1 DISCUSSION OF GRASPING CHALLENGES

We defined three main challenges as part of the grasping problem in the introduction (see Chapter 0). We will now revisit them and discuss how our thesis approached them.

GRASP DECISIONS ARE HIGH-DIMENSIONAL: Any approach that synthesizes robot grasps needs to deal with the high dimensionality of grasping motions. In this thesis, we used two ideas to guide the process of reducing this dimensionality: exploiting hand compliance and environmental stiffness. We extracted a reduced set of compliance modes based on the regularities found in interactions between hand and object. Furthermore, we used prototypical interactions between hand and environment to enhance the grasping repertoire. As a result, this discretized action space can be searched effectively while still being expressive enough to allow for general and robust grasping behavior.

We can now revisit and extend our funnel view of grasping (Sec. 2.3). Originally, this view included funnel transformations that were due to hand compliance. By incorporating the idea of contact with the environment during grasping, additional funnel transformations can be exploited. Fig. 9.1 is an updated version of Fig. 2.12. It shows grasp planning as the

sequencing and selection of appropriate funnel transformations which are due to compliant interactions between hand, object, and environment.

CONTACT IS HARD TO MODEL: Contact phenomena depend on a multitude of parameters and are computationally expensive to evaluate from first principles. Simulating the effects of contact in the context of grasping does not easily scale to large time horizons and many number of contacts. Thus, many grasp approaches resort to static models or approximations and do not include seemingly irrelevant aspects such as the environment. We also heavily simplify the evaluation of interactions between hand and object by reducing them to a few hand-specific compliance modes. But instead of completely ignoring the environment we identify reoccurring high-level contact interactions between hand and environment. These include the surface-constrained, edge-constrained, and wall-constrained grasps.

In summary, we do not model contact at the level of local small-scale interactions between rigid bodies. Instead we heavily discretize the effect on the object level and extract few reliable interaction phenomena. By doing so, we can also incorporate interactions between hand, object, and environment rather than focusing only on interactions between hand and object.

INFORMATION IS ALWAYS INCOMPLETE: We exploit hand adaptability to reduce the amount of information required. We do this by matching compliance modes to prototypical shapes which do not need to perfectly represent the object geometry as shown in Sec. 3.2. Furthermore, we extract only grasp-relevant information as shown in the algorithm presented in Sec. 3.1. The compliant hands which create large contact patches when enveloping the object also limit the required knowledge. These grasps are robust to significant variations in mass, center of mass, and inertia tensor of the object. Finally, our grasp strategies that exploit the environment succeed for a variety of different objects as was shown in Sec. 7.1.3. By exploiting the regularities in the environment we can further reduce the need to extract detailed object knowledge for grasping.

9.2 LIMITATIONS AND DIRECTIONS FOR FUTURE RESEARCH

We already emphasized the technical limitations of the presented algorithms in each chapter. In the following we list a number of more general limitations of the approach presented in this thesis. These limitations also serve as a starting point for possible future work.

STRONG THEORETICAL BACKGROUND MISSING: We explained the most prominent aspects of the classical model of grasp analysis in Sec. 1.2. This theory has evolved over years and is characterized by a sound and powerful mathematical language. In contrast, our funnel view of grasping sketched in Sec. 2.3 is much more vague and described through analogies. It still lacks a mathematical foundation which allows it to be used in a generative manner.

Our intention with this view was to emphasize the importance of low-level hand adaptability in high-level grasp planning. In the past, the classical view on grasping has not been able to transfer its insights from theory to robot grasping in the real world, dealing with realistic sensor measurements and limited prior object knowledge (see Sec. 1.2.5). We do not think that our funnel view contradicts the classical theory. Ultimately, both views should be merged. A mathematical formalization would simplify this process. There are formal frameworks that investigate this direction such as the theory of soft synergies (Bicchi et al., 2011). We think that it will be beneficial go further in this direction.

TOO SPECIFIC FOR CERTAIN HAND DESIGNS: Although we showed our grasping strategies on different robotic hands, the different actions were always hand-engineered based on the specific morphology and actuation. This process might be tedious when trying to use our approach with novel hands and knowledge about their particular behavior is missing. Algorithm 6.5 can be used to automatically adapt grasp parameters and it will also take the hand design into account. But this search is uninformed and thus only suited to search a rather restricted volume of parameter values locally.

DISCRETIZATION OF GRASP ACTIONS IS TOO RESTRICTIVE: Similar to the previous limitation is the fact that the set of discretized grasp actions that was presented might be too limited and again, requires human engineering intuition to be found. Our strategies were usually structured by moving the wrist and finishing a grasp by closing all finger simultaneously. In some cases, a richer low-level action vocabulary might be useful.

The Barrett Hand is able clamp objects between its fingers (called the “abduction grip” according to the taxonomy by Feix et al. (2009)). This is useful for small objects that cannot be enveloped and pressed against the palm. It is often more stable than using the fingertips, since the side of the fingers provide a larger homogeneous contact area. Another potentially useful grasp is the precision grasp with the RBO Hand 2 which brings index finger and thumb nail into contact. In this case the thumb actuator needs to be inflated first, followed by the index actuator. In general, an appropriate grasp vocabulary needs to find the sweet spot between

expressiveness and feasibility of the resulting search space.

DISCRETIZATION OF ENVIRONMENT IS TOO RESTRICTIVE: The planning of ECEs depends not only on the internal DOF of the hand but also on what the environment affords. Although our experiments showed that the environment could often be discretized into ECs, there might be cases in which it fails. A bowl is an example, since objects can be slid along its surface but it might not be detected as slideable at a certain curvature. This problem can be circumvented by introducing new ECs. A similar problem occurs when we are dealing with more cluttered scenarios.

LIMITATIONS DUE TO PROBLEM FACTORIZATION: We clearly suggest to break down the boundaries between hand design, control and planning in robot grasping. Still, our attempt to overcome these boundaries relies on human intuition: We observe and analyze effects in one of the original factors and try to transfer these insights. This process was facilitated by closely working together with hand designers in the context of the SOMA * project.

A less biased way to achieve this is to formulate problems such that they span multiple factors, i.e., the factorization is part of the solution. Deimel et al. (2017) search in the combined space of hand designs and grasping strategies. Although their approach is still limited in the amount of design and motion parameters, it is a promising direction to pursue. The same distinction holds for representations used in planning, control and perception. In this case the field of deep reinforcement learning with its “pixel-to-torque” mantra offers valuable future research directions.

