

Functional Autonomy Techniques for Manipulation in Uncertain Environments

Thesis by
Joseph Bowkett

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

The logo for the California Institute of Technology (Caltech), featuring the word "Caltech" in a bold, orange, sans-serif font.

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2020
Defended March 13, 2020

© 2020

Joseph Bowkett

ORCID: 0000-0002-3101-489X

All rights reserved except where otherwise noted

ACKNOWLEDGEMENTS

First and foremost, my thanks go to Joel, who numbers among the most conscientious and affable humans I have met on this planet. His patience and good-natured guidance made the graduate school experience infinitely more tolerable than it would otherwise have been.

I am eternally indebted to many of the fine folk at the Jet Propulsion Laboratory, who saw fit to offer me work, technical instruction, and social diversion. In particular, the denizens of the 198-B7 lab, both past and present, taught me more about robotics than I believe I could have learnt in any other forum. With any luck I'll be able to repay that debt in-kind after starting there as an employee.

Much of my work undertaken through JPL revolved around the Robotics Collaborative Technology Alliance, and across 7 integration meetings in a range of cities I was incredibly fortunate to have the opportunity to work alongside many superb engineers, roboticists, and researchers from General Dynamics, the Army Research Laboratory, the University of Washington, Carnegie Mellon University, the University of Pennsylvania, and many others. Despite occasionally trying circumstances, they always managed to maintain a cheery outlook, and somehow made the drudge of implementing a million software fixes across more than a million lines of code an enjoyable experience.

Of course, I never would have made it to Caltech in the first place were it not for the unwavering support of my family and friends in New Zealand. In particular, this thesis is dedicated to my parents, for their dedication to ensuring I had all the educational resources and anything else I could want for to pursue a fulfilling career and life, even if it meant flying halfway around the world to see me.

The first year of classes at Caltech was easily the most challenging of my life, but the comradery and collaboration of my 2014 intake classmates of the MCE and GALCIT departments made it survivable, nigh on enjoyable. The many late nights spent in various libraries or the Keith-Spalding building finishing problem sets while eating 3am orders of thai food will be one of my most enduring memories of Caltech.

Orientation week of that first year, around a poker table and otherwise, is also when I met many of the sterling individuals with whom I spent my time outside of studies. Among them, my thanks go to the Fighting Pinecones, skiing, and squash buddies, for managing to get me up and out of the lab every once in a while. The most special

of mentions must go to the Fremont lads whom, with the ever available supply of star tangled banglers, witty banter, and debate on any topic under the Sun (or beyond it), proved the perfect cocktail for keeping this stressed grad student sane.

Thanks also to the MCE & JPL administrative staff, whose guidance and patience dealing with my frequent work trips for integration meetings and conferences certainly saved the logistical day numerous times, and allowed the grad school process to pass as smoothly as it could have.

Last but not least, the PhD experience would not be complete without the amazing lab mates I had the pleasure of working with, from both the Burdick and Ames groups. The many diversions discussing geopolitical affairs, the minutia of programming languages, and the merits of different text editors or desk heights, served as welcome stimulation when work, at times, proved monotonous or stressful.

Funding Sources

Research relating to Chapter 3 of this thesis was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (80NM0018D004). Work on Chapters 4 and 5 was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0016. Government sponsorship acknowledged.

ABSTRACT

As robotic platforms are put to work in an ever more diverse array of environments, their ability to deploy visuomotor capabilities without supervision is complicated by the potential for unforeseen operating conditions. This is a particular challenge within the domain of manipulation, where significant geometric, semantic, and kinetic understanding across the space of possible manipulands is necessary to allow effective interaction. To facilitate adoption of robotic platforms in such environments, this work investigates the application of functional, or behavior level, autonomy to the task of manipulation in uncertain environments. Three functional autonomy techniques are presented to address subproblems within the domain.

The task of reactive selection between a set of actions that incur a probabilistic cost to advance the same goal metric in the presence of an operator action preference is formulated as the Obedient Multi-Armed Bandit (OMAB) problem, under the purview of Reinforcement Learning. A policy for the problem is presented and evaluated against a novel performance metric, disappointment (analogous to prototypical MAB's regret), in comparison to adaptations of existing MAB policies. This is posed for both stationary and non-stationary cost distributions, within the context of two example planetary exploration applications of multi-modal mobility, and surface excavation.

Second, a computational model that derives semantic meaning from the outcome of manipulation tasks is developed, which leverages physics simulation and clustering to learn symbolic failure modes. A deep network extracts visual signatures for each mode that may then guide failure recovery. The model is demonstrated through application to the archetypal manipulation task of placing objects into a container, as well as stacking of cuboids, and evaluated against both synthetic verification sets and real depth images.

Third, an approach is presented for visual estimation of the minimum magnitude grasping wrench necessary to extract massive objects from an unstructured pile, subject to a given end effector's grasping limits, that is formulated for each object as a "wrench space stiction manifold". Properties are estimated from segmented RGBD point clouds, and a geometric adjacency graph used to infer incident wrenches upon each object, allowing candidate extraction object/force-vector pairs to be selected from the pile that are likely to be within the system's capability.

PUBLISHED CONTENT AND CONTRIBUTIONS

Joseph Bowkett, Matt Burkhardt, and Joel W. Burdick (2016). “Combined Energy Harvesting and Control of *Moball*: A Barycentric Spherical Robot”. 2016 International Symposium on Experimental Robotics. In: *Springer Proceedings in Advanced Robotics*, vol 1. pp. 71–83. DOI: 10.1007/978-3-319-50115-4_7.

J.B. built the test apparatus, undertook harvesting experiments to validate prior simulations. Also designed, simulated, and experimentally tested strategies for magnet control. Not included in thesis as is outside the functional autonomy narrative.

Joseph Bowkett and Rudranarayan Mukherjee (2017). “Comparison of control methods for two-link planar flexible manipulator”. 2017 ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference In: *13th International Conference on Multibody Systems, Nonlinear Dynamics, and Control* DOI: 10.1115/DETC2017-67937.

J.B. formulated reduced dynamics for the problem, designed and implemented the two control schemes, tested them in simulation, then demonstrated their use on a purpose built experimental platform. Not included in thesis as is outside the functional autonomy narrative.

Joseph Bowkett, Renaud Detry, and Larry H. Matthies (2018). “Semantic Understanding of Task Outcomes: Visually Identifying Failure Modes Autonomously Discovered in Simulation”. 2018 IEEE International Conference on Robotics and Automation Workshop: *Multimodal Robot Perception: Representing a Complex World* URL: <https://natanaso.github.io/rcw-icra18/>.

J.B. developed the task space formulation, designed and scripted task simulations with depth map generation and processing, then constructed, trained and evaluated the visual classification model. Constitutes the early portions of Chapter 4.

Luca Massari, Calogero M. Oddo, Edoardo Sinibaldi, Renaud Detry, Joseph Bowkett, and Kalind C. Carpenter (2019). “Tactile sensing and control of robotic manipulator integrating fiber bragg grating strain-sensor”. In: *Frontiers in Neurorobotics* vol 13. DOI: 10.3389/fnbot.2019.00008.

J.B. aided in the analysis of test data as it pertained to grasping. Paper provides a motivating argument for the inclusion of proprioceptive inference in manipuland comprehension, referenced in Chapter 5.

William Reid, Gareth Meirion-Griffith, Sisir Karumanchi, Blair Emanuel, Brendan Chamberlain-Simon, Joseph Bowkett, and Michael Garrett (2019). “Actively Articulated Wheel-on-Limb Mobility for Traversing Europa Analogue Terrain”. In: *12th Conference on Field and Service Robotics* vol 6.

J.B. aided in maintenance and upgrade of experimental platform, as well as some lab experiments. Multiple mobility mode demonstrated in paper are a motivating example for algorithm described in Chapter 3.

Joseph Bowkett, Joel W. Burdick, and Renaud Detry (2019). “Visual Extraction Effort Estimation for Grasp Selection Among Unstructured Massive Objects”. 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems Workshop: *Autonomous Object Manipulation* URL: <https://manipulation-iros-workshop.github.io/>.

J.B. formulated wrench manifold representation, developed test cases in simulation, developed means of parameter estimation, collected datasets, coded a number of modules and behaviors employed by the experimental platform for the purpose of the paper and others, and evaluated Constitutes the early portions of Chapter 5.

Sahana Prasanna, Luca Massari, Edoardo Sinibaldi, Renaud Detry, Joseph Bowkett, Kalind Carpenter, and Calogero Maria Oddo (2019). “Neuromorphic tactile sensor array based on fiber Bragg gratings to encode object qualities”. In: *Optics and Photonics for Information Processing XIII* vol 11136. DOI: 10.1117/12.2530733.

J.B. contributed to construction of experimental apparatus, furnished control software for actuation, and provided input for the analysis of test data as it pertained to grasping. Paper provides a motivating argument for the inclusion of proprioceptive inference in manipuland comprehension, referenced in Chapter 5.

Peyman Tavallali, Sisir Karumanchi, Joseph Bowkett, William Reid, and Brett Kennedy (2020). “A Reinforcement Learning Framework for Space Missions in Unknown Environments”. To appear in: *2020 IEEE Aerospace Conferences*, Big Sky, MT.

J.B. aided in development of the framework, developed experimental apparatus for extended demonstration of algorithm. Paper describes an earlier iteration of PIU-ED policy documented in Chapter 3.

Joseph Bowkett, Peyman Tavallali, William Reid, Jay Jasper, Blair Emanuel, Brett Kennedy, and Sisir Karumanchi (2020). “The Obedient Multi-Armed Bandit: Selecting Operating Modes in Uncertain Environments with Bounded Exploration”. Submitted to: *2020 Workshop on the Algorithmic Foundations of Robotics*, Oulu, Finland.

J.B. formulated the novel problem space, adapted existing MAB policies to the new problem, developed new policy to address problem in presence of non-stationary distributions, compared policies against novel success metric, developed exteroceptive extension of policy, realized new experimental platform to demonstrate policy with exteroception. Paper constitutes much of the earlier sections of Chapter 3.

Chad C. Kessens, Long Quang, Joseph Bowkett, Yash Oza, Karl Schmeckpeper, Ajinkya Kamat, Jonathan Fink, Arnon Hurwitz, Matthew Kaplan, Philip R. Os-

teen, Trevor Rocks, John Rogers, Ethan Stump, Michael DiBlasi, Mark Gonzalez, Dilip Patel, Jaymit Patel, Shiyani Patel, Matthew Weiker, Renaud Detry, Sisir Karumanchi, Joel Burdick, Larry Mathies, Aditya Agarwal, Andrew Dornbush, Maxim Likhachev, Kostas Daniilidis, Sanjiban Choudhury, Aditya Vamsikrishna Mandalika, and Siddartha Srinivasa (2020). “Human-Scale Mobile Manipulation Using RoMan”. Presently submitted to: *Journal of Field Robotics, Special Issue on Robotics Collaborative Technology Alliance*, Wiley Online Library.

J.B. packaged planning software and developed behaviors for manipulation tasks undertaken by the platform, developed wrench reactive motion control employed for massive object interaction, acted as prime technical liaison between JPL and sponsors for the platform aiding in firmware and hardware issue diagnosis and correction, and was a key contributor to development of the capstone demonstration of the program, involving 7 integration trips to different collaborator sites. The experiments and behaviors described within the paper motivate the algorithm described in Chapter 5.

Joseph Bowkett, Sisir Karumanchi, Joel Burdick, Renaud Detry (2020). “Algorithms for Grasping and Forceful Manipulation of Novel Objects in Unstructured Environments”. Presently submitted to: *Journal of Field Robotics, Special Issue on Robotics Collaborative Technology Alliance*, Wiley Online Library.

J.B. package the grasp planning software described within, aided in the integration and extension of end effector motion control modules, as well as developed and implemented the grasp “region of interest” selection algorithms. The final section of the article constitutes much of Chapter 5.

TABLE OF CONTENTS

Acknowledgements	iii
Abstract	v
Published Content and Contributions	vi
Table of Contents	ix
List of Illustrations	xi
Nomenclature	xiv
Chapter I: Introduction	1
1.1 Motivation: Autonomy in Uncertain Environments	1
1.2 Problem Statement	5
1.3 Review of Existing Work	7
1.4 Structure of the Thesis	17
Chapter II: Background and Preliminaries	18
2.1 Homogeneous Transforms & Wrench Space	18
2.2 Multi-Armed Bandit Theory	20
2.3 Convolutional Neural Networks	21
2.4 Gaussian Process Implicit Surfaces	22
Chapter III: Reactive Discrete Operating Mode Selection	24
3.1 The Obedient Multi-Armed Bandit Problem	29
3.2 Policies for Stationary Distributions	32
3.3 Preferential Incremental Update Policy for Non-Stationary Case	35
3.4 Exteroceptively Informed Action-Value Decay	36
3.5 PIU Parameter Selection	38
3.6 Mobility Application	38
3.7 Excavation Application	41
3.8 Summary	46
Chapter IV: Semantic Task Outcome Classification	48
4.1 Semantic Task Outcome Model	51
4.2 Archetypal Cuboid Placement Task	52
4.3 Initial Segmentation Model	60
4.4 ResNet Classification Model	61
4.5 Placement Within Clutter Experiment	63
4.6 Object Stacking Experiment	65
4.7 Summary	67
Chapter V: Forceful Manipulation in Clutter	69
5.1 Wrench Space Stiction Manifold	73
5.2 Gaussian Process Implicit Surface Representation	77
5.3 Parameter Estimation	78
5.4 Grasp Candidate Selection	80
5.5 Summary	81

Chapter VI: Conclusions and Discussion	83
Bibliography	88
Appendix A: Calculations	103
A.1 Actuator Power Consumption Estimation	103
A.2 Wrench Reactive End Effector Deflection	104
A.3 End Effector Static Wrench Subtraction	107
Appendix B: Semantic Model Inference on Real Depth Images	108
B.1 Placement in Clutter	108
B.2 Stacking of Cuboids	111

LIST OF ILLUSTRATIONS

<i>Number</i>	<i>Page</i>
1.1 Structured operating environments robots have been deployed	1
1.2 Potential future applications of functional autonomy	2
1.3 RoboSimian competing in DARPA Robotics Challenge	3
1.4 Operator panel used during DRC trials for valve turning task	4
1.5 DARPA Autonomous Robotic Manipulation Challenges	4
2.1 Example deep neural network architectures	21
2.2 1D Gaussian Process posterior distribution example	22
2.3 Gaussian Process Implicit Surface potential field definition	23
3.1 Example planetary science applications for multi-modal operation . .	24
3.2 Overview of categories of Reinforcement Learning techniques	27
3.3 Probabilistic graphical models of MAB and OMAB formulations for both stationary and non-stationary distributions	30
3.4 Four Mobility Modes of RoboSimian: Driving, Inch-worming, Sculling and Walking	39
3.5 RoboSimian platform restrained by tether in a lab environment to impart artificial slip, and cost metrics accumulated in lab experiments	40
3.6 Comparison of “disappointment” metric trends across policies over MATLAB simulated long mobility runs	42
3.7 Three substrate excavation modes available on the SURROGATE multi-tool platform	43
3.8 Region of interest within sensor head camera field of view for external condition change detection	44
3.9 Action-values and mode sequence for substrate step change without exteroceptive decay (PIU)	45
3.10 Action sequence of PIU policy for substrate step change	45
3.11 Region of interest seen from sensor head before and after change of substrate	46
3.12 Action-values and mode sequence for substrate step change with exteroceptive decay (PIU-ED)	47
3.13 Action sequence of PIU-ED policy for substrate step change	47

4.1	Example outcome space for manipulation prototypical task of placement of object into receptacle in the presence of clutter	48
4.2	Probabilistic graphical model of corrective action selection for manipulation task outcome	51
4.3	Semantic task outcome model example output with three result modes for archetypal cuboid placement task	53
4.4	Example cuboid placement task scene at initialization of simulation, prior to BULLET physics engine imparting gravity and contact dynamics.	55
4.5	Slice of DBSCAN clustering within cuboid placement task outcome parameter space, highlighting distinction between objects resting with major axis horizontal on ground plane outside container, from those held above the edges of container.	56
4.6	Example depth images of outcome modes discovered in simulation for cuboid container placement task.	57
4.7	Synthetic depth maps at various stages of processing, in comparison with real depth map with ground plane subtracted	58
4.8	Synthetic depth maps demonstrating noise model and preprocessing applied prior to training and inference	59
4.9	Synthetic noisy depth maps, ground truth labels, and example model outputs for simulated cuboid placement task	61
4.10	Task outcome model applied to real depth image	62
4.11	Model demonstration task involving placement of yellow condiment bottle into container in presence of clutter objects from YCB dataset.	63
4.12	Example depth images of outcome modes discovered in simulation for condiment placement within YCB clutter task.	64
4.13	Inferred outcome modes from visual classification model of placement within clutter task.	65
4.14	Model demonstration task of stacking cuboid objects from the YCB dataset.	66
4.15	Example depth images of outcome modes discovered in simulation for YCB cuboid stacking task.	67
5.1	Example failure case of mobile manipulator	69
5.3	Example object support configuration in \mathbb{R}^2	73
5.4	A single inter-object wrench defined in the center of mass (COM) frame of the target object	75

5.5	Deterministic wrench space stiction manifold for example objects in \mathbb{R}^2 configuration space	76
5.6	Example probabilistic object support configuration in \mathbb{R}^2 , where parameters such as mass, friction coefficient, and contact angle are normally distributed	77
5.7	Wrench space of \mathbb{R}^2 example with contact parameters perturbed about normal distributions, and minimum magnitude extraction vector identified	78
5.8	Example singulation of objects within task scene using Locally Connected Convex Connected Patches algorithm	79
5.9	Depiction of pointcloud perceived center of mass shift induced by rotation of an object in the field of view	80
5.10	Test debris pile scene visualized in simulation, with grasps generated on selected object	81
5.11	Wrench space stiction surface Gaussian Process Implicit Surface representation with proprioceptive datapoint (in red) added after a failed extraction attempt along the red extraction vector for a particular object.	82
6.1	RoboSimian wheel-on-leg platform deployed on Europa analog terrain during field trial	84
6.2	Example of manipulation task with higher visual complexity, the folding of a piece of cloth.	86
A.1	Comparison of sum of individual actuator power consumption estimates with log of net power supplied to entire system, with hotel costs subtracted.	105

NOMENCLATURE

Throughout

- \mathbb{R} Real Numbers
- \mathbb{N} Natural Numbers
- \mathbb{H} Quaternions
- g_{a2b} Homogeneous transformation from frame b to frame a
- W_B^A Wrench of name A in frame B
- $\text{Ad}_{g_{a2b}}$ Adjoint transformation from frame b to frame a
- $\mathcal{U}(A)$ Uniform distribution over set A
- $\mathbb{1}_{\{condition\}}$ Indicator function, 1 when *condition* is true, 0 otherwise

Reactive Discrete Operating Mode Selection

- MAB Multi-Armed Bandit
- OMAB Obedient Multi-Armed Bandit
- \mathcal{A} Set of action/mode indices
- χ_i Reward distribution of action i
- μ_i Mean value of reward distribution of action i
- $a(t)$ Action index selected by agent at time t
- $r(t)$ Reward returned by environment at time t
- $Q_i(t)$ Action-value function of action i at time t
- P Index of action/mode commanded by an operator
- $c(t)$ Cost (progress/energy) incurred by agent over timestep t
- \bar{c}_i Empirically determined nominal/optimal cost for action i
- ρ Regret parameter of MAB theory
- λ Allowable deviation of commanded mode from nominal cost
- $\Delta(t)$ Measured deviation from nominal cost of action taken at time t
- D Disappointment parameter used to quantify efficacy of OMAB policy

$I(t)$	Active set of available actions at time t with minimum action-value
ε	ε -greedy poolicy parameter for exploration rate
Δ_i	Deviation from nominal cost for action i
T	Terminal time of the horizon over which policy operates
$N_i(t)$	Number of times action i has been sampled prior to time t
π_a^b	Mode selection policy of type a applied to problem b
α	Decay rate of action-values in incremental update rule
ω	Scalar progress of agent towards task goal
Ω	Terminal condition of progress for task complete
$\phi(t)$	Exteroceptive signal from environment at time t
$\alpha^\phi(t)$	Action-value forgetting factor induced by exteroceptive signal

Semantic Task Outcome Classification

CNN	Convolutional Neural Network
s_t	Complete state of the task space at time t
i_t	An image of the task space captured at time t
T	A task being undertaken by the manipulating agent
o_t	The outcome mode present in s_t for task T
a_t	The action chosen by the agent at time t
o_s	Outcome mode representing success within a task
π	A policy enacted to bring the task space to the success mode

Forceful Manipulation in Clutter

RCTA	Robotics Collaborative Technology Alliance
GPIS	Gaussian Process Implicit Surface
W_{com}	Net wrench imposed on an object in the Center Of Mass (COM) frame
\mathcal{W}_{com}	Wrench space stiction manifold
\mathcal{W}_{com}^{sb}	“Stiction breaking” wrench space exterior to stiction manifold, in COM frame
\mathcal{W}_g^{sb}	“Stiction breaking” wrench space exterior to stiction manifold, in grasp frame

\mathcal{W}_g^c	Wrench space interior to force constraints of system, in grasp frame
\mathcal{W}_g^v	Space of viable wrenches exterior to sb, interior to c.
g	Gravity
w^{net}	Net weight imposed on an object by gravity and incident contacts
i	Index of an inter-object contact incident upon a target object
\bar{r}_i	Vector describing contact point of contact i in COM frame of target object
\bar{n}_i	Normal force vector imposed upon target object at contact point i
\bar{f}_i	Frictional force imposed upon target object at contact point i
w_i^{net}	Net weight of object contacting target at contact point i
$rcga$	Robot Centric, Gravity Aligned frame
$ecga$	End effector Centric, Gravity Aligned frame

Chapter 1

INTRODUCTION

1.1 Motivation: Autonomy in Uncertain Environments

Autonomy within robotics has made steady progress over the past few decades, as advances in both sensing technologies, and computational capability, have enabled ever more complete representations of the world to be formulated. This has, in turn, allowed increasingly complex behaviors to be realized, ranging from spot welding with articulated arms in vehicle assembly [140] (Figure 1.1a), to vacuuming in the home [62] (Figure 1.1b).

Despite these advances, however, conventional robots are typically restricted to operating in structured environments, where the complexity of the scenes they must interpret is limited. Automotive factory floors are precisely configured, such that a target task will consistently be placed in an exact position relative to an arm, with no humans in proximity [124]; while autonomous vacuums only operate in confined regions of 2D surfaces, and may be tripped up by as little as an errant cable lying on the floor.



(a) Automotive manufacturing



(b) Home vacuuming

Figure 1.1: Structured operating environments robots have been deployed

For robots to operate in more complex environments they need to be capable of *functional autonomy*, which refers to the ability to intelligently perceive, deliberate, and execute tasks at the behavior level. Functional autonomy attempts to enable an agent to account for unexpected geometry, configurations, circumstances, and disturbances, while achieving a task as efficiently as it is capable.

Some of the most rapidly advancing technology in this field can be found in the autonomous car industry, where functional autonomy describes what is designated as SAE Level 4 (High Autonomy) meaning a vehicle may independently operate across the majority of environments that can reasonably be expected to be encountered. Despite the huge development effort behind these platforms, regulatory authorities have not yet been convinced of their efficacy and safety, demonstrating that the problem is not yet considered solved.

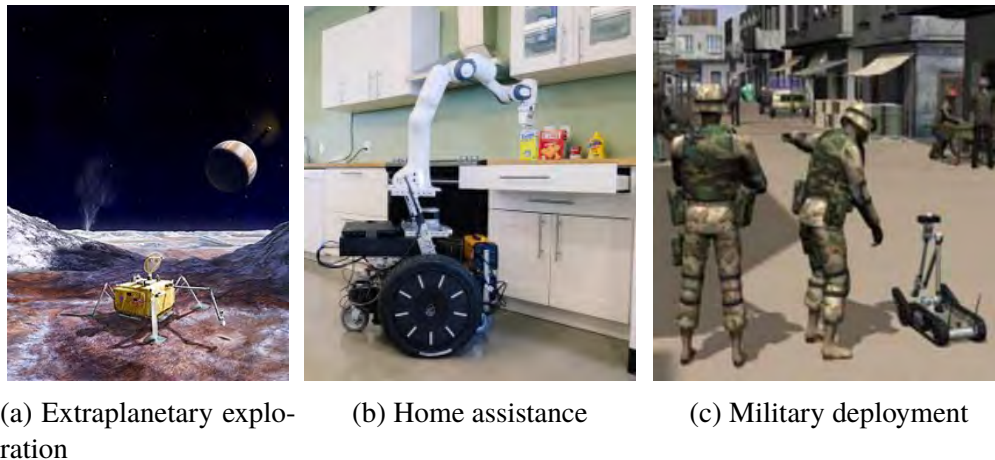
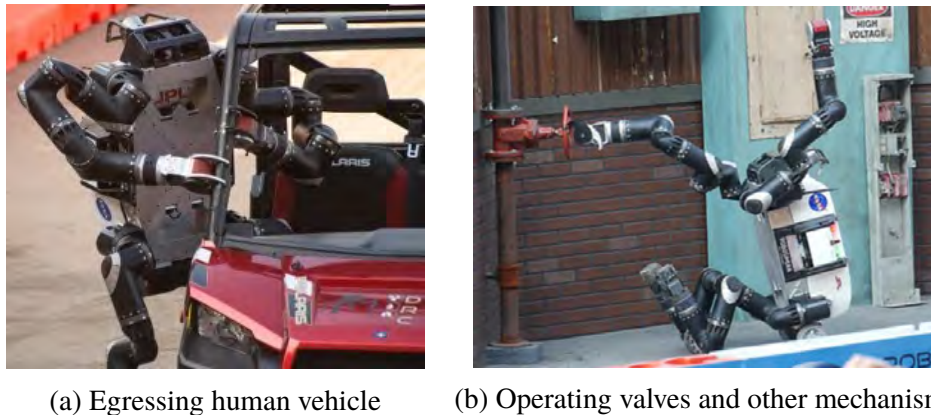


Figure 1.2: Potential future applications of functional autonomy

Recent years have seen increased interest in exploration of solar system bodies more distant than Mars (such as Europa in Figure 1.2a), where round trip communication latency can be up to 2 hours, and batteries are only expected to last for 11 days of operation, meaning the current NASA modus operandi of always having an operator in the loop severely restricts what these missions can achieve. Functional autonomy would allow systems to make their own task level decisions and achieve much greater science return than would be possible in a paradigm which must wait for operator input.

There is an aggressive push within industry to place robots in the home, where little or no operator input can be expected, and yet the home robot may encounter a huge range of objects and task configurations (Figure 1.2b).

The military has also been investing in ways to get robots out into the field, with programs such as the Army Research Lab's Robotics Collaborative Technology Alliance (RCTA) seeking to advance technologies to the point that platforms can be fielded as "team members rather than tools" (Figure 1.2c).



(a) Egressing human vehicle (b) Operating valves and other mechanisms

Figure 1.3: RoboSimian competing in DARPA Robotics Challenge

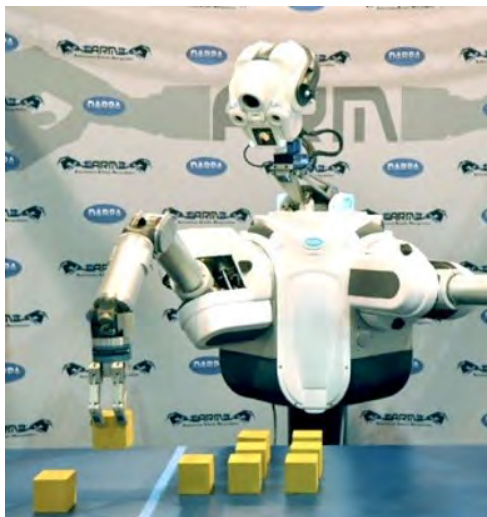
From 2012-2015, the DARPA Robotics Challenge (DRC) sought to advance capabilities of semi-autonomous platforms in industrial settings inaccessible to humans, such as immediately following the Fukushima nuclear disaster. NASA's Jet Propulsion Laboratory (JPL), designed the RoboSimian platform to compete within this competition (Figure 1.3), demonstrating the ability to drive and egress a human scale vehicle, operate valves and other mechanisms, and complete a number of other tasks [54, 65]. Despite several of the tasks using scripting to complete sequences of component actions, significant operator input was required to guide both positioning of the task frame, and recovery from any failure conditions. An example of the operator panel for the valve turning task can be seen in Figure 1.4. For that reason, the systems in the competition were still considered to be teleoperated, rather than capable of functional autonomy, meaning they could not be expected to operate independently in the aforementioned scenarios. The RoboSimian platform, and several of its derivatives, were later used in much of the work described within this thesis. Further detail of their kinematics and operation are included in the respective technical chapters.

An earlier competition, DARPA Autonomous Robotic Manipulation (DARPA ARM), attempted to demonstrate independent autonomy within a restricted set of environments, some of which are pictured in Figure 1.5 [58]. The block placement task relied on high contrast yellow blocks against a blue background, reducing the complexity for the vision system, with no confounding objects. The wheel unbolting and door opening tasks relied upon *a priori* models of their task space, allowing precise motion patterns to be pre-configured and exteroceptive input to be matched to geometric models. While these demonstrated capabilities were impressive, they

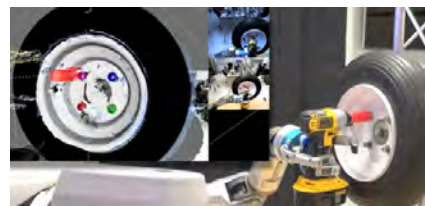


Figure 1.4: Operator panel used during DRC trials for valve turning task

may not transfer to uncertain and unstructured environments, with little to no *a priori* models, that would be encountered out in the wider world (or a different one).



(a) Block placement



(b) Wheel unbolting



(c) Door opening

Figure 1.5: DARPA Autonomous Robotic Manipulation Challenges

These competitions have markedly advanced the complexity of tasks that robotic platforms have been able to achieve autonomously, but demonstrate that there is still significant work to be done before they will be capable of operating without supervision in uncertain and unstructured environments.

1.2 Problem Statement

The aim of this work is to advance the ability of robotic manipulation platforms to operate in environments with uncertainty, which may be caused by lack of *a priori* models, non-stationarity of internal and external conditions, or latent environmental structure beyond a robot system's ability to exteroceptively perceive and understand.

This thesis examines three sub-areas of this problem, which are detailed below.

Reactive Discrete Operating Mode Selection

As robots are sent into progressively more complex environments, ensuring their ability to operate despite changing or poorly characterized environmental conditions at times necessitates equipping them with multiple operating modalities to achieve the same task.

An example of this is the extraplanetary mobility presently being investigated by a team at JPL [111], where different means of surface traversal may be needed to account for the variety of terrain that may be encountered in off-Earth deployments. Past NASA missions have also employed alternating means of locomotion, such as the Spirit rover which had its front right wheel cease motion 2 years into operation, requiring the operations team to drive the platform backwards [80].

At the same time, interest is shifting towards more far flung celestial bodies, such as Europa and Enceladus, where surface conditions can only be roughly approximated with present data [22, 86], and round trip communication time with Earth can be up to two hours.

Mission time for a lander on Europa or Enceladus is projected to be on the order of 11 days, due to lack of sufficient solar energy or a Radioisotope Thermoelectric Generator, meaning that science yield will be directly dependent on the speed of operation. Coupled with the significant round trip delay time, and minimal bandwidth to send back diagnostic data, future missions may need the ability to intelligently select between pre-designed operating modes in response to the conditions they encounter. This is a significant deviance from historical operating procedure for Mars missions, for example, where every component action is sequenced and evaluated by ground-staff, before upload to the remote platform.

The goal of this work is to formulate an algorithm for reactive mode switching in response to task efficiency metrics, while adhering to NASA doctrines for extraplanetary systems that require commands to be uploaded at intervals in sequences,

so as to remain palatable to mission designers.

Semantic Task Outcome Classification

Beyond the many challenges of synthesizing grasps and manipulating objects in uncertain environments, the ability to assess the *outcome* of a task execution can also prove complicated. This is particularly true when lack of *a priori* models makes it difficult to apply traditional visual pose recovery techniques to objects of interest within a scene, or when target manipulands (objects being manipulated) are deformable.

Recent advances in Convolutional Neural Networks (CNNs) have shown them capable of capturing contextual and relational information within a scene, much as is needed to discern the outcome of a given task conducted by a robotic agent. The challenge in their use is that they require vast amounts of data to train the weights of the network, on the order of tens of thousands of images. This necessitates laborious collection of images, or sourcing of suitable datasets from online resources which may not be able to provide the volume necessary to train a particular task. Synthetic RGB images have been used to train CNNs, but typically fail to transfer well to inference on real images, due to the difficulty of rendering photorealistic datasets. By contrast, the depth maps produced by cameras such as the Microsoft Kinect, or the Intel RealSense, are far easier to simulate, and past work has shown models trained on synthetic images can produce models capable of accurate classification on real datasets. This enables the use of physics simulations to both discover and generate synthetic depth maps of various task outcome modalities for selected archetypal robotic tasks.

