Technical Report 810

On Motion Planning with Uncertainty

Michael Andreas Erdmann

MIT Artificial Intelligence Laboratory

This blank page was inserted to preserve pagination.

On Motion Planning with Uncertainty

by

Michael Andreas Erdmann

B.S., University of Washington
(1982)

© Massachusetts Institute of Technology, 1984

Revised version of a thesis submitted to the Department of Electrical Engineering and Computer Science on August 17, 1984 in partial fulfillment of the requirements for the degree of Master of Science.

This empty page was substituted for a blank page in the original document.

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the Laboratory's Artificial Intelligence research is provided in part by the System Development Foundation, in part by the Office of Naval Research under Office of Naval Research contract N00014-81-K-0494, and in part by the Advanced Research Projects Agency under Office of Naval Research Contracts N00014-80-C-0505 and N00014-82-K-0344.

This empty page was substituted for a blank page in the original document.

On Motion Planning with Uncertainty

by

Michael Andreas Erdmann

Abstract

Robots must successfully plan and execute tasks in the presence of uncertainty. Uncertainty arises from errors in modelling, sensing, and control. Planning in the presence of uncertainty constitutes one facet of the general motion planning problem in robotics. This problem is concerned with the automatic synthesis of motion strategies from high level task specifications and geometric models of environments.

In order to develop successful motion strategies, it is necessary to understand the effect of uncertainty on the geometry of object interactions. Object interactions, both static and dynamic, may be represented in geometrical terms. This thesis investigates geometrical tools for modelling and overcoming uncertainty.

The thesis describes an algorithm for computing backprojections of desired task configurations. Task goals and motion states are specified in terms of a moving object's configuration space. Backprojections specify regions in configuration space from which particular motions are guaranteed to accomplish a desired task. The backprojection algorithm considers surfaces in configuration space that facilitate sliding towards the goal, while avoiding surfaces on which motions may prematurely halt.

In executing a motion from a backprojection region, a plan executor must be able to recognize that a desired task has been accomplished. Since sensors are subject to uncertainty, recognition of task success is not always possible. The thesis considers the structure of backprojection regions and of task goals that ensures goal recognizability.

The thesis also develops a representation of friction in configuration space, in terms of a friction cone analogous to the real space friction cone. The friction cone provides the backprojection algorithm with a geometrical tool for determining points at which motions may halt.

Thesis Supervisor: Professor Tomás Lozano-Pérez
Title: Associate Professor of
Electrical Engineering and Computer Science

Acknowledgments

I wish to express special thanks to my adviser Prof. Tomás Lozano-Pérez for his constant and generous advice, encouragement, and support. All the ideas in this thesis were inspired by discussions with Tomás. His enthusiasm for the field is unbounded. This work would have been impossible without his guidance and help.

Thanks to Prof. Patrick Winston for providing support, and for fostering the special environment of the A.I. Lab. The challenge and excitement of this environment have made the experience here a pleasurable one.

I would like particularly to thank Bruce Donald for his friendship and support. Bruce read a draft of the thesis, providing many illuminating insights and comments. He spent much time discussing this research, and was always willing to lend a hand in times of need.

Thanks to Steve Buckley and John Canny for their friendship. Thanks also for the many conversations about robotics in general and this research in particular.

Profs. Matt Mason and Warren Seering gave freely of their time to aid my understanding of friction. Thanks also to Prof. Mason for his insight into pre-images.

I would like to thank Prof. Eric Grimson for much support and advice over the past two years, including reading and commenting on a draft of this thesis.

Thanks to all the members of the robotics and vision community at the A.I. Lab, in particular, Dr. Mike Brady, Prof. Rod Brooks, Dr. Philippe Brou, Dr. Ellen Hildreth, Prof. John Hollerbach, Prof. Tomaso Poggio, Gideon Sahar, and Dr. Demetri Terzopoulos, for their assistance and encouragement at various times. Thanks to Philippe for the Dover utility program which made possible many of the figures in this thesis.

Thanks to Rich Doyle for his friendship and perspective.

Lori Sorayama and Jim Glass offered constant moral support. Thanks also to Lori for the race to finish.

Thanks to the System Development Foundation for providing the financial support that made this work possible.

Finally, I wish to thank my parents for their unconditional love and support. Thanks for believing in education.

Raison d'Être

In dreaming about robotics, one envisions sitting in the backyard on a hot Saturday afternoon, while one's home robot is mowing the lawn. One asks the robot to take a break and fetch oneself a cold glass of iced tea. Upon returning to his primary task, the robot discovers that the mower won't start, so he pulls out the spark plug, cleans it, then carefully replaces it.

This scene is one of the simplest one can imagine, yet it is still mainly a dream. Solving the problems suggested by this dream demands the resources of numerous fields, including Mechanical Engineering, Electrical Engineering, Physics, Linguistics, Mathematics, Computer Science, Psychology, and Artificial Intelligence. In order to move about, the robot must be structurally and dynamically sound. His various onboard computers must function in isolation and in cooperation. The task of pushing the lawn mower requires the ability to exert forces, and to predict the effect of exerting such forces. In order to understand the request for iced tea, the robot must be capable of accepting verbal human commands, and of translating those often vague commands into actions that successfully achieve the desires of the commanders. In particular, the robot must plan his actions in a manner that gracefully deals with failure and uncertainty. He should not trip while bringing the glass of tea, even while moving over irregular terrain, nor should he pour the tea into one's lap, even though one's hand may be wavering as one prepares to accept the drink. Cleaning and replacing the spark plug requires a delicate touch, in order to avoid changing the plug's gap size, and in order to avoid breaking the plug with excessive torque while screwing it back in. Consequently, the robot must be capable of planning motions that sense contact forces and that deal with friction.

All the robot's actions involve interactions of objects. The robot is pushing a lawn mower. The lawn mower is in contact with the ground. The robot is holding a glass of tea. The robot is manipulating a spark plug. The spark plug is in contact with the lawn mower. In each case, at least one object is interacting with another.

All the robot's tasks involve either direct interaction, or the planning of such interaction. For example, planning to bring the tea entails, in part, planning to move the tea from the kitchen to the backyard while avoiding collisions with walls and chairs. The robot must understand the kinematic interaction of objects in order to avoid such collisions. As another example, in order to replace the spark plug, the robot must plan, among other things, to slide the plug into its socket. In this case, the robot must understand the dynamical and frictional interactions of the mower and the plug.

The study of object interactions constitutes a central theme in realizing the dream expressed earlier. This study draws heavily on techniques in fields ranging from high school physics to abstract algebraic geometry. It is the merging of these fields, and the concomitant understanding of the world, which makes the task of fulfilling the dream enjoyable.

Table of Contents

Abstract 2

Acknowledgments 3

Raison d'Être 4

Table of Contents 5

Chapter 1. Introduction 11

1.1. Statement of the Problem 12
1.2. Brief Outline of the Approach 13
1.3. Issues and Goals 15
1.3.1. An Example 15
1.3.2. Compliant Motion and Force Sensing 19
1.3.3. Automatic Strategy Synthesis 20
1.4. Research Contributions and Outline of the Thesis 20
1.5. Previous Work 21
1.6. Relation to other Work in Robotics 23
1.6.1. A Research Agenda 23
1.6.1.1. Assembly Planning 23
1.6.1.2. Motion Planning 23
1.6.1.3. Uncertainty 25
1.6.2. Relation of this Thesis to the Research Agenda 26
1.6.3. Future Research 27

Chapter 2. Basic Problems in Planning Motions with Uncertainty 29

2.1. Configuration Space 32
2.1.1. The Two Dimensional Configuration Space of a Planar Object 32
2.1.2. The Three Dimensional Configuration Space of a Rotating Planar Object 35
2.1.3. Other Configuration Spaces 35
2.1.4. Forces in Configuration Space 36
2.2. Generalized Damper 37
2.2.1. The Generalized Damper Models Sliding 37
2.2.2. Other Dynamics 38

2.2.3. Comparison of Generalized Damper to other Dynamics 38
2.3. Backprojections 41
2.3.1. The Planner's Plan of Pre-Images 41
2.3.2. Approximating Pre-Images with Backprojections 42
2.3.3. A Two Dimensional Algorithm 44
2.3.3.1. Backprojection from an Edge 45
2.3.3.2. Backprojection Involving Sliding 46
2.3.3.3. Backprojection Obstructed by an Obstacle 47
2.3.3.4. An Algorithm 48
2.3.3.5. Comments 56
2.3.4. A Three Dimensional Algorithm 59
2.3.4.1. Basic Tenets 61
2.3.4.2. Backprojection Along Surfaces — An Example 62
2.3.4.3. Backprojection Along Surfaces — Non-Linear Equations 63
2.3.4.4. A Formal Procedure 64
2.3.5. A Three Dimensional Slice Algorithm 66
2.3.5.1. Statement of the Algorithm 66
2.3.5.2. An Example 71
2.3.6. Relation of Backprojection to Backchaining 75
2.3.7. Relation of Backprojection to Pre-Images 77
2.4. Friction 83
2.4.1. A Real Space Friction Cone 84
2.4.2. A Configuration Space Friction Cone 86
2.4.3. Properties of the Configuration Space Friction Cone 89
2.4.3.1. General Comments 89
2.4.3.2. Motion Ambiguities 90
2.4.3.3. Multiple Points of Contact 92
2.5. Summary 93

Chapter 3. Backprojections 95

Symbol Table 96
3.1. Chapter Goal and Overview 98
3.2. Pre-Images 98
3.2.1. Basic Definitions 102
3.2.2. Directional Pre-Images 104
3.3. Comments on Computability 105
3.4. Solving the Pre-Image Equation 105
3.4.1. Idempotency 106
3.4.2. Maximal Pre-Images 107