Solving optimization problems that span hand design, grasp planning, control and perception might still be far-fetched. But it is ultimately needed to overcome the local minima imposed by wrong problem factorizations.



We believe that mastering the exploitation of compliance in robot hands and stiffness in the environment are core competencies on the path to general grasping and manual dexterity. The investigations, algorithms, and results presented in this thesis all revolved around these two principles. We hope that they will inspire and guide future research in grasping and manipulation.

*<http://soma-project.eu/>

Bibliography

- Agha-Mohammadi, A.-A., Chakravorty, S., and Amato, N. M. FIRM: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements. *The International Journal of Robotics Research*, 33(2):268–304, 2014.
- Agrawal, S. and Goyal, N. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135, 2013.
- Aleotti, J. and Caselli, S. A 3D shape segmentation approach for robot grasping by parts. *Robotics and Autonomous Systems*, 60(3):358–366, 2012.
- Allard, J., Cotin, S., Faure, F., Bensoussan, P.-J., Poyer, F., Duriez, C., Delingette, H., and Grisoni, L. Sofa - an open source framework for medical simulation. In *Proc. 15th Conf. Medicine Meets Virtual Reality (MMVR)*, volume 125, pages 13–18. IOP Press, February 2007.
- Alterovitz, R., Siméon, T., and Goldberg, K. Y. The stochastic motion roadmap: A sampling framework for planning with Markov motion uncertainty. In *Proc. of Robotics: Science and Systems (RSS)*, volume 3, pages 233–241, 2007.
- Amend, J., Brown, E., Rodenberg, N., Jaeger, H., and Lipson, H. A positive pressure universal gripper based on the jamming of granular material. *IEEE Transactions on Robotics*, 28(2):341–350, April 2012.
- Ankerst, M., Kastenmüller, G., Kriegel, H.-P., and Seidl, T. 3D shape histograms for similarity search and classification in spatial databases. In *Int. Symp. on Spatial Databases*, pages 207–226. Springer, 1999.
- Antonova, R., Cruciani, S., Smith, C., and Kräig, D. Reinforcement learning for pivoting task. *arXiv preprint arXiv:1703.00472*, 2017.
- Asfour, T., Azad, P., Györfi, F., and Dillmann, R. Imitation learning of dual-arm manipulation tasks in humanoid robots. *International Journal of Humanoid Robotics*, 5(02):183–202, 2008.
- Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.
- Bae, J.-H., Park, S.-W., Park, J.-H., Baeg, M.-H., Kim, D., and Oh, S.-R. Development of a low cost anthropomorphic robot hand with high capability. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4776–4782. IEEE, 2012.
- Bai, H., Hsu, D., Lee, W. S., and Ngo, V. A. Monte Carlo value iteration for continuous-state POMDPs. In *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, pages 175–191. Springer, 2010.
- Bai, Y. and Liu, C. K. Dexterous manipulation using both palm and fingers. In *Proc. 2014 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014.
- Balasubramanian, R., Xu, L., Brook, P. D., Smith, J. R., and Matsuoka, Y. Human-guided grasp measures improve grasp robustness on physical robot. In *Proc. 2010 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2294–2301, 2010.

- Bard, C. and Troccaz, J. Automatic preshaping for a dexterous hand from a simple description of objects. In *Proc. 1990 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, page 865–872, 1990.
- Barr, A. H. Rigid physically based superquadrics. In *Graphics Gems III (IBM Version)*, pages 137–159. Elsevier, 1992.
- Berard, S., Trinkle, J., Nguyen, B., Roghani, B., Fink, J., and Kumar, V. davinci code: A multi-model simulation and analysis tool for multi-body systems. In *Proc. 2007 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2588–2593, April 2007.
- Berenson, D., Diankov, R., Nishiwaki, K., Kagami, S., and Kuffner, J. Grasp planning in complex scenes. In *Proc. 2007 IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, pages 42–48, 2007.
- Berenson, D., Srinivasa, S., and Kuffner, J. Task space regions: A framework for pose-constrained manipulation planning. *The International Journal of Robotics Research*, 30(12):1435–1460, 2011.
- Bernstein, N. The co-ordination and regulation of movements. *The co-ordination and regulation of movements*, 1966.
- Bhushan, B. *Introduction to tribology*. John Wiley & Sons, 2013.
- Biagiotti, L., Lotti, F., Melchiorri, C., and Vassura, G. How far is the human hand. *A review on anthropomorphic robotic end-effectors*, 2004.
- Bicchi, A., Gabiccini, M., and Santello, M. Modelling natural and artificial hands with synergies. *Phil. Trans. R. Soc. B*, 366(1581):3153–3161, 2011.
- Bohg, J. and Kragic, D. Learning grasping points with shape context. *Robotics and Autonomous Systems*, 58(4):362–377, 2010.
- Bohg, J., Johnson-Roberson, M., León, B., Felip, J., Gratal, X., Bergström, N., Kragic, D., and Morales, A. Mind the gap – robotic grasping under incomplete observation. In *Proc. 2011 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 686–693, 2011.
- Bohg, J., Morales, A., Asfour, T., and Kragic, D. Data-driven grasp synthesis—a survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2014.
- Bone, G. M., Lambert, A., and Edwards, M. Automated modeling and robotic grasping of unknown three-dimensional objects. In *Proc. 2008 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 292–298, 2008.
- Bonilla, M., Farnioli, E., Piazza, C., Catalano, M., Grioli, G., Garabini, M., Gabiccini, M., and Bicchi, A. Grasping with soft hands. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pages 581–587. IEEE, 2014.
- Book, W. J., Le, S., and Sangveraphunsiri, V. Bracing strategy for robot operation. In *Theory and Practice of Robots and Manipulators*, pages 179–185. Springer, 1985.
- Boothroyd, G. *Assembly Automation and Product Design*. CRC Press, 1991.
- Borras, J. and Asfour, T. A whole-body pose taxonomy for loco-manipulation tasks. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 1578–1585. IEEE, 2015.
- Brock, O. and Valero-Cuevas, F. Transferring synergies from neuroscience to robotics. Comment on "Hand