The goal of this work is to demonstrate a depth based task outcome classification model that is capable of differentiating between success and distinct failure modes discovered autonomously in simulations of representative manipulation tasks.

Forceful Manipulation in Clutter

In order for robotic agents to interact with the wide range of object configurations they may expect to encounter in a unconstrained environments, they must manage the problem of lifting manipulands whose weight or bulk might exceed the capabilities of the system. This may occur, for example, when lifting an object that is of manageable mass in isolation, but is subject to restraining forces imparted upon it via contacts with other objects within a pile structure. These may include the effect of these incident objects mass, as well as frictional forces induced during the

dynamic act of dislodging the candidate object for removal.

individual objects lying incident upon one another pose a combined mass

While prior art has looked at means of synthesizing grasps on previously unseen objects, emphasis has not been placed on the mass and configuration of such objects; in particular, how they impact the effort necessary for a system to extract or manipulate a component of it.

Even the most advanced sensing methods presently available within robotics experience a fair amount of noise, and the incomplete information they furnish produces significant uncertainty in the representation of a scene. If noisy exteroception (external perception) is used to select a candidate object for manipulation, yet moving it exceeds the capability of the system, this proprioceptive (internal/self) information should be used to update the understanding of the scene.

The goal of this portion of the work is to formulate a probabilistic representation of the forces restraining candidate manipulands, that is capable of incorporating noisy measurements from different modalities. This formulation will then be investigated for suitability in *selecting* grasp regions from unstructured piles of objects that are within the limitations of the system, so that grasps synthesized on these regions are more likely to succeed at composite tasks such as deconstructing a pile of massive and/or interlocked objects.

1.3 Review of Existing Work

Functional Autonomy

As the range of complex tasks for which robotic platforms have been designed has burgeoned, so too has the variety of reasoning methods used to imbue them with the ability to act autonomously. While the field of Artificial Intelligence as progressed independently for the past few decades, wider availability of platforms with advanced sensorimotor capabilities has seen renewed interest in applying its teachings to robotics [59].

The problem of planning may be the most common type of autonomy, and encompasses many task spaces including motion, perception, navigation, manipulation, and communication, among others. While some recent efforts have attempted to formulate domain independent planning schemes with a view to executing complex multi-domain tasks, these are still predominantly addressed using domain specific solutions [45].

As autonomous robotic systems are deployed into increasingly uncertain and dynamic environments, the incidence of task failures is only going to increase while sensing and reasoning remain fallible. The robotics field has attempted to address what is typically termed *fault diagnosis and recovery* (FDR), but this increase in likelihood of failure has merited renewed research interest in recent years. Other fields, however, such as that of industrial automation, has a longer history of fruitful research on what they term *execution monitoring*, which may be able to guide robotics in years to come [106].

One key focus area of high-level autonomy research is the formulation of system level goals, one level above that of functional autonomy. This includes the ability to explicitly express goals, automatically formulate them in response to outside stimuli, and manage them dynamically [136]. This capability is crucial to the execution of compound tasks, and allows an agent to intelligently select between component behaviors, such as might be individually managed with functional autonomy.

The analysis of task/behaviour spaces has occurred in parallel across several fields, including computer vision, robotics, and AI; particularly with the advance of imitation learning techniques that seek to replicate tasks carried out by example agents through observation [72]. Some of this work has been informed by neurobiological research investigating the links between visual analysis and motor representation in the nervous system [114]. Robust action representation and recognition allows a system to interact more readily with other agents, be they organic or machine, and will likely prove critical in both high-level and functional/behavior level autonomy in coming years. A common example of this comes as certain areas of industry seek to deploy robots within human-collaborative environments, such as workplaces and homes. As a result of the added complexity of planning within the safety constraints imposed by proximity of delicate humans, the subject of human-aware navigation has seen an increased emphasis in research [73].

In order for modern robotics platforms to leverage in real time the breadth of sensing and actuation technologies available to them, their architectures have become increasingly complex. Selection and design of such architectures suffers from competing design goals, in particular modularity and hierarchy, and trade-offs can have significant impact on the resultant behavior and capability of an autonomous system [68].

Perhaps the ultimate recent example of functional autonomy deployed on robotic platforms with limited operator access or communication is that of extraterrestrial

rovers, as sent primarily to Mars and the moon. Minimal communication bandwidth has meant these platforms need to be capable of long distance traversal through terrain not previously seen by operators, in order to produce greater science yield than can be had by navigating purely by direct observation of operators [139]. As space agencies look towards exploring more remote regions of the solar system, where communication is further reduced and missions' duration likely shorter, the need for functional autonomy in both navigation and invasive interaction is increasing.

Reactive Discrete Operating Mode Selection

Much of the modern research into autonomy falls under the domain of machine learning, and a subfield of this that has become increasingly common within robotics in recent years is that of Reinforcement Learning (RL), which focuses on in-situ learning from experience. Reinforcement learning derives its origins from two distinct research fields, optimal control theory, and biological trial-and-error learning, which were brought together in the 1980s [128]. The origins of optimal control stem from work in the 19th century by Hamilton [50] and Jacobi [60], whose eponymous equation was extended by Bellman to produce a necessary and sufficient condition for a given controller to be optimal with respect to a selected objective function. The term “Dynamic Programming” was coined to describe the class of methods capable of solving the Hamilton-Jacobi-Bellman equation [10]. Bellman also demonstrated the use of discrete time stochastic processes in optimal control, commonly formulated as Markov Decision Processes (MDP) [11]. The MDP formalism originates from the concept of a Markov Chain, first postulated by the Russian mathematician Andrey Markov [89], where each state transition in a sequence of actions satisfies the Markov Property [90].

One of the earliest connections between optimal control and learning came from Werbos, who proposed a concept called “heuristic dynamic programming”, applied to the problem of pattern recognition in weather forecasting that was treated as a system (appendix of [145]). Following this work, Werbos became a proponent for the use of dynamic programming to understand cognitive function in mammals [146]. Modern Reinforcement Learning is frequently posed using the MDP formalism, with perhaps the earliest learning example coming by Watkins in his 1989 thesis “Learning from Delayed Rewards” [142]. This was then demonstrated through the “Q-learning” theorem, a means for agents to learn to act optimally in Markovian domains [143]. Bertsekas and Tsitsiklis later extended this notion by explicitly incorporating neural networks into dynamic programming [13], to create what they

termed “Neuro-Dynamic Programming”.

The RL field rapidly branched out into a variety of approaches to suit different problem constraints, but they may broadly be broken into subgroups by the observability of problem states, being fully or partially observable, or alternatively unobservable. Fully or partially observable state space method may then be further subdivided by their temporal representation, being either discrete, continuous, or tabular [128]. Tabular methods address problems where the dimensionality of potential state and action spaces are small enough that approximate value functions may be represented in arrays rather than calculated [63]. They have included Q-Learning from the Bellman equation [31], Temporal Difference algorithms such as State-Action-Reward-State-Action (SARSA) [118], and Monte Carlo algorithms [130]. Discrete time methods determine the next action step at discrete points in time without use of a lookup table, and have similarly employed Q-Learning, as well as Deep Q Networks (DQN) [96]. Continuous action learning generalizes some these concepts to continuous time, with examples including Deterministic Policy Gradient (DPG) [125], and William’s REINFORCE algorithm [148].

Of greater relevance to this thesis are problems where the states of a problem space are *unobservable* to the agent, such that only a limited *reward* variable is available to infer the outcome of an action. One primary means of formulation for unobservable states, following on from the MDP, is that of the Hidden Markov Model (HMM), the earliest description of which comes from Baum et al. [8]. Some early applications of the HMM included speech recognition [61, 109], and, more recently, extensive use in the analysis of sections of biological DNA sequences [16, 38]. A selection of the problems the model has been applied to seek to infer the hidden internal states from the observable reward, and techniques addressing this include the Baum–Welch algorithm [7] (otherwise known as forward-backward), and the Viterbi algorithm, originally developed to decode convolutional codes in telecommunications [138].

By contrast, the problems in this thesis chapter are characterized by the hidden internal states that are of negligible importance compared to the optimization of the observable outcome. In particular, those where finite resources must be allocated between a discrete set of actions in order to optimize a given outcome, where parameters of each action are not, or poorly, characterized *a priori*. This is an archetypal example of the “explore exploit” tradeoff problem within the field of learning, and is typically modelled with Multi-Armed Bandit (MAB) theory, or K-armed bandit theory, in reference to a set of “one-armed bandit” slot machines with probabilistic

outcomes at gambling establishments. Early work on the prototypical form the problem focused on producing strategies for an optimal rate of convergence to the action with the highest mean outcome, with Lai and Robbins demonstrating a solution for populations with univariate density functions in 1985 [75]. Katehakis and Robbins then provided a proof for the case of normal populations with known variance [66], followed shortly thereafter by a solution for non-parametric distributions from Burnetas and Robbins [25].

Various methods to find approximate solutions to the MAB problem have also been investigated. A seminal advance was the ϵ -greedy algorithm, where a scalar parameter determines a static rate of exploration [142]. Some groups investigated pure exploration [23], but most techniques beyond moved to temporal dependence, such as the ϵ -first algorithm, where a scalar parameter directs a single transition from exploration to exploitation in a finite set of actions [132]. Another example comes from Cesa-Bianchi, who proposed the ϵ -decreasing algorithm, that varies its ϵ parameter as a function of iteration, so as to transition from early exploration to later exploitation [27]. Other groups employed adaptive methods, such as Tokic's "Value-Difference Based Exploration", which varies the ϵ parameter as a function of the agent's uncertainty about the environment, using temporal-difference error [131].

A particularly relevant offshoot of MAB theory is that of Contextual Multi-Armed Bandit theory, which seeks to bridge MAB theory and full Reinforcement Learning by introducing an observable state, outside of the pure reward distribution. One of the earliest examples comes from Wang et al. 2005, who proposed allowing an agent to observe a random variable that contains some correlation with the reward before each decision point [141]. Langford and Zhang developed this further into their Epoch-greedy algorithm, and demonstrated its application to matching internet based ads with users [76], which has since become a key focus of the Contextual MAB field. The contextual dependency between arms of the "bandit" was investigated by Pandey et al. [102], who propose an optimal MDP-based policy in the discounted reward case. Lu, Pal, and Pal investigated lower regret bounds for contextual MAB as a function of dimensions of the information space, again applied to the problem of maximizing internet ad revenue from viewers [85]. For a more extensive review of the contextual MAB field the reader is directed to Zhou 2015 [151].

A key assumption within most of the above works is that of stationarity of the under-

lying distributions of the systems being explored, and proven optimal policies were premised on this assumption. The limitations of existing strategies was identified in [6], where Auer et al. proposed the “UCB1” bandit algorithm where regret grows at most logarithmically, but it wasn’t until 2006 that an approach to non-stationarity within the MAB formulation was proposed by Kocsis and Szepesvári, who investigated the performance of an Upper Confidence Bound (UCB) policy when relaxing the stationarity assumption [67], that was originally proposed by Agrawal [1]. Garivier and Moulines followed on from this earlier work by analyzing both discounted and sliding-window UCB, establishing upper bounds on the expected regret via the number of times a suboptimal arm is played [44]. In the same year, Alaya-Feki et al. showed that a MAB could be applied to the non-stationary problem of Dynamic Spectrum Access in telecommunications [3]. Burtini et al. combined the concept of contextual MAB with the relaxation of stationarity, to propose a weighted least squares technique for optimization of web advertising among drifting interests within the target base [26]. The ability to incorporate knowledge of the trend within underlying distribution drift was contributed by Bouneffouf and Féraud, who presented an algorithm named Adjusted Upper Confidence Bound (A-UCB), motivated by web based problems such as individual music recommendation with the context of general population scale trends [21].

While much of the research in Multi-Armed Bandit theory has focused on predicting consumer demand trends, it has also seen application within the field of robotics, along with a range of other Reinforcement Learning algorithms. Kroemer, Detry, Piater, and Peters applied UCB methods to the problem of selecting grasps posed as a continuum-armed bandit to address the uncertainty in the high-dimensional grasp space [71]. More recent examples of MAB theory in grasping include Dex-Net 1.0 from Mahler et al. [88], and in 2D grasping from Laskey et al. [77]. The problem of selecting between pre-programmed state-machines within a library was investigated by Matikainen et al. [91], with the motivating example of vacuuming the floor on a room with unknown layout necessitating different coverage techniques. Pini et al. applied existing MAB algorithms (including ad-hoc, UCB, and ϵ -greedy) to the issue of task partitioning with swarms of robot agents [108]. Koval et al. posed the selection from a finite set of object rearrangement trajectories within an MAB formulation [69]. Human Robot Interaction (HRI) has also been approached in this way, when Leite proposed selecting actions for an empathetic robot based on the observed responses of a human user, to maximize social connection [81].

Semantic Task Outcome Classification

Visual classification has long been studied under the auspices of machine learning. Much of the early work on machine learning focused on the emulation of biological organism's capability for reasoning, such as described in Hebb's 1949 book *The Organization of Behavior* [53], which presents theories on the structure of communication between neurons. Some researchers, such as A.L. Samuel from IBM who coined the phrase *Machine Learning*, investigated the application of this machine reasoning to games of logic, in the case of Samuel being checkers [120]. Still other groups focused on the promise of automated machine perception, perhaps earliest among them being Rosenblatt, who conceived of the Perceptron, a machine intended to be capable of recognizing patterns in a similar fashion to that of the neural cortex in biology [116]. While Rosenblatt did not succeed in recognizing faces as was the desired outcome, the stage was set for later work, wielding far greater computational power, to do so. For the next few decades, many efforts then aimed towards explicit pattern classification, such as that of the nearest neighbor rule proposed by Cover and Hart [30].

One key concept leading to modern neuromimetic pattern recognition algorithms was the extension of the perceptron model to include multiple hidden layers. A chief example comes from Fukushima who conceived of the Neocognitron, a multi-hidden layer network that was invariant to positional shift within an input image [43]. Also of relevance was the development of *backpropagation*, popularized by Rumelhart et al. [117], which allowed multi-layer networks to adjust their hidden layers in response to new training data in what is termed the "backward propagation of errors". This was applied successfully to a number of problems, perhaps most influentially by LeCun et al. who demonstrated backpropagation to be an effective gradient-based learning technique [78] as applied to handwritten character recognition with their model LeNet5 [79]. This concept was generalized in the 2003 book "Hierarchical neural networks for image interpretation" [9], but did not see much further development in the early 2000s, until the advent of internet accessible image databases from modern, cheap cameras, coupled with the rapid increase in parallel computing capabilities within processors.

Not until 2012 did the field of deep learning for pattern recognition, in particular Convolutional Neural Networks (CNNs), take prime focus, when Krizhevsky, Sutskever, and Hinton released their AlexNet model [70] that far surpassed the existing state of the art classification of the ImageNet benchmarking database [32].

Research then rapidly accelerated, with all manner of variations on previous models being explored, such as the use of smaller convolutional units with deeper layers from Simonyan and Zisserman [126]. He et al. later proposed the use of feeding forward outputs from earlier layers to be combined with later layers in a network in a model they termed ResNet, or Residual Network [52], which some describe as a “Network within network”, and showed marked improvements in classification accuracy. The majority of modern networks are based off these and other architectures and sub-architectures, including those found in this thesis.

Forceful Manipulation in Clutter

Grasp selection in unstructured environments has proven a challenging task, and is complicated further when lacking *a priori* knowledge of manipuland shape and its mass properties.

Much of the earliest research on manipulation focused on grasp synthesis; the problem of finding a single or set of hand configurations that suitably constrain a target object relative to the agent, subject to any manipulation task specific constraints. Any advanced interaction with an environment first requires suitable contact to be made, for which reason, robust grasping is a pivotal concern for autonomous robotics, and has therefore seen many different approaches developed. While this thesis addresses the selection of object candidates for grasping, the generation of grasps on previously unseen objects experiences similar challenges in the need to predict resultant wrenches from what is frequently incomplete geometric information. Means of grasp synthesis have historically been divided into two camps; that of explicit analytical representations, and more recent data-driven or empirical approaches [18].

Analytic approaches typically focus upon representation of the ability of a given end effector to both restrain, and manipulate, objects relative to a wrist or palm [14]. Restraint of a manipuland is characterized by the closure properties of a grasp, comprised of a set contacts, often modelled as points that are either frictionless, frictional, or soft. Perhaps the most important concept within deterministic analysis is that of grasp force or wrench closure, where contact forces can counteract all other external forces such as gravity. While Reuleaux analyzed efficacy of fixtures and jigs in 1875 [113], it was Salisbury and Roth that showed that a grasp may be considered stable when the stiffness matrix that characterizes the contacts is positive definite [119]. Nguyen postulated that a grasp could be considered *force*

closed if and only if it is in equilibrium for any arbitrary wrench [100], which can necessitate arbitrarily high normal contact forces [74]. A stronger condition is that of form closure, described by Trinkle as existing if and only if it is force closed with frictionless contacts [133]. The ability to achieve these conditions on general geometries, as should be considered in object agnostic grasping, was investigated by Mishra, Schwartz, and Sharir who were the first to find an upper bound on the frictionless contacts needed to achieve form closure on general 3D objects with piecewise smooth boundaries [95].

Up until the turn of the millennium, the majority of grasping research focused on robotic grasping centered around model-based and mechanics-based approaches [18], at which point a shift towards empirical or data-driven methods occurred. This may have been in part due to the rapid progression of computational power available to research labs, as well as the emergence of the simulation platform Graspit! [94]. Empirical approaches initially used classical metrics derived from analytical formulations such as the ϵ -metric proposed by Ferrari and Canny [41]. While this was mathematically rigorous, and tied in with with prior art, recent papers have found these grasp success metrics do not transfer well from simulation to the real world [144], due to their fragility [35]. A number of research groups then pivoted towards using learning techniques to teach suitable grasping through experience. One seminal example came from Saxena et al. who skipped building 3D representations of objects altogether, opting instead to learn to identify workable grasp points from two monocular images taken at different points of view, demonstrating the ability to find viable grasps on novel objects [122]. Bone, Lambert, & Edwards used both camera and laser-on-wrist sensing to capture multiple images of a candidate object from different points of view, and compose a geometric and color space representation of points [19]. At this point, the advent of structured light depth sensors, such as the low price Microsoft Kinect, caused a shift towards use of this modality, which afforded great density of geometric information and were simple to deploy. Bohg et al. divide data-driven grasp synthesis approaches into *known*, *familiar*, and *unknown* objects; all three of which must be accounted for if designing a truly object agnostic algorithm [18]. On the topic of unknown objects, they further divide methods into those that 1) approximate the full shape of an object, 2) methods that generate grasps based on low-level features and a set of heuristics, and 3) methods that rely mostly on the global shape of the partially observed object hypothesis. This provides valuable insight into the techniques that have proven fruitful in planning with incomplete geometric, and little to no *a priori*, information. Bohg and Kragic themselves ap-

plied supervised learning to prototypical grasp points on example objects using the concept of shape context and then tested their transfer to novel objects [17].

One mathematical structure that has seen some recent use in modelling the uncertainty of geometric representation of manipulands is that of a Gaussian Process (GP), which generalizes the scalar normal distribution to a collection of variables, such as may describe a manifold [110]. This may be combined with the formulation of an *implicit surface*, first proposed in 1999 by Turk and O'Brien [135], where the 0-level set of an N dimensional potential field describes the surface of an object in N dimensions. Dragiev et al. employed a Gaussian Process Implicit Surface (GPIS) to represent the geometry of a manipuland [36, 37]. They also define an implicit shape field as the negative gradient of the implicit shape potential, which captures the expected normal direction of the surface. Ottenhaus et al. then demonstrated how a GPIS could be used to capture sparse haptic information when representing objects explored through proprioception [101]. The concept was adapted by Li et al. to employ a thin plate covariance function as it was shown to have preferable behavior at boundaries of the object [83]. Burkhardt et al. used proprioception to infer object mass then applied this thin plate GPIS surface representation to plan grasp points for a second end effector in dual arm manipulation to equitably distribute wrench between the platform's arms [24].

While many of the above works have addressed a variety of means of grasp synthesis, largely agnostic to considerations beyond force or form closure (be it analytical or empirical), attention has more recently turned toward selection of grasp *regions*, with suitability to particular tasks or affordances. An affordance may be informally considered as an 'opportunity for action', as were first proposed by psychologist J.J. Gibson in 1966 [46], and are used to describe the actions an agent may take with a given object or environment. They have been employed in the study of robotic traversal and object avoidance, grasping, and object manipulation [56]. One of the first to combine higher level reasoning with lower-level geometric planning were Antanas et al., who proposed using symbolic object parts to semantically reason about pre-grasp configurations for particular tasks [4]. Detry, Papon, and Matthies then presented a model that computes a distribution of geometrically compatible 6D grasp poses from a depth map, and then applies a CNN-based semantic model to select those configurationally compatible with a given task [34]. The selection of grasp type and location during a *handover* task were also recently investigated by Cini et al., who had human subjects pass and receive a range of objects [28].

Much of the study of manipulation affordances has emphasized grasp selection re- spective of the end-goal of composite or collaborative tasks, such as handing over an object to another agent, or pouring from a container [34]. In unconstrained envi- ronments, where the mass of potential manipulands may vary greatly, the affordance of lifting may arguably be considered more important, as inability to lift an object typically precludes any other action. Some early works applied pure proprioception to identify the center of mass (COM) of an object in-hand [5], but there has been little use of exteroceptive sensing means to predict and inform this process. One example is found from Kanoulas et al., who address the question of wrench mini- mizing grasp selection on a single object by exteroceptively predicting the COM and then iteratively lifting and updating the estimate with wrist torque measurements [64].

Unconstrained manipulation environments are commonly cluttered, with many po- tentially graspable or confounding objects present within a scene. Much focus has been placed on this problem within the context of the ‘Amazon picking challenge’, which sought to advance logistics technology capable of sorting through heteroge- neous boxes of consumer items. One of the key takeaways from early iterations of the challenge was the importance of combining reactive control with deliberative planning [29]. Development by various teams led to the demonstration of the ability to recognize and grasp both known and novel objects in cluttered environments [149], though these operate with manipulands well within the grasping system’s capability. Earlier work from Boularias, Bagnell, and Stentz used reinforcement learning to sequentially select pushing and grasping actions in order to remove rubble from a pile [20].

1.4 Structure of the Thesis

Chapter 2 describes the preliminaries of the mathematics for the following chapters. Chapter 3 lays out a Reinforcement Learning formulation that allows an agent to select between a discrete set of operating modes, in response to both changing environmental and intrinsic efficiency, while attempting to adhere to direction from an operator. Chapter 4 describes a method for synthetic generation of data to train depth CNN models for differentiating the outcomes of manipulation tasks. Chapter 5 details a model for representing the forces restraining massive objects in piles to enable sequencing of forceful manipulation actions. Chapter 6 summarizes the conclusions of each chapter, and suggests future avenues of work that could be explored.

Chapter 2

BACKGROUND AND PRELIMINARIES

This chapter introduces the mathematical background and some algorithmic functions utilized in later chapters.

2.1 Homogeneous Transforms & Wrench Space

This section reviews the basics of representing homogeneous transformations, used to describe relative poses in two or three dimensions, common across all chapters of the thesis. It also describes the “wrench” representation of forces and torques within two and three dimensions, along with means of manipulating them, including transformation between different reference frames. Definitions follow those from Murray, Li, Sastry [98], and are noted in the three dimensional case.

The specification of a position within three dimensions, $(q_1, q_2, q_3) \in \mathbb{R}^3$, is represented in *homogeneous coordinates* in \mathbb{R}^4 as

$$\bar{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ 1 \end{bmatrix}. \quad (2.1)$$

Such a point may be subjected to an *affine* transform, a linear mapping preserving straight lines and planes, from coordinate frame A to coordinate frame B . This *homogeneous transformation* is parameterized by a vector describing the position of the origin of frame B relative to the origin of frame A , $p_{ab} \in \mathbb{R}^3$, and a rotation matrix describing the orientation of frame B relative to frame A from the special orthogonal group, $R_{ab} \in SO(3)$. The resulting configuration space is denoted the *Special Euclidean Group*, given by

$$SE(3) := \{(p, R) : p \in \mathbb{R}^3, R \in SO(3)\} = \mathbb{R}^3 \times SO(3). \quad (2.2)$$

Given a point q denoted $q_a \in \mathbb{R}^3$ in coordinate frame A , and $q_b \in \mathbb{R}^3$ in coordinate frame B , we may transform between these two representations via

$$q_a = p_{ab} + R_{ab}q_b . \quad (2.3)$$

This transformation may be constructed in the *homogeneous representation*, which allows associative composition of transformations through matrix multiplication, as well as inversion to enable inverse transformations

$$g_{ab} := \begin{bmatrix} R_{ab} & q_{ab} \\ 0 & 1 \end{bmatrix} \in SE(3) . \quad (2.4)$$

A point in homogeneous coordinates may then be transformed between coordinate frames through multiplication with the homogeneous transformation matrix between those frames

$$\bar{q}_a = g_{ab}\bar{q}_b , \text{ where } \bar{q}_a, \bar{q}_b \in \mathbb{R}^4 . \quad (2.5)$$

The effect of a generalized force acting on a rigid body may be decomposed into the linear component as a pure force, and the angular component as a pure moment, acting on a point in a given reference frame. This generalized force may be represented via a vector in \mathbb{R}^6 , termed a *wrench*, given as

$$W = \begin{bmatrix} f \\ \tau \end{bmatrix} \quad \begin{array}{l} f \in \mathbb{R}^3 \quad \text{pure force} \\ \tau \in \mathbb{R}^3 \quad \text{pure moment} \end{array} \quad (2.6)$$

In this thesis, a wrench is denoted with a subscript describing the coordinate frame within which it is described, and a superscript (if present) giving a contextual name, such as

$$W_B^A := \text{Wrench named A in coordinate frame B} \quad (2.7)$$

A wrench defined at a given point may be said to be equivalent to a wrench defined at a different point if they produce the same net work for every possible rigid body motion, allowing wrenches to be rewritten relative to a different coordinate frames. This wrench translation between frames makes use of the *adjoint transformation*, which operates on the wrench dual space of *twists*, that describes rigid body velocities, and are also represented in \mathbb{R}^6 , where $\hat{\cdot}$ is the isomorphism from \mathbb{R}^3 to 3x3 skew symmetric matrices [98]. The adjoint transformation for twists is constructed as

$$\text{Ad}_{g_{ab}} := \begin{bmatrix} R_{ab} & \hat{q}_{ab}R_{ab} \\ 0 & R_{ab} \end{bmatrix}, \quad (2.8)$$

Given a homogeneous transform between frames A and B , $g_{ab} \in SE(3)$, a wrench described in frame B may be transformed into frame A via the dual of the adjoint transformation, being the transpose

$$W_a = \text{Ad}_{g_{ab}}^T \cdot W_b \quad (2.9)$$

2.2 Multi-Armed Bandit Theory

The explore-exploit problem of multi-armed bandit theory seeks to optimize an outcome over a sequence of actions, where each action returns some stochastic *reward* from an unknown underlying distribution. It may be considered a one-state Markov decision process, and is a subset of the Reinforcement Learning field. The work in this thesis focuses on the discrete case of action selection, and formulations will be posed as such, following [137].

Actions are drawn from $K \in \mathbb{N}^+$ options, or arms in the slot machine analogy, with each possessing an associated reward likelihood drawn from a real distribution, represented as a set of distributions, B , where

$$B := \{R_1, \dots, R_K\}. \quad (2.10)$$

Each of these real distributions possesses a mean, denoted $\mu_i := \mathbb{E}[R_i]$. The objective of most MAB algorithms is to minimize the *regret*, ρ , which is the difference between the reward per trial, \hat{r}_t , accumulated over some number of trials T , and the maximum possible mean reward that could have been collected by sampling purely from the optimal option, if it were known

$$\mu^* = \max_k \{\mu_k\}, \quad (2.11)$$

$$\rho := T\mu^* - \sum_{t=1}^T \hat{r}_t. \quad (2.12)$$

A strategy, or policy, is termed to have *zero-regret* when, as the number of rounds played tends to infinity $T \rightarrow \infty$, the average regret per round tends to zero $\rho/T \rightarrow 0$.

Such an outcome demonstrates that a given strategy eventually converges to the optimal action/arm in the case case of stationary reward distributions.

2.3 Convolutional Neural Networks

A Convolutional Neural Network (CNN), or ConvNet, is a class of deep neural network, or feedforward neural network, within the field of deep learning, used extensively in recent times in the analysis of visual information. Deep neural networks attempt to represent complex functions by feeding input variables through a set of weights/activations to predict an output, being distinct from regular neural networks in that they may contain “hidden” layers, between the input and output layer, as seen in Figure 2.1a. Work in the field of physiology found neurons in the the visual cortex of cats and monkeys would respond to small regions of the receptive field, effectively biological convolution [57]. This concept was adopted in the development of the “neocognitron”, which introduced convolutional and downsampling layers to the multi-layered neural network paradigm [43]. Convolution allowed the capture of features in “shift invariant” fashion, as the weights of a kernel are shared across an entire layer, as seen in Figure 2.1b.

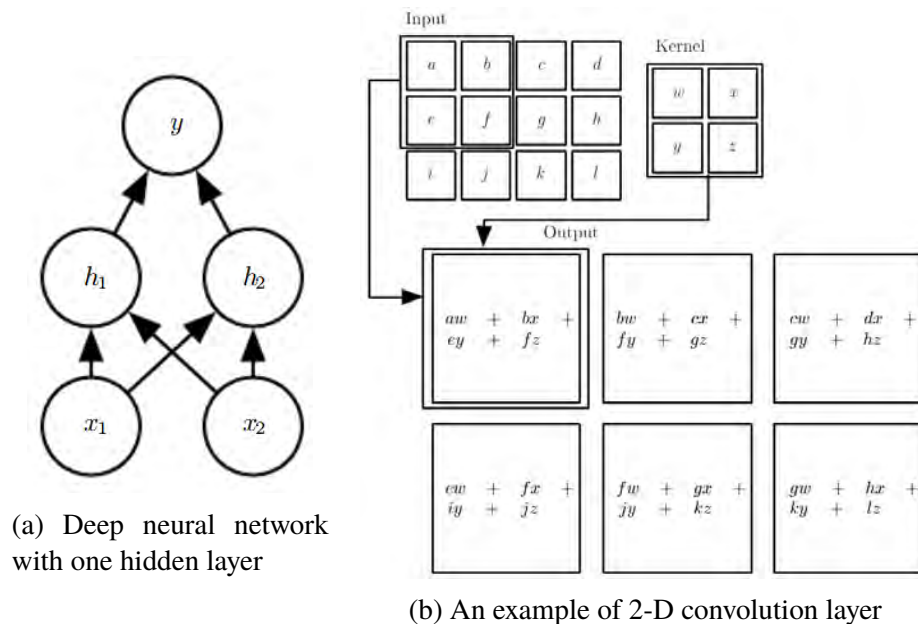


Figure 2.1: Example deep neural network architectures [47]

Many other varieties of network-architectures and processing functions have since been introduced, including rectification (such as Rectified Linear Units), average/-max pooling layers, fully connected layers, dropout layers, which have contributed

to network's ability to accurately classify spatial information across a range of modalities. The reader is directed towards [47] for further detail.

2.4 Gaussian Process Implicit Surfaces

A Gaussian Process is a collection of random variables indexed in either time or space (a stochastic process), such that every finite linear combination of those variables is Gaussian distributed, otherwise known as a multivariate normal distribution. For example, in the discrete time case, a stochastic process $X_t; t \in T$ is Gaussian if and only if for every finite set of indices $t_1, \dots, t_k \in T$ the vector $\mathbf{X}_{t_1 \dots t_k} = (X_{t_1}, \dots, X_{t_k})$ is a multivariate Gaussian random variable. This allows a GP to be considered a *distribution over functions*, which is fully specified by its mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$, constructed by [147] as

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})], \quad (2.13)$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[f(\mathbf{x} - m(\mathbf{x}))f(\mathbf{x}' - m(\mathbf{x}'))], \quad (2.14)$$

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \quad (2.15)$$

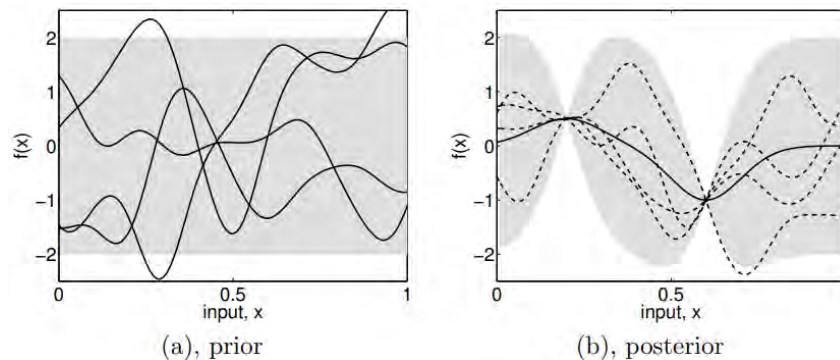


Figure 2.2: Left: Four 1D functions drawn from a prior Gaussian distribution, with the shaded region depicting twice the standard deviation of the distribution; Right: Posterior distribution after two data points are added, with dotted lines drawn from distribution about the mean solid line, with variance collapsed about the new points [147]

Gaussian processes are data-driven, and allow a user to select a covariance function that suitably represents the smoothness of the data (the correlation between proximal data points) for a given training dataset, then produce a posterior distribution over

the function's value after introduction of a new data point (such as depicted in Figure 2.2).