3.5. Formal Definition of Backprojection 108
3.5.1. Preliminary Definitions 110
3.5.2. Ignoring Termination Predicates 112
3.5.3. Basic Assumptions and Definitions 114
3.5.3.1. History 115
3.5.3.2. Errors 115
3.5.3.3. Trajectories 115
3.5.3.4. Termination Predicates and Forward Projections . . 118
3.5.3.5. Goals in Velocity Space 121
3.5.3.6. Characterization of $S(p_0^*, R, \{G_\alpha\})$ 121
3.5.3.7. Definition of $\overline{S}(p, \{G_\alpha\})$ 122
3.5.4. Simple Comparison of Pre-Images and Backprojections . . 122
3.6. The Property of Being Self-Contained 123
3.7. The Power of Termination Predicates 125
3.7.1. The Standard Termination Predicate 126
3.7.2. The Termination Predicate with State 126
3.7.3. The Termination Predicate with no Sense of Time 127
3.7.4. The Termination Predicate without History or Time 127
3.7.5. No Sense of Time vs. No History 129
3.7.6. Comments on the Loss of Power 130
3.8. The Relationship Between Backprojections And Pre-Images
— Basic Issues 131
3.8.1. Reachable Goals 132
3.8.2. Backprojecting from the Reachable Goals 132
3.8.2.1. Straight Backprojection 132
3.8.2.2. Backprojection Intersected with the
Forward Projection 134
3.8.2.3. Backprojection from
Successful Termination Points 136
3.8.3. Backprojection in Phase Space 138
3.8.4. Almost Simple Pre-Images 139
3.9. First Entry Sets and the Structure Equation 140
3.9.1. First Entry Points 141
3.9.2. Properties of First Entry Sets 142
3.9.3. A Structure Equation 143
3.9.4. Maximality of the Extension $A(R)$ 144
3.10. Selection of Goal Sets 146
3.10.1. Review of the Grand Scheme 146
3.10.2. The Burden on First Entry Sets and Forward Projections . 148
3.10.3. Moral 152
3.10.4. Compatible Goals and Interpretations 153
3.10.4.1. Constraints on the Forward Projection 153
3.10.4.2. A Practical Note 155
3.10.4.3. Constraints on the Goal Sets 158
3.10.4.4. Another Practical Note 161

3.10.4.5. An Example Illustrating Separating Velocities . . . 165
3.11. Conclusion . . . 167
3.12. Appendix to Chapter 3 —— Computability . . . 170

Chapter 4. Friction . . . 179

4.1. Introduction . . . 180
4.2. Planning and Simulation . . . 181
4.3. One-Point Contact of a Planar Peg-In-Hole Problem . . . 184
4.3.1. Review, Assumptions, and Notation . . . 184
4.3.2. Peg-In-Hole Insertion . . . 185
4.3.3. Choice of Reference Point . . . 186
4.3.4. Tangents . . . 187
4.3.5. Equations of Motion without Friction . . . 189
4.3.5.1. Statics . . . 190
4.3.5.2. Normal Reaction Force . . . 190
4.3.6. A Configuration Space Friction Cone . . . 193
4.3.7. Comments . . . 197
4.4. General Form of Planar One-Point Contact . . . 198
4.4.1. Vector Decomposition . . . 199
4.4.2. Equations of Motion . . . 199
4.4.3. Relative Motions of the Reference Point and the Point of Contact . . . 200
4.4.3.1. Sliding and Normal Contact Velocities and Accelerations . . . 201
4.4.3.2. Contact Conditions . . . 202
4.4.3.3. General Second Variation Constraint . . . 203
4.4.3.4. Second Variation Constraint for Type B Surfaces . . 203
4.4.3.5. Second Variation Constraint for Type A Surfaces . . 205
4.4.4. Reaction Forces . . . 206
4.4.4.1. Fictitious Accelerations and Forces . . . 206
4.4.4.2. Sliding at the Contact Point . . . 207
4.4.4.3. Explicit Computation of Reaction Forces . . . 207
4.4.5. Motion Ambiguities . . . 210
4.4.5.1. Condition under which the Friction Cone Dips below the Tangent Plane . . . 211
4.4.5.2. An Example . . . 212
4.4.5.3. A Numerical Example . . . 213
4.4.5.4. Inadequacy of Motion Equations with Friction . . . 214
4.5. Multiple Points of Contact . . . 216
4.5.1. Equations of Motion on the Intersection of Surfaces . . . 216
4.5.2. Three-Point Contact Example . . . 219
4.5.2.1. The Vector Decomposition . . . 219
4.5.2.2. The Friction Cones . . . 220

4.5.3. Classes of Contacts 220
4.5.4. Predicting Reaction Forces 221
4.5.4.1. Consistency of Second Variation Constraints 221
4.5.4.2. Fictitious Forces and Perpendicular Projections . . . 222
4.5.5. An Example Demonstrating Constraint Inconsistency . . . 224
4.5.6. Reaction Force Computation for a Two Dimensional Contact Set 225
4.5.6.1. Form of the Reaction Force 225
4.5.6.2. Legal Friction Cone Edges 226
4.5.6.3. An Algorithm for Projecting onto the Composite Friction Cone 228
4.5.7. Indeterminate Accelerations 228
4.5.7.1. An Example 229
4.5.7.2. Common Tangent 231
4.5.7.3. Explicit Equations of Motion 232
4.5.7.4. A Continuum of Reaction Forces 233
4.6. Friction in Six Dimensional Configuration Space 234
4.6.1. Configuration Space 234
4.6.2. One-Point Contact 235
4.6.3. Friction 237
4.6.4. Reaction Force 241
4.6.5. Application 247
4.7. Summary 251

Chapter 5. Conclusion 253

5.1. Summary 253
5.2. Suggestions for Future Work 255
5.2.1. Model Uncertainty 255
5.2.2. Implementation of Backprojections for Full Six Dimensions 255
5.2.3. Termination Predicates with Time 255
5.2.4. Representation of Infinite Sets 256
5.2.5. Constraints Between Forward Projections and Goal Sets . . 256
5.2.6. Frictional Ambiguities 256
5.2.7. Second Order Systems 256
5.2.8. Probabilistic Uncertainty 257
5.2.9. Computability 257

References 258

This empty page was substituted for a blank page in the original document.

Chapter One

Introduction

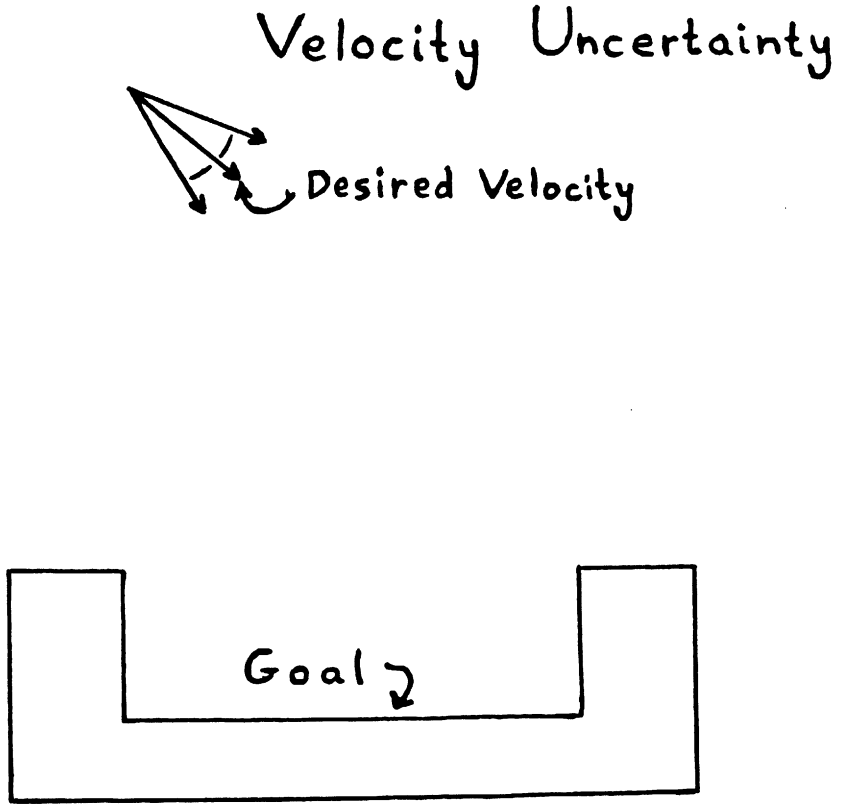


Figure 1.1. The task is to move a point to the edge at the bottom of the trough. The commanded velocity uncertainty is a cone about the desired velocity.

---

1. Introduction

1.1. Statement of the Problem

Robots must successfully plan and execute tasks in the presence of uncertainty. Uncertainty arises from errors in modelling, sensing, and control. Planning in the presence of uncertainty constitutes one facet of the general motion planning problem in robotics. This problem is concerned with the automatic synthesis of motion strategies from high level task specifications and geometric models of environments.

In order to develop successful motion strategies, it is necessary to understand the effect of uncertainty on the geometry of object interactions. Object interactions, both static and dynamic, may be represented in geometrical terms. This thesis investigates geometrical tools for modelling and overcoming uncertainty.