BIBLIOGRAPHY

- synergies: Integration of robotics and neuroscience for understanding the control of biological and artificial hands" by M. Santello et. al. *Physics of Life Reviews*, 17:27–32, 2016.
- Brost, R. C. and Christiansen, A. D. Probabilistic analysis of manipulation tasks: A conceptual framework. *The International Journal of Robotics Research*, 15(1):1–23, 1996.
- Brown, E., Rodenberg, N., Amend, J., Mozeika, A., Steltz, E., Zakin, M. R., Lipson, H., and Jaeger, H. M. Universal robotic gripper based on the jamming of granular material. *Proc. of the National Academy of Sciences*, 107(44):18809–18814, 2010.
- Bry, A. and Roy, N. Rapidly-exploring random belief trees for motion planning under uncertainty. In *Proc. 2011 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 723–730, 2011.
- Burridge, R. R., Rizzi, A. A., and Koditschek, D. E. Sequential composition of dynamically dexterous robot behaviors. *The International Journal of Robotics Research*, 18(6):534–555, 1999.
- Calinon, S. and Billard, A. Incremental learning of gestures by imitation in a humanoid robot. In *Proc. 2007 IEEE/ACM Int. Conf. on Human-Robot Interaction*. ACM, 2007.
- Calli, B., Wisse, M., and Jonker, P. Grasping of unknown objects via curvature maximization using active vision. In *Proc. 2011 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2011.
- Campbell, M., Hoane, A. J., and Hsu, F.-h. Deep blue. *Artificial intelligence*, 134(1-2):57–83, 2002.
- Catalano, M. G., Grioli, G., Farnioli, E., Serio, A., c. Piazza, and Bicchi, A. Adaptive synergies for the design and control of the Pisa/IIT SoftHand. *The International Journal of Robotics Research*, 33(5):768–782, 2014.
- Chakraborty, N., Berard, S., Akella, S., and Trinkle, J. C. A geometrically implicit time-stepping method for multibody systems with intermittent contact. *The International Journal of Robotics Research*, 33(3):426–445, 2014.
- Chang, L., Srinivasa, S., and Pollard, N. Planning pre-grasp manipulation for transport tasks. In *Proc. 2010 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2697–2704, 2010.
- Chaumette, F., Boukir, S., Bouthemy, P., and Juvin, D. Structure from controlled motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(5):492–504, 1996.
- Chavan-Dafle, N., Rodriguez, A., Paolini, R., Tang, B., Srinivasa, S., Erdmann, M., Mason, M. T., Lundberg, I., Staab, H., and Fuhlbrigge, T. Extrinsic dexterity: In-hand manipulation with external forces. In *Proc. 2014 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2014.
- Cherif, M. and Gupta, K. K. Planning quasi-static fingertip manipulations for reconfiguring objects. *Robotics and Automation, IEEE Transactions on*, 15(5):837–848, 1999.
- Christopoulos, V. N. and Schrater, P. R. Grasping objects with environmentally induced position uncertainty. *PLoS Computational Biology*, 5(10):e1000538, October 2009.
- Chu, W., Li, L., Reyzin, L., and Schapire, R. Contextual bandits with linear payoff functions. In *Proc. of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.
- Cignoni, P., Rocchini, C., and Scopigno, R. Metro: Measuring error on simplified surfaces. In *Computer Graphics Forum*, volume 17, pages 167–174. Wiley Online Library, 1998.
- Ciocarlie, M., Miller, A., and Allen, P. Grasp analysis using deformable fingers. In *Proc. 2005 IEEE/RSJ*

- Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 4122–4128, 2005.
- Ciocarlie, M. T. and Allen, P. K. Hand posture subspaces for dexterous robotic grasping. *The International Journal of Robotics Research*, 28(7):851–867, 2009.
- Correll, N., Bekris, K. E., Berenson, D., Brock, O., Causo, A., Hauser, K., Okada, K., Rodriguez, A., Romano, J. M., and Wurman, P. R. Analysis and observations from the first amazon picking challenge. *IEEE Transactions on Automation Science and Engineering*, 2016.
- Coumans, E. Bullet physics simulation. In *ACM SIGGRAPH 2015 Courses*, page 7. ACM, 2015.
- Cutkosky, M., Jourdain, J., and Wright, P. Skin materials for robotic fingers. In *Proc. 1987 IEEE Int. Conf. on Robotics and Automation (ICRA)*, volume 4, pages 1649–1654, 1987.
- Deimel, R. and Brock, O. A novel type of compliant, underactuated robotic hand for dexterous grasping. In *Proc. of Robotics: Science and Systems (RSS)*, Berkeley, USA, July 2014.
- Deimel, R. and Brock, O. A novel type of compliant and underactuated robotic hand for dexterous grasping. *The International Journal of Robotics Research*, 35(1-3):161–185, 2016.
- Deimel, R., Eppner, C., Álvarez-Ruiz, J., Maertens, M., and Brock, O. Exploitation of environmental constraints in human and robotic grasping. In *Int. Symp. on Robotics Research (ISRR)*, 2013.
- Deimel, R., Irmisch, P., Wall, V., and Brock, O. Automated co-design of soft hand morphology and control strategy for grasping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1213–1218, September 2017.
- Detry, R., Kraft, D., Kroemer, O., Bodenhausen, L., Peters, J., Krüger, N., and Piater, J. Learning grasp affordance densities. *Paladyn*, 2(1):1–17, March 2011.
- Diankov, R. *Automated Construction of Robotic Manipulation Programs*. PhD thesis, Carnegie Mellon University, Robotics Institute, 2010.
- Dogar, M. R. and Srinivasa, S. S. A planning framework for non-prehensile manipulation under clutter and uncertainty. *Autonomous Robots*, 33(3):217–236, 2012.
- Dogar, M. and Srinivasa, S. Push-grasping with dexterous hands: Mechanics and a method. In *Proc. 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, page 21232130, 2010.
- Dollar, A. M. and Howe, R. D. The highly adaptive SDM hand: Design and performance evaluation. *The International Journal of Robotics Research*, 29(5):585–597, 2010.
- Dragiev, S., Toussaint, M., and Gienger, M. Gaussian process implicit surfaces for shape estimation and grasping. In *Proc. 2011 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2845–2850, 2011.
- Dune, C., Marchand, E., Colloret, C., and Leroux, C. Active rough shape estimation of unknown objects. In *Proc. 2008 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3622–3627, 2008.
- Egerstedt, M. Behavior based robotics using hybrid automata. In *Hybrid Systems: computation and control*. Springer, 2000.
- Ekvall, S. and Kragic, D. Learning and evaluation of the approach vector for automatic grasp generation and planning. In *Proc. 2007 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 4715–4720, 2007.
- Ekvall, S. and Kragic, D. Interactive grasp learning based on human demonstration. In *Robotics and Automata*