Drawing on the description from [24], given a set of training data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, a posterior distribution of the update function when a new data point \mathbf{x}_* is introduced is given by $f(\mathbf{x}_*) \sim \mathcal{N}(\bar{f}_*, \Sigma_*)$ where

$$\bar{f}_* = \mathbf{k}_*^T (K + \sigma_n^2 I)^{-1} \mathbf{y}, \quad (2.16)$$

$$\Sigma_* = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T (K + \sigma_n^2 I)^{-1} \mathbf{k}_*, \quad (2.17)$$

and $\mathbf{k}_* = \{k(\mathbf{x}_i, \mathbf{x}_*)\}_{i=1}^n$, $K \in \mathbb{R}^{n \times n}$ whose i^{th} column is \mathbf{x}_i , $\mathbf{y} = \{y_i\}_{i=1}^n$, and σ_n^2 is the variance of the surface measurement [147].

A stochastic process, Gaussian or otherwise, may be generalized to higher dimensions to allow representation of variables across multivariate spatial domains, such as across a surface or through a volume. One example of this employed in this thesis is that of a Gaussian Process Implicit Surface, which uses a potential field across \mathbb{R}^{n+1} to represent a manifold in \mathbb{R}^n , as suggested by [36]. Positive values within the field represent the exterior of the manifold, and negative values the interior, allowing the 0-level set of the potential function to implicitly describe the manifold, as seen in Figure 2.3. A further use of this potential field is that the positive gradient at the 0-level set, $\nabla f(\mathbf{x})$, predicts the normal of the manifold, as employed in Chapter 5.

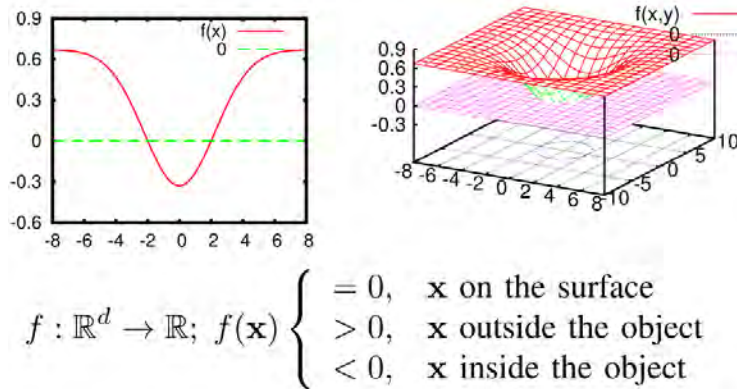


Figure 2.3: Above left: Implicit representation of 1D object in $[-2, 2]$; Above right: 2D object described by the blue ellipse in the center of the plane; Below: Gaussian Process Implicit Surface potential field definition [36]

Chapter 3

REACTIVE DISCRETE OPERATING MODE SELECTION

Introduction

One of the key challenges that complicates deployment of semi-autonomous rovers to explore other bodies in the solar system is the lack of *a priori* knowledge on the conditions within which they will be required to operate. A prime example comes from the icy Saturnian moon Enceladus, which has been found to harbor scientifically significant conditions potentially conducive to life [93], though it has only been observed through long distance imaging from flyby missions such as Cassini [22]. Visible wavelength imagery from these flybys attain resolutions of between 3m and 14km per pixel [115], while other sensing modalities such as the Visible and Infrared Mapping Spectrometer can be in the order of kilometers [22].



Figure 3.1: Left: JPL’s RoboSimian platform traversing rough Europa analog terrain in Death Valley, CA [111]. The platform possesses a total of 32 Degrees of Freedom (DoFs), spread across four articulated limbs with wheels at the distal joint. Right: JPL’s SURROGATE platform executing a “sweeping” excavation mode on regolith simulatant with a multi-tool mounted on 7 DoF limb.

This creates a significant challenge in characterizing the surface conditions that a robotic mission may encounter, both in topography and terramechanics. One approach to address the uncertainty is to equip platforms with redundant means of operation so as to react to conditions on the surface, such as is presently being developed on the Jet Propulsion Laboratory’s RoboSimian platform seen in Figure 3.1 left, such as driving, inchworming, and sculling. At the same time, the round trip

communication delay over such great distances, coupled with exceptionally low data rates, mean that Earth based operators can only be supplied with minimal data and imagery from the platform. Remote operators may also only provide sporadic input to surface operations. With a severely limited mission lifetime due to likely reliance on battery power, decisions must be taken on-board about which of the redundant operating modes should be applied, while attempting to adhere to an operator's most recent command.

To address this particular operational challenge and others like it, this paper presents a framework for a platform to execute tasks within what we term an "Obedient Multi-Armed Bandit" (OMAB) problem, where an agent must attempt to select between a set of available actions only when performance for a directed behavior is off-nominal. This approach allows the remote agent to conform to an operator command as long as a reward or cost signal is within expected bounds from *a priori* knowledge. The prior can be derived from empirical characterization or theoretical analysis. When unexpected deviations occur, the agent can explore alternative modes. In our two icy world exploration examples, this takes the form of 1) selecting between mobility modes for surface traversal with a reward signal of distance over energy consumed, and 2) selecting between digging modes for excavation with a reward signal of volume removed over energy consumed.

A further complication in these applications is that the conditions experienced by the platform may change both spatially, as it traverses across varying terrain types, and temporally, as temperature change or mechanical wear alter the interaction with the environment. In the MAB literature this is referred to as "non-stationarity", where the underlying reward distributions associated with each action are not fixed.

The formulation is developed so as to be consistent with JPL space-flight method of operations, where command sequences are up-linked to a platform on a per-sol basis, whereupon execution takes place over the course of the sol with result of actions returned for analysis before the next command cycle. In the advent of an error or warning condition, the platform will cease operation and wait for user input, to minimize risk of damage to the system. This slow command and response cycle is facilitated by presence of a consistent and long-lived power source for such missions, typically sun-facing solar arrays or a Radioisotope Thermoelectric Generator (RTG). Missions too distant in the solar system to derive sufficient solar power, and too mass and budget limited to contain an RTG, must operate on a more aggressive schedule to achieve adequate scientific yield, and be capable of making

some decisions on-board that are typically deferred to operators based, at times, over a billion kilometers away. The severely limited communication rates achievable at such distances also means that the quantity of data and imagery that can be sent back to operators for the purpose of decision making is minimal, further highlighting the need for on-board deliberation. Despite working with limited information, operators may still attempt to remotely deduce the most suitable operating mode for the system, due to either predicted efficacy or external factors such as scientific yield or tool wear, and hence would command the platform to employ the chosen mode, but to deviate intelligently if it proves ineffective. As off-nominal efficiency indicates the agents interaction with the environment is outside conditions characterized as ideal, it may suggest that undue wear or damage could be accumulating due to a modes continued use. This motivates a desire, when the operators choice is off-nominal, to employ the most efficient mode available that performs within some tolerance of its nominal efficiency.

The remote planetary platform itself will have a much more complete set of sensory data available to it than its remote operators, since the operational data sent back to Earth is limited. Even a rich set of onboard sensory information may still not be sufficient to accurately predict the efficacy of each available operating mode in a given set of environmental conditions. A similar problem has been identified in Martian surface operations, where wheel slip on soft substrate can lead to inefficient traversal, increasing the risk of becoming irrecoverably stuck [15, 80]. Some prior work at JPL attempted to visually predict regions of slip on Martian surface analogs [92]. However, while some success at classifying terrain types and risk of slip at a distance was demonstrated, the results utilized significant prior knowledge of the terrain substrates that might be encountered. Such prior knowledge would be absent in excursions to less studied bodies such as Enceladus. For this reason, the approach to the problem proposed in this chapter is posed as a reactive mode selection method, rather than a deterministic and predictive concept.

Relation to Prior Work

Multi-Armed Bandit theory is a subfield of the domain of Reinforcement Learning (RL), within which an agent attempts to optimize a feedback policy through the use of online measurements that result from its actions. RL may be broadly categorized by the observability of problem states, being fully or partially observable, or alternatively unobservable. Fully or partially observable state space methods may be further subdivided by their temporal representation, being either discrete,

continuous, or tabular [128]. A visual representation of this categorization may be seen in Figure 3.2, demonstrating the position of Obedient Multi-Armed Bandit theory within it. One of the key representations used in RL is that of the Markov Decision Process (MDP), first adopted for control of discrete stochastic processes by Bellman [11]. MAB theory may be considered a single-state MDP where each action is a separate state transition back to the original state.

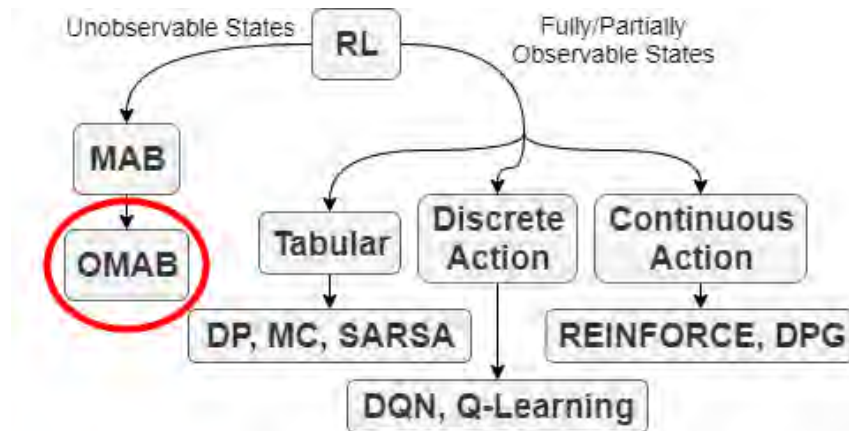


Figure 3.2: Overview of categories of Reinforcement Learning techniques. The approach proposed in this chapter targets a new variant of the Multi-Armed Bandits (MAB) formulation, which addresses problems where a limited set of resources must be allocated between mutually exclusive choices so as to maximize cumulative reward.

While fully and partially observable state problems have been addressed with tabular techniques (e.g. SARSA [118]), discrete techniques (e.g. Q-Learning [96]), and continuous action learning (e.g. DPG [125]), the problem addressed by Multi-Armed Bandit theory involves *unobservable* states, where the underlying reward distributions are usually unknown *a priori*. One primary means of formulation for unobservable states, an extension of the MDP, is that of the Hidden Markov Model (HMM) [8], early applications of which included speech recognition [61], and analysis of sections of biological DNA sequences [16]. These problems focus on determining the hidden states, however, as compared to MAB where the ultimate goal of the agent is to optimize the cumulative reward signal.

Early work on the prototypical form of the MAB problem focused on strategies for an optimal rate of convergence to the action with the highest mean outcome [75], with approximate solutions being developed such as the ϵ -greedy algorithm, where a scalar parameter determines a static rate of exploration [142], along with other variants [23, 27, 132]. A particularly relevant body of work is that of Contextual

Multi-Armed Bandits, that introduce an observable state outside of the reward distribution that can be used to predict outcomes, such as proposed by Wang et al. [141]. Pandey et al. investigated the contextual dependency between arms of the “bandit”, and presented an optimal MDP-based policy in the discounted reward case [102].

The key distinction between the above works and the OMAB approach is that, rather than focusing singularly on the optimization of the reward signal, the agent accommodates preference for an operator specified mode (i.e. *is obedient*), and only deviates when the performance of that mode is from nominal expected performance. To achieve this, it also leverages *a priori* knowledge of the expected behavior of each mode, so as to determine what is off-nominal behavior, while also informing the choice of next most suitable selection.

A common assumption among all of the above papers is that the underlying reward distribution for each mode is stationary, which limits application to dynamic environments where the true action-value may shift temporally or spatially. Some of the earliest work addressing this came from Kocsis and Szepesvári [67], who investigated the behavior of an extension to the Upper Confidence Bound policy (UCB) [1], called UCB1 [6], when payoff sequences drift. Garivier and Moulines then studied non-stationary payoffs for discounted and sliding-window UCB, establishing upper bounds on the expected regret via the number of times a suboptimal arm is played [44]. Perhaps closest to this work is that of Burtini et al. who address the problem of selecting between advertising strategies with the non-stationary distributions of consumer sentiment, while accounting for contextual information in the form of demographic and other information [26]. This work, by contrast, addresses non-stationary deviation from nominal operation of previously characterized actions, while also being restructured to facilitate specified mode preference.

Structure of the Chapter

Section 3.1 describes the OMAB algorithm and how it relates to Multi-Armed Bandit theory, by presenting the distinctions between them in the stationary distribution case, along with a means of quantifying policy success named “disappointment”. Section 3.2 then discusses how existing MAB policies for stationary distributions may transfer to the OMAB formulations, and derives disappointment results for these policies in the stationary case. Section 3.3 presents a new policy, termed Preferential Iterative Update (PIU), that uses exponential decay of past action-values

to address non-stationary distributions, to which Section 3.4 proposes an extension that allows an external signal to inform the decay rate so as to more rapidly respond to perceptible environmental condition changes. Section 3.6 then demonstrates an application of the PIU policy to the problem of mobility mode selection for planetary exploration, first through short distance laboratory experiments and then longer traverse simulations leveraging empirical data. Finally, Section 3.7 demonstrates the improved behavior afforded by exteroceptive action-value decay, though experimental application to multi-modal surface excavation on the SURROGATE platform, in the case of substrate step change.

3.1 The Obedient Multi-Armed Bandit Problem

The OMAB problem seeks to select between a discrete set of actions, or operating modes, chosen at discrete time steps, attempting to adhere to an operator specified preference except when the cost for that action is off-nominal, in which case alternatives are explored using *a priori* knowledge of their nominal cost. The distinction between MAB and OMAB can be seen visually in Figure 3.3 for both stationary and non-stationary distribution cases, where the additional information of nominal cost per action, and mode preference, inform the selection of subsequent actions.

Stationary Formulation

In the prototypical MAB, the “arms” of the problem are represented as $n \in \mathbb{N}$ actions available to the agent at every discrete time step, $t \in \{\mathbb{N} \mid t < T\}$, where $T \in \{\mathbb{N}, \infty\}$ is the terminal time of the analysis horizon. At each time step the agent selects an index representing one of n actions/modes, $a(t) \in \mathcal{A} := \{1, \dots, n\}$, and executes the corresponding action, for which the environment returns a scalar reward, $r(t)$. Each action has an associated stationary reward distribution, χ_i , and mean, $\mu_i = \mathbb{E}[\chi_i] \forall i \in \mathcal{A}$, that is unknown to the agent. The reward for a given time step is drawn from the distribution of the action that was taken in that timestep, $r(t) \sim X_{a(t)}$. The goal of MAB theory is to design a policy, denoted π , that will maximize the future cumulative reward, $R(t)$, by selecting amongst the available actions over the remaining time horizon. This may include a discount factor, $\gamma \in [0, 1]$, that prioritizes immediate over future rewards, defined as

$$R(t) := \sum_{s=t}^T \gamma^{s-t} r(s) . \quad (3.1)$$

The expected value of each action is often tracked using the notion of an ‘action-

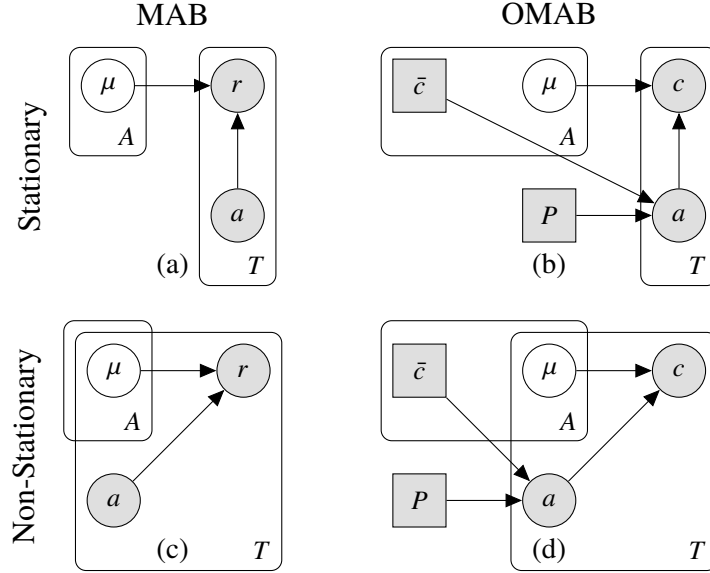


Figure 3.3: Probabilistic graphical models of MAB and OMAB formulations for both stationary and non-stationary distributions.

A : Action space, T : Time steps, μ : Underlying reward/cost distribution, a : Action taken, r : Measured reward, P : Preferred mode, \bar{c} : Nominal action cost, c : Measured cost.

Shaded/Unshaded = Observed/Latent variables, Circle/Square = Probabilistic/Prescribed parameters.

value function' that captures the anticipated cumulative reward for each action if it were selected for the remainder of the analysis horizon, given by

$$Q_i(t) = \mathbb{E}\{R(t) \mid a(s) = a_i \forall s \in \{t, \dots, T\}\}. \quad (3.2)$$

The Obedient Multi-Armed Bandit extends this classical framework in two ways. First, it reformulates the objective from a *reward* to a deviation from nominal *cost*. This facilitates the process of determining when the operator's commanded mode performance is off-nominal, thus necessitating exploration of alternative modes. Cost may represent the efficiency of achieving the given task, such as *Joules/meter* of vehicle traversal, or *Joules/cm³* of substrate excavated, and is denoted at each time step by $c(t)$. Second, OMAB adds two additional parameters to the formulation. The mode/action commanded by the operator is denoted $P \in \mathcal{A}$, and the *a priori* information on the nominal cost of each mode (characterized empirically under ideal conditions) is denoted \bar{c}_i , which represents the minimum cost for each action across conceivable environment states \mathcal{S} , i.e. $\bar{c}_i \leq \mu_i \forall \mathcal{S}$.

Quantifying Obedience

The efficacy of any given policy applied to a MAB problem is typically parameterized by the notion of “regret”. Regret represents the difference between the realized cumulative reward, and that which could have been achieved by sampling from the action with the true highest mean ($\mu^* := \max_{i \in \mathcal{A}} \mu_i$) over the entire time horizon, as defined by

$$\rho = T\mu^* - \sum_{t=1}^T r(t). \quad (3.3)$$

MAB policies seek to minimize regret through a combination of exploration and exploitation of the reward distributions. By contrast, the Obedient Multi-Armed Bandit does not target optimization of the reward, but rather attempts to select modes with minimal deviation from nominal operating performance, with a preference for the operator’s specified operating mode. For this purpose we define an analog to regret termed *disappointment* that parameterizes the priority of mode P through a scalar, $\lambda \in \mathbb{R}^+$, such that

$$D := \sum_{t=1}^T \Delta(t) - \lambda \mathbb{1}_{\{a(t)=P\}}. \quad (3.4)$$

where λ is chosen to represent the deviation from nominal performance, in the same units as cost, allowed of the commanded mode before others will be explored, and $\Delta(t)$ is the deviation from nominal cost of the action sampled at a given time instant,

$$\Delta(t) := c(t) - \bar{c}_{a(t)}, \quad (3.5)$$

From this we can see that, when the deviation from nominal cost of specified mode P is less than λ , disappointment will be negative (indicating operator *satisfaction*), but as soon as it exceeds this value, disappointment will grow positively. While $\Delta(t)|_{a(t)=P} > \lambda$, any other mode that performs nominally ($\Delta(t) = 0$) will result in lower disappointment than the operator specified mode. We also wish to prioritize use of modes with lower nominal cost when deviating from the operator specification, but this can be captured within the design of the exploration policy rather than through explicit inclusion in the obedience metric.

3.2 Policies for Stationary Distributions

One of the earliest and simplest policies for stationary distribution MAB is ε -greedy, which selects between 1) exploring any randomly chosen action, and 2) exploiting the action with the current highest action-value function, at a relative rate defined by a constant parameter ε . To implement this greedy behavior, the discount value for future cost is taken as $\gamma = 0$, and the action-value function becomes an average over all samples taken of an action prior to time t , defined as

$$N_i(t) = \sum_{s=1}^t \mathbb{1}_{\{a(s)=i\}} , \quad (3.6)$$

$$Q_i(t) = \frac{\sum_{s=1}^t r(s) \mathbb{1}_{\{a(s)=i\}}}{N_i(t)} . \quad (3.7)$$

The ε -greedy policy is then defined as in Equation 3.8, where $U(\mathcal{A})$ is a uniform distribution across all actions. The policy is initialized by sampling each of the available actions (playing each of the arms) precisely once. In the limit of time, after the action with the highest true mean reward, a^* , has been identified ($\arg \max_i Q_i(t) = a^*$) which, by the law of large numbers, will occur in the stationary distribution case, the regret of the ε -greedy policy continues to grow with $O(T)$, as demonstrated in Equation 3.9, where $\bar{\mu}_{i \in \mathcal{A}}$ is the mean of the true reward distribution means, π denotes a policy, and $\rho|_{\pi_{\varepsilon\text{-greedy}}}$ is the reward under the greedy policy.

$$\pi_{\varepsilon\text{-greedy}} : a(t+1) = \begin{cases} \arg \max_i Q_i(t) & \text{Pr} = 1 - \varepsilon \\ \sim U(\mathcal{A}) & \text{Pr} = \varepsilon \end{cases} , \quad (3.8)$$

$$\begin{aligned} \rho|_{\pi_{\varepsilon\text{-greedy}}} &\xrightarrow{T \rightarrow \infty} T\mu^* - (1 - \varepsilon)T\mu^* - \varepsilon T\bar{\mu}_{i \in \mathcal{A}} \\ &\xrightarrow{T \rightarrow \infty} \varepsilon T(\mu^* - \bar{\mu}_{i \in \mathcal{A}}) . \end{aligned} \quad (3.9)$$

A similar strategy may be employed within the Obedient Multi-Armed Bandit by employing an averaging action-value function over historical samples, as described in Equation 3.10. We then define the *active set* at time t , $I(t) \subseteq \mathcal{A}$ in Equation 3.11, which contains all actions of the present minimum action-value, and may frequently be of dimension greater than one given there may be multiple actions performing nominally at once. The ε -greedy policy for OMAB is then given by Equation 3.12,

where exploiting the minimum nominal cost over the active set of actions allows accommodation of the previously describe priority given to more nominally efficient modes during exploration. As the action-value is derived from the deviation from nominal cost, which is previously known, initialization consists of setting all actions values to zero save the preferred mode, as in Equation 3.13, and commencing the policy without having to sample every action first.

$$Q_i^{\text{OMAB}}(t) = \frac{\sum_{s=1}^t \Delta(s) \mathbb{1}_{\{a(s)=i\}}}{N_i(t)} - \lambda \mathbb{1}_{\{i=O\}}, \quad (3.10)$$

$$I(t) := \arg \min_{i \in \mathcal{A}} Q_i^{\text{OMAB}}(t), \quad (3.11)$$

$$\pi_{\varepsilon\text{-greedy}}^{\text{OMAB}} : a(t+1) = \begin{cases} \arg \min_{i \in I(t)} \bar{c}_i & \text{Pr} = 1 - \varepsilon \\ \sim U(\mathcal{A}) & \text{Pr} = \varepsilon \end{cases}, \quad (3.12)$$

$$Q_i(0) = -\lambda \mathbb{1}_{\{i=P\}} \quad \forall i \in \mathcal{A}. \quad (3.13)$$

With this policy the OMAB success metric, termed disappointment, similarly experiences an $O(T)$ trend in the limit of time, dependent on the difference between true and nominal mean for each action, $\Delta_i := \mu_i - \bar{c}_i \in \mathbb{R}^+$, and whether the preferred mode is within the minimum set of action-values, $a^* := \arg \min_i [\Delta_i - \lambda \mathbb{1}_{\{i=P\}}]$, as shown by

$$\begin{aligned} D |_{\pi_{\varepsilon\text{-greedy}}^{\text{OMAB}}} &\xrightarrow{T \rightarrow \infty} (1 - \varepsilon)T [\Delta_{a^*} - \lambda \mathbb{1}_{\{a^*=P\}}] + \varepsilon T \left[\bar{\Delta}_{i \in \mathcal{A} \setminus a^*} - \frac{\lambda}{n-1} \mathbb{1}_{\{a^* \neq P\}} \right] \\ &\xrightarrow{T \rightarrow \infty} \begin{cases} (1 - \varepsilon)T(\Delta_P - \lambda) + \varepsilon T \bar{\Delta}_{i \in \mathcal{A}} & a^* = P \\ (1 - \varepsilon)T \Delta_{a^*} + \varepsilon T (\bar{\Delta}_{i \in \mathcal{A}} - \frac{\lambda}{n-1}) & a^* \neq P \end{cases}. \end{aligned} \quad (3.14)$$

The key distinction is that the gradient of this trend may be positive or negative, due to the possibility that the prescribed action returns a negative disappointment value. As expected in the obedient case ($a^* = P$), the gradient of the trend (indicating growing disappointment/satisfaction) will be primarily dependent (given typical values of $\varepsilon \ll 1$) on the cost deviation of the preferred mode, Δ_P , relative to the selected λ parameter. In the disobedient case, even if there exists an alternative mode

that behaves close to nominal ($\Delta_{a^*} \rightarrow 0$), the continued sampling of alternatives modes at rate ε results in linear disappointment growth.

An improved ‘upper confidence bound’ regret strategy for the MAB stationary distribution case, named UCB1, was proposed by Auer et al. [6], where the exploration rate diminishes logarithmically with time, as given by

$$\pi_{\text{UCB1}}: a(t+1) = \arg \max_i \left[Q_i(t) + \sqrt{\frac{2 \ln t}{N_i(t)}} \right], \quad (3.15)$$

UCB1 is initialized in the same fashion as ε -greedy by playing all actions once prior to engaging the policy. Auer et al. show in their 2002 Theorem 1 that, in the limit of execution time for every suboptimal action, $j \in \mathcal{A} \setminus a^*$, the expectation of the number of times it is played has an upper bound of

$$\mathbb{E} [N_j(T)] \leq \frac{8 \ln T}{(\mu^* - \mu_j)^2} + 1 + \frac{\pi^2}{3}. \quad (3.16)$$

When applied to the OMAB formulation, a similar upper bound on suboptimal action selections may be derived, reflecting the modified action-value function, resulting in expectation of number of plays for $j \in \mathcal{A} \setminus a^*$ given by

$$\pi_{\text{UCB1}}^{\text{OMAB}}: a(t+1) = \arg \min_i \left[Q_i(t) - \sqrt{\frac{2 \ln t}{N_i(t)}} \right], \quad (3.17)$$

$$\mathbb{E} [N_j(T)] \leq \frac{8 \ln T}{(\Delta_{a^*} - \Delta_j - \lambda(-1)^{\mathbb{1}_{\{a^*=P\}}})^2} + 1 + \frac{\pi^2}{3}. \quad (3.18)$$

As with the ε -greedy OMAB policy, initialization is given by Equation 3.13, and no *initial* plays are necessary, which would be prohibitive in space applications. The policy produces an upper bound on disappointment that scales at $O(\ln T)$ for suboptimal actions, and $O(T)$ for the optimal action a^* , of

$$D|_{\pi_{\text{UCB1}}} \leq \left[\sum_{i: \mathcal{A} \setminus a^*} \frac{8 \ln T}{\Delta_{a^*} - \Delta_j - \lambda(-1)^{\mathbb{1}_{\{a^*=P\}}}} \right] + T (\Delta_{a^*} - \lambda \mathbb{1}_{\{a^*=P\}}) + \left(1 + \frac{\pi^2}{3} \right) \sum_{j \in \mathcal{A}} \Delta_{a^*} - \Delta_j - \lambda(-1)^{\mathbb{1}_{\{a^*=P\}}}. \quad (3.19)$$

The continued $O(T)$ is a result of disappointment being formulated around minimum deviation from nominal cost, rather than difference from maximally achievable reward. There will always be a linear component in disappointment that depends on that minimum deviation, as well as the $-\lambda$ component when the optimal mode is also that which was commanded.

Non-Stationary Formulation

Lesser studied within the field of MAB, but of higher relevance to the task of navigating and sampling from poorly characterized environments, is the problem of selecting actions with non-stationary underlying distributions. Non-stationarity may derive from both temporal and spatial changes in the interaction between the system and the environment. Within the context of icy planet exploration, temporal changes may reflect shifting surface temperatures as a function of solar illumination that modify terramechanics underfoot, and spatial changes may occur as a system transitions between two varieties of substrate while traversing, or in depth during excavation.

One further key practical consideration is the degradation of the tool surfaces used to affect each of the modes, which may contribute a monotonic component to the cost distributions. As a result, the underlying distributions governing the rewards garnered through execution of the available actions becomes a function of time, or the number of samples taken, as depicted within the graphical models of MAB and OMAB in the bottom of Figure 3.3.

3.3 Preferential Incremental Update Policy for Non-Stationary Case

To facilitate tracking of both spatially and temporally variable cost distributions, I propose to adapt an incremental update rule for the action-value formulation detailed in Sutton and Barto 2011 [128], which is a form of exponential smoothing filter. The key distinction in the update policy proposed here is that the action-value functions of *all* actions is updated at every time step, as compared to only the action that was sampled at a current time step. The update value, however, *is* dependent on whether or not the given action was executed on the last time step, as defined by $q_i(t)$ in Equation 3.20. The action-value equation then takes the form seen in Equation 3.21, where rate of tracking in the incremental update step is dictated by a user specified parameter, $\alpha \in (0, 1)$, and is applied to the action-value of every action at every time step.

$$q_i(t) := \begin{cases} \Delta(t) - \lambda \mathbb{1}_{\{i=P\}} & a(t) = i \\ -\lambda \mathbb{1}_{\{i=P\}} & a(t) \neq i \end{cases}, \quad (3.20)$$

$$Q_i^{\text{PIU}}(t+1) = Q_i^{\text{PIU}}(t) + \alpha [q_i(t) - Q_i^{\text{PIU}}(t)] \quad \forall i \in \mathcal{A}. \quad (3.21)$$

As with the application of the ε -greedy to the OMAB problem, the policy selects from the *active set* of actions at each time step (those that minimize the action-value), and executes the action with the lowest nominal cost, via

$$I(t) := \arg \min_{i \in \mathcal{A}} Q_i^{\text{PIU}}(t), \quad (3.22)$$

$$\pi_{\text{PIU}} : a(t+1) = \arg \min_{i \in I(t)} \bar{c}_i. \quad (3.23)$$

Due to the presence of an operator preference and the use of an action-value incremental update, this procedure is termed a *Preferential Incremental Update* (PIU) policy, and is described below in full as Algorithm 1. The remainder of the chapter demonstrates the application of PIU to the problems of multi-modal robotic surface traversal, and robotic sample excavation.

3.4 Exteroceptively Informed Action-Value Decay

The baseline PIU algorithm is purely reactive to the instantaneous cost signal received for executing a given action, but the ability of the system to react to changing underlying cost distributions may be accelerated if there are external stimuli correlated with that change. Examples may include any means of exteroception, such as visual and other wavelength light, or proprioceptive, such as tactile interrogation of the terramechanic properties of the environment.

These visual signatures can be leveraged by making the incremental update rate, α , a function of the rate of change of this external signal. Let some external signal that is loosely correlated with environmental changes be denoted $\phi(t)$, which is measured at each time t . We then redefine the α decay factor as a positive semi-definite difference function of the change of this signal between timesteps, $\phi_{\Delta}(t) : \Phi \rightarrow \mathbb{R}$, for example a norm $\phi_{\Delta}(t) := \|\phi(t) - \phi(t-1)\|$, and bounding it at chosen minimum value ψ below which the original decay α applies as in Equation 3.24.

Algorithm 1: OMAB PIU policy. $t \in \mathbb{N}$ represents a temporal index, ω represents a scalar measurement of progress within a given autonomous task undertaken by an agent, and Ω represents value of ω necessary to declare task complete. \mathcal{A} represents a set of actions available to the agent, which executes $a(t)$ at each temporal index t , incurring a visible cost $c(t)$ drawn from a hidden distribution for the selected action, while advancing ω . Agent has *a priori* knowledge of the minimum possible cost associated with each action, $\bar{c}_i \forall i \in \mathcal{A}$, and is directed to prefer action P up to the cost bound of preference parameter λ .