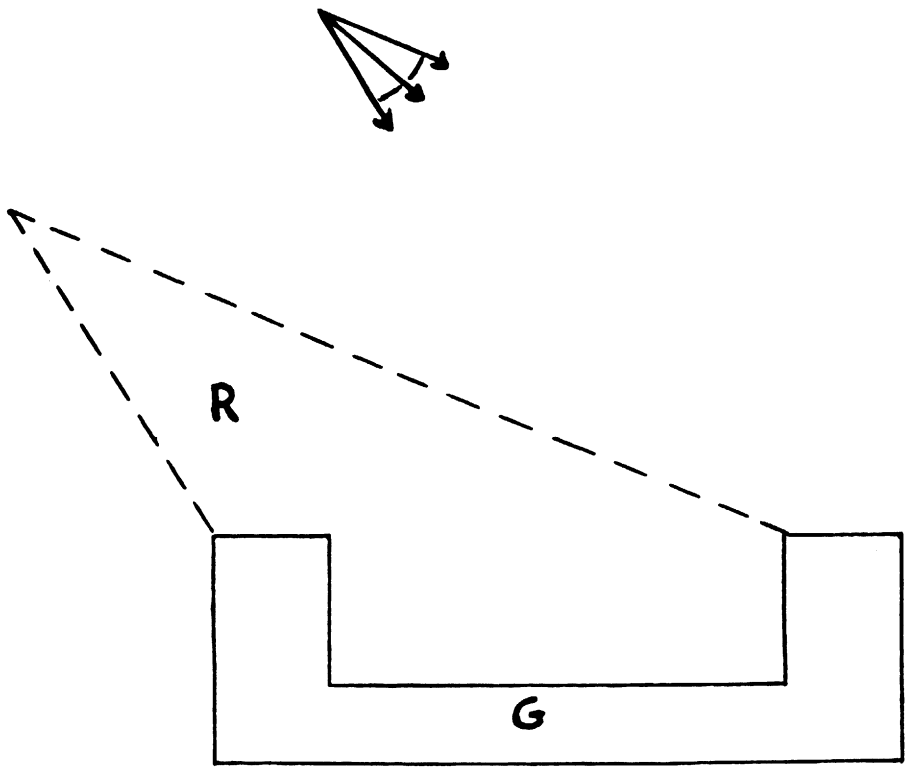


Figure 1.2. Region from which all motions are guaranteed to achieve the goal edge. The velocity uncertainty cone is shown. Motions either directly hit the goal edge, or slide along intervening edges before hitting the desired goal edge.

1.2. Brief Outline of the Approach

The basic approach models assembly tasks as a set of geometrical relations. These geometrical relations specify the desired configuration of objects upon task completion. The planning portion of the approach involves determining all possible initial configurations from which a motion is guaranteed to achieve the desired task configuration in the presence of uncertainty.

In the example of Fig. 1.1 the objective is to move a point to the edge at the bottom of the trough. Moving a point is not an artificial problem, since tasks involving object motions can be reduced to tasks involving point motions in the object's configuration space. This reduction will be explained in more detail later.

The nominal commanded velocity is down and to the right. The effective commanded velocity lies in some uncertainty cone about the desired velocity, as shown in Fig. 1.1. Assume that none of the effective commanded velocities cause a motion to stick on any of the trough's edges. In particular, assume that the effective commanded velocities all lie outside each of the edges' friction cones.

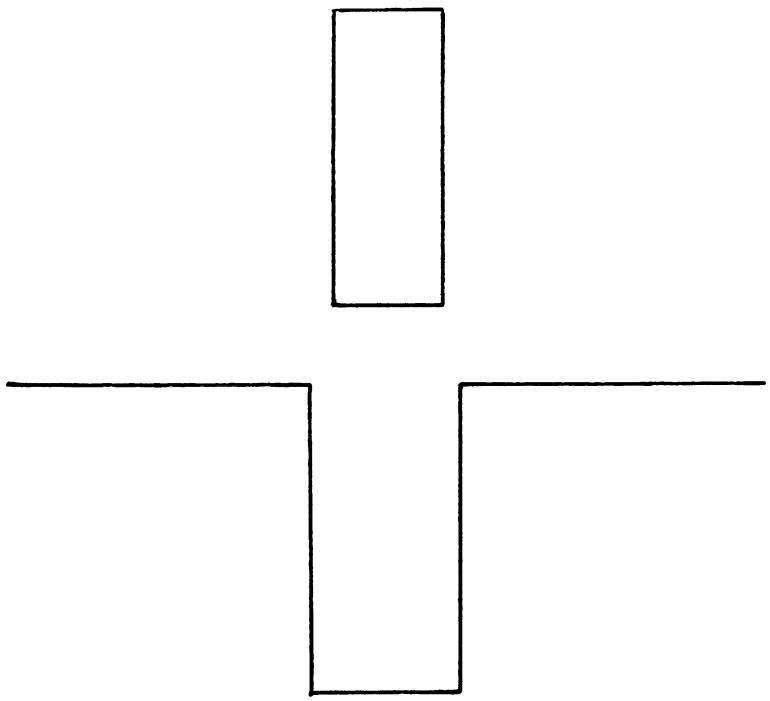


Figure 1.3. A peg-in-hole task.

Fig. 1.2 displays a region from which any point is guaranteed to achieve the goal edge, given the commanded velocity uncertainty cone. The region is bounded by the dashed lines drawn in the figure, and by edges from the environment. A point in the region is guaranteed either to directly attain the goal edge with a motion through free space, or to indirectly attain the goal edge by sliding along intervening edges.

A planning scheme should construct regions of the type shown in Fig. 1.2. At issue in forming these regions is the determination of sliding surfaces and termination conditions. The planner must take advantage of surfaces that guide a motion towards the desired goal. Additionally, the plan executor must be able to recognize success once a goal has been achieved.

This thesis presents a backprojection algorithm for computing regions from which particular motions are guaranteed to accomplish a geometrically specified task in the presence of uncertainty. The thesis considers the type of termination conditions and the structure of goal sets that permit successful recognition of task completion. Finally, the thesis investigates the representation of friction. An

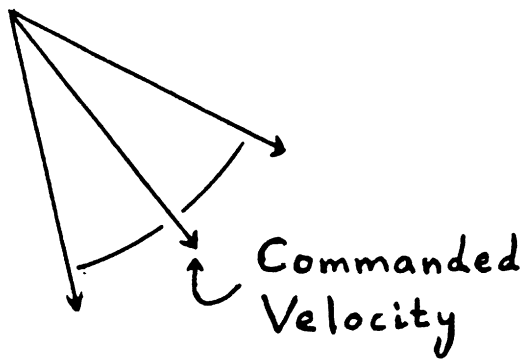


Figure 1.4. Velocity uncertainty cone about the commanded velocity.

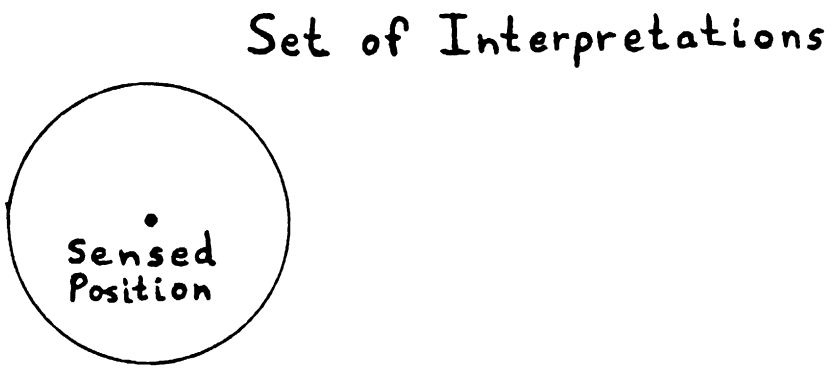


Figure 1.5. Position uncertainty about the sensed position.

---

understanding of friction is necessary in order to select surfaces along which motions are guaranteed to slide to the goal.

1.3. Issues and Goals

1.3.1. An Example

Consider the classical planar peg-in-hole problem of Fig. 1.3 The task is to move the peg from its initial position and orientation down into the hole until the bottom of the peg is resting in the bottom of the hole. Given perfect sensing and control, this task can be accomplished simply by commanding a motion of the peg straight down. The motion is terminated once the position sensor indicates that the peg is in the bottom of the hole.

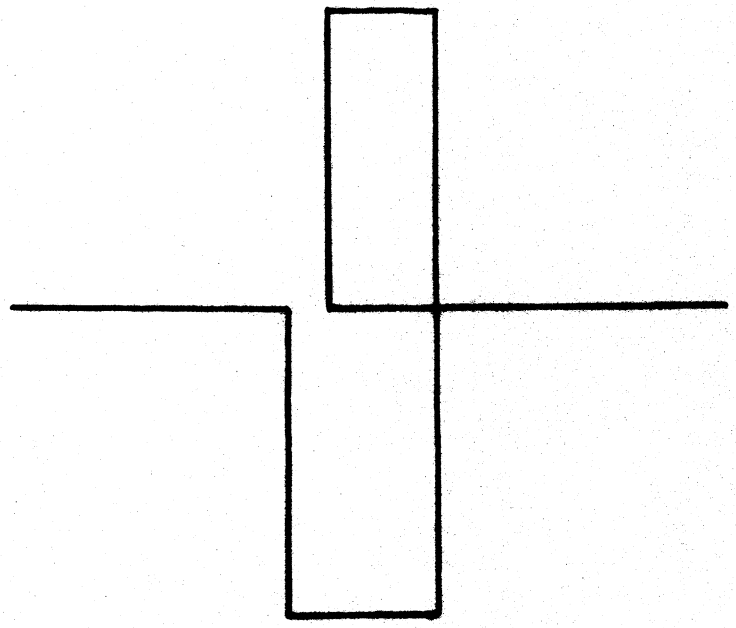


Figure 1.6. The peg is stuck at the top right corner of the hole.

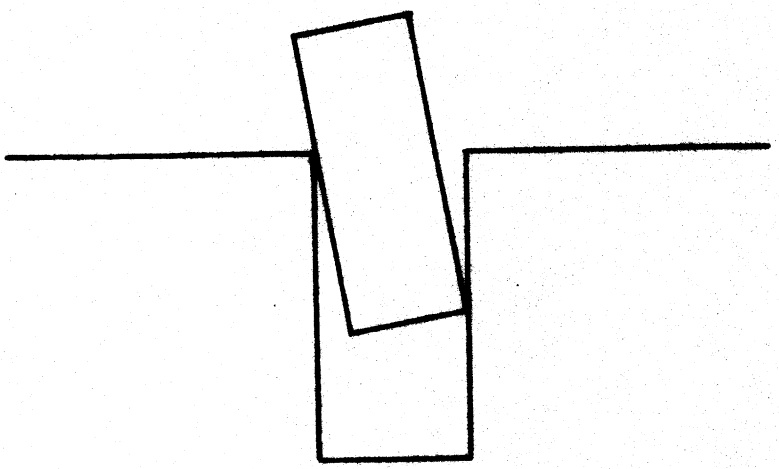


Figure 1.7. The peg is stuck in two-point contact resulting from angular misalignment.