BIBLIOGRAPHY

- tion, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on, volume 4, pages 3519–3524. IEEE, 2004.
- Erdmann, M. A. and Mason, M. T. An exploration of sensorless manipulation. *IEEE Journal of Robotics and Automation*, 4(4):369–379, 1988.
- Erdmann, M. Using backprojections for fine motion planning with uncertainty. *The International Journal of Robotics Research*, 5(1):19–45, 1986.
- Ernst, H. A. Mh-1, a computer-operated mechanical hand. In *Proc. of Spring Joint Computer Conference, AIEE-IRE '62 (Spring)*, pages 39–51, New York, NY, USA, May 1962. ACM.
- Feix, T., Pawlik, R., Schmiedmayer, H., Romero, J., and Kragic, D. A comprehensive grasp taxonomy. In *Proc. of Robotics: Science and Systems (RSS)*, 2009.
- Feix, T., Bullock, I., and Dollar, A. Analysis of Human Grasping Behavior: Object Characteristics and Grasp Type. *IEEE Transactions on Haptics*, 7(3):311–323, July 2014.
- Felip, J. and Morales, A. Robust sensor-based grasp primitive for a three-finger robot hand. In *Proc. 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1811–1816, 2009.
- Felzenszwalb, P. F. and Huttenlocher, D. P. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004.
- Ferrari, C. and Canny, J. Planning optimal grasps. In *Proc. 1992 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2290–2295 vol.3, 1992.
- Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., Lally, A., Murdock, J. W., Nyberg, E., Prager, J., et al. Building watson: An overview of the deepqa project. *AI magazine*, 31(3):59–79, 2010.
- Feurer, M., Klein, A., Eggensperger, K., Springenberg, J. T., Blum, M., and Hutter, F. Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems*, pages 2755–2763. 2015.
- Fikes, R. E. and Nilsson, N. J. Strips: A new approach to the application of theorem proving to problem solving. *Artificial intelligence*, 2(3-4):189–208, 1971.
- Finkemeyer, B., Kröger, T., and Wahl, F. M. Executing assembly tasks specified by manipulation primitive nets. *Advanced Robotics*, 19(5):591–611, 2005.
- Fischinger, D., Weiss, A., and Vincze, M. Learning grasps with topographic features. *The International Journal of Robotics Research*, 34(9):1167–1194, 2015.
- Fischler, M. A. and Bolles, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- Fox, A. and Hutchinson, S. Exploiting visual constraints in the synthesis of uncertainty-tolerant motion plans. *IEEE T. Robotics and Automation*, 11(1):56–71, 1995.
- Goldfeder, C., Allen, P. K., Lackner, C., and Pelossof, R. Grasp planning via decomposition trees. In *Proc. 2007 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 4679–4684, 2007.
- Goldfeder, C., Ciocarlie, M., Peretzman, J., Dang, H., and Allen, P. K. Data-driven grasping with partial sensor data. In *Proc. 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, page 1278–1283, 2009a.

- Goldfeder, C., Ciocarlie, M., Dang, H., and Allen, P. K. The columbia grasp database. In *Proc. 2009 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1710–1716, 2009b.
- Grioli, G., Catalano, M., Silvestro, E., Tono, S., and Bicchi, A. Adaptive synergies: An approach to the design of under-actuated robotic hands. In *Proc. 2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 356–364, 2012.
- Gualtieri, M., ten Pas, A., Saenko, K., and Platt, R. High precision grasp pose detection in dense clutter. In *Proc. 2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 598–605, 2016.
- Guenter, F., Hersch, M., Calinon, S., and Billard, A. Reinforcement learning for imitating constrained reaching movements. *Advanced Robotics*, 21(13):1521–1544, 2007.
- Han, L. and Trinkle, J. C. Dextrous manipulation by rolling and finger gaiting. In *Proc. 1998 IEEE Int. Conf. on Robotics and Automation (ICRA)*, volume 1, 1998.
- Hershey, J. and Olsen, P. Approximating the Kullback Leibler divergence between Gaussian mixture models. In *Proc. 2007 Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 4, pages IV–317–IV–320, April 2007.
- Herzog, A., Pastor, P., Kalakrishnan, M., Righetti, L., Asfour, T., and Schaal, S. Template-based learning of grasp selection. In *Proc. 2012 IEEE Int. Conf. on Robotics and Automation (ICRA)*, page 2379–2384, 2012.
- Herzog, A., Pastor, P., Kalakrishnan, M., Righetti, L., Bohg, J., Asfour, T., and Schaal, S. Learning of grasp selection based on shape-templates. *Autonomous Robots*, 36(1-2):51–65, January 2014.
- Hillenbrand, U. and Roa, M. A. Transferring functional grasps through contact warping and local replanning. In *Proc. 2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2963–2970, 2012.
- Hirose, S. Connected differential mechanism and its applications. In *Proc. Int. Conf. on Advanced Robotics (ICAR)*, 1985.
- Hirose, S. and Umetani, Y. The development of soft gripper for the versatile robot hand. *Mechanism and Machine Theory*, 13(3):351–359, 1978.
- Hoffmann, J. and Nebel, B. The ff planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 14:253–302, 2001.
- Holz, D. and Behnke, S. Fast range image segmentation and smoothing using approximate surface reconstruction and region growing. In *Proc. 12th Int. Conf. on Intelligent Autonomous Systems (IAS)*, Jeju Island, Korea, June 2012.
- Horowitz, M. and Burdick, J. Interactive non-prehensile manipulation for grasping via pomdps. In *Proc. 2013 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 3257–3264, 2013.
- Howard, W. S. and Kumar, V. Stability of planar grasps. In *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 2822–2827. IEEE, 1994.
- Howard, W. S. and Kumar, V. On the stability of grasped objects. *IEEE Transactions on Robotics and Automation*, 12(6):904–917, 1996.
- Hsiao, K., Chitta, S., Ciocarlie, M., and Jones, E. Contact-reactive grasping of objects with partial shape information. In *Proc. 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1228–1235, 2010.