Initialization

- Empirically determine $\bar{c}_i \forall i \in \mathcal{A}$ before deployment
- Set progress $\omega = 0$, goal Ω , and decay factor α
- Initialize $t = 0$, $a(0) = P$, the operator preference
- Initialize action-value functions $Q_i(0) = -\lambda \mathbb{1}_{\{i=P\}}$

Policy

while $\omega < \Omega$ **do**

- Take action $a(t)$, measure goal progress increment ξ and incurred cost $c(t)$.
- $q_i(t) = (c(t) - \bar{c}_i) \mathbb{1}_{\{i=a(t)\}} - \lambda \mathbb{1}_{\{i=P\}}$
- $Q_i(t+1) = Q_i(t) + \alpha [q_i(t) - Q_i(t)]$
- $I(t+1) = \arg \min_{i \in \mathcal{A}} Q_i(t)$
- $a(t+1) = \arg \min_{i \in I(t+1)} \bar{c}_i$
- $t \leftarrow t + 1$
- $\omega \leftarrow \omega + \xi$

end

$$\alpha^\phi(t) := \begin{cases} \alpha & \phi_\Delta(t) \leq \psi \\ \alpha \phi_\Delta(t) & \phi_\Delta(t) > \psi \end{cases}. \quad (3.24)$$

The action-value of the PIU policy is be updated to reflect this decay-rate to produce a *PIU-External Decay* (PIU-ED) policy, utilizing the same active-set and action update policy from Equations 3.22 and 3.23, as given by

$$Q_i^{\text{PIU-ED}}(t+1) = Q_i^{\text{PIU-ED}}(t) + \alpha^\phi(t) [q_i(t) - Q_i^{\text{PIU-ED}}(t)] \quad \forall i \in \mathcal{A}. \quad (3.25)$$

3.5 PIU Parameter Selection

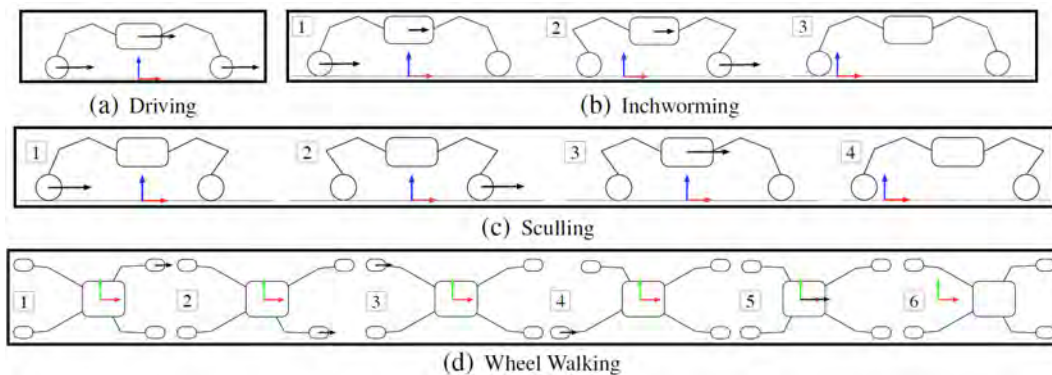
Of primary import in the application of the PIU policy is the selection of the λ parameter, which we recall is in the units of the cost for each available action. This value encapsulates the strength of preference of the operator (either human or higher level planning in an autonomous system) for the directed mode, given it determines the deviation from the nominal performance of the mode that must occur before other alternatives are explored. A suitable guide on the magnitude of the λ value may be taken from the difference between nominal costs of the various modes. If the preference of the operator is weak, where perhaps the directed mode was merely that assumed to operate closest to nominal, but more efficient means would be preferred; then the operator may select a small λ value relative to the difference between nominal cost of the preferred mode and the mode closest in value (10% of the delta). By contrast, if the operator has a strong preference for a mode, such as if it allows more fruitful collection of scientific data during icy body exploration, then the λ parameter might be set close to the difference between nominal cost of the preferred mode and the next closest (90% of the delta).

The α forgetting factor determines the rate at which the most recent measurements modify the expected value of a mode, captured as the action-value function. It should be determined as a function of the expected rate of change of environmental or self-state conditions, normalized against the execution duration of each mode selection window, while also accounting for likelihood of variance in the measurement at each timestep due to noise. As the action-value function acts as an exponential decay between the prior values and the most recent measurement, the rate of adaptation to new values may be captured with the *exponential half-life*, $\ln(2)/\alpha$. This represents the time at which the action-value will reach $1/2(C_1 + C_2)$, where C_1 and C_2 are an initial and final underlying cost. If the environmental conditions are only expected to change at a rate of 50% over 100 time steps, and may be subject to measurement noise on smaller timeframes, then a value of $\alpha = 100/\ln(2)$ will track the environmental changes at their expected rate of change, while smoothing shorter term fluctuations in measurements.

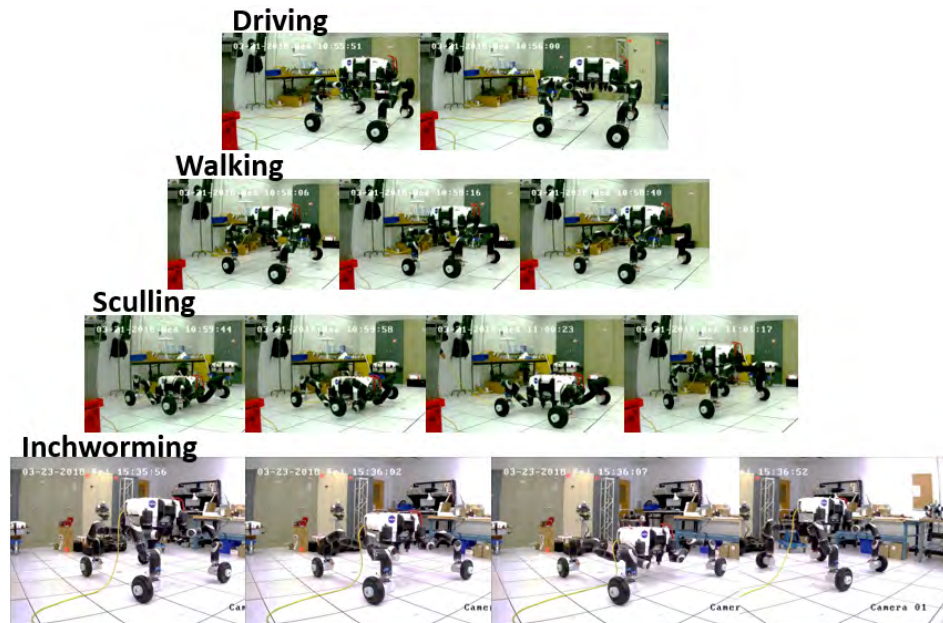
3.6 Mobility Application

The first example application of an OMAB policy is the selection of mobility modes during icy body surface traversal. Experiments were undertaken using the JPL RoboSimian platform pictured in Figure 3.1 left which, through use of wheels mounted to the distal joint of four 8 DoF articulated limbs, is capable of

the locomotion modes of driving, inchworming, sculling, and wheel walking, as depicted in Figure 3.4a.



(a) Illustration of the motion states with the wheel-on-leg platform



(b) Images of each motion state executed in lab environment

Figure 3.4: Four Mobility Modes of RoboSimian: Driving, Inch-worming, Sculling and Walking. A detailed description of modes is discussed in [111].

Experiments

The PIU policy was first evaluated through demonstration of its ability to deviate from the nominally optimal mobility mode of driving with the RoboSimian platform. This was conducted within a controlled laboratory environment as pictured in Figure 3.5 left, where a tether was employed to impart artificial slip to the platform after executions of the preferred and nominally optimal driving. Figure 3.5 right top depicts the nominal costs of each mode, represented by the kJ/m gradient of

the graph, where distance is the goal/ ω attribute being accumulated. It also displays the cost profile for a nominal driving action in red, and a subsequent driving action with artificial slip in blue. Figure 3.5 right illustrates that the PIU policy employs the preferred driving mode for two action cycles (corresponding to the cost profiles depicted above), then switches to the next most nominally efficient mode, inchworming, for one cycle, before returning to a driving mode. This rapid exploration and mode transition occurs due to a high α value being employed within the PIU policy, given the need to demonstrate transitions within the short traverse distance available in the lab environment.

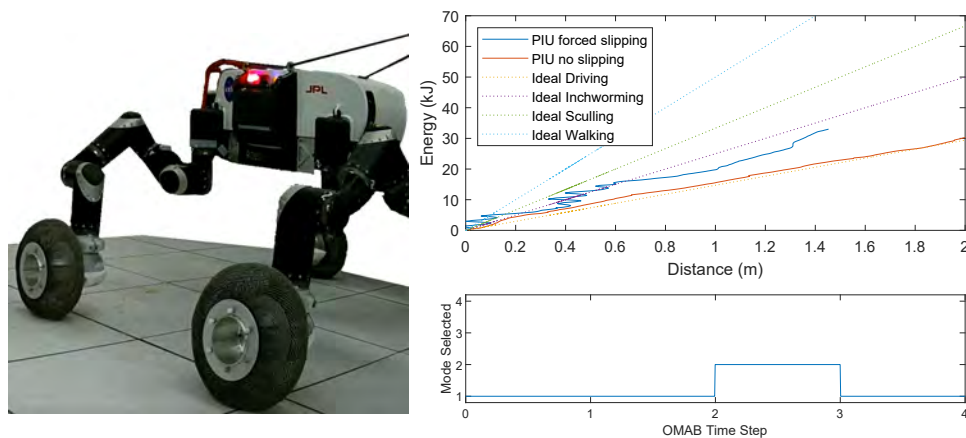


Figure 3.5: Left: RoboSimian platform in a lab environment restrained by tethers to impart artificial slip while testing the PIU algorithm. Right Top: The kJ/m cost utilized by the policy when applied to mobility mode selection is represented by the gradient of the distance/energy plot. Solid red and blue lines show cost measurement across a single nominal and off-nominal (slip induced) driving action respectively, while dotted lines represent the ideal cost gradient for the four available actions. Right Bottom: Modes employed by PIU policy over four time steps in lab environment with driving as the preferred mode. Steps 1 & 2 correspond to the nominal and off-nominal driving cost measurements above respectively. An off-nominal cost prompts the PIU policy to explore the next most nominally efficient mode of inchworming. [1=driving, 2=inchworming, 3=sculling, 4=wheel walking]

Long Traverse Simulation

Since to the maximum experimental traverse distance that could be realized was limited by the length of the power tether, experimental values for ideal and off-nominal performance in each of the modes [111, 112] were used to evaluate the behavior of the policy in a MATLAB simulation of a long distance traverse. Comparisons were also made relative to the existing policies of ϵ -greedy and UCB. The simulated scenario consisted of an icy body surface mobility platform which in practice would

likely experience sudden changes in the underlying surface and ice distributions due to transition between substrate types, or slow degradation of efficacies due to temporal environment change or tool wear. Nominal costs for driving, inchworming, sculling, and wheel walking were 15, 25, 32.5, and 50 kJ/m respectively, as measured experimentally. Policy parameters were set to standard values of $\varepsilon = 0.1$, $\alpha = 0.1$, $\lambda = 7 kJ/m$, and the preferred mode specified as driving. Results displayed in Figure 3.6.

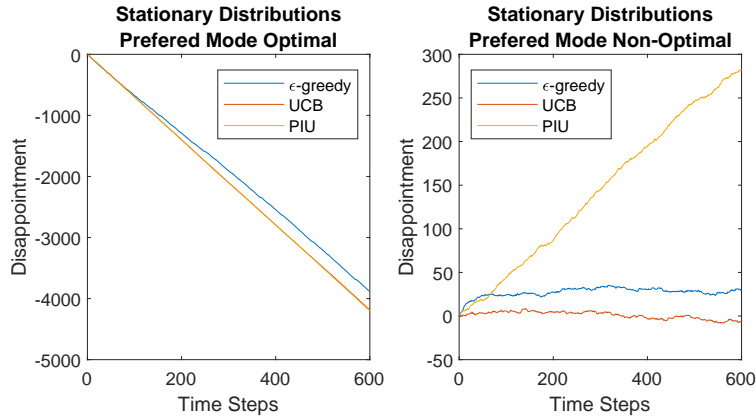
Figure 3.6 top demonstrates disappointment metric trends when the cost distributions for each action are fixed/stationary. The left plot represents underlying distributions where the preferred mode is behaving nominally, and illustrates that PIU and UCB achieve negative disappointment superior to ε -greedy when the preferred mode is also optimal, due to both predominantly sampling the optimal mode while the ε -greedy algorithm continues to explore at a linear rate. The right plot highlights the primary inferior case for PIU performance; when cost distributions are stationary and the preferred mode is suboptimal, the α decay of the action-values means the preferred mode continues to be sampled frequently (the agent is *optimistic*), unlike ε -greedy or UCB which predominantly exploit the optimal action. Note, however, the shallow gradient of the rising disappointment relative to that of the comparison policies under other scenarios.

Figure 3.6 center depicts simulation results using the same parameters and cost distributions as above, but with a step change in conditions. The preferred mode transitions from being optimal to non-optimal in the left plot, and from preferred mode optimal to non-optimal on the right. The incremental update and forgetting property of PIU allows it to adapt more rapidly than the linear exploration of ε -greedy, and logarithmic decaying exploration of UCB in both these cases.

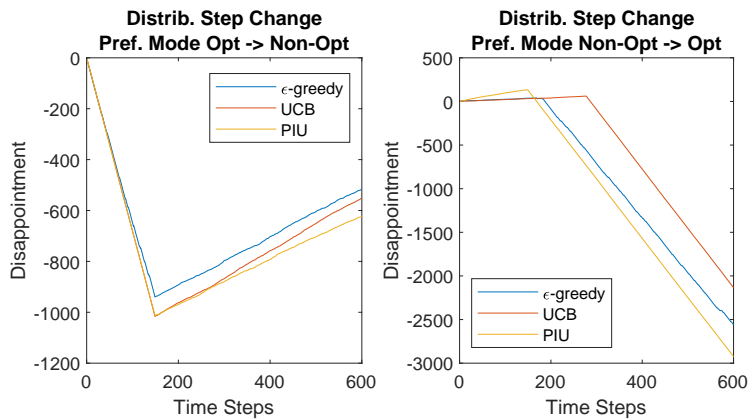
Figure 3.6 bottom again shows a simulation result using the same parameters and initial distributions over actions and preference. However, the preferred mode linearly drifts from optimal to non-optimal on the left, and optimal to non-optimal on the right, between time steps $t=150$ and $t=300$. Once again, the forgetting feature of the incremental update allows the PIU policy to adjust its action selection earlier than the alternative policies, intended as they are for stationary distributions.

3.7 Excavation Application

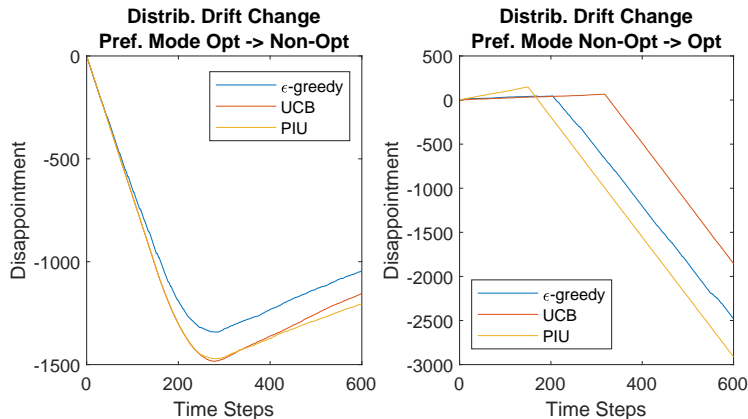
The second example application of OMAB policy is motivated by the important space robotics application of substrate excavation. Experiments were carried out



(a) Left: Stationary cost distributions when the preferred mode is optimal, resulting in negative disappointment (satisfaction) being accumulated. Right: Stationary cost distributions when preferred mode is non-optimal, highlighting the inferior case for PIU where it continues to optimistically sample the preferred mode, slowly accruing disappointment.



(b) Left: Cost distributions with step change at $t=150$ transitioning preferred mode from optimal to non-optimal. Right: Cost distribution with step change at $t=150$ from preferred optimal to preferred non-optimal.



(c) Left: Cost distributions with preferred mode drift from optimal to non-optimal between $t=150,300$. Right: Cost distributions with preferred mode drift from non-optimal to optimal between $t=150,300$.

Figure 3.6: Comparison of “disappointment” metric trends across policies over MATLAB simulated long mobility runs (where lower is better), utilizing empirical measurements of nominal and off-nominal mode efficiencies. $\varepsilon = 0.1$, $\alpha = 0.1$, $\lambda = 7 \text{ kJ/m}$

with the JPL platform SURROGATE (pictured in 3.1 right), which is equipped with a “multi-tool” capable of executing three different excavation modes, as pictured in Figure 3.7. These modes include: 1) sweeping, with a static 20cm wide flat edge profile; 2) raking, with an 8cm wide rounded finned edge that oscillates during progression; and 3) biting, that employs a pair of steel teeth pressed down against a substrate to fracture and dislodge material via a lateral shearing force. The goal (ω) within the excavation task is volumetric removal of the substrate, in cm^3 , as measured by the difference in profile swept by successive excavation passes. When combined with the accumulated energy expended during execution, estimated using the method documented in Appendix A.1, this produces a cost metric of kJ/cm^3 .



Figure 3.7: Three substrate excavation modes available on the SURROGATE multi-tool platform. Left: Sweeping mode with 20cm wide tool. Center: Raking mode with 8cm wide oscillating tool. Right: “Chomping” mode with two steel teeth engaging surface 8cm apart.

Substrate Step Change with Exteroceptive Feedback

The characteristic forgetting factor in PIU-ED to accelerate adaptation to perceptible changes in environment was evaluated by introducing a step change in substrate mechanics. The exteroceptive signal was taken as a vector in $\bar{\mathbf{p}}(t) \in \mathbb{R}^3$ representing the mean RGB values of pixels, $\mathbf{p}_i \in \mathbb{N}^3$, within a specific region of interest (ROI) of the platform’s sensor head camera field of view, as defined in Equation 3.26, where w and h are the width and height of the region. The difference function, $\phi_{\Delta}(t)$, employed by Equation 3.24 is then defined as the L2 norm between mean ROI pixel values of successive images, captured at the beginning of each time step when the policy calculates the next action, as per

$$\bar{\mathbf{p}}(t) := \frac{1}{wh} \sum_{i \in ROI} \mathbf{p}_i(t), \quad (3.26)$$

$$\phi_{\Delta}(t) := \|\bar{\mathbf{p}}(t) - \bar{\mathbf{p}}(t-1)\|_2. \quad (3.27)$$

While more advanced comparative methods could be employed in the presence of subtler differences in environmental condition changes, the mean pixel norm proved sufficient for the exaggerated substrate changes adopted in the experimental setup, given the primary aim of exercising the policy. Even in the presence of minor shadows, the mean pixel difference function value is minimal while the substrate remains the same. The difference function proved suitably invariant to shadows introduced in the course of action execution within the same substrate, as seen in Figure 3.8 left and right where the mean pixel values are (186.3, 175.0, 130.7) and (182.3, 170.8, 126.9) respectively, resulting in an L2 norm of 6.9.

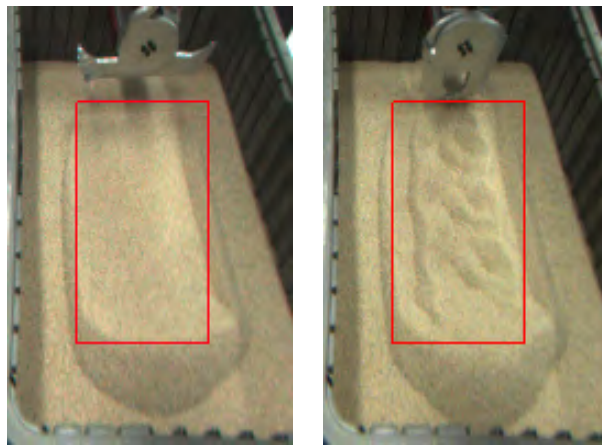


Figure 3.8: Left: Sensor head camera view of excavation area after sweep action with ROI used for determination of exteroceptive condition change highlighted. Right: ROI after completion of a raking action.

A step change in the experimental substrate mechanics was introduced through the addition of a 5mm thick foam board (within a retaining frame) atop the homogeneous medium grit construction sand (seen in Figure 3.8). Nominal efficiencies of the sweeping, raking, and chomping modes within the homogenous subsurface material was empirically determined *a priori* as 0.025, 0.05, and 0.1 kJ/cm^3 respectively, and the operator action preference is specified as the median efficiency mode of raking, with a cost bound preference of $\lambda = 0.05 \text{ kJ/cm}^3$, and decay factor of $\alpha = 0.1$. Figure 3.9 plots the calculated action-values of all actions at each time step, along with the action sequence chose by the PIU policy. Figure 3.10 then depicts the minimal progress made by the initially selected raking mode and sweeping mode actions, due to their inability to penetrate the tougher surface medium. Their resulting action values therefore increased, causing the lowest nominally efficient mode chomping to be selected for the third action. In this third mode the robot manages to pierce the surface material, resulting in closer to nominal efficiency, and producing a lower

action-value than the alternatives, causing it to be selected again for the fourth time step.

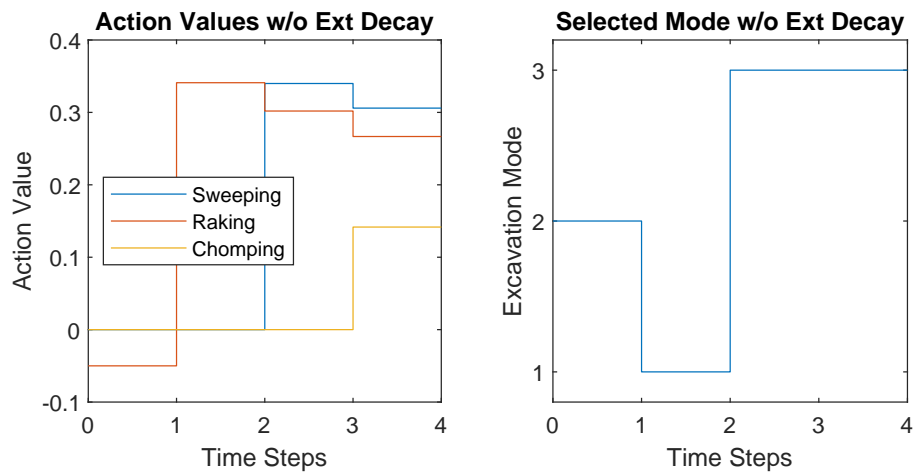


Figure 3.9: Left: Action-value functions for all actions across the timesteps of the excavation run, where $\lambda = 0.05kj/cm^3$ and $\alpha = 0.1$, demonstrating closer to nominal performance of chomping action causing it to be re-selected (due to lowest action-value), despite lower substrate now being exposed. Right: Action sequence selected by PIU policy in absence of exteroceptively informed decay. [1=sweeping, 2=raking, 3=chomping]

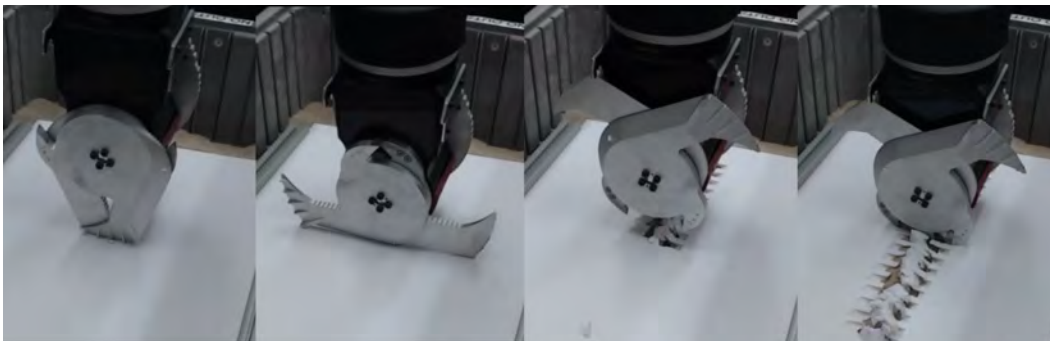


Figure 3.10: Action sequence of PIU policy (left to right) for substrate step change in presence of operator preference for raking. [raking, sweeping, chomping, chomping]

While the PIU policy successfully iterated through available operating modes to find an effective mode to handle the solid substrate that was presented, this mode proved inefficient once that substrate was broken, due to the now exposed granular medium. After breaking through, raking affords a greater efficiency, and is the preferred by the operator. Figure 3.11 depicts the sensor head camera ROI before and after the chomping action is undertaken. The mean pixel values are (250.6, 252.2, 252.6) and (240.1, 241.4, 241.1) respectively, resulting in an L2 error norm of 18.9.

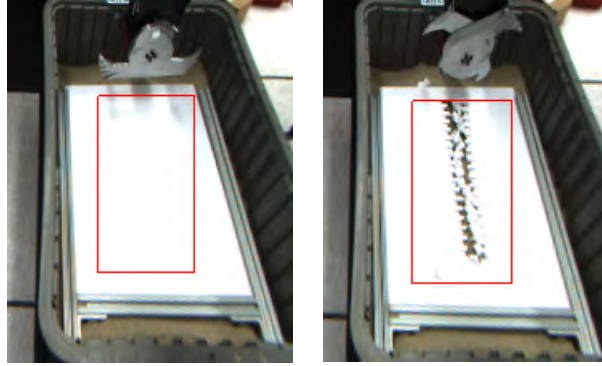


Figure 3.11: Left: Platform sensor head camera ROI initial view of substrate. Right: ROI after chomping action exposes subsurface substrate change to granular medium.

This perceptible change in conditions enables the PIU-ED policy to accelerate the forgetting of prior action-values for each mode, prompting faster exploration of the operator preferred mode. The PIU-ED policy was applied to the same substrate configuration, utilizing the same nominal efficiencies and policy parameters λ and α , but employing exteroceptive decay $\alpha^\phi(t)$ with a signal deadband of $\psi := 0.56$ and $\phi_\Delta(t) := 0.056 \|\bar{\mathbf{p}}(t) - \bar{\mathbf{p}}(t-1)\|_2$. This produced the action-values and mode sequence seen in Figure 3.12 left and right respectively, with actions undertaken depicted in Figure 3.13. Between the third and fourth time steps (with ROI pictured in Figure 3.11 left and right respectively), the exteroceptive difference function exceeds the baseline ψ , causing the action-values of the previously sampled raking and sweeping modes to decay more rapidly than under the PIU policy. The PIU-ED policy then selects the operator preferred mode of raking, which is able to act upon the granular material exposed by the prior chomping action.

3.8 Summary

A novel formulation of the prototypical Multi-Armed Bandit theory has been presented, that adds the notion of an action preference and nominal action-value measurements known *a priori*, and is termed the Obedient Multi-Armed Bandit (OMAB) problem. A success metric for the problem, analogous to MAB regret, is formulated and termed disappointment. Existing MAB policies were adapted to the OMAB formulation and order of disappointment growth shown analytically for the case of stationary cost distributions. Utility of the policy was demonstrated within the applications of multi-modal surface mobility, and substrate excavation.

Future extensions of this work may include learning mappings from measured

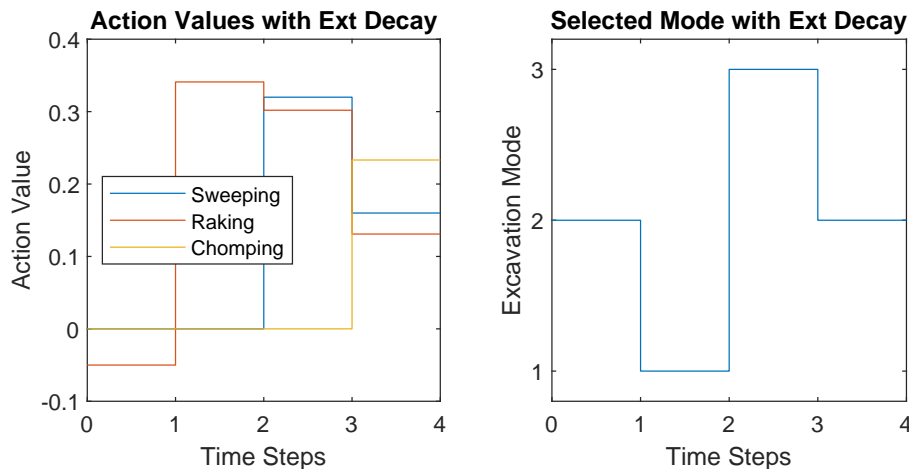


Figure 3.12: Left: Action-value functions for all actions across the timesteps of the excavation run, where $\lambda = 0.05kJ/cm^3$, $\alpha = 0.1$, $\psi := 0.56$, and $\phi_{\Delta}(t) := 0.056 \|\bar{\mathbf{p}}(t) - \bar{\mathbf{p}}(t-1)\|_2$, where perceptible visual changes between time steps 3 and 4 causes exteroceptive decay to lower action-value of operator preferred raking below the alternatives. Right: Action sequence selected by PIU-ED policy. [1=sweeping, 2=raking, 3=chomping]



Figure 3.13: Action sequence of PIU-ED policy (left to right) for substrate step change in presence of operator preference for raking. [raking, sweeping, chomping, raking]

efficacy to discernible visual parameters to allow forward planning with knowledge gained in situ during traversal. Also of interest is formulating a means to update the nominal cost measure for each action, \bar{c}_i , over time so as to more accurately capture monotonic efficacy change caused by actuator degradation or otherwise.

Chapter 4

SEMANTIC TASK OUTCOME CLASSIFICATION

Introduction

Over the past three decades, the robotics community has constructed a solid understanding of the geometric aspects of manipulation – motion planning, grasp (hand/wrist pose) planning, and manipulation control. By contrast, the *semantic* aspect of manipulation remains poorly understood. Concepts related to task success generalize poorly under the strictly geometric metrics that we currently use.

As our field pushes further into unstructured environments, such as human workplaces and homes, the incidence of manipulation failures is set to increase. For robots to be able to operate effectively in these environments, they must possess the ability to identify and correct any failures in tasks they are given, be these due to insufficient planning data, unforeseen impediments, or adversarial interference.



(a) Task success

(b) Failure requiring regrasp

(c) Partial failure?

Figure 4.1: Example outcome space for manipulation prototypical task of placement of object into receptacle in the presence of clutter, where the object of interest is the yellow condiment bottle. Task success is designated by the entirety of the manipulant being encompassed by the bounds of the container in the XY plane.

Figure 4.1 illustrates the need to distinguish modes within the outcome space through the example of the archetypal manipulation task of placing objects into a container in the presence of clutter, where the yellow condiment bottle must come to rest fully encompassed by the blue container in the XY plane for success to be achieved (as in Fig. 4.1a). Fig. 4.1b demonstrates a clear failure mode, which may be described as *object outside container*. The manipulating agent must plan a new grasping and placement action from scratch, lifting the manipulant against gravity before releasing it, in order to rectify the failure. Fig. 4.1c, however, represents a distinct

failure mode of *object on edge of container*, where imparting a small push upon the manipuland in the direction of the container centroid may suffice to achieve task success, without the need to plan an entire grasp and place sequence. Enabling an agent to autonomously recognize these distinctions in failure modes, and equipping them with a mapping to sufficient corrective actions, may allow them to operate more efficiently by reducing the time, and computational or kinetic effort, necessary to rectify a failed task attempt.

In the last decade, the advancement of deep networks within the field of machine learning has offered a means to encode mappings between visual and semantic signatures, such as is necessary to infer the requisite corrective actions for a given task outcome space. One challenge posed by the adoption of such networks, however, is their dependence on large volumes of training data that may be prohibitively expensive to collect, particularly in the field of manipulation where task spaces may be arduous to reset for each experimental iteration. While some groups have resorted to large scale collection of data [82], developing models in simulation offers an avenue of more expedient development, though this approach exposes a shortcoming of deep networks: sensitivity to systemic differences between training and evaluation datasets. Producing photorealistic synthetic images in color (RGB) is one example where deep networks have struggled to generalize to real scenes, but the widespread adoption of depth enabled (or RGBD) cameras within the robotics field has afforded an alternative exteroceptive sensing mode and, when suitable noise models are applied, provides an easier means of generating realistic synthetic data.

This chapter proposes a method of exploring a task outcome space through physics simulation, where only task success is explicitly encoded; clustering parameters in the outcome space to autonomously discover distinct failure modes, allowing a supervisor to associate corrective actions with each; and train a visual classifier on synthetic *depth* maps so as to identify these corrective actions, with a suitable noise model to allow transfer to real task scenes.

Relation to Prior Work

In prior art, much of the existing work on analysis of manipulation task outcomes focused on prediction, so as to minimize the chances of failure. Pastor et al. use reinforcement learning to acquire new motor skills from demonstration for the tasks of striking a pool ball and manipulating a box with chopsticks, while also learning to

predict the performance of a given skill [105]. Moldovan et al. also employ learning techniques, though applied to generalizing affordance models across different object [97]. Through development of a statistical framework, Paolini et al. seek to maximize likelihood of post-grasp manipulation task success by building a model of sensory requirements for a successful execution offline [103]. They then apply it to the tasks of placing, dropping, and inserting objects. Nguyen and Kemp train a pair of support vector machine classifiers to map from a registered pointcloud to 3D locations that are likely to succeed for a given manipulation task [99]. More prior work has focused on the detection of binary failure during the execution of a safety critical task, for instance Pile, Wanna, and Simaan who demonstrate the ability to detect the onset of electrode tip folding during insertion of a cochlear implant [107].