Now suppose that the assumption of perfect sensing and control is relaxed. In particular, suppose that the actual direction of motion lies in some cone about the desired direction, as indicated in Fig. 1.4. The precise direction is unpredictable and may vary during the motion. In addition, the orientation of the peg is not guaranteed to remain constant throughout the motion. Thus the peg may tilt slightly as it moves. Furthermore, the position sensor is subject to error. A particular sensor reading serves merely to indicate that the actual position of the peg lies in some circle centered at the given sensor value (see Fig. 1.5). With a large enough position error, it is impossible to distinguish the left side of the hole from the right side, on the basis of position information alone. It may even be impossible to distinguish the bottom of the hole from the top.

Given these uncertainties in sensing and control, it is clear that the original solution to the peg in hole problem may fail. The peg may drift to the left as it moves downward, thereby getting stuck on the left corner at the top of the hole. Or it may drift to the right, sticking at the right corner. See Fig. 1.6. Given the uncertainty in the position sensor, it is impossible to execute corrective action once sticking is detected. This is because the corrective action depends on knowing the side at which sticking occurred.

Worse than being unable to correct errors should they occur, is the inability to detect success. In the given example, it is possible that the peg may move smoothly into the hole, as desired by the original plan. Unfortunately, given a large error in the position sensor, the plan executor cannot be certain that the peg is actually resting at the bottom of the hole, and not stuck at one of the corners.

Finally, it is also possible that the peg may jam while entering the hole. In the presence of friction, a slight angular misalignment of the peg may cause it to stick as it makes contact with both walls of the hole. See Fig. 1.7. Thus there are several potential states of the assembly in which the peg has stopped moving. Only one of these is the desired goal state, yet all are indistinguishable from each other.

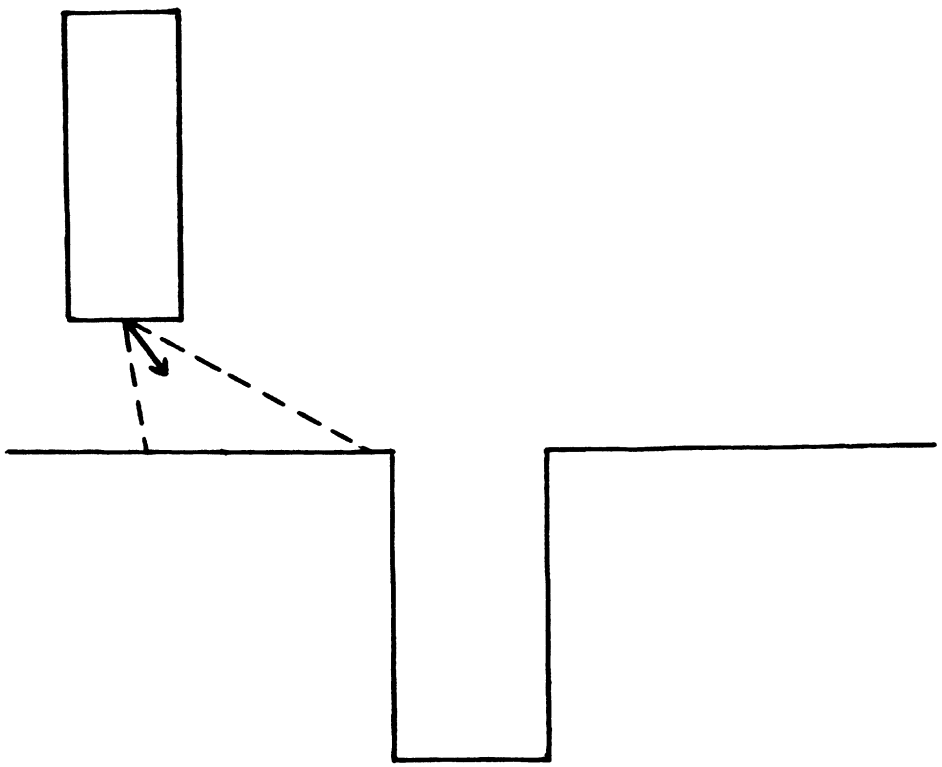


Figure 1.8. From this starting position, all velocities in the commanded uncertainty range indicated by the dashed lines, are guaranteed to move the peg onto a surface from which it can slide towards the hole.

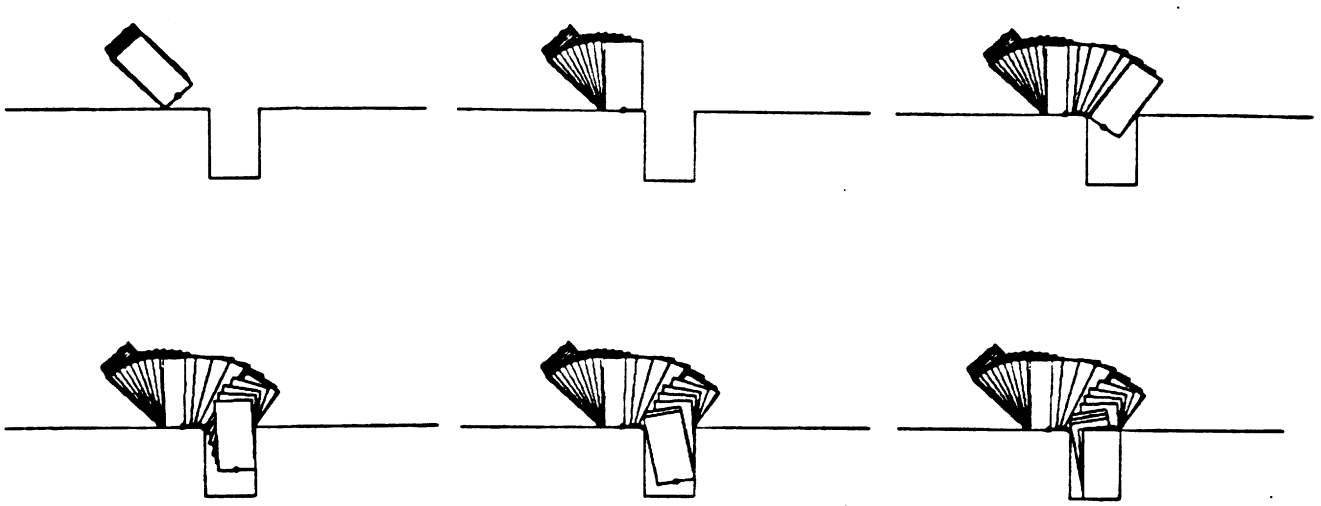


Figure 1.9. Simulation of a peg moving into a hole. The commanded velocity is down and to the right throughout the entire motion.

One solution to this problem (see Inoue (1974)) deliberately introduces an offset of the peg from the hole, such as moving the peg slightly to the left of the hole. Then a motion is commanded which points downward and to the right. By choosing the offset and commanded motion properly, this strategy guarantees that the peg will hit the surface above and to the left of the hole, regardless of where, within the error cone about the desired motion direction, the actual motion direction lies. See Fig. 1.8.

Upon hitting the surface above the hole, the peg slides along the surface until it encounters the hole, then tumbles down towards the goal. See Fig. 1.9. Recall that the position sensors may not be accurate enough to distinguish the bottom of the hole from its sides, hence are useless in signalling success. Instead, termination is detected by introducing force sensors. These sensors can be used both to detect collisions as well as to sense direction of motion. In particular, the force sensors can detect that the peg's motion has stopped once it hits the bottom of the hole. Assuming that the commanded motion has been so chosen that the peg does not jam while entering the hole, this condition signals success.

1.3.2. Compliant Motion and Force Sensing

The previous example demonstrates the following points:

- Surfaces may be used to guide motions. This technique is known as *compliant motion* (see Whitney (1977) and Mason (1981, 1983)).
- Force sensors may be used to detect collisions and to sense motion. This technique is known as *guarded moves* (see Will and Grossman (1975)).

The method of commanding directions and sensing forces stands in sharp contrast to the method of commanding and sensing positions. An object which has been commanded to move in a certain direction will move in that direction unless obstructed by a surface. When obstructed by a surface the object complies to that surface by moving tangentially to the surface, using the tangential component of the commanded direction. Friction may influence the direction and magnitude of this tangential motion. By commanding motions, a strategy takes advantage of the constraints in the environment to naturally guide the object to its goal.

Pure position control strategies are particularly subject to failure in the presence of uncertainty. Force sensing can improve performance, thereby enlarging the class of solvable tasks. For example, suppose a position control system tries to place an object onto a surface. If the actual surface lies above the position at which the system believes the surface to lie, then the system will try to push the object through the surface. Such action can generate excessive forces, and cause jamming or breakage. If the surface lies below the position believed by the position control system, then the object will be placed slightly above the surface. In this case the system does not achieve the desired assembly state.

In contrast, a system employing force sensing would accomplish the previous task by executing a guarded move. It would command a motion direction towards the surface, terminating the motion upon detecting a collision with the desired

surface. The precise location of the surface is irrelevant to the success of the motion. Thus force sensing strategies can accomplish tasks at accuracies that exceed the inherent resolution of the position controller and the position sensors.

1.3.3. Automatic Strategy Synthesis

A task planner must automatically synthesize strategies from high level task specifications and geometrical models, in the presence of uncertainty. Thus the planner must exploit surfaces along which compliant motions will guide the task to successful completion. It must recognize surfaces on which sticking or jamming can occur, so that it can choose motions that avoid premature termination. Finally, the planner must employ termination predicates, such as force and position sensors, to signal successful task completion. Although the previous examples have treated position and force sensors in isolation, it is clear that the two types of sensors should interact.

The requirements of the planner may be summarized by:

- The planner should exploit surfaces that facilitate sliding towards the goal.
- The planner should avoid surfaces on which sticking may occur.
- The planner should choose termination conditions that use both force and position sensors.