BIBLIOGRAPHY

- Hsiao, K., Kaelbling, L. P., and Lozano-Pérez, T. Grasping POMDPs. In *Proc. 2007 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 4685–4692, 2007.
- Huebner, K. and Kragic, D. Selection of robot pre-grasps using box-based shape approximation. In *Proc. 2008 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1765–1770, 2008.
- Ijspeert, A. J., Nakanishi, J., and Schaal, S. Learning attractor landscapes for learning motor primitives. In *Advances in Neural Information Processing Systems*, pages 1547–1554. 2003.
- Jacobson, D. H. and Mayne, D. Q. *Differential dynamic programming*. Elsevier, New York, 1970.
- Javdani, S., Klingensmith, M., Bagnell, J. A., Pollard, N. S., and Srinivasa, S. S. Efficient touch based localization through submodularity. In *Proc. 2013 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1828–1835, 2013.
- Jeannerod, M. Intersegmental coordination during reaching at natural visual objects. *Attention and performance IX*, pages 153–168, 1981.
- Ji, X. and Xiao, J. Automatic generation of high-level contact state space. *The International Journal of Robotics Research*, 20:238–244, 1999.
- Ji, X. and Xiao, J. Planning motions compliant to complex contact states. *The International Journal of Robotics Research*, 20(6):446–465, 2001.
- Johnson, A. E. *Spin-images: a representation for 3-D surface matching*. PhD thesis, Carnegie Mellon University Pittsburgh, PA, 1997.
- Kaiser, D., Stein, T., and Peelen, M. V. Object grouping based on real-world regularities facilitates perception by reducing competitive interactions in visual cortex. *Proceedings of the National Academy of Sciences*, 111(30):11217–11222, 2014.
- Kalakrishnan, M., Righetti, L., Pastor, P., and Schaal, S. Learning force control policies for compliant manipulation. In *Proc. 2011 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2011.
- Kapandji, I. A. Cotation clinique de l'opposition et de la contre-opposition du pouce. *Annales de Chirurgie de la Main*, 5(1):68–73, 1986.
- Kappler, D., Bohg, B., and Schaal, S. Leveraging big data for grasp planning. In *Proc. 2015 IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2015.
- Kappler, D., Chang, L., Przybylski, M., Pollard, N., Asfour, T., and Dillmann, R. Representation of pre-grasp strategies for object manipulation. In *Proc. 2010 IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2010.
- Kappler, D., Chang, L. Y., Pollard, N. S., Asfour, T., and Dillmann, R. Templates for pre-grasp sliding interactions. *Robotics and Autonomous Systems*, 60(3):411–423, 2012.
- Karpathy, A., Miller, S., and Fei-Fei, L. Object discovery in 3D scenes via shape analysis. In *Proc. 2013 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2013.
- Kasper, A., Xue, Z., and Dillmann, R. The kit object models database: An object model database for object recognition, localization and manipulation in service robotics. *The International Journal of Robotics Research*, 31(8):927–934, 2012.

- Kavraki, L. E., Svestka, P., Latombe, J.-C., and Overmars, M. H. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, 1996.
- Kazemi, M., Valois, J.-S., Bagnell, J. A., and Pollard, N. Robust object grasping using force compliant motion primitives. In *Proc. of Robotics: Science and Systems (RSS)*, Sydney, Australia, July 2012.
- Kazemi, M., Valois, J., Bagnell, J. A., and Pollard, N. Human-inspired force compliant grasping primitives. *Autonomous Robots*, 37(2):209–225, 2014.
- Khoshelham, K. and Elberink, S. O. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012.
- King, J., Klingensmith, M., Dellin, C., Dogar, M., Velagapudi, P., Pollard, N., and Srinivasa, S. Pregrasp manipulation as trajectory optimization. In *Proc. of Robotics: Science and Systems (RSS)*, 2013.
- Kitaev, N., Mordatch, I., Patil, S., and Abbeel, P. Physics-based trajectory optimization for grasping in cluttered environments. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3102–3109. IEEE, 2015.
- Klingbeil, E., Rao, D., Carpenter, B., Ganapathi, V., Ng, A. Y., and Khatib, O. Grasping with application to an autonomous checkout robot. In *Proc. 2011 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.
- Kootstra, G., Popovic, M., Jorgensen, J. A., Kuklinski, K., Miatliuk, K., Krasic, D., and Kruger, N. Enabling grasping of unknown objects through a synergistic use of edge and surface information. *The International Journal of Robotics Research*, 31(10):1190–1213, August 2012.
- Koval, M. C., Pollard, N. S., and Srinivasa, S. S. Pre-and post-contact policy decomposition for planar contact manipulation under uncertainty. *The International Journal of Robotics Research*, 35(1-3):244–264, 2016.
- Krainin, M., Henry, P., Ren, X., and Fox, D. Manipulator and object tracking for in-hand 3d object modeling. *The International Journal of Robotics Research*, 30(11):1311–1327, 2011.
- Kroemer, O. B., Detry, R., Piater, J., and Peters, J. Combining active learning and reactive control for robot grasping. *RAS*, 58:1105–1116, September 2010. ACM ID: 1838783.
- Kroemer, O., Daniel, C., Neumann, G., van Hoof, H., and Peters, J. Towards learning hierarchical skills for multi-phase manipulation tasks. In *Proc. 2015 IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2015.
- Krut, S. A force-isotropic underactuated finger. In *Proc. 2005 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2314–2319, 2005.
- Kuffner, J. J. and LaValle, S. M. RRT-connect: An efficient approach to single-query path planning. In *Proc. 2000 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 995–1001, 2000.
- Kumar, V., Todorov, E., and Levine, S. Optimal control with learned local models: Application to dexterous manipulation. In *Proc. 2016 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 378–383, 2016.
- Kurniawati, H., Hsu, D., and Lee, W. S. SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces. In *Proc. of Robotics: Science and Systems (RSS)*, 2008.
- Lampe, T. and Riedmiller, M. Acquiring visual servoing reaching and grasping skills using neural reinforcement learning. In *Proc. 2013 Int. Joint Conf. on Neural Networks (IJCNN)*, 2013.