While focusing on predicting outcome during execution greatly improves the likelihood of task success on the first attempt, we believe it necessary to consider the failures that will inevitably occur in these ever more general environments. Similar in nature to this work, Hanheide et al. described unexpected failures in motion planning as a mismatch between expectation and experience [51]. This work goes beyond Hanheide et al. by seeking to infer semantic meaning from observation of a manipulation task outcome space, rather than introspection of the actions that may have caused it. Visual verification of task success was investigated by Erkent et al. [39] by checking for task completion while using visual servoing on various tasks. However, the authors do not attempt to classify the types of failure if success is not detected in a given time. Saran et al. explored viewpoint selection for visually determining binary task failure [121], which is complementary to the work described here.

Structure of the Chapter

Section 4.1 describes the formulation of the model in generality. Section 4.2 details an example application for the model, in the form of the archetypal manipulation task of placing objects into a container. The application process includes the means of simulating the task outcome space, the clustering of workspace parameters to discover distinct modalities of failure, and the generation and processing of synthetic depth maps for deep network training. Section 4.3 then presents the initial segmentation model employed to confer visual signature interpretation through segmentation, providing results for the cuboid container placement task in simulation and with real images. Section 4.4 adapts the visual model by shifting to a more recent ResNet based classification architecture, application of which are detailed in

Sections 4.5 and 4.6 for the problems of single object placement within clutter, and stacking of cuboid objects, respectively.

4.1 Semantic Task Outcome Model

This chapter proposes a semantic task outcome model that leverages contact/physics simulation to parse the structure of a given behavioral domain and to extract a symbolic characterization of the nature of possible failures (or *failure modes* of the task). In turn, the model leverages an image classifier to capture the sensory context of a manipulation task, and to ground failure modes in perceptual data.

The model identifies the failure modes of a given task by executing randomly-perturbed variations of reference trajectories provided by the model designer/instructor in simulation, and grouping those executions according to proximity in a space consisting of geometric measurements effected on end-of-task scene configurations. These groupings may then be associated with a suitable corrective action, furnished by the instructor, that should bring the task to completion with a minimum expenditure of time and/or effort in the case of nominal execution. With an instantiation of the model suitably trained to operate on a particular task, this would enable an autonomous agent to visually detect the symbolic failure modes discovered through clustering, and employ the corrective action recommended by the instructor to allow task completion.

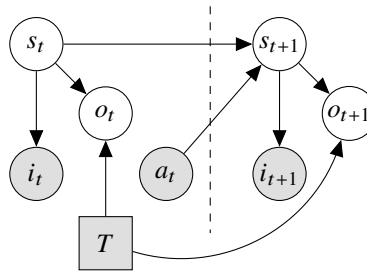


Figure 4.2: Probabilistic graphical model of corrective action selection for manipulation task outcome, where dashed line divides timesteps.

s_t : State of the environment at time t , i_t : Image of the environment at time t , T : Task agent is directed to achieve, o_t : Present state or outcome mode of task T at time t , a_t : Action selected at time t to attempt to rectify any failure.

Shaded/Unshaded = Observed/Latent, Circle/Square = Probabilistic/Prescribed.

In this regard, the model may be likened to a Partially Observable Markov Decision Process (POMDP), in that the agent receives a partial observation of the complete task space in the form of an image. A probabilistic graphical model representation of this concept is depicted in Figure 4.2, where the state of the task space at a given

time step is represented by s_t , and the evolution of this state is modified by the action chosen by the agent at that time step from the set of available actions, $a_t \in \mathcal{A}$. The present outcome that has eventuated is a member of the set of known modes, $o_t \in \mathcal{O}$, and is wholly determined by the state of the environment at the present time step, s_t , and the task that is being undertaken, T . The agent only receives partial information on the full state of the task space, in the form of an image, i_t , and must infer the present outcome mode from that knowledge. The goal of the agent is to apply actions that will bring the outcome mode to the state of success, defined as $o_s \in \mathcal{O}$, through prior knowledge derived from the trained model, and observation of the sampled image, giving a policy function space of $\pi : I \times T \rightarrow \mathcal{A}$.

The responsibility of the image classifier is to identify whether a given perceptual representation of a scene indicates success or failure of the task, and in the case of failure, identify a failure mode. When a failure occurs, it can be taken as evidence that the information used to plan the task was either flawed or incomplete. For this reason, determining the outcome of general tasks in unstructured environments may benefit greatly from an unbiased assessor that ignores any *a priori* knowledge of the workspace. Accordingly, visual signature interpretation is implemented with a Convolutional Neural Network – a model known for its capacity to capture high-variance environmental parameters, which estimates the posterior probability $P(o_t|T, i_t)$ of each mode being present.

The semantic mapping of outcome modes to suitable corrective actions is furnished by the instructor, having inspected the context of each mode provided by the clustered simulation outcomes, as is described by

$$\arg \max_{a_t} P(o_{t+1} = o_s | o_t, a_t) . \quad (4.1)$$

The policy enacted by the agent is therefore the composition of the present outcome mode inferred by the visual classifier, and the semantic mapping to a suitable corrective action, given by

$$\pi = \arg \max_{a_t} P(o_{t+1} = o_s | T, i_t, a_t) . \quad (4.2)$$

4.2 Archetypal Cuboid Placement Task

The initial example chosen for application of the model was that of the archetypal manipulation problem of an object-container insert [2, 84]. This captures many of

the salient challenges encountered within dexterous manipulation, in particular, relative positioning of manipulands and confounding/motivating objects within the task space, while minimizing superfluous complications in the posing of the problem. Complexity was further reduced by employing variable sized cuboids as objects of interest, so as to allow computationally efficient large scale simulation during early development of the model, while the receiving container was constricted to a rectangular cross section box.

Placement of each object was deemed a success if the entirety of an object came to rest within the \mathbb{R}^3 bounds of the receiving box; any other result was deemed a failure. Variation in task execution trajectories was introduced by modulating the pose from which each object was dropped. The initial position of each cuboid was offset in a plane parallel to the ground, by a vector drawn randomly from a uniform distribution defined on $\{(x, y) : x, y \in [-2W, 2W]\}$, where W is the nominal width of the container. The orientation of each object was specified in Roll Pitch Yaw coordinates, drawn from the distribution $(R, P, Y) \sim \mathcal{U}([0, \pi]^3)$.

Simulation and clustering of this task space (described in subsequent subsections) served to expose three outcome modes within this prototypical manipulation example, as are demonstrated in Figure 4.3. These were comprised of *object in box* (success) *object in the box but sticking out* (failure mode 1), and *object fell outside of the box* (failure mode 2). Of primary importance is the emergence of two distinct failure modes, of which responsibility for identifying suitable corrective actions falls to the instructor or operator of the mode during synthesis.

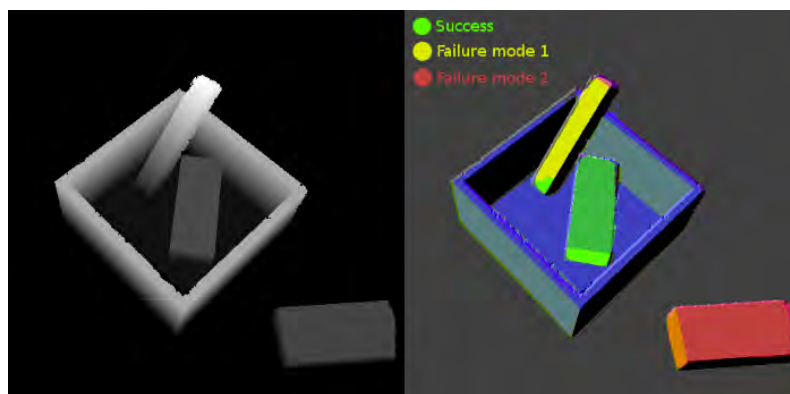


Figure 4.3: Left: Greyscale render of synthetic depth image (without noise) of simulation outcome. Right: Task outcome labels provided by model for archetypal cuboid placement in container task: green corresponds to a successfully-executed insertion task, yellow and red correspond to two modes of failure.

As the cuboid colored red (failure mode 2) has fallen entirely outside the confines of the container, it would be necessary to raise the object against gravity above the edge of the container, in order to release it within the bound of the box in the XY plane. This necessitates planning a new force closure grasp on the object, accompanied by planning of a new placement trajectory accounting for collision avoidance of the grasped object, the instructor may therefore designate a corrective action of *regrasp object and replan placement*.

The cuboid in yellow, however, already has its center of mass resting within the bounds of the container. Increasing the the cuboid's Z position of its center of mass is not necessary for it to undertake a motion into a successful task state. Therefore, the instructor may specify a corrective action of *push protruding object towards center of container*, as that will likely be sufficient to achieve task success, without requiring a computationally and temporally expensive grasp, move, and place behavior.

This initial demonstration of cuboid placement is framed around the ability of an autonomous agent to assess the present state of a potentially compound manipulation task, involving placement of multiple manipulands, as might be expected if such an agent were called upon to correct flawed work of another platform. As such, visual classification is elevated to the level of segmentation of outcome modes (described in Section 4.3), furnishing pixelwise labels of a scene as depicted in Figure 4.3 right, and allowing an agent to spatially identify the presence of multiple modes in one image. In turn, this allows the semantic scene comprehension to be fused with geometric planning methods in order to execute rectifying actions of the present task space for any failure cases that may be present.

Outcome Space Simulation

The outcome space of each manipulation task is explored through randomized simulations, populated with representative manipulands and target/confounding objects particular to a task. These simulations were computed in the open source 3D creation suite Blender, which employs the BULLET physics engine to compute dynamic motion and elastic contacts between objects.

In the case of the cuboid placement task, objects are generated at their uniformly distributed initial pose, inter-object collisions are detected and rectified, then objects are released to fall under the influence of gravity. The simulation terminates upon a condition being met of the velocity magnitude of all dynamic objects within the

scene slowing below a minimum threshold. An example of an initial state of one simulation run can be seen in Figure 4.4, where cuboids for placement are in gray, the container is blue, and the supporting ground plane in brown. The wire-frame pyramid shape represents the viewpoint of the camera observing the scene.

After development of the template task scene within the visual interface of the Blender suite, the software and physics engine were compiled into a python library to allow parallelization of perturbed trajectory simulations across many-core server processors. For the case of the archetypal cuboid placement task, 10,000 variants of the reference scenario were simulated across the parameter set, requiring 1.5 days computation on a 6 core Xeon X5690 CPU. Parameters perturbed across the dataset to produce a wide spread of outcomes include:

- Number of cuboids placed, $n \in [1, 5]$
- Dimensions of each cuboid within scene, $d_i \sim \mathcal{U}([0.1m, 0.3m]^3 \forall i \in [1, \dots, n])$
- Pose of each cuboid relative to center of container (as defined in Sec. 4.2)
- Scaling factor of nominal container $d_c \sim \mathcal{U}([0.7, 1.3]^3)$
- Pose of container drawn uniformly across the task ground plane in $SE(2)$

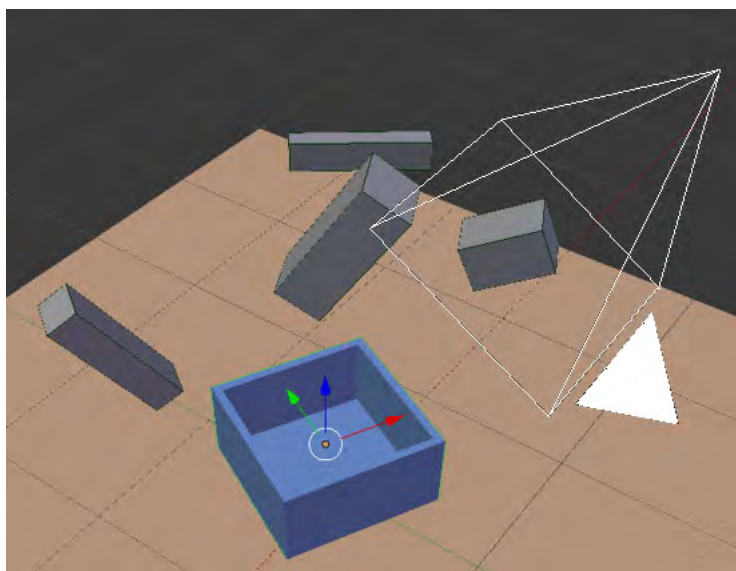


Figure 4.4: Example cuboid placement task scene at initialization of simulation, prior to BULLET physics engine imparting gravity and contact dynamics.

Autonomous Outcome Mode Clustering

As previously intimated, one key advantage of exploring the outcome space of a given manipulation task is the ability to easily interrogate the myriad geometric parameters that may describe the space, rather than laborious hand calculation of such parameters necessary in experimentally derived data. Successful outcomes are explicitly identified using rules furnished by the instructor, but the parameters of all remaining failure cases may then be interrogated for any form of structure that may be used to inform recovery actions.

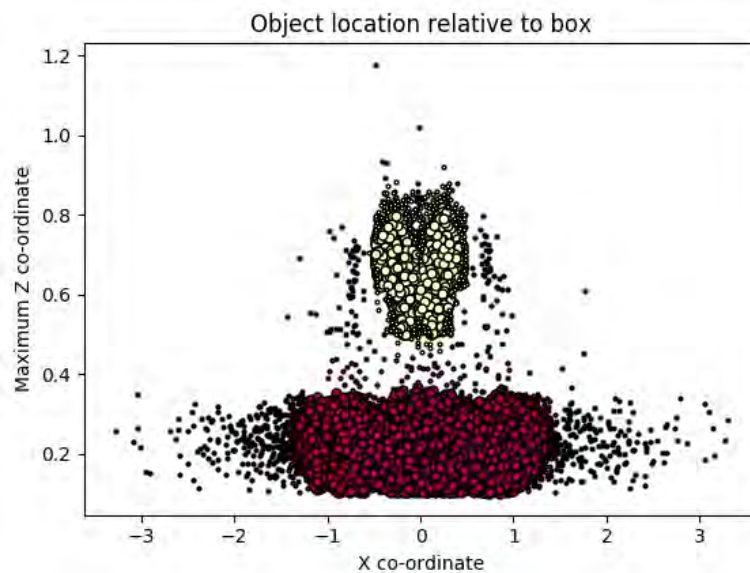


Figure 4.5: Slice of DBSCAN clustering within cuboid placement task outcome parameter space, highlighting distinction between objects resting with major axis horizontal on ground plane outside container, from those held above the edges of container.

A variety of means could be employed for this interrogation of the outcome parameter space, but the technique chosen for application within this model was that of Density-based Spatial Clustering of Applications with Noise (DBSCAN) [40], as implemented within the Sci-Kit python library. This employs nearest neighbour calculations to identify closely packed regions of data points within the parameter space (with multiple neighbours within a bound), while denoting points in low density regions as outliers, and points along the edge of a group with minimal neighbours as their boundary. In the example of cuboid placement task, the clustering parameters included among others:

- Location relative to container,

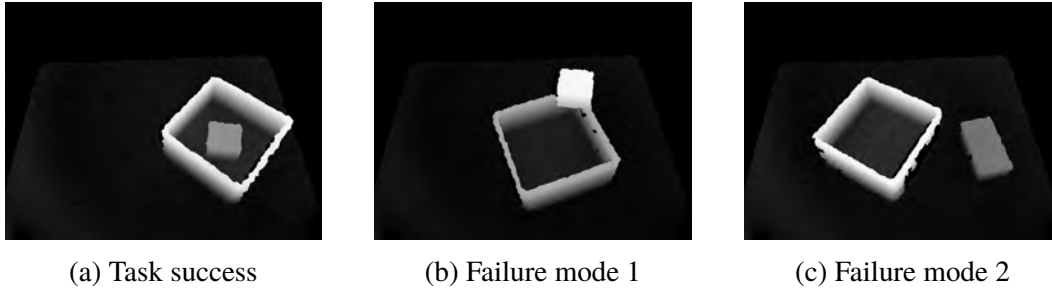


Figure 4.6: Example depth images of outcome modes discovered in simulation for cuboid container placement task.

Outcome mode	Success	Object on edge	Object outside
Corrective action	No action	Push towards center	Regrasp and place

Table 4.1: Instructor specified corrective actions for cuboid placement task outcome modes discovered in simulation.

- Final orientation
- Number of container contacts
- Number of inter-object contacts
- Maximum Z co-ordinate of cuboid vertices

The highest scoring cluster set was of dimension 2. Distinction between the two members was derived from the last of the parameters listed, the maximum Z co-ordinate of the object after placement, as depicted in Figure 4.5. Examples of depth maps of the three output modes produced through clustering may be seen in Figure 4.6.

Responsibility for prescribing suitable corrective actions for each of the discovered modes then falls to the instructor. This imbues the model with the means to interpret the semantic understanding of the output space that will be furnished by a visual classifier, as previously denoted $\arg \max_{a_t} P(o_{t+1} = o_s | o_t, a_t)$. An example of corrective actions that might be recommended by the instructor for the cuboid placement task can be found in Table 4.1.

Depth Map Generation and Processing

Once symbolic failure modes have been identified in the clustering of the simulated dataset, synthetic depth maps must be generated from these task scenes to act as training data for the visual classification network. This is achieved through

application of a realistic depth-camera sensor model named BlenSor [48], that employs pixel-wise ray tracing to produce structured point clouds from scenes in the Blender creative suite. These pointclouds are then converted to grayscale bitmap representations with the Point Cloud Library, where Z position of a point in the tabletop frame is mapped to pixel intensity, producing images as seen in Figure 4.7a.

It may be observed that this initial naive synthesis creates artificially precise edges, so a realistic noise-model is then applied to introduce regional fluctuations in depth values typical of the sensing modality, demonstrated after bitmap conversion in Figure 4.7b.

As is common practice in computer vision for tabletop manipulation, the ground plane upon which objects rest is then subtracted in order to eliminate visual signatures irrelevant to the task space, the result seen in Figure 4.7c, comparing well with the real depth map captured in Figure 4.7d. This is accomplished with the algorithm RANdom SAmple Consensus (RANSAC) [42], that estimates Cartesian plane parameters of the form $ax + by + cz + d = 0$ by sampling from the points

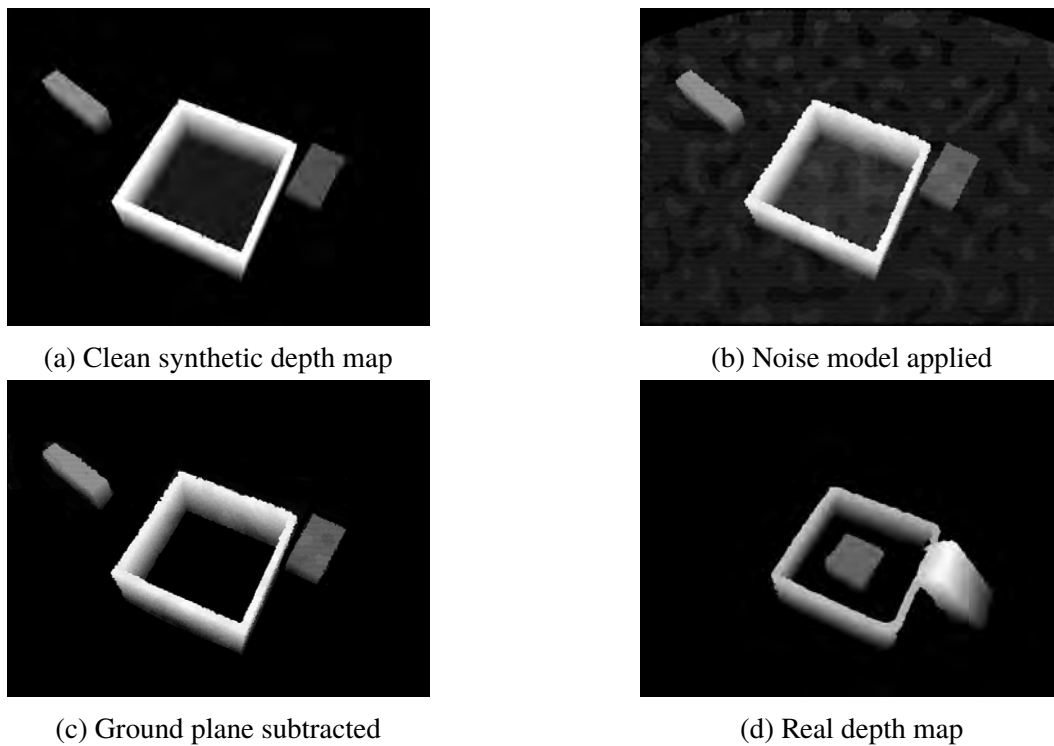
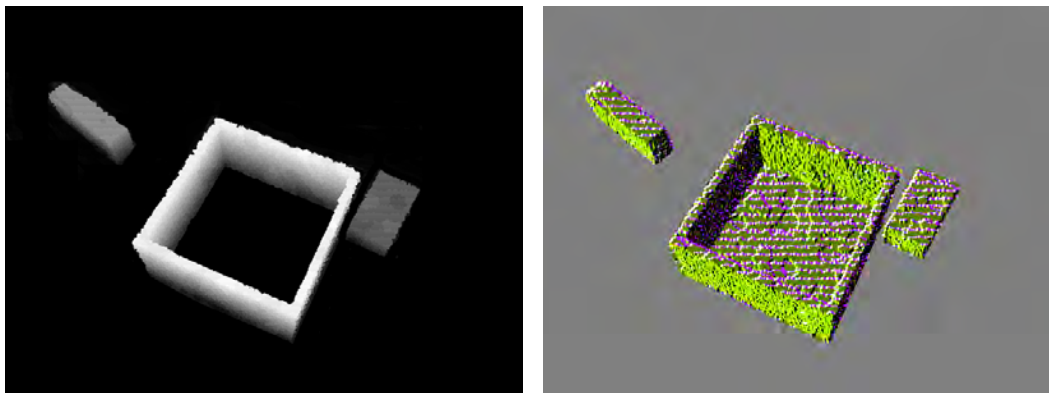


Figure 4.7: Synthetic depth maps at various stages of processing, in comparison with real depth map with ground plane subtracted

within the cloud, then eliminating points within some range of the highest likelihood plane, which is a function of the Point Cloud Library (PCL).

Once converted into a suitable 2D bitmap form, input images must be processed into channels from which a deep network can derive sufficient context to classify the contents. While this has been studied extensively within the domain of color/RGB images, this remains an open question for the employ of *depth* enabled exteroception. Some prior art has attempted to apply CNNs to depth values directly [104], and yet others have employed additional geometric parameters, such as camera pose relative to the ground frame, to enrich depth data across multiple channels (such as HHA [49]). For the purposes of the visual outcome model, we are primarily interested in the relative spatial positioning of items within the confined task space, as opposed to their absolute position within the world frame and so instead opt to employ a Sobel gradient operator, to produce three channels from the grayscale depth, as suggested in [33], which also afford a level of invariance to net depth. Pixels are then normalized about the value of half the bit, or color, depth.



(a) Synthetic depth map with noise model

(b) Sobel operator preprocessed image

Figure 4.8: Synthetic depth maps demonstrating noise model and preprocessing applied prior to training and inference. Visualization of the preprocessed image maps the Sobel operators \mathbf{G}_x , \mathbf{G}_y , and $\mathbf{G}_{\|\nabla\|}$ to the R, G, and B color channels respectively.

The Sobel x -gradient and y -gradient operators, G_x and G_y , are defined in Equations 4.3 and 4.4 respectively, while the magnitude of the Sobel gradient is simply the L2 norm of these operators, as given in Equation 4.5, where \mathbf{I}_m represents the 2D image space, and B_d is the bit-depth or range of pixel values of the image. Figure 4.8 presents a synthetic depth map before and after processing.

$$\mathbf{G}_x := \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * \mathbf{I}_m, \quad (4.3)$$

$$\mathbf{G}_y := \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{I}_m, \quad (4.4)$$

$$\mathbf{G}_{\|\nabla\|} = \sqrt{\mathbf{G}_x^2 + \mathbf{G}_y^2}. \quad (4.5)$$

4.3 Initial Segmentation Model

The visual classification model employed to infer semantic outcome modes for the archetypal cuboid placement in container task was that of a Convolutional Neural Network (described in generality in section 2.3), in particular the fully-convolutional MultiNet architecture proposed by Teichmann et al. [33, 129]. As it produces a segmentation output, this served to furnish pixel-wise labelling of an input image, allowing an agent to infer spatial context, and localise each of the present outcome modes within the task frame. As previously intimated, this was implemented so as to allow the agent to evaluate the present state of an environment in which multiple similar tasks had already been attempted to mixed success. The agent might then fuse geometric information from the depth map with the spatial semantic context of the labelled output image, in order to apply corrective actions to multiple manipulands within the scene. To further facilitate fusion of spatial semantic context with geometric recovery planning, the container itself was labelled in the synthetic scenes, such that corrective actions could infer target locations if not otherwise given.

The network was trained on 95% of the 10,000 synthetic images produced through simulation (5% withheld for validation), with ground truth class labels generated within the software suite, such as the two examples seen in Figure 4.9 right.

To measure the network’s performance, we evaluated the network’s ability to predict the presence or absence of each of the three outcome classes (success, failure mode 1, failure mode 2), by applying the model to the 5% validation set, and pixel-wise comparing the inferred mode labels to the ground truth labels of the validation set generated in simulation. The results showed that in this canonical experiment, the network identified the correct presence or absence of all outcome modes in 86.8%

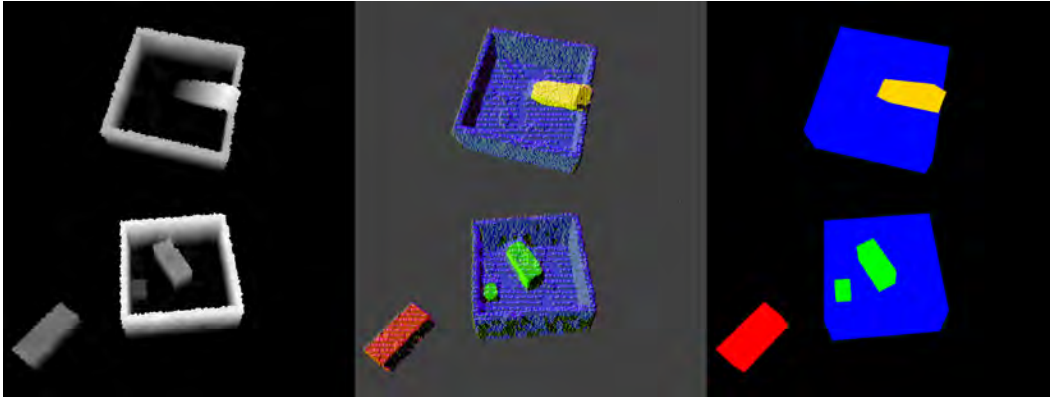


Figure 4.9: Left: Synthetic depth maps of simulated task outcome spaces. Center: Overlay of segmented classes produced by CNN segmentation model. Right: Ground truth labels used for model training. [Green=Task Success, Yellow=Failure mode 1, Red=Failure mode 2]

of images in the synthetic validation set, interpreted as when only labels of the true modes were present, and a majority of pixel labels matched those of the ground-truth image. Each image inference required 139ms on a Titan X GPU. The first row of Fig. 4.9 shows an example of failure mode 1: the object is not entirely within the volume of the container. The second row shows two successful inserts (in green), and one failure of mode 2 (in red).

The ability of the model to transfer to real task scenes was then evaluated through processing of depth maps captured with a Microsoft Kinect camera. A sample set of 15 representative scenes was captured, with a uniform spread of present outcome modes across the set. Of the 15 images, 9 were correct by the aforementioned evaluation metric applied to simulation validation results, 2 marginal but classed incorrect, and the remainder incorrect, resulting in a successful inference rate of 60% (as compared to a random guess outcome of 33%). An example of successful and unsuccessful inference can be seen in Figure 4.10 top and bottom respectively, the latter of which correctly identified failure mode 2 across the majority of pixels of the rightmost object, but failed to classify the object with task success inside the box by designating the majority of it failure mode 1.

4.4 ResNet Classification Model

The initial visual model was aimed at spatial recognition of a potential multiplicity of manipulands in various states of task outcome, and served to demonstrate that the model was capable of transferring from simulated data to real task scenes. Focus then shifted to interpreting the outcome of a single task execution attempt,

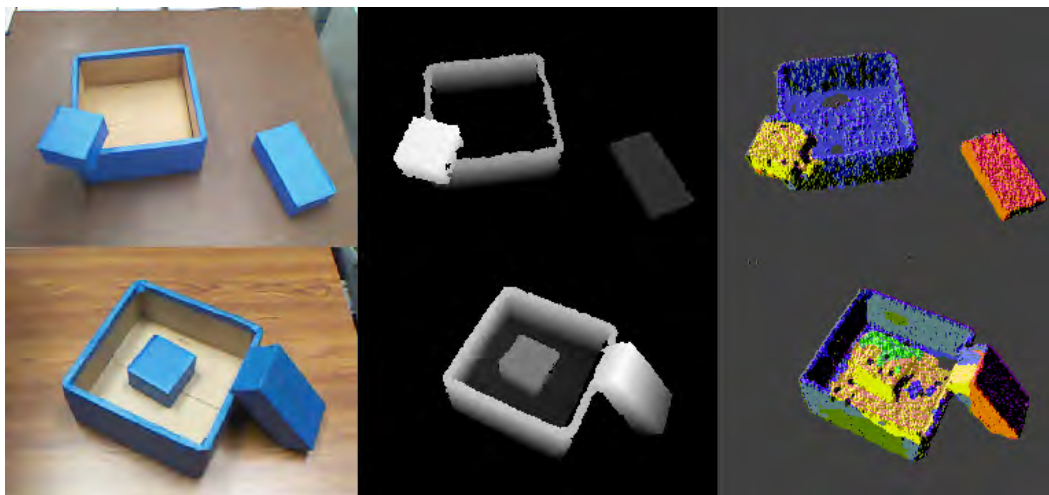


Figure 4.10: Left: RGB images of real scenes from Kinect camera. Center: Grayscale renders of processed depth images from Kinect. Right: Overlays of segmented classes produced by CNN.

Top: Example of successful inference of failure modes 1 and 2. Bottom: Example of failed inference, where true outcomes were success and failure mode 2.

[Green=Task Success, Yellow=Failure mode 1, Red=Failure mode 2]

being more conducive to the POMDP formulation of the agent acting and then interpreting that singular outcome, rather than a set of outcomes produced by an external party.

To achieve this end, the visual classification model was altered from that of segmentation, producing the pixelwise labels seen previously, to a fully-connected classification architecture. This provides a scalar probability of the presence of each of the available visual signature/outcome modes within an imaged scene, described in the POMDP model as $P(o_t|T, i_t)$.

The particular architecture employed was that of a Deep Residual Network, or ResNet [52], which enjoys the advantage of being relatively more modern than the Multi-Net formulation. This is a variety of CNN that employs edges inspired by pyramidal cells in the cerebral cortex, where additional connections are added to the prototypical form that skip over one to several layers, producing a *residual* of convolved signatures earlier in the model.

While the cuboid placement task was of excessive simplicity by design, and served to illustrate the viability of the model in a minimal representative context, the ResNet classification architecture was applied to more realistic manipulation tasks detailed in the sections that follow. The ground-plane removal assumption was also removed

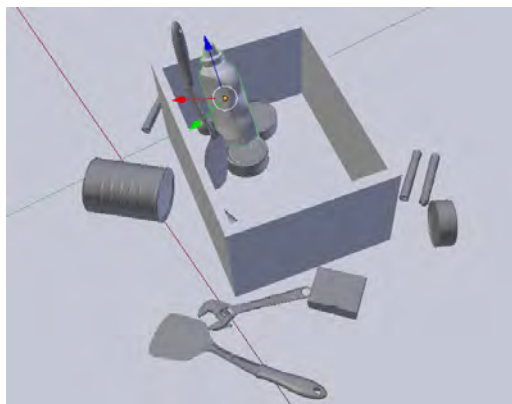
for the subsequent models, to present a stronger case for its ability to generalize to real task scenes without extensive tuning or preprocessing.

4.5 Placement Within Clutter Experiment

The first of the more visually advanced tasks to be investigated was an extension of the archetypal manipulation task of object placement; in particular, the placement of a particular object (a yellow condiment bottle) within a container, in the presence of confounding objects or “clutter” drawn from the Yale-CMU-Berkeley (YCB) object dataset. Specification of successful task outcome imitated that of the cuboid placement task, save for the distinction that the *centroid* of the manipuland must come to rest within the \mathbb{R}^3 confines of the container, as opposed to its entirety. Example outcome scenes for this task can be viewed in Figure 4.11, where the left real image demonstrates a failure of presently uncharacterized variety, and the right synthetic image with manipuland pose displayed represents a successful outcome, despite the extremity of the manipuland extending beyond the confines of the container in the Z -axis. Note that the height of the manipuland is such that standing on end at the base of the box is sufficient to raise the centroid above the edge, thus constituting a failure. Similarly to the cuboid placement task, it may also be held in a raised position by the clutter already present, causing the centroid to rest above the bounds of the container.



(a) Real task scene



(b) Simulated task scene

Figure 4.11: Model demonstration task involving placement of yellow condiment bottle into container in presence of clutter objects from YCB dataset.