1.4. Research Contributions and Outline of the Thesis

This thesis develops tools that provide a planner with the capabilities needed to construct motion strategies that avoid sticking surfaces, while exploiting surfaces that cause sliding to the desired goal state. The contributions in particular are:

- The thesis presents an algorithm for computing motion strategies. The strategies are guaranteed to succeed in the presence of uncertainty.
- The thesis investigates the theoretical framework which forms the basis for the motion synthesis algorithm.
- Finally, the thesis develops a geometrical representation of friction in configuration space which constitutes an integral component of the planner's ability to avoid sticking surfaces.

Chapter 2 provides a fairly detailed but non-technical view of the thesis. This chapter should be read before considering the more formal material of Chapters 3 and 4. Chapter 2 describes tools used for the automatic synthesis of motion strategies in the presence of uncertainty. The chapter provides a review of configuration space and generalized damper dynamics. The details of the backprojection algorithm are presented. Finally, the chapter introduces a representation of friction in configuration space.

Chapter 3 investigates the formal foundation of the backprojection algorithm. The relationship of backprojections to the pre-images proposed by Lozano-Pérez, Mason, and Taylor (1983) is discussed. It is shown that backprojections can serve as

basic building blocks for pre-images, under restricted assumptions on termination predicates. Finally, a method for selecting good goal sets is presented.

Chapter 4 develops a representation of friction in configuration space. The representation is analogous to the classical friction cone in real space. Using this representation, the chapter considers the prediction of motions, both in single and multiple point contact. Examples demonstrating motion ambiguity under certain conditions are also discussed.

The remainder of the current chapter is devoted to reviewing previous work relevant to the thesis. Additionally, the chapter places the thesis within the context of other research in robotics, in particular, motion planning.

1.5. Previous Work

The limitations on position tolerances imposed by pure position control systems stimulated research into techniques for overcoming uncertainty in position sensors. Inoue (1974) used force feedback to successfully accomplish peg-in-hole operations at tolerances exceeding the inherent positional accuracy of his manipulator. Specifically, he used force control to accurately align parts by sliding on contact surfaces. Force exertion may also be used during trajectory execution to maintain contact with surfaces whose exact locations are unknown.

Force control is one form of a technique known as compliant motion. The principle underlying compliance consists of utilizing task constraints to guide assembly operations, although the constraints themselves may not be known precisely. Whitney (1977) described generalized dampers as one means of achieving compliance. Raibert and Craig (1981) described a combination of position and force control, known as hybrid control. Salisbury (1980) has considered generalized stiffness control for six degrees of freedom. In particular, he considered transforming into joint coordinates stiffness specifications made in arbitrary cartesian hand coordinates. Mason (1981, 1983) provided a complete exposition of compliance, including damping, stiffness, and hybrid control. See Mason (1983) for a thorough review of work in the area of compliance.

Force control strategies must account for friction, in order to avoid assembly states which cause premature termination of planned motions. Much work has been done in modelling friction. Simunovic (1975, 1979) and Whitney (1982) have determined conditions under which jamming or wedging can occur for peg-in-hole insertions. In particular, Whitney (1982) has derived constraints on forces during two-point contact that ensure smooth entry of the peg into the hole. Ohwovoriole and Roth (1981), and Ohwovoriole, Hill, and Roth (1980) have also analyzed the possible states in which an assembly may jam or wedge. They derive both kinematic and force constraints that must be satisfied. Burridge, Rajan, and Schwartz (1983) have considered various classes of motions possible during frictional contact. Mason (1982) has considered pushing and grasping operations in the presence of friction. He derives conditions for successful task completion in the presence of uncertainty. The methods of these authors apply generally to the real space analysis of assembly operations.

It is often possible for one sensor to disambiguate the readings provided by another sensor. The guarded moves of Will and Grossman (1975) constitute one technique for achieving accurate positioning. By moving one object towards another object until a collision is detected, for instance, by a force or velocity sensor, the two objects can be brought into contact, even though the position of neither is known accurately. Guarded moves and compliant motions are complementary techniques. Guarded moves attain contact, while compliant motions maintain contact, both in the presence of position (and control) uncertainty.

Compliant and guarded motion strategies attempt to account for uncertainty implicitly by using constraints in the environment as silent guides. These techniques do not, however, address the question of recognizing the explicit effect of different errors on the uncertainty of object interactions. This explicit recognition is necessary in order to determine the conditions under which specific strategies, such as compliant or guarded move strategies, are applicable.

The skeleton strategies advanced by Lozano-Pérez (1976) and by Taylor (1976) offered a means of relating error estimates to strategy specifications in detail. By considering constraints on error ranges, it is possible to determine the values of parameters required in a skeleton strategy to insure success. Brooks (1982) also considered symbolic constraints in computing errors. His constraints can be used both to provide error estimates, as well as to identify plan variables whose values must be constrained for successful task completion.

Dufay and Latombe (1984) have considered a system that develops strategies for dealing with uncertainty by considering multiple traces of executed plans. The system consists of a training phase that generates execution traces, and an induction phase that transforms the traces into a program capable of dealing with uncertainty. During the training phase, the system may augment the original plan with error recovery steps as unexpected events occur. By examining the different types of modifications introduced into the original plan, the induction phase can determine the range of possible outcomes of a desired motion. The induction phase can therefore generate a program able to deal with uncertainty by explicitly testing for the different motion outcomes resulting from uncertainty.

Recently Lozano-Pérez, Mason, and Taylor (1983) proposed a formal solution to the problem of automatically synthesizing fine motion strategies. Given a set of goals, they constructed regions from which, in the presence of uncertainty, it was always possible to reach one of the goals using a single control command. By recursively backchaining from the initial goal set, they were able to define a sequence of motion commands that was guaranteed to move an object from its initial position into one of the desired goals. Mason (October 1983) has shown this method to be both correct and bounded-complete. However, at present there does not exist an effective procedure for computing these motion commands. Chapter 3 discusses this work in more detail.

1.6. Relation to other Work in Robotics

1.6.1. A Research Agenda

In his Master's thesis, Lozano-Pérez (1976) provided an outline for research in robotics. The outline appeared in his description of the conceptual phases which are involved in task planning. There are three such phases. In the first phase, known as the *Assembly Planning Phase*, a general plan is developed. This plan specifies the classes of operations which are to be performed. For example, it may specify that object A is to be placed on top of object B, without specifying precisely how to accomplish this operation. In the second phase, known as the *Pick And Place Phase*, the grasp positions and collision free motions of objects are determined. Continuing the previous example, this phase determines a path which permits placing A on B without colliding with any objects while moving A. Finally, the third phase, known as the *Feedback Planning Phase*, incorporates sensing into the previous plans, in order to deal with uncertainty and error. For example, this phase might employ position and force sensing to ensure that A is stably placed on top of B.

1.6.1.1. Assembly Planning

Brooks (1982), Lozano-Pérez (1976), and Taylor (1976) have considered the *Assembly Planning Phase*. Brooks (1982) has developed a symbolic error analysis system to be used in a plan checker. The system may be used to modify a plan to ensure that it will succeed.

Lozano-Pérez (1976) and Taylor (1976) have investigated planning systems that expand partial task specifications into complete sequences of operations. Each operation possesses certain preconditions and achieves certain postconditions. The planner attempts to generate a sequence of operations that accomplishes a task subject to the constraints imposed on the operations by the geometry of the world. The plan checker of Brooks (1982) is a means of verifying satisfaction of these constraints, and of introducing new operations so as to guarantee satisfaction.

See also Lozano-Pérez (1983) for a thorough review of work in the area of task planning.

1.6.1.2. Motion Planning

Much work has been done in connection with the *Pick and Place Phase* outlined in Lozano-Pérez's thesis, in particular with regard to the motion planning problem. The motion planning problem is that of moving an object from an initial configuration through a collection of obstacles to a goal configuration, while avoiding collisions with the obstacles. The problem is to discover whether a path between the initial and goal configurations exists, and if so, to find such a path. The objects are generally assumed to be either polyhedral or algebraic.

Investigation of this problem finds its roots in the works of Brooks (1983), Lozano-Pérez (1981, 1983), Lozano-Pérez and Wesley (1979), Reif (1979), Schwartz and Sharir (1981, 1982, 1983), and Udupa (1977). Udupa (1977) first transformed the obstacle avoidance problem from a task involving the motion of an object to

the task of moving a point. The basic principle was to shrink the moving object to a point, while growing obstacles to account for this shrinking.

Lozano-Pérez introduced into robotics the notion of *configuration space*. This representation of the motion planning problem reduces the problem of moving an object through an obstacle filled environment to the problem of moving a point through a higher dimensional space, namely the configuration space of the moving object. The configuration space of an object is the parameter space representing the degrees of freedom of the object. Obstacles in real space constitute constraints on the object's degrees of freedom. These constraints can be represented as obstacle surfaces in the object's configuration space.

Reif (1979) showed that the motion planning problem, for a robot with an arbitrary number of degrees of freedom in the form of linkages, is *P-Space*-hard. Schwartz and Sharir (1982) have exhibited an algorithm for the motion planning problem which is exponential in the number of degrees of freedom, but, for fixed degrees of freedom, runs in polynomial-time in the number of obstacle faces.

Hopcroft, Joseph, and Whitesides (1982) have considered the motion of planar linkages. Their investigation showed that deciding whether an arm with an arbitrary number of links can be moved from one configuration to another while remaining inside a given region is *NP*-hard. For circular regions they showed this problem to be solvable in polynomial time.

Brooks and Lozano-Pérez (1983) have implemented a path planning algorithm for the problem of a planar polygonal object with two translational and one rotational degrees of freedom. The configuration space for this problem is a three dimensional space. A general rigid object in the real world has three translational and three rotational degrees of freedom. Thus the configuration space is a six dimensional space. Lozano-Pérez (1981) has also suggested a method for planning collision free paths for the full six dimensional motion planning problem, using three dimensional slice projections.

In his Master's thesis, Donald (1984) described a complete algorithm, which he also implemented, for the full six dimensional motion planning problem. The algorithm explicitly uses the coherence of configuration space obstacles to plan paths, by using operators that slide parallel to five dimensional surfaces[1] and to the intersection of such surfaces in configuration space. Donald also described the form of the surface and intersected surface equations, as well as the applicability regions in rotation space for these surfaces.