BIBLIOGRAPHY

- Langford, J. and Zhang, T. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in neural information processing systems*, pages 817–824, 2008.
- Laskey, M., Mahler, J., McCarthy, Z., Pokorny, F. T., Patil, S., van den Berg, J., Krivic, D., Abbeel, P., and Goldberg, K. Multi-armed bandit models for 2D grasp planning with uncertainty. In *Proc. 2015 IEEE Int. Conf. on Automation Science and Engineering (CASE)*, 2015.
- LaValle, S. M. Rapidly-exploring random trees: A new tool for path planning. Technical report, Department of Computer Science, Iowa State University, 1998.
- LaValle, S. M. *Planning algorithms*. Cambridge university press, 2006.
- Lenz, I., Lee, H., and Saxena, A. Deep learning for detecting robotic grasps. In *Proc. of Robotics: Science and Systems (RSS)*, Berlin, Germany, June 2013.
- Levine, S. and Abbeel, P. Learning neural network policies with guided policy search under unknown dynamics. In *Advances in Neural Information Processing Systems*. 2014.
- Levine, S. and Koltun, V. Guided policy search. In *Proc. 30th Int. Conf. on Machine Learning (ICML)*, 2013.
- Levine, S., Finn, C., Darrell, T., and Abbeel, P. End-to-end training of deep visuomotor policies. *arXiv preprint arXiv:1504.00702*, 2015.
- Levine, S., Pastor, P., Krizhevsky, A., and Quillen, D. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *arXiv preprint arXiv:1603.02199*, 2016.
- Li, Q., Elbrechter, C., Haschke, R., and Ritter, H. Integrating vision, haptics and proprioception into a feedback controller for in-hand manipulation of unknown objects. In *Proc. 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013.
- Li, W. and Todorov, E. Iterative linear quadratic regulator design for nonlinear biological movement systems. In *Proc. 2004 Int. Conf. on Informatics in Control, Automation and Robotics (ICINCO)*, 2004.
- Lin, Q., Burdick, J., and Rimon, E. Computation and analysis of compliance in grasping and fixturing. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, volume 1, pages 93–99. IEEE, 1997.
- Littlefield, Z., Kurniawati, H., Bekris, K., and Klimenko, E. The importance of a suitable distance function in belief-space planning. In *Int. Symp. on Robotics Research (ISRR)*, 2015.
- Lozano-Pérez, T., Mason, M. T., and Taylor, R. H. Automatic synthesis of fine-motion strategies for robots. *The International Journal of Robotics Research*, 3(1):3–24, March 1984.
- Lynch, K. M. and Park, F. C. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017.
- Mahler, J., Pokorny, F. T., Hou, B., Roderick, M., Laskey, M., Aubry, M., Kohlhoff, K., Kröger, T., Kuffner, J. J., and Goldberg, K. Dex-net 1.0: A cloud-based network of 3D objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In *Proc. 2016 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1957–1964, 2016.
- Mahler, J., Liang, J., Niyaz, S., Laskey, M., Doan, R., Liu, X., Ojea, J. A., and Goldberg, K. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In *Proc. of Robotics: Science and Systems (RSS)*, 2017.

- Maldonado, A., Klank, U., and Beetz, M. Robotic grasping of unmodeled objects using time-of-flight range data and finger torque information. In *Proc. 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2586–2591, Taipei, Taiwan, October 2010.
- Malisiewicz, T. and Efros, A. A. Improving spatial support for objects via multiple segmentations. In *British Machine Vision Conference*, 2007.
- Mason, M. T. Grope and hope. Invited talk at the Workshop on Multimodal Manipulation Under Uncertainty in Dagstuhl, October 2015.
- Mason, M. T. The mechanics of manipulation. In *Proc. 1985 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 544–548, 1985.
- Mason, M. T. *Mechanics of Robotic Manipulation*. MIT Press, Cambridge, MA, August 2001.
- Mason, M. T. *Manipulator grasping and pushing operations*. PhD thesis, Massachusetts Institute of Technology, 1982.
- McDermott, D., Ghallab, M., Howe, A., Knoblock, C., Ram, A., Veloso, M., Weld, D., and Wilkins, D. Pddl—the planning domain definition language. 1998.
- MekaBot, M. H₂ compliant hand datasheet, 2009.
- Melchior, N. A. and Simmons, R. Particle RRT for path planning with uncertainty. In *Proc. 2007 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1617–1624, 2007.
- Miller, A. and Allen, P. Graspit! a versatile simulator for robotic grasping. *Robotics & Automation Magazine, IEEE*, 11(4):110–122, 2004.
- Miller, A., Knoop, S., Christensen, H., and Allen, P. Automatic grasp planning using shape primitives. In *Proc. 2003 IEEE Int. Conf. on Robotics and Automation (ICRA)*, volume 2, pages 1824–1829 vol.2, 2003.
- Minsky, M. *The Society of Mind*. Simon & Schuster, Inc., New York, NY, USA, 1986.
- Montana, D. J. The condition for contact grasp stability. In *Proc. 1991 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 412–417, 1991.
- Montesano, L. and Lopes, M. Active learning of visual descriptors for grasping using non-parametric smoothed beta distributions. *Robotics and Autonomous Systems*, 60(3):452–462, March 2012.
- Morales, A., Asfour, T., Azad, P., Knoop, S., and Dillmann, R. Integrated grasp planning and visual object localization for a humanoid robot with five-fingered hands. In *Proc. 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 5663–5668, October 2006.
- Morales, A., Prats, M., Sanz, P., and Pobil, A. P. An experiment in the use of manipulation primitives and tactile perception for reactive grasping. In *Robotics: Science and Systems, Workshop on Robot Manipulation: Sensing and Adapting to the Real World, Atlanta, USA*, 2007.
- Moravec, H. *Mind children: The future of robot and human intelligence*. Harvard University Press, 1988.
- Mordatch, I., Popović, Z., and Todorov, E. Contact-invariant optimization for hand manipulation. In *Proc. 2012 ACM SIGGRAPH Conf.*, 2012.
- Morrow, J., Shin, H.-S., Phillips-Grafflin, C., Jang, S.-H., Torrey, J., Larkins, R., Dang, S., Park, Y.-L., and Berenson, D. Improving soft pneumatic actuator fingers through integration of soft sensors, position and