The task was simulated within the BULLET physics engine across 5,000 perturbed trajectories, where clutter objects were placed first, followed by insertion of the object of interest, as would occur in a placement attempt by a manipulation agent. The

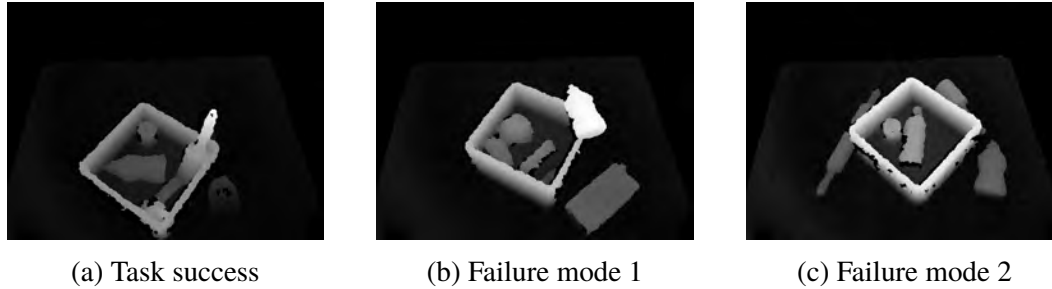


Figure 4.12: Example depth images of outcome modes discovered in simulation for condiment placement within YCB clutter task.

outcome spaces of each scenes were then clustered with the DBSCAN algorithm, employing the same parameters as for the cuboid placement task. Perhaps unsurprisingly, the highest scoring clustering of parameter groups differentiated between locations of the manipuland centroid relative to the container in the XY plane, with poses outside the XY bound of the container forming one group and those held above the Z limit of the container forming another, with examples of each mode seen in Figure 4.12. Examples of suitable instructor supplied corrective actions for the task are then given in Table 4.2.

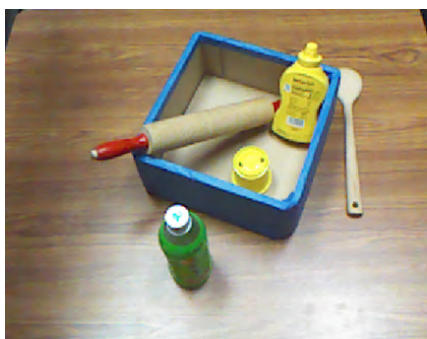
Outcome mode	Success	Object on edge	Object outside
Corrective action	No action	Push towards center	Regrasp and place

Table 4.2: Instructor specified corrective actions for condiment placement within clutter task outcome modes discovered in simulation.

Synthetic depth maps were then produced of the 5,000 simulated task outcomes, and the ResNet visual-signature classification network trained on 95% of the images, with 5% withheld for validation, with ground-truth class/mode flags furnished by the physics simulation software. Each image inference with the ResNet model requires 31ms on a Titan X GPU. The efficacy of the visual classifier within simulation was evaluated by comparing the highest likelihood outcome mode inferred for each of the validation images with the ground-truth label calculated in simulation, which resulted in an accuracy of 84%.

The ability of the model to transfer to real scenes for the placement in clutter task was then evaluated through application to 15 depth maps captured with the Microsoft Kinect camera. Evaluation of the inferred outcome modes with aforementioned criteria for the simulated validation set produced 10 correct outcomes, and 5 incorrect, giving a classification accuracy of 66.7% (as compared to a random guess

outcome of 33%). The color and depth maps with inference results of the real validation scenes may be found in Appendix B.1, with examples of correct and incorrect outcomes seen in Figure 4.13 top and bottom respectively.



(a) Inferred success with 0.78 likelihood, correct



(b) Inferred failure mode 2 with 0.74 likelihood, incorrect

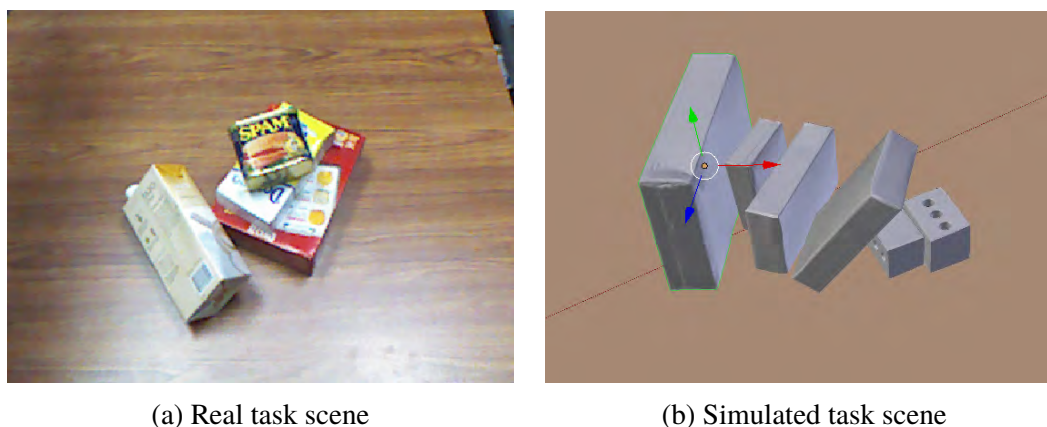
Figure 4.13: Inferred outcome modes from visual classification model of placement within clutter task.

4.6 Object Stacking Experiment

The next task used for demonstration of the model was that of vertically stacking cuboid objects from the YCB dataset. As with the previous classification only application of the visual model, it is assumed the manipulation agent is presented with a partially complete compound task; that is, there are already a number of items stacked. The agent then attempts to place a new item atop without disturbing the pre-existing objects, with task success specified as the novel object coming to rest wholly supported by the preceding stack member. Real and simulated examples of uncharacterized failures can be seen in Figure 4.14.

The task was simulated within the BULLET physics engine across 5,000 perturbed trajectories, where a pile of varied objects was placed in a stacked formation and allowed to come to rest. An additional cuboid was then released a short distance above the stable stack, and contact/physics simulation allowed to progress until all object motion was below a minimal threshold. Perturbations were introduced in the number and type of objects in the initial stack, the relative positioning and orientation between subsequent members of the initial stack, and lastly the release pose plus type of the final item placed in the action being inspected.

The parameter space of the simulation outcomes that did not meet success criteria was then clustered across a variety of parameters extracted from the final state of each scene, including:



(a) Real task scene

(b) Simulated task scene

Figure 4.14: Model demonstration task of stacking cuboid objects from the YCB dataset.

- Area of convex hull of object centroids in XY plane
- Number of object-pair contacts, normalized by total number of objects
- Sum of Z-coordinates of each object centroid relative to Z-coordinate of supporting object in initial stack
- Total number of ground contacts among objects

Clustering of the outcome dataset yielded three distinct groups within parameter space; success, where all items remained sequentially supported; failure mode 1, where the novel object came to rest aside from the stack, or semantically *placed object fell off pile*; and lastly failure mode 2, where multiple objects contact the ground and centroids no longer monotonically increase in height, or *placed object knocked over the pile*. Examples of these can be seen in Figure 4.15 a, b, and c respectively. The instructor might then supply the corrective modes defined in Table 4.3.

Outcome mode	Success	Object fell next to intact stack	Object caused stack to deconstruct
Corrective action	No action	Regrasp and place last object	Clear workspace and re-initiate stacking

Table 4.3: Instructor specified corrective actions for YCB cuboid stacking task outcome modes discovered in simulation.

As with the previous task, synthetic depth maps were produced of the 5,000 simulated outcomes, however, image preprocessing was modified to utilize a raw depth

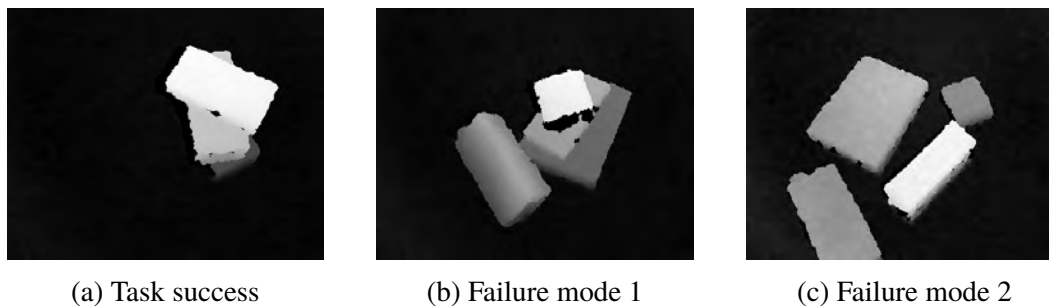


Figure 4.15: Example depth images of outcome modes discovered in simulation for YCB cuboid stacking task.

channel rather than the magnitude of the depth gradient. This was due to an expectation that it may confer additional geometric context of the height of the stack upon the classifier input, which might be of importance given the hierarchical relative positioning of the initial configuration. The ResNet visual-signature classification network was then trained on 95% of these images, with 5% withheld for validation, with ground-truth class/mode flags furnished by the physics simulation software. Evaluating the resulting visual classifier against the simulated validation set, matching highest likelihood inferred outcome mode to ground-truth class labels showed the network to have an accuracy of 82%.

Transfer of the model to real depth maps uncovered a shortcoming of the modified input channels, however. Replacing the magnitude of Sobel gradient channel with the raw depth value removed the invariance to systemic differences in depth between the synthetic and real datasets, reducing the generality of the classifier. Over 15 real depth maps with an even distribution of true outcome modes, all were classified by the model as failure mode 2, where placement of the most recent item caused the structure to fail, save for a single correct classification of success. As a result of this 5/15 or 33% of inferences were correct, which of course is equal to the random chance likelihood. This could likely be rectified by retraining the model after reverting the input channels of the network, though was beyond the period of performance of the project. The color and depth maps with inference results of the real validation scenes may be found in Appendix B.2

4.7 Summary

In this chapter, a model to prescribe semantic meaning to task outcome modes autonomously discovered in simulation was presented. Synthetically generated datasets allowed an instructor instantiating the model for a given task to discover

symbolic failure modes, and assign corrective actions that an autonomous agent should undertake if they are encountered. Two varieties of visual classifier were presented to allow the agent to infer semantic meaning of a task outcome, and application to three representative manipulation tasks demonstrated with simulated and real dataset validation.

Chapter 5

FORCEFUL MANIPULATION IN CLUTTER

Introduction

Grasp selection in unstructured environments has proven a challenging task for the robotics community, and is complicated further when a system lacks *a priori* knowledge of the shape and mass properties of objects it may be called upon to manipulate. While indiscriminate, randomized grasp and lift motions can be coupled with proprioceptive wrench guarding to eventually find a viable removal candidate (if one exists), the aim of this task is to decompose a pile of potentially massive objects time efficiently. This necessitates leveraging exteroception to infer the composition and structure of the pile, allowing the system to more rapidly identify grasps that comply with the force and torque limits of the manipulation system.



Figure 5.1: Left: Army Research Lab's 'RoMan' mobile manipulation platform with broken proximal joint on Robotiq 3-finger gripper. Right: Example debris pile of massive objects, where lifting two or more items simultaneously may exceed manipulator payload limits.

To dislodge a candidate extraction object from the structure, the system must be capable of breaking the *stiction* restraining it. This describes the static friction that must be overcome to enable motion of a given stationary object relative to its contacting surroundings, and is a portmanteau of static and friction. Failure to suitably predict or detect excessive stiction or mass of a candidate manipuland also has the potential to cause catastrophic damage to a system, as occurred in Figure 5.1 left, which could render an autonomous agent inoperative.

Prior art has sought to address the problems of object agnostic grasp synthesis [17, 19, 87], grasping of known and unknown objects amongst clutter [20, 149], as well as lifting of massive objects with wrench constrained end effectors or actuators [64]. This work seeks to address the intersection of these, in particular the disassembly of unstructured piles of massive objects (e.g. Figure 5.1 right), where lifting one object may induce lifting or pulling of other objects, which in turn increases the required grasp wrench, and may exceed the capabilities of the manipulation system.

While operating in a workspace where the geometric, inertial, and contact properties of all present objects might allow a deterministic calculation of the force structure of a pile to be posed, this representation seeks to address the circumstance where little to no information on the workspace is available *a priori*, save for the common manipulation sensing modality of vision with 2.5D depth (example seen in Figure 5.8 left and center). For this purpose, the force structure is formulated with a probabilistic representation, which allows it to capture the uncertainty inherent in the estimations of the parameters quantifying and shaping the structure, while affording the capability to include improved data where available.

Chief among these estimations is the problem of singulating objects within the pile and determining adjacency, which is assumed to be furnished by any chosen segmentation algorithm (e.g. Locally Convex Connected Patches [127] in Figure 5.8 right), such that the formulation is agnostic to the visual processing employed. Coupled with rough estimates of the mass as a function of volume, and contact between the singulated objects, the formulation provides a *best guess* of which objects may be within the system’s ability to dislodge, as well as the *direction* of extraction that may induce the least resistance, termed the *extraction vector*. A



Figure 5.2: Left: RGB image of example debris pile containing aluminum truss segment, safety barrier, 4x4 wood section, and pallet. Center: Point cloud of workspace captured with RealSense D435. Right: Singulated object candidates with geometric adjacency from Locally Convex Connected Patches algorithm. [127]

manipulating agent may then select an object-extraction vector pair from among these candidates to execute.

One key source of additional structural information is the resisting wrenches experienced by the system when it attempts to grasp and extract a selected object, as is commonly measured via a wrist mounted force-torque sensor. In the case that the wrench is within the capability of the system, the extraction attempt may proceed to completion; but, in the event of measured forces exceeding safety limits, the agent may add this data point to the representation of the grasped object and select the next most viable extraction vector for that object, or a different object entirely. In this regard, it may be described as facilitating informed proprioceptive *probing* of the pile, directed toward the task of identifying viable objects for removal.

Relation to Prior Work

Of perhaps closest relevance to this work is that of Boularias et al. [20], who approach the problem of grasping objects in dense clutter with no prior information through the application of reinforcement learning. The robot learns online how to manipulate objects through trial and error, in particular through the application of *pre-grasping* actions that seek to expose objects for easier geometric access to suitable grasps, without giving consideration to the mass of candidate manipulands. By contrast, the formulation presented in this chapter does not require any learning through repeated application, but seeks to leverage geometric context that can be provided by vision algorithms, even in the absence of any other prior information, and apply a model of contact physics to predict viable grasps, without extensive interaction with the environment.

Recent work by Holladay et al. looked at representing the kinematic and wrench constraints of the use of different tools to enable *forceful manipulation* [55]. While this leverages the concept of a wrench space surface to describe the forces that must be applied to a given tool to operate it, the analysis assumes exact prior knowledge of the manipulands and their interaction with the environment, which cannot be assumed working in unstructured environments.

Zhang and Trinkle propose to use a particle filter to simultaneously estimate the physical parameters of an object and track it while it is being pushed. The dynamic model of the object is formulated as a mixed nonlinear complementarity problem [150]. While they address uncertainty in the physical properties of a manipulant, they depend primarily on tactile information over the course of a motion to infer

these properties, and are only concerned with the in-hand manipulation of a single object, rather than selection of viable grasps among a set of candidate manipulands as is the focus of the work in this chapter.

Berenson et al. investigated planning articulated arm motions in the presence of wrench constraint manifolds in the arm configuration space, such as might be imposed when lifting massive objects between two end effector poses [12]. This is distinct from the work in this chapter in that it focuses on wrenches imposed on the end effector by the mass of a manipuland during large scale motion, rather than predicting the restraining wrench imposed on a manipuland by its surroundings, but is complementary to the work presented here in that it could inform arm trajectories once an object is extracted.

The problem of identifying object properties in-hand through properceptive inference was investigated by Burkhardt et al. [24], who also employed a Gaussian Process Implicit Surface (GPIS) in their representation. They focused on localising the center of mass through changing the orientation of a grasped object with respect to gravity, through wrench measurements from the wrist mounted force-torque sensor, while also inferring the geometry of the object through probing. In contrast to the work presented here, the GPIS was applied to the geometric representation of the object's surface, rather than in describing the wrenches restraining an object. They also rely solely on proprioceptive information, rather than attempting to leverage any information gleaned from visual sensing modalities as in this chapter.

Structure of the Chapter

Section 5.1 proposes a means of representing the forces restraining an object within a pile in a deterministic fashion, employing Newtonian mechanics and Coulomb friction to predict a conservative upper bound on the wrench necessary to dislodge a candidate manipuland. Section 5.2 then presents a means of describing this representation probabilistically, in order to account for uncertain parameters, leveraging the concept of a Gaussian Process Implicit Surface. Section 5.3 proposes various means of estimating the parameters employed by the representation through processing of an RGBD view of the task scene, including object singulation and adjacency, center of volume, mass, and contact poses. It also presents means of incorporating proprioceptive data points acquired through extraction attempts, as well as any *a priori* parameters that might be available to the manipulation agent in the case of some manipulands being known. Section 5.4 then suggests possible criteria for

selecting between candidate object-extraction vector pairs predicted to be within the system's capability.

Note: This chapter is a partial draft of a section for a journal article that will be submitted to a JFR special issue on May 1st 2020. Its inclusion in the thesis (as a chapter or appdenix) will be dependent on the committee's input and, in the case of affirmation, will be updated with the final version prior to that date.

5.1 Wrench Space Stiction Manifold

Dislodging and object from an pile requires overcoming both the mass of the object in grasp, and the forces imparted upon it by surrounding objects. These forces may be comprised of both a component of their own mass, as well as the frictional force between the objects that resists relative motion. The magnitude and number of these forces that must be overcome to induce motion in the object varies as a function of the direction a manipulating agent attempts to impart a grasp wrench, termed the *extraction vector*. A rudimentary example of this may be seen in the \mathbb{R}^2 configuration space example depicted in Figure 5.3 from the cases of 1) lifting the central object directly upward, and 2) moving it purely in the horizontal plane.

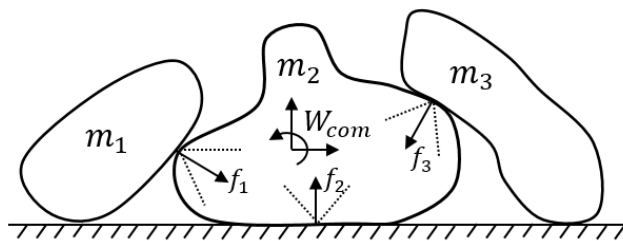


Figure 5.3: Incident forces of two contacting objects upon a central object of interest, where W_{com} describes the net wrench imposed upon it in the Center of Mass (COM) frame, m_i is the mass of each object, and each f_i represents the friction cone of a contact wrench imposed on the target object by its surroundings.

In the first case, vertical motion requires that the force of gravity on the mass of the central object must be overcome to produce non-zero acceleration, in addition to displacing the left and right objects lying upon it by overcoming f_1 and f_3 . It does not, however, require overcoming the frictional or supporting force of the ground contact (assuming no adhesive properties), as this contact is broken upon inducing motion.

In the second case, the mass of the central object only influences the force required to induce motion insofar as it contributes to the friction between the object and

the ground plane, the horizontal component of f_2 . Depending on the direction of motion, one of the contact forces from the incident objects (f_1 & f_3) will oppose motion, while the other will contribute to it.

A third case of note is that of pressing the central object downwards, where the presence of an impenetrable supporting plane will result in zero motion no matter the magnitude of wrench imposed.

Due to this variation in the magnitude of wrenches necessary to induce motion with the direction in which they are applied, this work proposes to probabilistically represent the upper bound of stiction and mass forces restraining each object as a manifold, denoted in the center of mass frame as \mathcal{W}_{com} , in the \mathbb{R}^3 or \mathbb{R}^6 wrench spaces of 2D or 3D object configuration spaces respectively. This representation is termed a *wrench space stiction manifold*, which forms a bound between the set of wrenches imposed upon the center of mass that *would not* cause motion (the manifold interior), and those that *would* cause motion (the exterior), denoted the *stiction breaking* wrench space \mathcal{W}_{com}^{sb} . The minimum distance of this manifold from the origin represents the smallest wrench that could be applied to dislodge the given object from the pile.

When provided with a possible grasp pose on the object by an external planner, this manifold may then be transformed from the COM wrench space of the object in question, to the *grasp* wrench space for that candidate grasp, as given by

$$\mathcal{W}_g^{sb} = \text{Ad}_{g_{g2com}}^T \mathcal{W}_{com}^{sb} . \quad (5.1)$$

The intersection of \mathcal{W}_g^{sb} and the *grasp wrench constraint* manifold, \mathcal{W}_g^c (defined by the force-torque limitations of the system), therefore provides the space of extraction-vectors for the given grasp pose that are both within the system's capability, and would dislodge the manipuland from its surroundings, defined as

$$\mathcal{W}_g^v := \mathcal{W}_g^{sb} \cap \mathcal{W}_g^c . \quad (5.2)$$

In the case that $\mathcal{W}_g^{sb} \cap \mathcal{W}_g^c = \{0\}$, the formulation predicts that there are no extraction-vectors within the capability of the system that will cause the manipuland to move at that particular grasp point.

To construct the manifold for each object within the pile, we define the forces acting on that object in the frame of its center of mass (COM), which is initially

assumed to be coincident with its center of volume, and oriented in alignment with the inertial frame. The force of gravity therefore acts directly through this frame, while the forces imposed by each of the inter-object contacts acts through the contact point. The following formulation is presented in the context of analysis of a single target object within the pile, where each of the inter-object contacts it is subject to is enumerated i , and is comprised of the vectors depicted in Figure 5.4. The force imposed between objects acts through the contact point denoted \bar{r}_i for the i 'th contact, and is comprised of both a normal and frictional component, \bar{n}_i and \bar{f}_i respectively. The latter of these is defined with the Coulomb friction model, and may lie anywhere in the orthogonal space of the normal vector, where $D \in \{2, 3\}$ is the dimension of the configuration space such that

$$\bar{f}_i \in \mathbb{R}^D, \quad \text{s.t. } \bar{f}_i \cdot \bar{n}_i = 0, \quad |\bar{f}_i| \leq \mu |\bar{n}_i|. \quad (5.3)$$

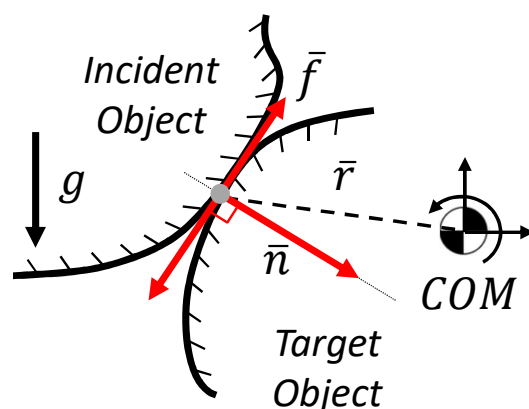


Figure 5.4: A single inter-object wrench defined in the center of mass (COM) frame of the target object. In the static equilibrium of a pile, the effect of gravity on the mass of the incident object causes it to impart a normal force \bar{n} on the target object at the contact point \bar{r} , as well as a Coulomb friction force \bar{f} , under the influence of gravity g .

In a similar fashion to the ground plane contact, these inter-object contacts are directed, in that only objects below the line of gravity with respect to their pair must overcome the contact wrench to break stiction, whereas the reverse is not as the contact is simply broken. This results in the object adjacencies derived from the exteroceptive algorithm of choice being regarded as a *directed wrench graph*, where the direction of edges indicates the direction of gravity, with the originating node imparting a force downwards upon the supporting object.

As the forces supporting an object may be statically indeterminate, this initial Newtonian analysis assumes the maximum friction force to provide an upper bound on the expected extraction effort. Also in keeping with this methodology, the magnitude of the normal force \bar{n}_i between objects is taken to be full net weight experienced by the object incident for that contact, w_i^{net} . This is based on an assumption that the pile is loosely stacked in the XY plane, i.e. that there are no bounding walls imparting significant lateral forces, such that the likelihood of any object being wholly supported by normal forces acting within $\pi/4$ radians of horizontal is low (the angle at which two symmetric supports in \mathbb{R}^2 would each exert mg Newtons to retain the object against gravity).

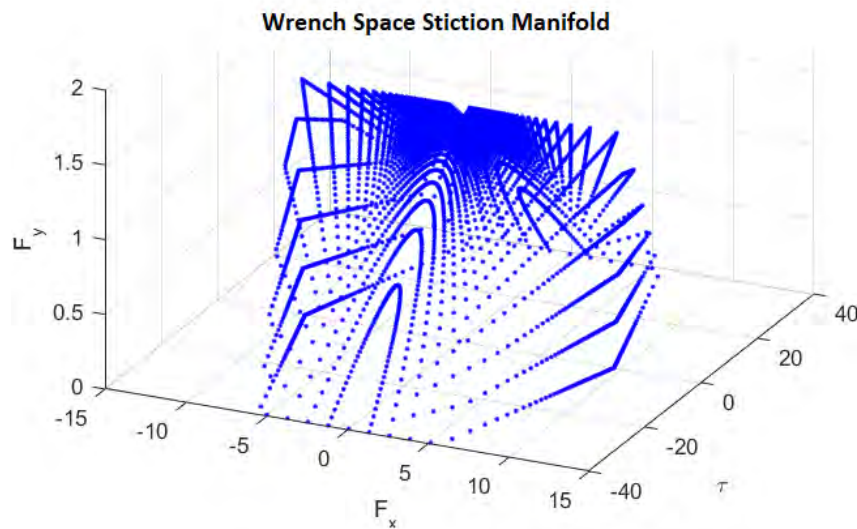


Figure 5.5: Deterministic wrench space stiction manifold for example objects in \mathbb{R}^2 configuration space. Points are generated across θ and γ to aid in visualization, while true representation is derived from a uniform sampling approximation algorithm

Due to this dependence on the net weight of the objects above each, the order of objects for which the stiction manifold is calculated must progress hierarchically down the directed wrench graph in height. For the root node of the graph, the net weight will simply be the effect of gravity on its assumed mass, mg . This is taken to be the sum of its own weight, and the vertical component of all forces predicted to be imparted upon it via the directed wrench graph. If there are no other objects detected as adjacent to the object being inspected, it is assumed that the entirety of its weight is supported by some unseen contact beneath its center of mass.

Points on the unit sphere used in sampling of the manifold are precalculated to provide approximate uniformity via a spiral-based sampling method [123], however,

to aid in static visualization of the manifold, the surfaces presented utilize positions equispaced across angle and azimuth in the unit sphere, such that straight lines allow the structure to be resolved. For each of the extraction directions/vectors represented by these points on the unit sphere, the sum of the forces acting on the target object in that orientation produces the distance of the stiction surface from the origin. Figure 5.5 depicts the stiction manifold restraining the central object in the \mathbb{R}^2 case presented in Figure 5.3, using the angle azimuth visualisation.

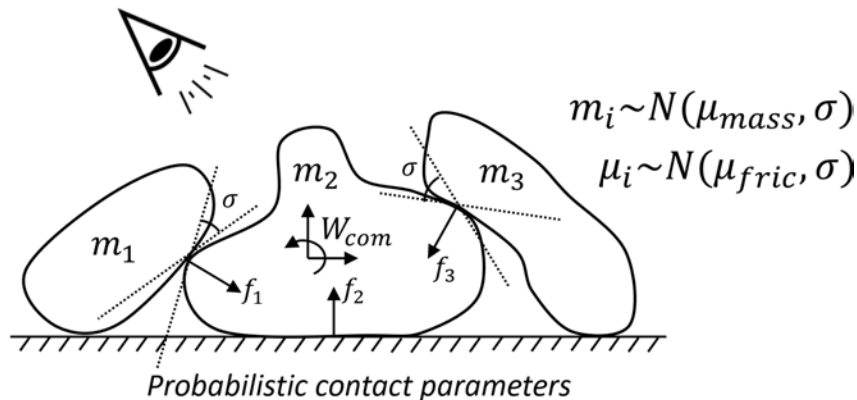


Figure 5.6: Example probabilistic object support configuration in \mathbb{R}^2 , where parameters such as mass, friction coefficient, and contact angle are normally distributed.

5.2 Gaussian Process Implicit Surface Representation

The stiction manifold presented in the previous section assumed complete knowledge of the parameters that define it. However, when employing exteroception in previously unseen environments there is significant uncertainty in the estimation of any of these parameters, such as orientation of contacts, as well as object mass and friction coefficient, as depicted in Figure 5.6. To capture this, the parameters furnished by exteroception are perturbed about a normal distribution during the calculation of wrench magnitude for each of the sampled direction points on the unit sphere. This produces a distribution of points over the manifold in wrench space describing the forces necessary to dislodge the target object from the pile, i.e. to *break stiction*. Applying this perturbed parameter sampling to the \mathbb{R}^2 example produces the set of wrench space data points seen in Figure 5.5.

The structure described in Section 2.4, that of a Gaussian Process Implicit Surface, may then be applied to infer a mean and variance across this manifold. It allows an estimate of the minimum magnitude wrench that would dislodge the target object from its surroundings, through search for the point on the GPIS closest to the origin. An example of such a minimum magnitude vector on the GPIS for the \mathbb{R}^2 three

object configuration space representative problem can be seen represented by the red arrow in Figure 5.5.

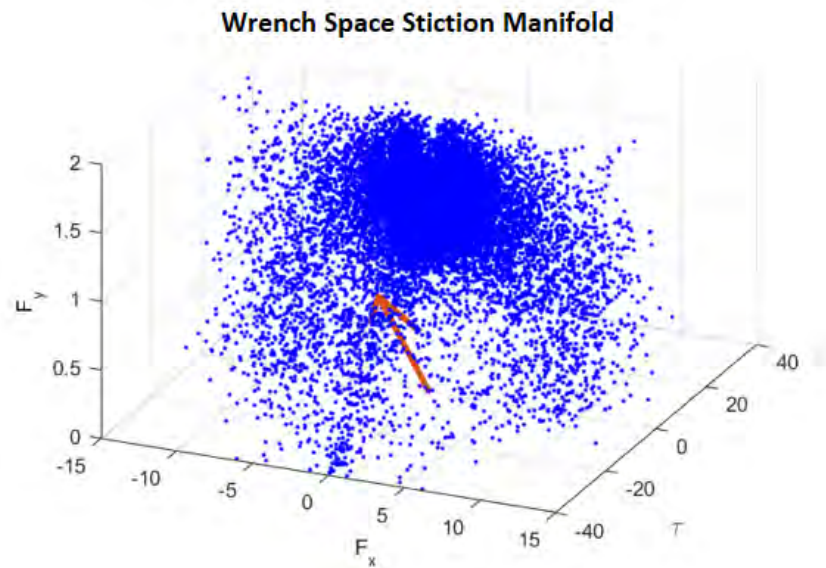


Figure 5.7: Wrench space of \mathbb{R}^2 example with contact parameters perturbed about normal distributions, and minimum magnitude extraction vector identified by red arrow.

5.3 Parameter Estimation

This section proposes means of estimating the parameters utilized within the formulation. While the representation itself is agnostic of the techniques used to infer these parameters, their selection can have large bearing on the accuracy and stability of the inferred stiction manifold, thus necessitating their discussion.

Object Singulation and Adjacency

The first step in establishing parameters for describing the structure of the pile is to attempt to visually distinguish distinct components within, and their geometric relation to one another. The former of these processes is typically termed *segmentation*, and there are a number of algorithms in existence to address it. While the formulation is posed so as to be agnostic to the specific exteroceptive processing techniques applied to it, for the purpose of this demonstration of the formulation the algorithm selected was that of Locally Convex Connected Patches [127], the output of which can be seen in Figure 5.8. This groups points within the cloud into “supervoxels”, and then compares the normals of adjacent supervoxels to determine regions of convexity within the cloud.

These regions of convexity may then be grouped, producing a pixelwise labeling of object candidates, as well as inferring geometric adjacency of candidates through presence of member points within immediate proximity. An example segmentation and adjacency map of a representative manipulation scene is depicted in Figure 5.8.

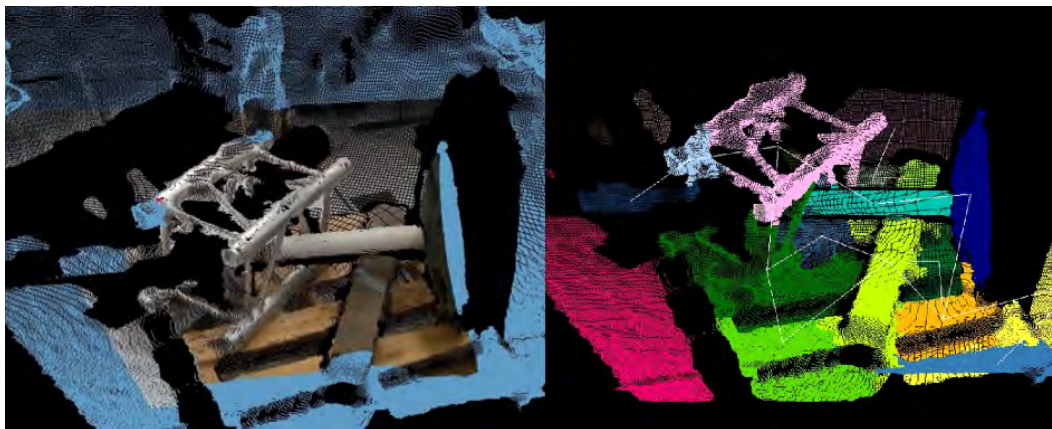


Figure 5.8: Example singulation of objects within task scene using Locally Connected Convex Connected Patches algorithm [127]. Left: Raw RGBD image from robot sensor head. Right: Color coded segmentation of task scene regions into object candidates, with adjacencies between visualized as white lines between centroids.