Ó'Dúnlaing and Yap (1982), and Ó'Dúnlaing, Sharir and Yap (1982) have considered planar motion problems using retraction to Generalized Voronoi diagrams. Brooks (1983) has implemented a planar path planner which models the free space between obstacles as Generalized Cones. Donald (1984) has also defined and considered Generalized Voronoi Manifolds in configuration space for path planning with six degrees of freedom.

[1]Throughout this thesis 'surface' refers to any submanifold. No particular dimension should be attributed a priori to the word 'surface'.

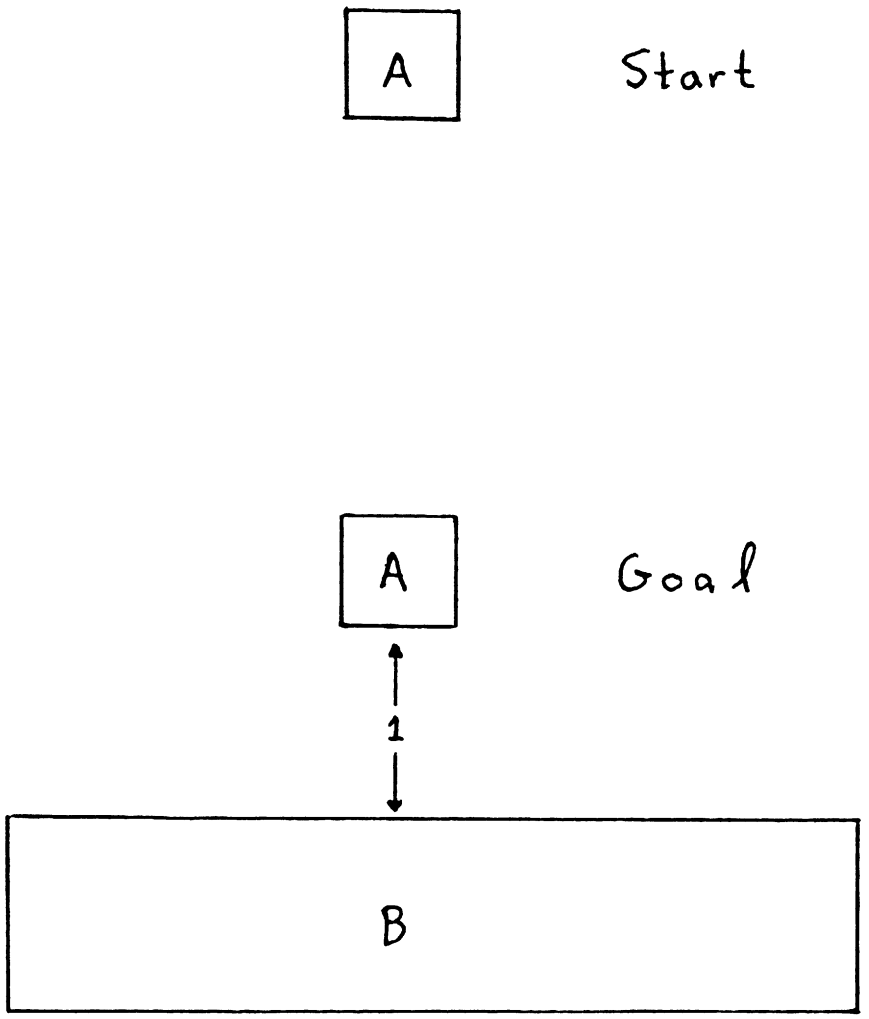


Figure 1.10. The task is to move object A one meter above object B, in the presence of position sensing uncertainty. A single motion cannot solve this problem. A strategy that first bumps into B, then moves up one meter, can successfully accomplish the task.

Hopcroft and Wilfong (1984) have investigated the motion of objects in contact. They have shown that if two objects in contact at one configuration can be moved to another configuration in which the two objects are again in contact, then there exists a motion between the two configurations during which the objects are always in contact.

1.6.1.3. Uncertainty

Some work has been done within the area of Lozano-Pérez's third conceptual planning phase, the *Feedback Planning Phase*. This phase accounts for uncertainty and error in the plan steps found by the *Assembly Planning* and *Pick And Place Phases*. Lozano-Pérez's LAMA system (1976) uses geometric simulation of plan steps to decide on possible motion outcomes. The simulation makes explicit the possible errors that might occur. By introducing sensing at the error points and by

restricting the values of parameters in the skeleton plans, it is possible to recover from or explicitly avoid errors during the execution of planned motions.

Taylor (1976) provided an algebra system which considered constraints on error ranges to restrict the values of parameters in skeleton strategies, in order to ensure successful motions. Brooks (1982) has considered symbolic constraints in computing errors. By manipulating the constraint dependencies between successive plan steps, his system can be used both to provide error estimates, as well as to identify plan variables whose values must be constrained for successful task completion.

Dufay and Latombe (1984) developed a system that induces the possible effects of uncertainty from execution traces of a proposed plan. The system generates a modified plan that accounts for uncertainty. This plan uses conditional motions which explicitly test for the error states learned from the execution traces.

Recently Lozano-Pérez, Mason, and Taylor (1983), and Mason (October 1983) proposed a formal system which directly incorporates uncertainty information into the planning process. Their work forms the foundation of this thesis, and will be presented in more detail later.

1.6.2. Relation of this Thesis to the Research Agenda

Previous work has tended to view the three conceptual phases of planning as distinct and consecutive phases, primarily because of the difficulty of understanding the problems within these phases even in isolation. Thus work on motion planning generally assumed perfect knowledge and perfect control. Incorporation of error handling has occurred as an afterthought. Having first developed a plan under the assumption of a perfect world, the plan was only secondarily augmented with sensing steps and constraints on plan variables to ensure the plan's success. For example, the error analysis system of Brooks (1982) was intended as a plan checker. In other words, the system expected a given plan along with error information. The system then either augmented the plan with sensing steps to account for the error information, or, if it was unable to guarantee the plan's success, it labelled the plan as infeasible.

In effect, within the planning process, information has tended to flow from the second conceptual phase, the *Pick And Place Phase*, to the third conceptual phase, the *Feedback Planning Phase*. While this approach can solve many problems, it ultimately limits the class of tasks that a robot can be expected to perform. In general, a planning scheme must explicitly take account of error and uncertainty while deciding on the motions to be performed, as this information may affect the structure of the plan.

For example, consider the problem in Fig. 1.10. The object A is to be moved from its initial position to a point one meter above object B. Given perfect sensing and control, this is a trivial task, requiring a single direct motion from the initial to the desired goal configuration. If, however, the assumption of perfect sensing is relaxed, so that the initial position is not known precisely, then the plan fails. Furthermore, it is impossible to spruce up the given motion with additional sensing, so as to insure success. However, if the structure of the plan is altered, then a

two step motion, which first bumps object A into object B, and then moves to the desired goal, will succeed. A similar example, in which the goal is a region rather than a point, can be used to show that a plan which succeeds under perfect control must be restructured in order to succeed under error tainted control. More complicated examples abound.

This thesis proposes to take explicit account of error and uncertainty during the planning process. The thesis develops geometrical tools which allow a planner to construct motions that are guaranteed to succeed in the presence of uncertainty.

1.6.3. Future Research

Additional work is required in the *Assembly Planning Phase*. Lozano-Pérez, Mason, and Taylor (1983) note that small changes in geometry can effect large changes in strategy. This effect is compounded by the presence of uncertainty. Once again, it is necessary to take explicit account of error and uncertainty while deciding on the classes of operations to be performed for a given task. In particular, the order of operations may be affected by uncertainty. Additionally, aids such as jigs or scaffolds may be required to insure success of operations subject to uncertainty. See also Chapter 5.

This empty page was substituted for a blank page in the original document.

Chapter Two

Basic Problems in Planning Missions with Uncertainty

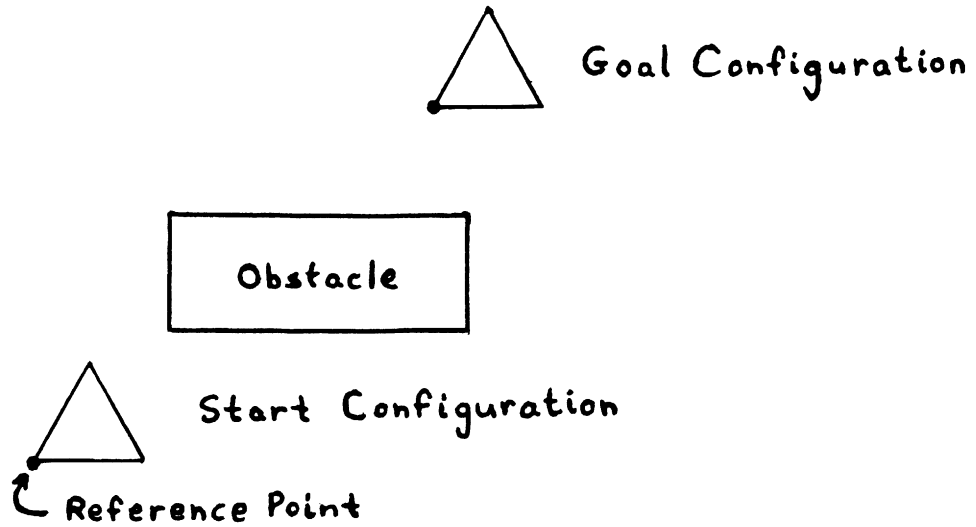


Figure 2.1. The task is to move the triangle from the start to the goal configuration. The rectangle is a stationary obstacle.

2. Basic Problems in Planning Motions with Uncertainty

Briefly recall the requirements of a motion planner (see Sec. 1.3.3). It should exploit surfaces that cause sliding to the goal. It should avoid surfaces on which sticking may occur, and it should employ both force and position sensors in choosing termination conditions for the plan executor. This thesis develops tools required by the planner. This chapter is devoted to an overview of the ideas behind the tools. It serves both as a refresher of several established concepts in robotics as well as a primer on new research presented in this thesis. Later chapters discuss the tools in more detail. In particular, the later chapters build the mathematical foundation for the research presented in the third and fourth sections of the current chapter.