BIBLIOGRAPHY

- force control, and rigid fingernails. In *Proc. 2016 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 5024–5031, 2016.
- Mülling, K., Kober, J., Kroemer, O., and Peters, J. Learning to select and generalize striking movements in robot table tennis. *The International Journal of Robotics Research*, 32(3):263–279, 2013.
- Murray, R. M., Li, Z., Sastry, S. S., and Sastry, S. S. *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- Nakanishi, J., Cory, R., Mistry, M., Peters, J., and Schaal, S. Operational space control: A theoretical and empirical comparison. *The International Journal of Robotics Research*, 27(6):737–757, 2008.
- Napier, J. R. The prehensile movements of the human hand. *Bone & Joint Journal*, 38(4):902–913, 1956.
- Nieuwenhuisen, M., Stückler, J., Berner, A., Klein, R., and Behnke, S. Shape-primitive based object recognition and grasping. *ROBOTIK* 2012, 2012.
- Oberlin, J. and Tellez, S. Autonomously acquiring instance-based object models from experience. In *Int. Symp. on Robotics Research (ISRR)*, 2015.
- Odhner, L. U., Walker, C., and Dollar, A. M. Simplifying robot hands using recursively scaled power grasps. In *Proc. 2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2909–2914, 2012.
- Odhner, L., Ma, R., and Dollar, A. Open-loop precision grasping with underactuated hands inspired by a human manipulation strategy. *IEEE Transactions on Automation Science and Engineering*, 10(3):625–633, 2013.
- Ong, S. C., Png, S. W., Hsu, D., and Lee, W. S. Planning under uncertainty for robotic tasks with mixed observability. *The International Journal of Robotics Research*, 29(8):1053–1068, 2010.
- O'Regan, J. K. and Noë, A. A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(5):939–973, 2001.
- Osada, R., Funkhouser, T., Chazelle, B., and Dobkin, D. Shape distributions. *ACM Trans. Graph.*, 21(4):807–832, October 2002.
- Pas, A. t. and Platt, R. Localizing grasp affordances in 3-d points clouds using taubin quadric fitting. *arXiv preprint arXiv:1311.3192*, 2013.
- Peters, J., Mülling, K., and Altun, Y. Relative entropy policy search. In *Proc. AAAI Conf. on Artificial Intelligence*. Atlanta, 2010.
- Pfeifer, R. and Iida, F. Morphological computation: Connecting body, brain and environment. *Japanese Scientific Monthly*, 58(2):48–54, 2005.
- Phillips-Grafflin, C. and Berenson, D. Planning and resilient execution of policies for manipulation in contact with actuation uncertainty. In *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2016.
- Pinto, L. and Gupta, A. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. *arXiv preprint arXiv:1509.06825*, 2015.
- Platt, R., Kaelbling, L., Lozano-Pérez, T., and Tedrake, R. Simultaneous localization and grasping as a belief space control problem. In *Int. Symp. on Robotics Research (ISRR)*, volume 2, 2011.
- Platt Jr, R., Tedrake, R., Kaelbling, L., and Lozano-Pérez, T. Belief space planning assuming maximum likelihood. *Journal of Machine Learning Research*, 13:1073–1099, 2012.

- hood observations. In *Proc. of Robotics: Science and Systems (RSS)*, 2010.
- Pokorny, F., Stork, J., and Kragic, D. Grasping objects with holes: A topological approach. In *Proc. 2013 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1100–1107, May 2013.
- Polygerinos, P., Wang, Z., Overvelde, J. T. B., Galloway, K. C., Wood, R. J., Bertoldi, K., and Walsh, C. J. Modeling of soft fiber-reinforced bending actuators. *IEEE T-RO*, 31(3):778–789, June 2015.
- Porta, J. M., Vlassis, N., Spaan, M. T., and Poupart, P. Point-based value iteration for continuous POMDPs. *Journal of Machine Learning Research*, 7:2329–2367, 2006.
- Prattichizzo, D. and Trinkle, J. C. Grasping. In Siciliano, B. and Khatib, O., editors, *Springer Handbook of Robotics*, pages 671–700. Springer Berlin Heidelberg, 2008.
- Prentice, S. and Roy, N. The belief roadmap: Efficient planning in linear POMDPs by factoring the covariance. In *Int. Symp. on Robotics Research (ISRR)*, pages 293–305. Springer, 2010.
- Prieur, U., Perdereau, V., and Bernardino, A. Modeling and planning high-level in-hand manipulation actions from human knowledge and active learning from demonstration. In *Proc. 2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2012.
- Przybylski, M., Asfour, T., and Dillmann, R. Unions of balls for shape approximation in robot grasping. In *Proc. 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1592–1599, 2010.
- Reuleaux, F. *The Kinematics of Machinery: Outlines of a Theory of Machines. German original (1875). Translated by A. Kennedy*. MacMillan and Co., London, 1876.
- Rickert, M. and Gaschler, A. Robotics Library: An object-oriented approach to robot applications. In *Proc. 2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 733–740, Vancouver, BC, Canada, September 2017.
- Rickert, M., Sieverling, A., and Brock, O. Balancing exploration and exploitation in sampling-based motion planning. *IEEE Transactions on Robotics*, 30(6):1305–1317, 2014.
- Righetti, L., Kalakrishnan, M., Pastor, P., Binney, J., Kelly, J., Voorhies, R. C., Sukhatme, G. S., and Schaal, S. An autonomous manipulation system based on force control and optimization. *Autonomous Robots*, 36(1-2):11–30, 2014.
- Ritter, H., Haschke, R., and Steil, J. J. A dual interaction perspective for robot cognition: grasping as a "rosetta stone". In *Perspectives of neural-symbolic integration*, pages 159–178. Springer, 2007.
- Roa, M. A. and Suárez, R. Grasp quality measures: review and performance. *Autonomous Robots*, 38(1):65–88, 2015.
- Robbins, H. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1985.
- Rodriguez, A. *Shape for Contact*. PhD thesis, Carnegie Mellon University, 2013.
- Rodriguez, A. and Mason, M. T. Grasp invariance. *The International Journal of Robotics Research*, 31(2):236–248, February 2012.
- Rodriguez, A., Mason, M. T., and Srinivasa, S. S. Manipulation capabilities with simple hands. In *Int. Symp. on Experimental Robotics (ISER)*, pages 285–299. Springer, 2014.