Center of Volume

Of crucial importance within the formulation is the specification of the Center of Mass (COM) frame (initially assumed taken as center of volume (COV), until detected otherwise), as all other force and contact poses are defined within it. Ascertaining the center of mass of an object from limited view vision can be challenging, and the majority of applications and libraries simply employ a mean of all points within the cloud representing the object. This introduces high sensitivity of the estimated COV to slight perturbations in the viewing angle, or pose of the object, as can be seen depicted in Figure 5.9.

As the orientation of the object changes, the density of points on the two sides within view shifts as a result of the even angular density of depth resolution. This biases the mean pixel value estimate for COV toward the face with most points, potentially causing large drifts in the perceived COM frame if adopted. While this does not present a problem when only processing a single view of the workspace for a single execution of the model, if extended to multiple iterations (i.e. successively removing multiple objects) then the ability to match objects between views to track progress may be confounded by perceived shifts in center of mass.

To address this, the center of volume may be calculated as the centroid of the *convex hull* of the points comprising the visual representation of the object. While more computationally expensive, this reduces sensitivity to slight changes in object-camera pose and lighting, that could influence the density of points across object faces. Computational burden is lessened by the volume (for mass prediction) already having been calculated through determination of the convex hull.

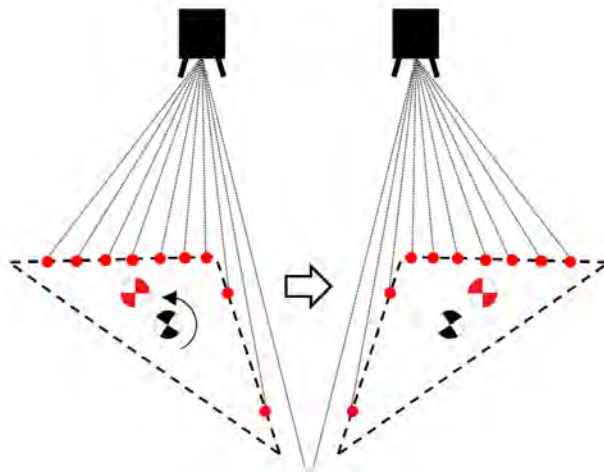


Figure 5.9: Depiction of pointcloud perceived center of mass shift induced by rotation of an object in the field of view in \mathbb{R}^2 configuration space. If the object is rotated about its true center of mass (in black), the density of points shifts from one face to another, resulting in the naïve visual approximation of mean pixel shifting significantly (in red).

5.4 Grasp Candidate Selection

Once the GPIS formulation has been calculated for each of the objects within the pile, the minimum magnitude extraction vector for each may be compared to the force capabilities of the manipulation system intended to interact with them. For the set of objects within the pile that have an extraction vector within the systems capabilities, external criteria may then be employed to judge which is most suitable for an extraction attempt. This will commonly include reachability criteria that define the bounds of the manipulation workspace.

An example real debris pile decomposition task scene, captured with RGBD camera, is depicted as visualized in offline simulation in Figure 5.10 left. With the pipe object laying incident upon the others, though not prominent in height as might be naively used to select amongst objects, the algorithm designates it as a candidate for removal and grasp poses may be planned for it using a geometric planner, as pictured in Figure

5.10 right.

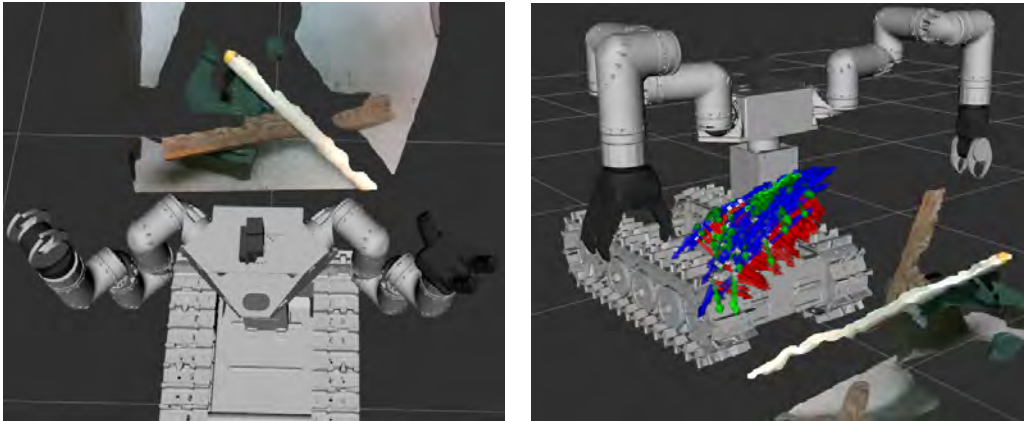


Figure 5.10: Left: Test debris pile scene as captured by RealSense RGBD camera visualized in simulation, comprised of plastic tubing, wood section, and tool crate. Right: Grasp poses generated on selected pipe object using a geometric grasp planner.

Proprioceptive Datapoints

If a candidate grasp-vector extraction pair suggested by the algorithm fails, this provides us with an additional *proprioceptive* datapoint that can be used to refine the accuracy of the wrench stiction representation. An example of this is seen in Figure 5.11, where an extraction attempt along the red extraction vector for the inspected object fails, so the red datapoint is entered in the dataset at the maximum wrench magnitude that was exerted before the attempt aborted. The GPIS representation of the wrench stiction surface may then be recalculated, accounting for this additional datapoint, and a new extraction vector for the object selected. If sufficient extraction vectors are sampled to cause the entire wrench stiction surface to lie at greater magnitude than our system is capable of exerting, then the object is considered immovable, and a new candidate object selected for extraction.

5.5 Summary

This chapter presented a means of representing the forces restraining an object within an unstructured pile as a *wrench space stiction manifold*, to enable a manipulation platform to identify objects that are within its force capability to extract. It then proposed a means of formulating this manifold within the probabilistic structure afforded by a Gaussian Process Implicit Surface, such as to capture uncertainty within the parameters that define the shape and magnitude of the latent force structure within the pile. Approaches for estimating the relevant parameters were investi-

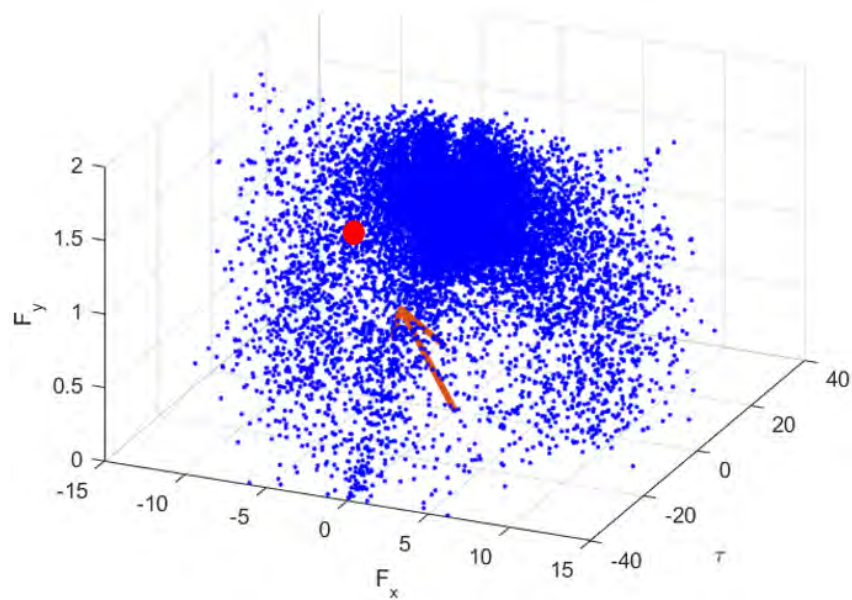


Figure 5.11: Wrench space stiction surface Gaussian Process Implicit Surface representation with proprioceptive datapoint (in red) added after a failed extraction attempt along the red extraction vector for a particular object.

gated, and then criteria suggested for selecting among the object-extraction vector candidates predicted by the formulation to be within its capability.

Chapter 6

CONCLUSIONS AND DISCUSSION

Reactive Discrete Operating Mode Selection

A novel formulation of the prototypical Multi-Armed Bandit theory has been presented, that adds the notion of an action preference and nominal action-value measurements known *a priori*, and is termed the Obedient Multi-Armed Bandit (OMAB) problem. A success metric for the problem, analogous to MAB regret, was formulated and termed disappointment. Existing MAB policies were adapted to the OMAB formulation and order of disappointment growth shown analytically for the case of stationary cost distributions. Utility of the policy was demonstrated within the applications of multi-modal surface mobility, and substrate excavation. Empirical data from mobility experiments was then used in simulation to compare the behavior of the policy against existing MAB policies in the cases of step-change and slow drift non-stationary cost distributions.

The use of longer time-frame simulations proved crucial in investigating the strengths and weaknesses of the proposed PIU policy, as compared to the adaptations of existing MAB policies. One key demonstration of the shortcomings of the algorithm was in the case of stationary cost distributions where the operator commanded mode was non-optimal. The in-built tendency of PIU to optimistically re-sample that mode, as the previously measured action-value decayed below zero, leads it to accumulate disappointment at a higher rate than the stationary policies, given their design causing them to prioritize pursuing the previously inferred optimum. The shallow gradient at which disappointment is accumulated, however, may be considered a worthy price to pay given the improved response produced by PIU in the case of cost distributions shifting; as was the focus of the policy design.

The incremental update aspect of PIU, otherwise known as exponential smoothing, allowed it to rapidly detect a step or drift change in mode efficiencies, as compared to ϵ -greedy or UCB, whose response is delayed by the equal priority given across all prior measurements for each action. This is particularly true in the case of the commanded mode transitioning from off-nominal to nominal, given the λ bias term causing more frequent re-sampling of that mode, the positive side to the trade-off of the optimistic sampling that caused accumulation of disappointment in the

off-nominal stationary case.

On the continuum between the purely reactive mode selection of PIU, and full blown exteroceptive look-ahead planning, the logical progression was to afford the policy some minimal signature from external sensing to motivate mode exploration, and this was developed within the PIU-Exteroceptive Decay policy extension. As demonstrated in the excavation example, this proved capable of detecting distinct step changes in the visual features of the substrate, and re-sampling the commanded mode accordingly. It may, however, struggle to identify less stark or drifting changes in the visual signatures of the environment, as might be correlated with a drift in underlying cost distributions. For that reason, it could be of use to compare signatures over a range of time horizons in order to identify slow change, though this would increase the computational and memory requirements of the algorithm, which at present are minimal.

The PIU and PIU-ED policies will shortly undergo longer distance experimental testing for the mobility application through utilization on impending field trials with the full RoboSimian platform in Europa analog terrain, such as pictured in Figure 6.1.



Figure 6.1: RoboSimian wheel-on-leg platform deployed on Europa analog terrain during field trial.

Future extensions of this work are expected to include learning mappings from measured efficacy to discernible visual parameters to allow forward planning with knowledge gained in situ during traversal. This will effectively further bridge the gap between the purely reactive selection of the PIU policy, and look-ahead planning

where full terrain classification at distance might be expected. Also of interest is formulating a means to update the nominal cost measure for each action, \bar{c}_i , over time so as to more accurately capture monotonic efficacy change caused by changes in the system itself, a prime example being actuator or toolbit degradation.

Semantic Task Outcome Classification

In this chapter, a model to prescribe semantic meaning to task outcome modes autonomously discovered in simulation was presented. Synthetically generated datasets allowed an instructor instantiating the model for a given task to discover symbolic failure modes, and assign corrective actions that an autonomous agent should undertake if they are encountered. Two varieties of visual classifier were presented to allow the agent to infer semantic meaning of a task outcome, and the model applied to three representative manipulation tasks with simulated and real datasets used for validation.

One key critique of this work might be that the tasks demonstrated may lend themselves well to employing pose recovery methods on the objects in question, for which a large variety of algorithm exist within the computer vision community. An autonomous agent, suitably equipped with knowledge of these poses, could then apply the same explicit geometric rules that were used to distinguish the failure modes in simulation to determine a suitable corrective action. While this is particularly true of the cuboid placement task, the power of this method is that it should transfer to varieties of tasks where deep networks excel at classification, while concrete geometric representations are difficult to attain. A prime example of this comes from manipulation of deformable objects, such as the folding of cloth (as seen in simulation in Figure 6.2), or transfer and placement of liquids. Future demonstration of this method might be well served to illustrate applicability to such domains, where traditional discrete geometric representations of a task space struggle.

In addition, as previously intimated, if a task execution has resulted in failure then that may be taken as evidence that the prior information used to plan that attempt was flawed. Adopting a distinct task outcome evaluation modality offers a means of introspection for the agent, allowing it to identify failures or inaccuracies in its modelling of the task. Future work might seek to leverage this knowledge to allow an agent to update its assumptions of the environment in response to the failure modes identified, to improve chances of the specified corrective action succeeding in turn.

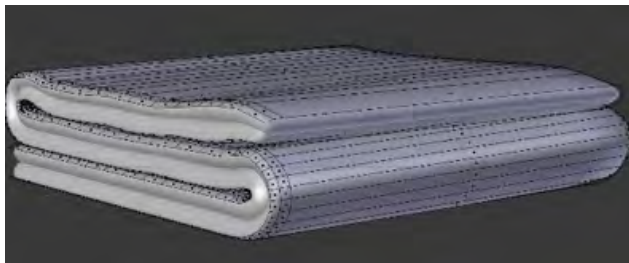


Figure 6.2: Example of manipulation task with higher visual complexity, the folding of a piece of cloth. The outcome modes of such a task could not easily be derived from pure pose recovery methods due to the deformable nature of the manipuland.

Forceful Manipulation in Clutter

This chapter presented a means of representing the forces restraining an object within an unstructured pile as a *wrench space stiction manifold*, to enable a manipulation platform to identify objects that are within its force capability to extract. It then proposed a means of formulating this manifold within the probabilistic structure afforded by a Gaussian Process Implicit Surface, such as to capture uncertainty within the parameters that define the shape and magnitude of the latent force structure within the pile. Approaches for estimating the relevant parameters were investigated, and then criteria suggested for selecting among the object-extraction vector candidates predicted by the formulation to be within its capability.

While the underlying formulation appears capable of representing the uncertainty inherent in exteroceptive sensing of previously unseen objects, its success in predicting grasps can only be as good as the techniques employed to derive that information. One of the key challenges of the problem uncovered by this investigation was that of estimating the pose of inter-object contacts. The frequent presence of occlusions blocking view of the regions of objects that impose an incident force upon others made sole reliance on the object adjacencies furnished by the exteroceptive algorithm limited in the range of configurations they could address. One potential means of addressing this may be adoption of *object completion* methods, such as presented by Tulsiani et al. [134], where the geometry of an object may be extrapolated into the occluded or otherwise unseen space through intelligent selection of mirroring planes. This may also aid in the estimation of object volume, given the present use of the convex hull of points from an object visible within a partial RGBD view is inherently limited in its ability to capture the spatial bounds of the object.

The use of a Newtonian formulation of the restraining wrenches, employing upper bounds on value estimates where applicable, results in a highly conservative

prediction of the forces necessary to extract each candidate object. The advantage to this formulation was its relative ease of computation, the restraining manifold simply being the sum of force constraints imposed by all incident objects, though it inevitably results in some objects being bound that may in fact be viable. A strategy to address this, and a suitable avenue for future work, would be to replace the upper bound Newtonian formulation with Lagrangian mechanics and applying the principle of virtual work; though this may require dynamic programming in order to be implemented in generality within software.

While the formulation presents how to incorporate data points furnished by proprioceptive probing, there are many other ways information gained through extraction attempts could be leveraged. In the instance that there are multiple similarly shaped and/or colored objects within the workspace, task execution on one may be used to infer mass or friction properties of the others, for example. This could also be incorporated with in-hand proprioceptive inference, such as presented by Burkhardt et al. [24], such that once an object is extracted, its inertial properties are resolved, so as to improve knowledge of similar objects within the structure.

One complication of relying on fallible exteroceptive processing algorithms, is that the singulation of object candidates within the force structure may be flawed. The probabilistic nature of the formulation may allow easy extension to incorporate a representation of the belief that a given inter-object contact, or edge in the directed wrench graph, may in fact describe a rigid connection between two bodies. Similarly, objects that are assumed to be granular components of the structure may, in fact, be comprised of multiple distinct items. A means of representing confidence for or against such a condition might be more challenging to parameterize, however, could perhaps be affected by expanding the level of exteroceptive sensing to include tracking of objects while force is applied to them. Relative motion between presumed connected components of an object might then be used to perceive their distinction.

BIBLIOGRAPHY

- [1] Rajeev Agrawal. Sample Mean Based Index Policies with $O(\log n)$ Regret for the Multi-Armed Bandit Problem Author (s): Rajeev Agrawal Source : Advances in Applied Probability , Vol . 27 , No . 4 (Dec . , 1995), pp . 1054-1078 Published by : Applied Probability Trust. *Advances in Applied Probability*, 27(4):1054–1078, 1995.
- [2] R Alami, T Siméon, and J.-P. Laumond. A Geometrical Approach to Planning Manipulation Tasks. *Proceedings International Symposium on Robotics Research*, (3):113–119, 1989.
- [3] Afef Ben Hadj Alaya-Feki, Eric Moulines, and Alain Lecornec. Dynamic spectrum access with non-stationary multi-armed bandit. *IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC*, pages 416–420, 2008. doi: 10.1109/SPAWC.2008.4641641.
- [4] Laura Antanas, Plinio Moreno, Marion Neumann, Rui Pimentel de Figueiredo, Kristian Kersting, José Santos-Victor, and Luc De Raedt. High-level Reasoning and Low-level Learning for Grasping: A Probabilistic Logic Pipeline. 2014. URL <http://arxiv.org/abs/1411.1108>.
- [5] Christopher G. Atkeson, Chae H. An, and John M. Hollerbach. Rigid Body Load Identification for Manipulators. *Proceedings of the IEEE Conference on Decision and Control*, (December):996–1002, 1985. ISSN 01912216. doi: 10.1109/cdc.1985.268649.
- [6] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256, 2002. ISSN 10495258.
- [7] B Y Leonard E Baum, T E D Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1): 164–171, 1970.
- [8] Leonard E. Baum and Ted Petrie. Statistical inference for finite state markov chains. *The Annals of Mathematical Statistics*, 37:1554–1563, 1966. URL https://projecteuclid.org/download/pdf_1/euclid.aoms/1177699147.
- [9] Sven Behnke. *Hierarchical neural networks for image interpretation*, volume 2766. Springer, 2003.
- [10] Richard Bellman. *Dynamic Programming*. Princeton University Press, 1957.

- [11] Richard Bellman. A Markovian Decision Process. *Journal of mathematics and mechanics*, 6(4):679–684, 1957. ISSN 0022-2518. doi: 10.1512/iumj.1957.6.56038.
- [12] Dmitry Berenson, Siddhartha S. Srinivasa, Dave Ferguson, and James J. Kuffner. Manipulation planning on constraint manifolds. i:625–632, 2009. doi: 10.1109/robot.2009.5152399.
- [13] Dimitri P Bertsekas and John N Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific Belmont, MA, 1996.
- [14] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. *Proceedings-IEEE International Conference on Robotics and Automation*, 1: 348–353, 2000. ISSN 10504729. doi: 10.1109/ROBOT.2000.844081.
- [15] Jeffrey J. Biesiadecki, Eric T. Baumgartner, Robert G. Bonitz, Brian K. Cooper, Frank R. Hartman, P. Christopher Leger, Mark W. Maimone, Scott A. Maxwell, Ashitey Trebi-Ollennu, Edward W. Tunstel, and John R. Wright. Mars Exploration Rover surface operations: Driving opportunity at Meridiani Planum. *IEEE Robotics and Automation Magazine*, 13(2):63–71, 2006. ISSN 10709932. doi: 10.1109/MRA.2006.1638017.
- [16] M. J. Bishop and E. A. Thompson. Maximum likelihood alignment of DNA sequences. *Journal of Molecular Biology*, 190(2):159–165, 1986. ISSN 00222836. doi: 10.1016/0022-2836(86)90289-5.
- [17] Jeannette Bohg and Danica Kragic. Learning grasping points with shape context. *Robotics and Autonomous Systems*, 58(4):362–377, 2010.
- [18] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasp synthesis-A survey. *IEEE Transactions on Robotics*, 30(2): 289–309, 2014. ISSN 15523098. doi: 10.1109/TRO.2013.2289018.
- [19] Gary M. Bone, Andrew Lambert, and Mark Edwards. Automated modeling and robotic grasping of unknown three-dimensional objects. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 292–298, 2008. ISSN 10504729. doi: 10.1109/ROBOT.2008.4543223.
- [20] Abdeslam Boularias, J Andrew Bagnell, and Anthony Stentz. Learning to Manipulate Unknown Objects in Clutter by Reinforcement. *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence Learning*, pages 1336–1342, 2015.
- [21] Djallel Bouneffouf and Raphael Féraud. Multi-armed bandit problem with known trend. *Neurocomputing*, 205:16–21, 2016. ISSN 18728286. doi: 10.1016/j.neucom.2016.02.052.

- [22] Robert H Brown, Roger N Clark, Bonnie J Buratti, Dale P Cruikshank, Jason W Barnes, Rachel ME Mastrapa, J Bauer, S Newman, T Momary, and KH Baines. Composition and physical properties of Enceladus' surface. *Science*, 311(5766):1425–1428, 2006.
- [23] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in multi-armed bandits problems. In *International Conference on Algorithmic learning theory*, pages 23–37, 2009. ISBN 3642044131. doi: 10.1007/978-3-642-04414-4{_}7.
- [24] Matt Burkhardt, Sisir Karumanchi, Kyle Edelberg, Joel W Burdick, and Paul Backes. Proprioceptive Inference for Dual-Arm Grasping of Bulky Objects Using RoboSimian. *IEEE International Conference on Robotics and Automation*, pages 4049–4056, 2018.
- [25] Apostolos N. Burnetas and Michael N. Katehakis. Optimal adaptive policies for sequential allocation problems. *Advances in Applied Mathematics*, 17(2): 122–142, 1996. ISSN 01968858. doi: 10.1006/aama.1996.0007.
- [26] Giuseppe Burtini, Jason Loepky, and Ramon Lawrence. Improving online marketing experiments with drifting multi-armed bandits. *ICEIS 2015 - 17th International Conference on Enterprise Information Systems, Proceedings*, 1 (April):630–636, 2015. doi: 10.5220/0005458706300636.
- [27] N Cesa-Bianchi and Paul Fischer. Finite-time regret bounds for the multi-armed bandit problem. *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 100–108, 1998.
- [28] F. Cini, V. Ortenzi, P. Corke, and M. Controzzi. On the choice of grasp type and location when handing over an object. *Science Robotics*, 4(27):1–18, 2019. ISSN 24709476. doi: 10.1126/scirobotics.aau9757.
- [29] Nikolaus Correll, Kostas E. Bekris, Dmitry Berenson, Oliver Brock, Albert Causo, Kris Hauser, Kei Okada, Alberto Rodriguez, Joseph M. Romano, and Peter R. Wurman. Analysis and observations from the first Amazon picking challenge. *IEEE Transactions on Automation Science and Engineering*, 15 (1):172–188, 2018.
- [30] T Cover and P Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967. ISSN 15579654. doi: 10.1109/TIT.1967.1053945.
- [31] Richard Dearden, Nir Friedman, and Stuart Russell. Bayesian Q-learning. *Proceedings of the National Conference on Artificial Intelligence*, pages 761–768, 1998.
- [32] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. (June):248–255, 2010. doi: 10.1109/cvpr.2009.5206848.

- [33] R Detry, J Papon, and L Matthies. Task-oriented grasping with semantic and geometric scene understanding. In *IROS*, 2017. doi: 10.1109/IROS.2017.8206162.
- [34] Renaud Detry, Jeremie Papon, and Larry Matthies. Task-oriented grasping with semantic and geometric scene understanding. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017. ISBN 9781538626825. doi: 10.1109/IROS.2017.8206162.
- [35] Rosen Diankov. *Automated Construction of Planning Knowledge-base*. PhD thesis, 2010. URL http://www.programmingvision.com/rosen_diankov_thesis.pdf.
- [36] Stanimir Dragiev, Marc Toussaint, and Michael Gienger. Gaussian process implicit surfaces for shape estimation and grasping. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 2845–2850, 2011. ISBN 9781612843865. doi: 10.1109/ICRA.2011.5980395.
- [37] Stanimir Dragiev, Marc Toussaint, and Michael Gienger. Uncertainty aware grasping and tactile exploration. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 113–119, 2013. ISSN 10504729. doi: 10.1109/ICRA.2013.6630564.
- [38] Richard Durbin, Sean R Eddy, Anders Krogh, and Graeme Mitchison. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998.
- [39] Ozgur Erkent, Dadhichi Shukla, and Justus Piater. Visual task outcome verification using deep learning. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 9 2017. ISBN 9781538626825. doi: 10.1109/IROS.2017.8206357.
- [40] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Second International Conference on Knowledge Discovery and Data Mining*, KDD'96. AAAI Press, 1996.
- [41] Carlo Ferrari and John Canny. Planning optimal grasps. *Proceedings - IEEE International Conference on Robotics and Automation*, 3(May):2290–2295, 1992. doi: 10.1109/robot.1992.219918.
- [42] Martin a Fischler and Robert C Bolles. 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. doi: 10.1145/358669.358692.

- [43] Kunihiro Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980. ISSN 03401200. doi: 10.1007/BF00344251.
- [44] Aurélien Garivier and Eric Moulines. On Upper-Confidence Bound Policies for Non-Stationary Bandit Problems. 2008. URL <http://arxiv.org/abs/0805.3415>.
- [45] Malik Ghallab, Dana Nau, and Paolo Traverso. *Automated Planning: theory and practice*. Elsevier, 2004.
- [46] James Jerome Gibson. The senses considered as perceptual systems. 1966.
- [47] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
- [48] Michael Gschwandtner, Roland Kwitt, Andreas Uhl, and Wolfgang Pree. BlenSor: Blender Sensor Simulation Toolbox. In *Advances in Visual Computing*, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-24031-7.
- [49] Saurabh Gupta, Ross B Girshick, Pablo Andrés Arbeláez, Jitendra Malik, Pablo Arbel, Pablo Andrés Arbeláez, Jitendra Malik, Pablo Arbel, Pablo Andrés Arbeláez, and Jitendra Malik. Learning Rich Features from {RGB-D} Images for Object Detection and Segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8695(PART 7):345–360, 2014. ISSN 16113349. doi: 10.1007/978-3-319-10584-0{_}23. URL http://dx.doi.org/10.1007/978-3-319-10584-0_23.
- [50] W.R. Hamilton. On the application to dynamics of a general mathematical method previously applied to optics. *British Association Report*, pages 513–518, 1834.
- [51] Marc Hanheide, Moritz Gobelbecker, Graham S. Horn, Andrzej Pronobis, Kristoffer Sjøo, Alper Aydemir, Patric Jensfelt, Charles Gretton, Richard Dearden, Miroslav Janicek, Hendrik Zender, Geert-Jan Jan Kruijff, Nick Hawes, Jeremy L. Wyatt, Moritz Göbelbecker, Graham S. Horn, Andrzej Pronobis, Kristoffer Sjöo, Alper Aydemir, Patric Jensfelt, Charles Gretton, Richard Dearden, Miroslav Janicek, Hendrik Zender, Geert-Jan Jan Kruijff, Nick Hawes, Jeremy L. Wyatt, Moritz Göbelbecker, Graham S. Horn, Andrzej Pronobis, Kristoffer Sjøo, Alper Aydemir, Patric Jensfelt, Charles Gretton, Richard Dearden, Miroslav Janicek, Hendrik Zender, Geert-Jan Jan Kruijff, Nick Hawes, Jeremy L. Wyatt, Moritz Göbelbecker, Graham S. Horn, Andrzej Pronobis, Kristoffer Sjøo, Alper Aydemir, Patric Jensfelt, Charles Gretton, Richard Dearden, Miroslav Janicek, Hendrik Zender, Geert-Jan Jan Kruijff, Nick Hawes, and Jeremy L. Wyatt. Robot task planning and explanation

- in open and uncertain worlds. *Artificial Intelligence*, 247:119–150, 8 2015. ISSN 00043702. doi: 10.1016/j.artint.2015.08.008. URL <http://dx.doi.org/10.1016/j.artint.2015.08.008>.
- [52] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90.
- [53] D. O. Hebb. *The Organization of Behavior; A Neuropsychological Theory*. John Wiley & Sons, 1949. doi: 10.2307/1418888.
- [54] Paul Hebert, Max Bajracharya, Jeremy Ma, Nicolas Hudson, Alper Aydemir, Jason Reid, Charles Bergh, James Borders, Matthew Frost, Michael Hagman, John Leichty, Paul Backes, Brett Kennedy, Paul Karplus, Brian Satzinger, Katie Byl, Krishna Shankar, and Joel Burdick. Mobile manipulation and mobility as manipulation - Design and algorithms of RoboSimian. *Journal of Field Robotics*, 32(2):255–274, 2015. ISSN 15564967. doi: 10.1002/rob.21566.
- [55] Rachel Holladay, Tomás Lozano-Pérez, and Alberto Rodriguez. Force-and-Motion Constrained Planning for Tool Use. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019. URL <https://mcube.mit.edu/tool-use/>.
- [56] Thomas E. Horton, Arpan Chakraborty, and Robert St. Amant. Affordances for robots: A brief survey. *Avant*, 3(2):70–84, 2012. ISSN 20826710.
- [57] D. H. Hubel and T. N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. *The Journal of Physiology*, 195(1):215–243, 1968. ISSN 14697793. doi: 10.1113/jphysiol.1968.sp008455.
- [58] Nicolas Hudson, Thomas Howard, Jeremy Ma, Abhinandan Jain, Max Bajracharya, Steven Myint, Calvin Kuo, Larry Matthies, Paul Backes, Paul Hebert, Thomas Fuchs, and Joel Burdick. End-to-end dexterous manipulation with deliberate interactive estimation. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 2371–2378, 2012. ISSN 10504729. doi: 10.1109/ICRA.2012.6225101.
- [59] Félix Ingrand and Malik Ghallab. Robotics and Artificial Intelligence: a Perspective on Deliberation Functions. *AI Communications*, 27(1):63–80, 2014.
- [60] Carl Gustav Jakob Jacobi. *Vorlesungen über dynamik*. G. Reimer, 1866.
- [61] Frederick Jelinek, Lalit R. Bahl, and Robert L. Mercer. Design of a Linguistic Statistical Decoder for the Recognition of Continuous Speech. *IEEE Transactions on Information Theory*, 21(3):250–256, 1975. ISSN 15579654. doi: 10.1109/TIT.1975.1055384.

- [62] Joseph L. Jones. Robots at the tipping point. *IEEE Robotics and Automation Magazine*, 13(1):76–78, 2006. ISSN 10709932. doi: 10.1109/MRA.2006.1598056.
- [63] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4: 237–285, 1996. ISSN 18196608.
- [64] Dimitrios Kanoulas, Jinh Lee, Darwin G. Caldwell, and Nikos G. Tsagarakis. Center-of-mass-based grasp pose adaptation using 3d range and force/torque sensing. *International Journal of Humanoid Robotics*, 15(4):1–26, 2018. ISSN 02198436. doi: 10.1142/S0219843618500135.
- [65] Sisir Karumanchi, Kyle Edelberg, Ian Baldwin, Jeremy Nash, Brian Satzinger, Jason Reid, Charles Bergh, Chelsea Lau, John Leichty, Kalind Carpenter, Matthew Shekels, Matthew Gildner, David Newill-Smith, Jason Carlton, John Koehler, Tatyana Dobrova, Matthew Frost, Paul Hebert, James Borders, Jeremy Ma, Bertrand Douillard, Krishna Shankar, Katie Byl, Joel Burdick, Paul Backes, and Brett Kennedy. Team robosimian: Semi-autonomous mobile manipulation at the 2015 DARPA robotics challenge finals. *Springer Tracts in Advanced Robotics*, 121:191–235, 2018. ISSN 1610742X. doi: 10.1007/978-3-319-74666-1{_}6.
- [66] Michael N. Katehakis and Herbert Robbins. Sequential choice from several populations. *Proceedings of the National Academy of Sciences of the United States of America*, 92(19):8584–8585, 1995. ISSN 00278424. doi: 10.1073/pnas.92.19.8584.
- [67] Levente Kocsis and Csaba Szepesvari. Bandit Based Monte-Carlo Planning. In *European conference on machine learning*, pages 282–293. Springer, 2006. ISBN 978-3-540-67602-7. doi: 10.1007/3-540-45164-1. URL <http://dblp.uni-trier.de/db/conf/ecml/ecml2000.html#BrazdilS00>.
- [68] David Kortenkamp, Reid Simmons, and Davide Brugali. Robotic Systems Architectures and Programming. In *Springer Handbook of Robotics*, number October, pages 283–306. 2016. ISBN 9783540303015. doi: 10.1007/978-3-540-30301-5.
- [69] Michael C. Koval, Jennifer E. King, Nancy S. Pollard, and Siddhartha S. Srinivasa. Robust trajectory selection for rearrangement planning as a multi-armed bandit problem. *IEEE International Conference on Intelligent Robots and Systems*, 2015-Decem:2678–2685, 2015. ISSN 21530866. doi: 10.1109/IROS.2015.7353743.
- [70] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. ISBN 9781420010749. doi: 10.1201/9781420010749.