The first two sections review the notions of configuration space and generalized damper. Configuration space abstracts the essential geometric constraints of object interactions. The generalized damper is a dynamical model that defines an equivalence between velocities and forces.

The third section defines backprojection regions and presents an algorithm for computing these regions. The planner can use backprojection regions in a backchaining process to develop motion strategies that are guaranteed to succeed in the presence of uncertainty. This section also discusses the relationship of backprojections to the LMT (Lozano-Pérez, Mason, and Taylor (1983)) planning system.

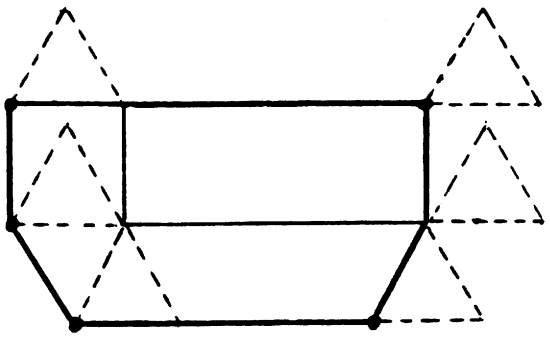


Figure 2.2. Construction of the configuration space obstacle for the task of Fig. 2.1.

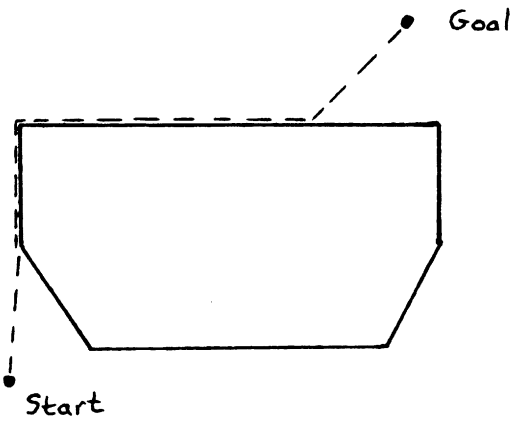


Figure 2.3. Path in configuration space between start and goal configurations.

Finally, the fourth section develops a representation of friction for configuration space. This representation is seen to be a friction cone with properties analogous to those of a real space friction cone. The friction cone permits the planner to use geometrical techniques in deciding whether sticking can occur at a point. This is invaluable to the planner's search for sliding surfaces.

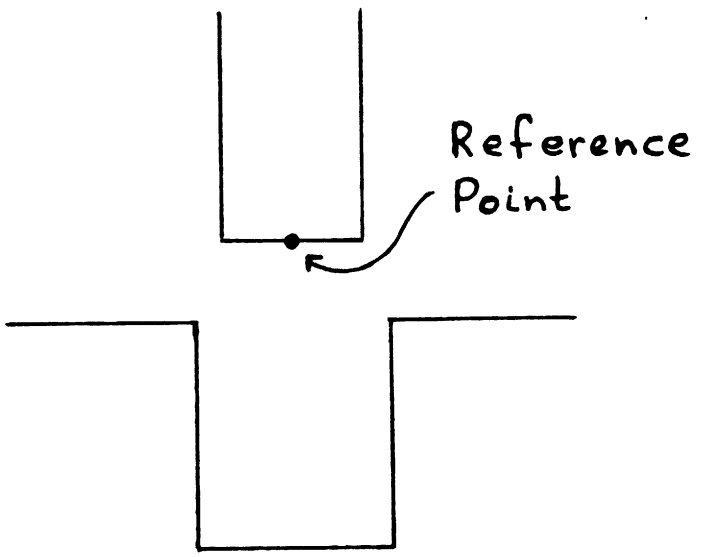


Figure 2.4. Peg-in-hole task. The reference point is at the bottom of the peg.

2.1. Configuration Space

The list of requirements indicates that the surfaces in the moving object's environment are critically important, as they represent constraints on the object's motion. It is therefore desirable to employ a geometric representation which makes these constraints explicit. The *configuration space* of a moving object provides a representation which highlights the relevant constraints on motion (see Arnold (1978), Lozano-Pérez (1981, 1983)).

The configuration space of an object is the parameter space representing the object's degrees of freedom. Obstacle surfaces in real space define constraints on the object's degrees of freedom. These constraints may be represented by hyper-surfaces in the object's parameter or configuration space. Thus the task of moving a complicated object in a world filled with other complicated objects is transformed into the simpler problem of moving a point in a higher dimensional space, namely the object's configuration space.

The next few subsections describe several common configuration spaces. The treatment is of intuitive nature. The precise mathematical definitions are not needed for the current exposition, but will be stated and used in later chapters.

2.1.1. The Two Dimensional Configuration Space of a Planar Object

Consider the environment of Fig. 2.1. The moving object is the triangle. It is allowed to translate in the plane, but is not permitted to rotate. Thus the triangle

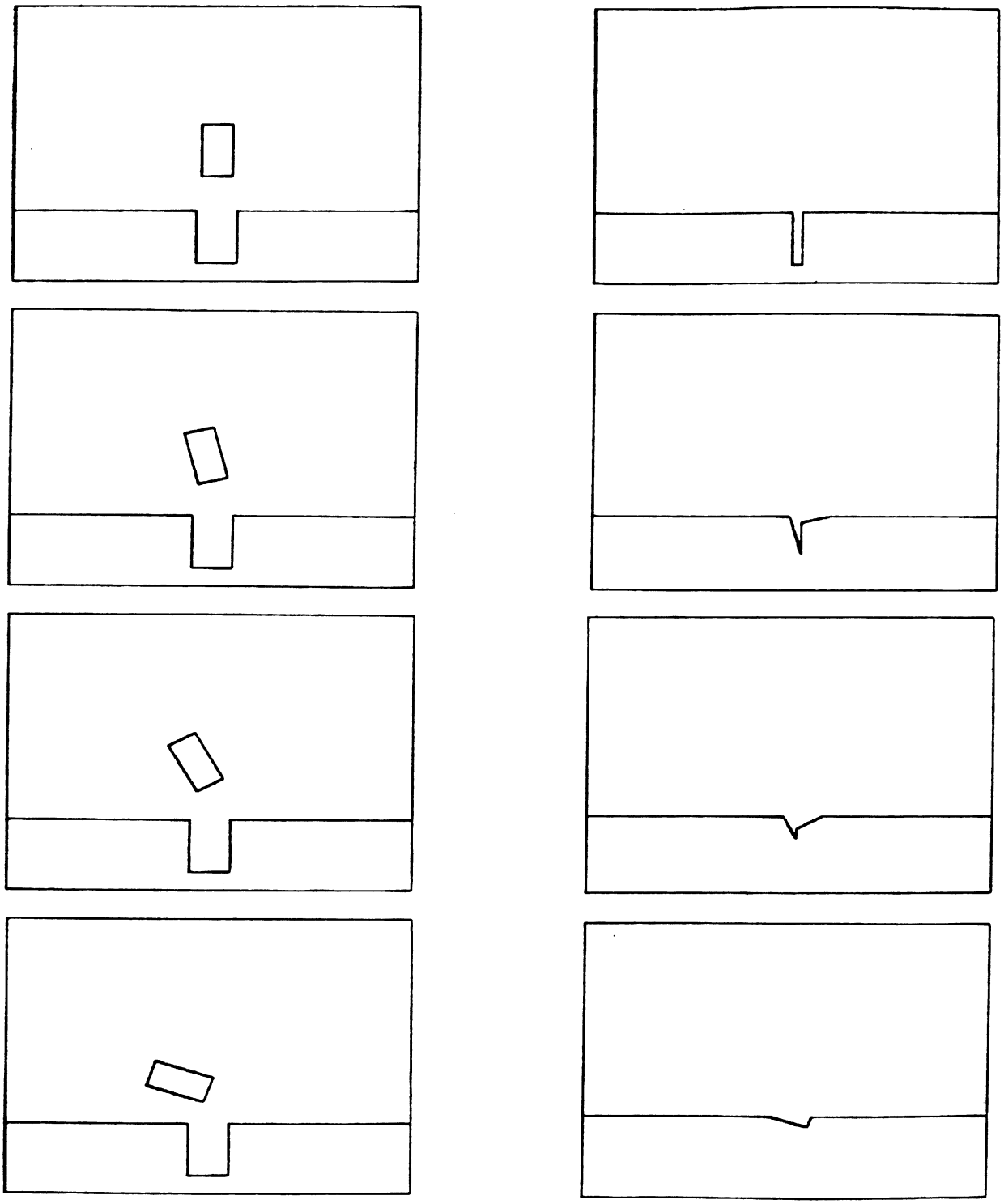


Figure 2.5. Several slices of the peg-in-hole configuration space, corresponding to different orientations of the peg. The real space is on the left, while the associated configuration space is on the right.