BIBLIOGRAPHY

- Romero, J., Feix, T., Kjellström, H., and Kragic, D. Spatio-temporal modeling of grasping actions. In *Proc. 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2103–2108, 2010.
- Rubinstein, R. The cross-entropy method for combinatorial and continuous optimization. *Methodology and computing in applied probability*, 1(2):127–190, 1999.
- Rublee, E. and Strasheim, T. Ecto: a data flow framework, <https://github.com/plasmadic/ecto>, 2011.
- Rusu, R. B., Blodow, N., and Beetz, M. Fast point feature histograms for 3D registration. In *Proc. 2009 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 3212–3217, 2009.
- Salganicoff, M., Ungar, L. H., and Bajcsy, R. Active learning for vision-based robot grasping. *Machine Learning*, 23(2-3):251–278, 1996.
- Santello, M., Flanders, M., and Soechting, J. F. Postural hand synergies for tool use. *The Journal of Neuroscience*, 18(23):10105 –10115, 1998.
- Sauer, E. L., Argall, B. D., Metta, G., and Billard, A. G. Iterative learning of grasp adaptation through human corrections. *Robotics and Autonomous Systems*, 60(1):55–71, 2012.
- Saxena, A., Driemeyer, J., and Ng, A. Y. Robotic grasping of novel objects using vision. *The International Journal of Robotics Research*, 27(2):157, 2008a.
- Saxena, A., Wong, L., and Ng, A. Y. Learning grasp strategies with partial shape information. In *Proc. 23rd Nat. Conf. on Artificial intelligence*, pages 1491–1494, 2008b.
- Schaal, S. Dynamic movement primitives-a framework for motor control in humans and humanoid robotics. In *Adaptive motion of animals and machines*, pages 261–280. Springer, 2006.
- Schulman, J., Ho, J., Lee, A. X., Awwal, I., Bradlow, H., and Abbeel, P. Finding locally optimal, collision-free trajectories with sequential convex optimization. In *Proc. of Robotics: Science and Systems (RSS)*, volume 9, pages 1–10, 2013.
- Scott, P. B. The ‘omnigripper’: A form of robot universal gripper. *Robotica*, 3(3):153–158, 1985.
- Shimoga, K. B. Robot Grasp Synthesis Algorithms: A Survey. *The International Journal of Robotics Research*, 15(3):230–266, June 1996.
- Shimoga, K. B. and Goldenberg, A. A. Soft materials for robotic fingers. In *Proc. 1992 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1300–1305, 1992.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.
- Siméon, T., Cortés, J., Sahbani, A., and Laumond, J.-P. A general manipulation task planner. In *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, pages 311–328. Springer, 2004.
- Sintov, A. and Shapiro, A. Swing-up regrasping algorithm using energy control. In *Proc. 2016 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 4888–4893, 2016.
- Sklansky, J. Finding the convex hull of a simple polygon. *Pattern Recognition Letters*, 1(2):79–83, 1982.
- Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

- Stansbury, D. E., Naselaris, T., and Gallant, J. L. Natural scene statistics account for the representation of scene categories in human visual cortex. *Neuron*, 79(5):1025–1034, 2013.
- Stilman, M. Task constrained motion planning in robot joint space. In *Proc. 2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3074–3081, 2007.
- Stulp, F., Theodorou, E., Buchli, J., and Schaal, S. Learning to grasp under uncertainty. In *Proc. 2011 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 5703–5708, 2011.
- Suzuki, S. et al. Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, 30(1):32–46, 1985.
- Sweeney, J. D. and Grupen, R. A model of shared grasp affordances from demonstration. In *Proc. 2007 IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, pages 27–35, 2007.
- Szita, I. and Lörincz, A. Learning tetris using the noisy cross-entropy method. *Neural Computation*, 18(12):2936–2941, 2006.
- Tangelder, J. W. and Veltkamp, R. C. A survey of content based 3d shape retrieval methods. In *Shape Modeling Applications, 2004. Proceedings*, pages 145–156. IEEE, 2004.
- Tedrake, R., Manchester, I. R., Tobenkin, M., and Roberts, J. W. Lqr-trees: Feedback motion planning via sums-of-squares verification. *The International Journal of Robotics Research*, 29(8):1038–1052, 2010.
- Theodorou, E. A., Buchli, J., and Schaal, S. Learning policy improvements with path integrals. In *Proc. 2010 Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, 2010.
- Thompson, W. R. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.
- Thrun, S. Monte carlo pomdps. In *Advances in Neural Information Processing Systems*, volume 12, pages 1064–1070. 1999.
- Tournier, M., Nesme, M., Gilles, B., and Faure, F. Stable Constrained Dynamics. *ACM Trans. Graph.*, 34:132:1–132:10, 2015.
- Toussaint, M., Ratliff, N., Bohg, J., Righetti, L., Englert, P., and Schaal, S. Dual execution of optimized contact interaction trajectories. In *Proc. 2014 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2014.
- Townsend, W. The barrett-hand grasper—programmably flexible part handling and assembly. *Industrial Robot: an international journal*, 27(3):181–188, 2000.
- Ückermann, A., Haschke, R., and Ritter, H. Real-time 3d segmentation of cluttered scenes for robot grasping. In *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*, pages 198–203. IEEE, 2012.
- Ulrich, N. T. *Grasping with mechanical intelligence*. PhD thesis, University of Pennsylvania, December 1989.
- Van Den Berg, J., Patil, S., and Alterovitz, R. Motion planning under uncertainty using iterative local optimization in belief space. *The International Journal of Robotics Research*, 31(11):1263–1278, 2012.
- van Hoof, H., Hermans, T., Neumann, G., and Peters, J. Learning robot in-hand manipulation with tactile features. In *Proc. 2015 IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2015.
- Vien, N. A. and Toussaint, M. Touch based POMDP manipulation via sequential submodular optimization.

BIBLIOGRAPHY

- In *Proc. 2015 IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, pages 407–413, 2015.
- Walker, I. D. A successful multifingered hand design – the case of the raccoon. In *Proc. 1995 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, volume 2, pages 186–193, 1995.
- Weisz, J. and Allen, P. Pose error robust grasping from contact wrench space metrics. In *Proc. 2012 IEEE Int. Conf. on Robotics and Automation (ICRA)*, St. Paul, Minnesota, USA, 2012.
- Wentworth, N., Benson, J. B., and Haith, M. M. The development of infants' reaches for stationary and moving targets. *Child development*, 71(3):576–601, 2000.
- Will, P. and Grossman, D. An experimental system for computer controlled mechanical assembly. *IEEE Transactions on Computers*, C-24(9):879–888, 1975.
- Wilson, R. H. and Latombe, J.-C. Geometric reasoning about mechanical assembly. *Artificial Intelligence*, 71: 371–396, 1994.
- Wonik Robotics Co., Ltd. RoboticsLab (Robotics Software Development Environment with Dynamics and Control Engines), 2010.
- Xu, Z., Deyle, T., and Kemp, C. 1000 trials: An empirically validated end effector that robustly grasps objects from the floor. In *Proc. 2009 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 2160–2167, May 2009.
- Yang, Y., Zhu, D., et al. Randomized allocation with nonparametric estimation for a multi-armed bandit problem with covariates. *The Annals of Statistics*, 30(1):100–121, 2002.
- Yoshikawa, T. Manipulability of robotic mechanisms. *The International Journal of Robotics Research*, 4(2):3–9, 1985.
- Zheng, J. Z., De La Rosa, S., and Dollar, A. M. An investigation of grasp type and frequency in daily household and machine shop tasks. In *Proc. 2011 IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 4169–4175, 2011.
- Zhou, L. A survey on contextual multi-armed bandits. *CoRR*, abs/1508.03326, 2015.
- Zhou, Y. and Hauser, K. 6dof grasp planning by optimizing a deep learning scoring function. In *Robotics: Science and Systems (RSS) Workshop on Revisiting Contact-Turning a Problem into a Solution*, 2017.