- [71] O. B. Kroemer, R. Detry, J. Piater, and J. Peters. Combining active learning and reactive control for robot grasping. *Robotics and Autonomous Systems*, 58(9):1105–1116, 2010. ISSN 09218890. doi: 10.1016/j.robot.2010.06.001. URL <http://dx.doi.org/10.1016/j.robot.2010.06.001>.
- [72] Volker Krüger, Danica Kragic, Aleš Ude, and Christopher Geib. The meaning of action: A review on action recognition and mapping. *Advanced robotics*, 21(13):1473–1501, 2007. URL <https://www.ntchosting.com/encyclopedia/databases/structured-query-language/>.
- [73] Thibault Kruse, Amit Kumar Pandey, Rachid Alami, and Alexandra Kirsch. Human-aware robot navigation: A survey. *Robotics and Autonomous Systems*, 61(12):1726–1743, 2013. ISSN 09218890. doi: 10.1016/j.robot.2013.05.007. URL <http://dx.doi.org/10.1016/j.robot.2013.05.007>.
- [74] Vijay R. Kumar and Kenneth J. Waldron. Force Distribution in Closed Kinematic Chains. *IEEE Journal on Robotics and Automation*, 4(6):657–664, 1988. ISSN 08824967. doi: 10.1109/56.9303.
- [75] T. L. Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985. ISSN 10902074. doi: 10.1016/0196-8858(85)90002-8.
- [76] John Langford and Tong Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. *Advances in Neural Information Processing*, pages 1–8, 2007. URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.143.8000&rep=rep1&type=pdf>.
- [77] Michael Laskey, Jeff Mahler, Zoe McCarthy, Florian T. Pokorny, Sachin Patil, Jur Van Den Berg, Danica Kragic, Pieter Abbeel, and Ken Goldberg. Multi-armed bandit models for 2D grasp planning with uncertainty. *IEEE International Conference on Automation Science and Engineering*, 2015-October: 572–579, 2015. ISSN 21618089. doi: 10.1109/CoASE.2015.7294140.
- [78] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to digit recognition, 1989.
- [79] Yann Lecun, Leon Bottou, Yoshua Bengio, and Patrick Ha. GradientBased Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. ISSN 00189219. doi: 10.1109/5.726791.
- [80] P Chris Leger, Whiteys Trebi-ollennu, John R Wright, Scott A Maxwell, Robert G Bonitz, Jeffrey J Biesiadecki, Frank R Hartman, Brian K Cooper, Eric T Baumgartner, and Mark W Maimone. Mars Exploration Rover Surface Operations : Driving Spirit at Gusev Crater. In *2005 IEEE Conference on Systems, Man and Cybernetics*, pages 1815–1822, 2005.

- [81] Iolanda Leite. Long-term interactions with empathic social robots. *AI Matters*, 5(1):13–15, 2015. doi: 10.1145/2735392.2735397.
- [82] Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. In *The International Journal of Robotics Research*, volume 37, page 421–436. 2017. ISBN 9783319501154. doi: 10.1007/978-3-319-50115-4_{_}16. URL http://link.springer.com/10.1007/978-3-319-50115-4_16.
- [83] Miao Li, Kaiyu Hang, Danica Kragic, and Aude Billard. Dexterous grasping under shape uncertainty. *Robotics and Autonomous Systems*, 75:352–364, 2016. ISSN 09218890. doi: 10.1016/j.robot.2015.09.008. URL <http://dx.doi.org/10.1016/j.robot.2015.09.008>.
- [84] Tomas Lozano-Perez, Joseph L. Jones, Emmanuel Mazer, Patrick A. O’Donnell, and W. Erick L. Grimson. Handey: a Robot System That Recognizes, Plans, and Manipulates. pages 843–849, 1987. doi: 10.1109/robot.1987.1087847.
- [85] Tyler Lu, Dávid Pál, and Martin Pál. Contextual Multi-Armed Bandits. *Thirteenth International Conference on Artificial Intelligence and Statistics*, 9:485–492, 2010. ISSN 15324435. URL <http://proceedings.mlr.press/v9/lu10a/lu10a.pdf>
<http://www.jmlr.org/proceedings/papers/v9/lu10a/lu10a.pdf>
<http://www.ualberta.ca/~dpal/papers/clicks/lipschitz-clicks.pdf>.
- [86] Baerbel K Lucchitta and Laurence A Soderblom. The geology of Europa. In *Satellites of Jupiter*, pages 521–555, 1982.
- [87] Jeffrey Mahler, Matthew Matl, Xinyu Liu, Albert Li, David Gealy, and Ken Goldberg. Dex-Net 3.0: Computing Robust Robot Suction Grasp Targets in Point Clouds using a New Analytic Model and Deep Learning. *arXiv preprint arXiv:1709.06670*, 2017.
- [88] Hou Brian Roderick Melrose Laskey Michael Aubry Mathieu Kohlhoff Kai Kroeger Torsten Kuffner James Goldberg Ken Mahler Jeffrey, Pokorny Florian T, Jeffrey Mahler, Florian T Pokorny, Brian Hou, Melrose Roderick, Michael Laskey, Mathieu Aubry, Kai Kohlhoff, Torsten Kroger, James Kuffner, and Ken Goldberg. Dex-Net 1.0: A Cloud-Based Network of 3D Objects for Robust Grasp Planning Using a Multi-Armed Bandit Model with Correlated Rewards. *Proceedings - IEEE International Conference on Robotics and Automation*, 0:1957–1964, 2016. ISSN 10504729. doi: 10.1109/ICRA.2016.7487342. URL <http://goldberg.berkeley.edu/pubs/icra16-submitted-Dex-Net.pdf>.

- [89] A. A. Markov. An example of statistical investigation of the text eugene onegin concerning the connection of samples in chains. *Science in Context*, 19(4):591–600, 2006. ISSN 02698897. doi: 10.1017/S0269889706001074.
- [90] Andrei Andreevich Markov. The theory of algorithms. *Trudy Matematicheskogo Instituta Imeni VA Steklova*, 42:3–375, 1954.
- [91] Pyry Matikainen, P. Michael Furlong, Rahul Sukthankar, and Martial Hebert. Multi-armed recommendation bandits for selecting state machine policies for robotic systems. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 4545–4551, 2013. ISSN 10504729. doi: 10.1109/ICRA.2013.6631223.
- [92] Larry Matthies, Daniel Helmick, and Pietro Perona. Learning and Prediction of Slip from Visual Information. *Journal of Field Robotics*, 24(3):205–231, 2007. doi: 10.1002/rob.
- [93] Christopher P. McKay, Carolyn C. Porco, Travis Altheide, Wanda L. Davis, and Timothy A. Kral. The possible origin and persistence of life on enceladus and detection of biomarkers in the plume. *Astrobiology*, 8(5):909–919, 2008. ISSN 15311074. doi: 10.1089/ast.2008.0265.
- [94] Andrew T. Miller and Peter K. Allen. Graspit: A versatile simulator for robotic grasping. *IEEE Robotics and Automation Magazine*, 11(4):110–122, 2004. ISSN 10709932. doi: 10.1109/MRA.2004.1371616.
- [95] B. Mishra, J. T. Schwartz, and M. Sharir. On the existence and synthesis of multifinger positive grips. *Algorithmica*, 2(1-4):541–558, 1987. ISSN 01784617. doi: 10.1007/BF01840373.
- [96] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with Deep Reinforcement Learning. pages 1–9, 2013. URL <http://arxiv.org/abs/1312.5602>.
- [97] Bogdan Moldovan, Plinio Moreno, Martijn Van Otterlo, José Santos-Victor, and Luc De Raedt. Learning relational affordance models for robots in multi-object manipulation tasks. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 4373–4378. IEEE, 2012. ISBN 9781467314039. doi: 10.1109/ICRA.2012.6225042.
- [98] Richard M Murray, Zexiang Li, and S Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press (1700), 1994. ISBN 9780849379819. doi: 10.1.1.169.3957. URL <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:A+Mathematical+Introduction+to+Robotic+Manipulation#0>.

- [99] Hai Nguyen and Charles C. Kemp. Autonomously learning to visually detect where manipulation will succeed. *Autonomous Robots*, 36(1-2):137–152, 2014. ISSN 09295593. doi: 10.1007/s10514-013-9363-y.
- [100] Van-Duc Nguyen. Constructing force-closure grasps. *The International Journal of Robotics Research*, 7(3):3–16, 1988.
- [101] Simon Ottenhaus, Martin Miller, David Schiebener, Nikolaus Vahrenkamp, and Tamim Asfour. Local implicit surface estimation for haptic exploration. *IEEE-RAS International Conference on Humanoid Robots*, pages 850–856, 2016. ISSN 21640580. doi: 10.1109/HUMANOIDS.2016.7803372.
- [102] Sandeep Pandey, Deepayan Chakrabarti, and Deepak Agarwal. Multi-armed bandit problems with dependent arms. *ACM International Conference Proceeding Series*, 227:721–728, 2007. doi: 10.1145/1273496.1273587.
- [103] Robert Paolini, Alberto Rodriguez, Siddhartha S. Srinivasa, and Matthew T. Mason. A data-driven statistical framework for post-grasp manipulation. *International Journal of Robotics Research*, 33(4):600–615, 2014. ISSN 17413176. doi: 10.1177/0278364913507756.
- [104] Jeremie Papon and Markus Schoeler. Semantic pose using deep networks trained on synthetic RGB-D. *Proceedings of the IEEE International Conference on Computer Vision*, 2015 Inter:774–782, 2015. ISSN 15505499. doi: 10.1109/ICCV.2015.95.
- [105] Peter Pastor, Mrinal Kalakrishnan, Sachin Chitta, Evangelos Theodorou, and Stefan Schaal. Skill learning and task outcome prediction for manipulation. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011.
- [106] Ola Pettersson. Execution monitoring in robotics: A survey. *Robotics and Autonomous Systems*, 53(2):73–88, 2005. ISSN 09218890. doi: 10.1016/j.robot.2005.09.004.
- [107] J. Pile, G. B. Wanna, and N. Simaan. Robot-assisted perception augmentation for online detection of insertion failure during cochlear implant surgery. *Robotica*, 35(7):1598–1615, 2017. ISSN 14698668. doi: 10.1017/S0263574716000333.
- [108] G Pini, A Brutschy, G Francesca, M Dorigo, and M Birattari. Multi-armed Bandit Formulation of the Task Partitioning Problem in Swarm Robotics. In *International Conference on Swarm Intelligence*, number May, pages 109–120, 2012.
- [109] Lawrence R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989. ISSN 15582256. doi: 10.1109/5.18626.

- [110] Carl Edward Rasmussen. Gaussian Processes in machine learning. In *Summer School on Machine Learning*, pages 63–71, 2003. doi: 10.1007/978-3-540-28650-9{_}4.
- [111] William Reid, Gareth Meirion-griffith, Sisir Karumanchi, Blair Emanuel, Joseph Bowkett, and Michael Garrett. Actively Articulated Wheel-on-Limb Mobility for Traversing Europa Analogue Terrain. In *Field and Service Robotics*. Springer, 2019.
- [112] William Reid, Blair Emanuel, Brendan Chamberlain-Simon, Sisir Karumanchi, and Gareth Meirion-griffith. Mobility Mode Evaluation of a Wheel-on-Limb Rover on Glacial Ice Analogous to Europa Terrain. In *2020 IEEE Aerospace Conference*, 2020.
- [113] F Reuleaux. Theoretische Kinematic, 1875. Translated as *Kinematics of Machinery*, 1963.
- [114] Giacomo Rizzolatti, Leonardo Fogassi, and Vittorio Gallese. Neurophysiological mechanisms and imitation of action. 2(September):1–10, 2001.
- [115] Th Roatsch, M. Wählisch, B. Giese, A. Hoffmeister, K. D. Matz, F. Scholten, A. Kuhn, R. Wagner, G. Neukum, P. Helfenstein, and C. Porco. High-resolution Enceladus atlas derived from Cassini-ISS images. *Planetary and Space Science*, 56(1):109–116, 2008. ISSN 00320633. doi: 10.1016/j.pss.2007.03.014.
- [116] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958. ISSN 0033295X. doi: 10.1037/h0042519.
- [117] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors David. *Neture*, 114(2):533–536, 1986. ISSN 02632241. doi: 10.1016/j.measurement.2017.09.025.
- [118] G A Rummery and M Niranjan. Online {Q}-Learning using Connectionist Systems. 1994.
- [119] J. K. Salisbury and B. Roth. Kinematic and force analysis of articulated mechanical hands. *Journal of Mechanical Design, Transactions of the ASME*, 105(1):34–41, 1983. ISSN 10500472. doi: 10.1115/1.3267342.
- [120] A. L. Samuel. Some studies in machine learning using the game of checkers. II-Recent progress. *Annual Review in Automatic Programming*, 6(PART 1): 1–36, 1969. ISSN 00664138. doi: 10.1016/0066-4138(69)90004-4.
- [121] Akanksha Saran, Branka Lakic, Srinjoy Majumdar, Juergen Hess, and Scott Niekum. Viewpoint selection for visual failure detection. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2017-Septe, pages 5437–5444. IEEE, 2017. ISBN 9781538626825. doi: 10.1109/IROS.2017.8206439.

- [122] Ashutosh Saxena, Justin Driemeyer, and Andrew Y. Ng. Robotic grasping of novel objects using vision. *International Journal of Robotics Research*, 27(2):157–173, 2008. ISSN 02783649. doi: 10.1177/0278364907087172.
- [123] Anton Semechko. S² Sampling Toolbox. URL <https://github.com/AntonSemechko/S2-Sampling-Toolbox>.
- [124] Jane Shi, Glenn Jimmerson, Tom Pearson, and Roland Menassa. Levels of human and robot collaboration for automotive manufacturing. *Performance Metrics for Intelligent Systems (PerMIS) Workshop*, pages 95–100, 2012. doi: 10.1145/2393091.2393111.
- [125] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. *31st International Conference on Machine Learning, ICML 2014*, 1:605–619, 2014.
- [126] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. pages 1–14, 2014. ISSN 09505849. doi: 10.1016/j.infsof.2008.09.005. URL <http://arxiv.org/abs/1409.1556>.
- [127] Simon Christoph Stein, Markus Schoeler, Jeremie Papon, and Florentin Worgotter. Object partitioning using local convexity. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 304–311, 2014. ISSN 10636919. doi: 10.1109/CVPR.2014.46.
- [128] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press, 2011.
- [129] Marvin Teichmann, Michael Weber, Marius Zoellner, Roberto Cipolla, and Raquel Urtasun. MultiNet: Real-time Joint Semantic Reasoning for Autonomous Driving. *arXiv preprint arXiv:1612.07695*, 2016. ISSN 10636919. doi: 10.1109/CVPR.2012.6248074. URL <http://arxiv.org/abs/1612.07695>.
- [130] Sebastian Thrun. Monte carlo pomdps. In *Advances in neural information processing systems*, pages 1064–1070, 2000.
- [131] Michel Tokic. Adaptive ϵ -greedy exploration in reinforcement learning based on value differences. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6359 LNAI:203–210, 2010. ISSN 03029743. doi: 10.1007/978-3-642-16111-7{_}23.
- [132] Long Tran-Thanh, Archie Chapman, Enrique Munoz De Cote, Alex Rogers, and Nicholas R. Jennings. E-First Policies for Budget-Limited Multi-Armed Bandits. *Proceedings of the National Conference on Artificial Intelligence*, 2:1211–1216, 2010.

- [133] Jeffrey C. Trinkle. On the Stability and Instantaneous Velocity of Grasped Frictionless Objects. *IEEE Transactions on Robotics and Automation*, 8(5): 560–572, 1992. ISSN 1042296X. doi: 10.1109/70.163781.
- [134] Shubham Tulsiani, Abhishek Kar, and Qixing Huang. Shape and Symmetry Induction for 3D Objects. (Section 3).
- [135] Greg Turk and James F. O’Brien. Shape transformation using variational implicit functions. *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1999*, pages 335–342, 1999. doi: 10.1145/311535.311580.
- [136] Swaroop Vattam, Matthew Klenk, Matthew Molineaux, and David W Aha. Breadth of Approaches to Goal Reasoning : A Research Survey. *Goal Reasoning: Papers from the ACS Workshop*, page 111, 2013.
- [137] Joann’es Vermorel and Mehryar Mohri. Empirical Evaluation of Multi-Armed Bandit Algorithms. *European conference on machine learning*, pages 437–448, 2005.
- [138] Andrew J Viterbi. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, 1967. doi: 10.1142/9789814287517{_}0004.
- [139] Richard Volpe. Rover functional autonomy development for the mars mobile science laboratory. *IEEE Aerospace Conference Proceedings*, 2:643–652, 2003. ISSN 1095323X. doi: 10.1109/AERO.2003.1235474.
- [140] Johanna Wallén. The history of the industrial robot. *Linkopings universitet*, page 18, 2008. URL <http://liu.diva-portal.org/smash/get/diva2:316930/FULLTEXT01.pdf>.
- [141] Chih Chun Wang, Sanjeev R. Kulkarni, and H. Vincent Poor. Bandit problems with side observations. *IEEE Transactions on Automatic Control*, 50(3):338–355, 2005. ISSN 00189286. doi: 10.1109/TAC.2005.844079.
- [142] Christopher Watkins. *Learning from delayed rewards*. PhD thesis, King’s College, Cambridge, 1989.
- [143] Christopher Watkins and Peter Dayan. Q-Learning. *Machine learning*, 8 (3-4):279–292, 1992. doi: 10.4018/978-1-59140-993-9.ch026.
- [144] Jonathan Weisz and Peter K. Allen. Pose error robust grasping from contact wrench space metrics. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 557–562, 2012. ISSN 10504729. doi: 10.1109/ICRA.2012.6224697.
- [145] Paul Werbos. Advanced forecasting methods for global crisis warning and models of intelligence. *General System Yearbook*, pages 25–38, 1977.

- [146] Paul J. Werbos. Building and Understanding Adaptive Systems: A Statistical/Numerical Approach to Factory Automation and Brain Research. *IEEE Transactions on Systems, Man and Cybernetics*, 17(1):7–20, 1987. ISSN 21682909. doi: 10.1109/TSMC.1987.289329.
- [147] Christopher K I Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.
- [148] Ronald J. Williams. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning*, 8(3):229–256, 1992. ISSN 15730565. doi: 10.1023/A:1022672621406.
- [149] Andy Zeng, Shuran Song, Kuan Ting Yu, Elliott Donlon, Francois R. Hogan, Maria Bauza, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, Nima Fazeli, Ferran Alet, Nikhil Chavan Dafle, Rachel Holladay, Isabella Morona, Prem Qu Nair, Druck Green, Ian Taylor, Weber Liu, Thomas Funkhouser, and Alberto Rodriguez. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. *International Journal of Robotics Research*, pages 1–16, 2019. ISSN 17413176. doi: 10.1177/0278364919868017.
- [150] Li Zhang and Jeffrey C. Trinkle. The application of particle filtering to grasping acquisition with visual occlusion and tactile sensing. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 3805–3812, 2012. ISSN 10504729. doi: 10.1109/ICRA.2012.6225125.
- [151] Li Zhou. A Survey on Contextual Multi-armed Bandits. 2015. URL <http://arxiv.org/abs/1508.03326>.

Appendix A

CALCULATIONS

A.1 Actuator Power Consumption Estimation

Energy consumption of the SURROGATE platform during the excavation experiments utilized a 7 DOF arm constructed of RoboSimian actuators, and a 3 joint CamHand to actuate the multi-tool. The CamHand supplies live current and voltage measurements across all three actuators, so power consumption can be calculated directly.

The Elmo motor controllers used within the RoboSimian actuators only synchronously report winding current, and bus voltage, which, being from different segments of the power electronics, may not be used to produce an instantaneous measure of power consumed at the actuator level. Knowledge of the winding constants of the motor, however, can be used to produce an estimate of the winding voltage as a function of rotation rate, starting from Kirchhoff's Voltage Law as seen in Equation A.1.

R_w = Winding resistance (Ohms)

K_e = Back EMF constant (Voltage · second / rad)

ω = Rotation rate (rad / second)

$$V_w = I_w R_w + \omega K_e , \quad (\text{A.1a})$$

$$P = I_w V_w . \quad (\text{A.1b})$$

The back EMF constant can be ascertained by virtue of being numerically identical to the torque constant of the motor as can be shown through conservation of energy where

K_t = Torque constant (Newton · meters / Amps)

$$P_{elec}^{in} = P_{mech}^{out} + P_{losses} , \quad (A.2a)$$

$$P_{elec}^{in} = I_w^2 R_w + I_w \omega K_e , \quad (A.2b)$$

$$P_{mech}^{in} = T \omega = I_w \omega K_t , \quad (A.2c)$$

$$P_{losses} = I_w^2 R_w , \quad (A.2d)$$

$$\implies I_w^2 R_w + I_w \omega K_e = I_w \omega K_t + I_w^2 R_w , \quad (A.2e)$$

$$\implies K_e = K_t . \quad (A.2f)$$

For example, the measured torque constant for the 200V winding RoboSimian actuators is given as 0.402 Nm/A, meaning the back EMF constant for the motor is 0.402 Vs/rad. Combined with a measured winding resistance of 3.9 Ω , this produces a power estimate of:

$$P_{elec} = 3.9 I_w^2 + 0.402 \omega I_w . \quad (A.3)$$

The accuracy of the power consumption estimate for the actuators was assessed through comparison to the net power draw of the entire RoboSimian platform, as logged by the power supply, with “hotel” costs from computers and ancillary electronics subtracted. Figure A.1 shows a plot of this comparison.

A.2 Wrench Reactive End Effector Deflection

Compliance of the end effector trajectory during a controlled motion is specified per axis for translation and rotation. For ease of operator use, the wrench upon which the compliance acts is defined in an "End effector Centric, Robot Aligned" (ecra) frame.

Deflection is calculated from the change in wrench experienced by the wrist mounted force torque sensor over the course of the motion; this is achieved by taring the wrench against that measured upon the controller being engaged. Exponential smoothing is used to filter the force torque wrench measurements and produce a damped deflection response, with smoothing factor $\alpha = 0.01$ in the 250Hz control

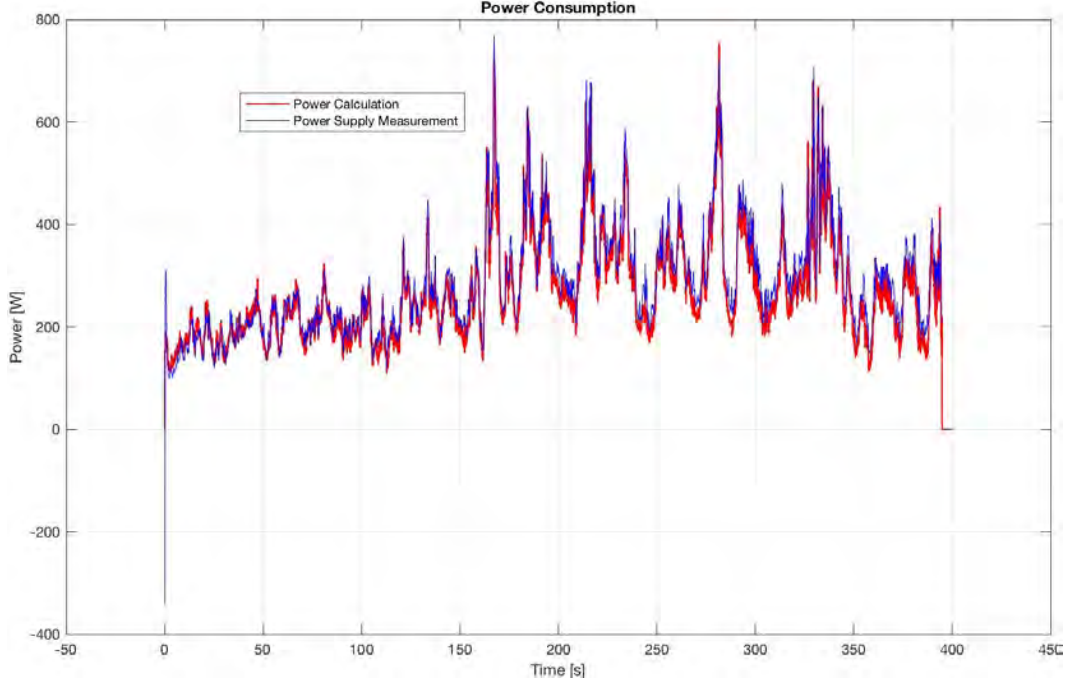


Figure A.1: Comparison of sum of individual actuator power consumption estimates with log of net power supplied to entire system, with hotel costs subtracted.

loop, where W_{ft}^{filt} is then used to calculate instantaneous deflection.

$$W_{ft}^{tare} = W_{ft}^{inst} - W_{ft}^{init} \quad (\text{A.4a})$$

$$W_{ft}^{filt} = (1 - \alpha)W_{ft}^{inst} + \alpha W_{ft}^{tare} \quad (\text{A.4b})$$

The adjoint of the homogeneous transform between two frames allows conversion of the dual space of wrenches between those frames, as given by Equation A.7 [98], which brings the wrench measured at the force torque sensor into the end effector frame, within which we must operate within payload limits of the fitted tool, as given by Equation A.8.

$$g = \begin{bmatrix} R & \bar{p} \\ 0 & 1 \end{bmatrix}, \quad (\text{A.5})$$

$$Ad_g = \begin{bmatrix} R & \hat{p}R \\ 0 & R \end{bmatrix}, \quad (\text{A.6})$$

$$W_a = Ad_{a2b}^T W_b, \quad (\text{A.7})$$

$$W_{ee} = Ad_{ee2ft}^T W_{ft} . \quad (\text{A.8})$$

The hybrid frame of EE position with robot orientation (if robot is assumed aligned with gravity this becomes the End effector Centric, Gravity Aligned (ECGA) frame), is defined with the label *hyb*:

$$Ad_{hyb2ee} = \begin{bmatrix} R_{robot2ee} & 0 \\ 0 & R_{robot2ee} \end{bmatrix} , \quad (\text{A.9})$$

$$W_{hyb} = Ad_{hyb2ee}^T W_{ee} . \quad (\text{A.10})$$

The operator may then specify compliance parameters as scalars along the diagonal of the compliance matrix:

$$C_{hyb} := \text{diag}(c_{f_x}, c_{f_y}, c_{f_z}, c_{\tau_x}, c_{\tau_y}, c_{\tau_z}) . \quad (\text{A.11})$$

The deflected end effector goal pose tracked towards by a linearly interpolating position controller is then given by Equation A.15:

$$T_{hyb}^{defl} = C_{hyb} W_{hyb} , \quad (\text{A.12})$$

$$T_{ee}^{defl} = Ad_{ee2hyb} T_{hyb}^{defl} , \quad (\text{A.13})$$

$$G_{ee2ee_{defl}} = e^{\hat{T}_{ee}^{defl}} , \quad (\text{A.14})$$

$$G_{robot2ee_{defl}} = G_{robot2ee} G_{ee2ee_{defl}} . \quad (\text{A.15})$$

A.3 End Effector Static Wrench Subtraction

A force torque sensor mounted within the wrist of a robot will be subject to the mass of the end effector assembly, impacting both static and dynamic measurements from the sensor.

When undertaking slow motions, such as excavation behaviors with a digging implement, static subtraction of the mass of the end effector (being distal to the FT sensor measurement) is sufficient to estimate the wrench experienced at the tooltip.

Parameters that may be used to determine the influence of the end effector on the force torque sensor reading may typically be comprised of:

W_{ft}^{raw} = Raw wrench as measured by force torque sensor

W_{ecga}^{ee} = Wrench exerted by static EE in “EE Centric, Gravity Aligned frame”

m = Mass of end effector (kg)

d = Distance of end effector (EE) center of mass from force torque sensor (m)

g = Gravity (m/s^2)

An example wrench for subtraction might be a simple point mass, measured as the total mass of the end effector, and positioned at its empirically or CAD sourced center of mass:

$$W_{ecga}^{ee} := \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (A.16)$$

In the case that center of mass of end effector is offset along Z axis of the force-torque frame by a distance d :

$$g_{ft2ecga} := \begin{bmatrix} I & d\hat{k} \\ 0 & 1 \end{bmatrix} . \quad (A.17)$$

The net wrench exerted upon the force torque center without the influence of the end effector mass is therefore:









$$W_{ft}^{net} = W_{ft}^{raw} - Ad_{g_{ft2ecga}}^T W_{ecga}^{ee} . \quad (A.18)$$











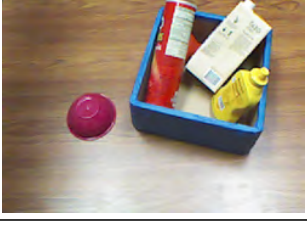

Appendix B










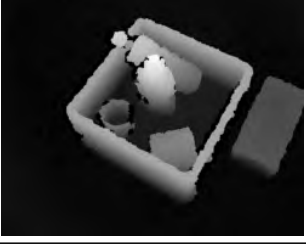
SEMANTIC MODEL INFERENCE ON REAL DEPTH IMAGES

B.1 Placement in Clutter

Below are displayed real images of outcome scenes for the task of placing a condiment bottle into container in the presence of clutter, drawn from the YCB dataset. Mode labels are presented in the order: ['failure1' 'failure2' 'success']

Cropped RGB Image	Cropped Depth Map	Inferred mode		
		True mode		
		Result		
		0.0898	0.2557	0.654
		0	0	1
		correct		
		0.8911	0.0594	0.0493
		1	0	0
		correct		
		0.0204	0.2033	0.7761
		0	0	1
		correct		
		0.21498	0.3158	0.4691
		0	1	0
		wrong		






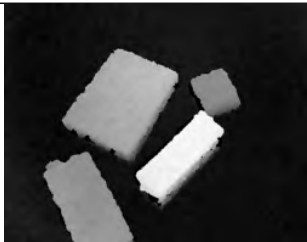


		0.9891	0.0051	0.0057
		0	1	0
		wrong		
		0.1438	0.2962	0.5599
		0	0	1
		correct		
		0.2539	0.2266	0.5194
		0	0	1
		correct		
		0.7475	0.1453	0.1071
		0	1	0
		wrong		
		0.0093	0.8155	0.1750
		1	0	0
		wrong		
		0.0892	0.2678	0.6428
		0	0	1
		correct		









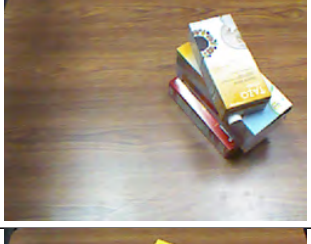



		0.0106	0.8622	0.1270
		0	1	0
		correct		
		0.5562	0.3207	0.1229
		1	0	0
		correct		
		0.2182	0.2929	0.4888
		0	0	1
		correct		
		0.2328	0.2894	0.4777
		0	0	1
		correct		
		4.2366	9.9520	4.3671
		0	0	1
		wrong		











B.2 Stacking of Cuboids

Below are displayed real images of outcome scenes for the task of stacking cuboid objects from the YCB dataset, where the model displayed bias towards failure mode

2. Mode labels are presented in the order: ['failure1' 'failure2' 'success']

Cropped RGB Image	Cropped Depth Map	Inferred mode		
		True mode		
Result				
		0.01847	0.9803	0.0011
		0	0	1
		wrong		
		2.83e-04	9.99e-01	8.01e-06
		1	0	0
		wrong		
		7.46e-19	1.00e+00	0.00e+00
		0	1	0
		correct		
		0.0792	0.5989	0.3218
		0	0	1
		wrong		

		0.1228	0.1670	0.7101
		0	0	1
		correct		
		0.07027	0.6534	0.2762
		1	0	0
		wrong		
		1.22e-02	9.96e-01	9.70e-04
		1	0	0
		wrong		
		2.57e-02	9.73e-01	8.31e-04
		1	0	0
		wrong		
		2.09-e03	9.96e-01	9.70e-04
		0	0	1
		wrong		
		6.24e-03	9.93e-01	3.82e-06
		1	0	0
		wrong		

		5.4e-12	1.0e+00	1.2e-26
		0	1	0
		correct		
		1.71e-03	9.98e-01	8.91e-06
		1	0	0
		wrong		
		9.81e-08	9.99e-01	7.97e-32
		0	1	0
		correct		
		0.0388	0.9580	0.0030
		0	0	1
		wrong		
		1.06e-05	9.99e-01	8.50e-10
		0	1	0
		correct		