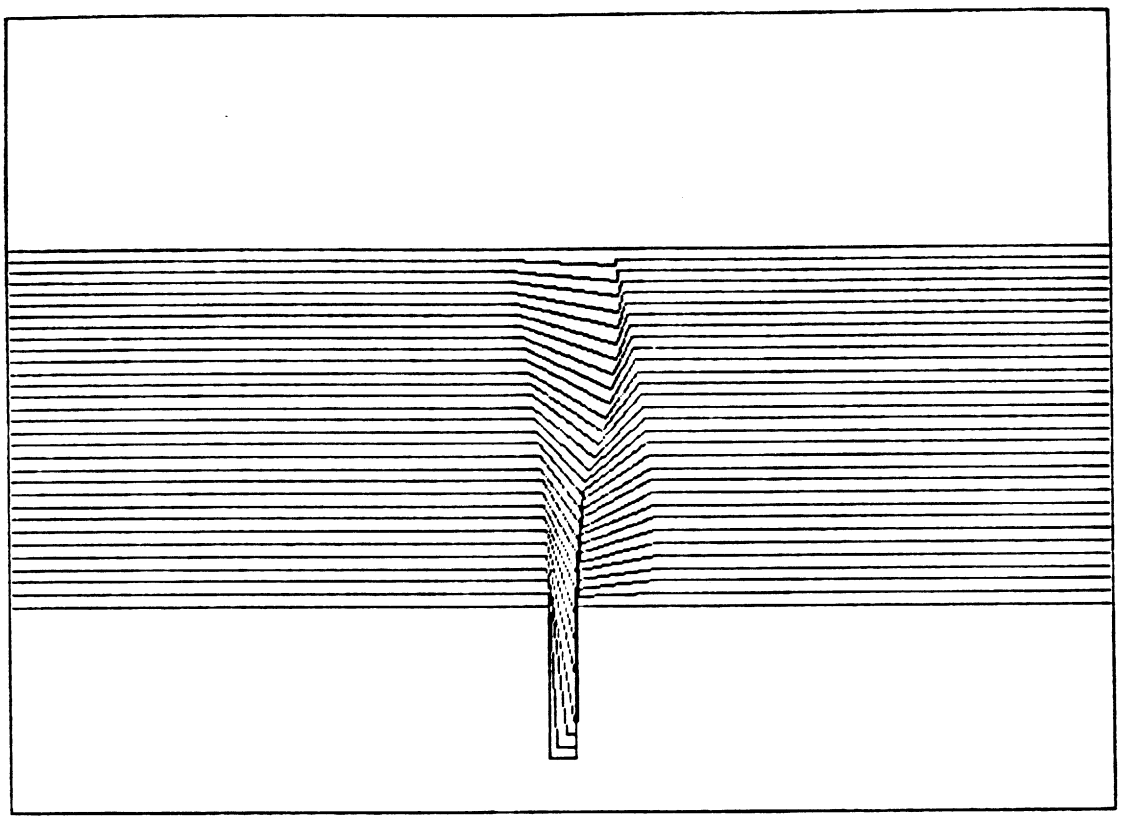


Figure 2.6. A portion of the configuration space surface corresponding to the peg-in-hole problem. The surface is shown as a collection of slices.

---

has two degrees of freedom. The task is to move the triangle from its given position to the indicated goal position.

In order to transform the problem into its configuration space representation, it is necessary to choose a reference point relative to which the object's parameters are measured. This is indicated in Fig. 2.1. Since the object can only translate, the configuration space is two dimensional. The configuration space is itself a plane, consisting of (x, y) configurations.

The constraints on the object's motion are represented in its configuration space by the configurations of the reference point that would cause collisions between the object and obstacles in the plane. The resulting constraints are depicted in Fig. 2.2.

The initial and goal positions of the triangle correspond to two points in the triangle's configuration space. One path between these two configurations is shown in Fig. 2.3. Since the path lies either in free space or on the boundary of configuration space obstacles, the motion of the triangle occurs either as free motion between real space obstacles or as sliding motion along the edges of real space obstacles.

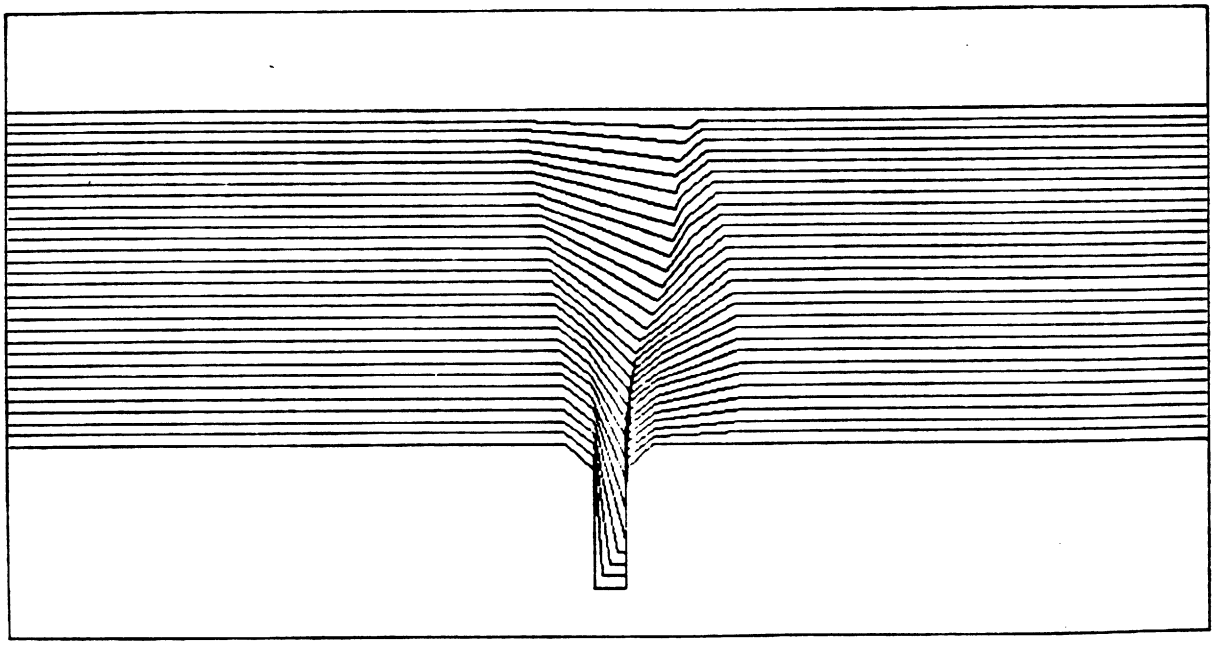


Figure 2.7. Configuration space of the peg-in-hole problem with a chamfered hole.

2.1.2. The Three Dimensional Configuration Space of a Rotating Planar Object

Consider now the planar peg-in-hole problem previously mentioned in Chapter 1. Assuming that the peg is allowed only to translate, but not to rotate, the configuration space is a two dimensional planar space, as in the previous example. Choose the reference point at the bottom of the peg, as in Fig. 2.4. For each fixed orientation of the peg, the resulting configuration space of the translating peg is a two dimensional (x, y) space. This space is depicted for several orientations in Fig. 2.5.

If now the peg is permitted to rotate as well as translate, then the peg has three degrees of freedom. Hence the peg's configuration space is a three dimensional space, consisting of (x, y, θ) configurations. For each fixed value of θ, the associated (x, y) cross section of this three dimensional space is just the two dimensional configuration space of the translating but not rotating peg. A few of these cross sections are shown in Fig. 2.6. Fig. 2.7 shows the change in the configuration space for the peg-in-hole problem if the hole is chamfered.

2.1.3. Other Configuration Spaces

A rigid object in real space has three translational degrees of freedom and three rotational degrees of freedom. Hence the configuration space is a six dimensional space.

A robot arm with six prismatic or revolute joints has six degrees of freedom. Thus its configuration may be specified by six parameters in a six dimensional configuration space.

As an aside, one should note that the configuration subspaces of rotational degrees of freedom are slightly different than the configuration subspaces of translational degrees of freedom. This is because rotations must wrap around. Furthermore, rotations need not commute.

2.1.4. Forces in Configuration Space

In real space, if one pushes on an object, it pushes back. Objects generate reaction forces that are parallel to the normal vectors of the objects' faces. Configuration space objects behave identically. Forces exist in configuration space as generalized forces. In real space a force vector pointing in a particular direction causes acceleration in that direction. The same statement applies in configuration space, except that the coordinates along which the acceleration is occurring, are parameters such as translation and orientation, rather than the usual three dimensional coordinates of everyday life. Conceptually, there is no difference.

Configuration space surfaces are similar to real space surfaces. They represent inviolable constraints. Thus they are as solid as real space surfaces. Additionally, they generate reaction forces in a fashion similar to the manner in which real space surfaces generate reaction forces. These reaction forces lie along the normals to the surfaces, just as for real space surfaces. Much of the reasoning and intuition known from everyday life carries over to configuration space surfaces.

The advantage of configuration space lies in its simplification of constraints. Configuration space makes explicit the geometrical constraints imposed on the motion of an object by obstacles in the object's environment. In real space, it is necessary to explicitly examine the interaction of every edge, vertex, and face of the moving object with its environment. In configuration space, it is sufficient to consider the interaction of a single point with a configuration space surface. Thus configuration space simplifies the analysis of motion.

The similarity between real and configuration space surfaces has led to formulations of sliding strategies in configuration space, in particular, of compliant motions and guarded moves. See (Mason (1981)) for seminal work in this area. This realization demonstrates the usefulness of configuration space as a conceptual tool in robotics. Not only does configuration space capture the geometrical constraints of obstacles in the real world, but it also offers a representation of the dynamics of object interactions which preserves real world intuition.

The previous paragraphs have naturally ignored a few minor details. For example, in order for the acceleration of generalized coordinates to correspond to the application of generalized forces, it is necessary to choose the reference point at the center of mass or at the center of compliance of the moving object. Additionally, the representation of real space friction in configuration space requires a bit of work. Some of these details are investigated more thoroughly in later sections and chapters.

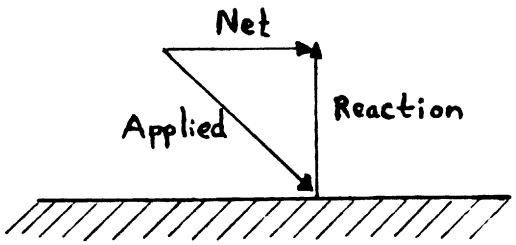


Figure 2.8. Velocity/Force computation under generalized damper dynamics, assuming no friction.

2.2. Generalized Damper

2.2.1. The Generalized Damper Models Sliding

As noted above, a planner should synthesize strategies that take advantage of intervening surfaces, planning motions that slide along the surfaces towards the goal. Furthermore, the control commands given should be motion directions rather than positions.

A convenient model for implementing sliding motions, while commanding velocities, is the *generalized damper* (Whitney (1977)). The model is specified by the following relationship between forces and velocities:

$$\mathbf{F} = \mathbf{B}(\mathbf{v} - \mathbf{v}_0), \tag{2.1}$$

where $\mathbf{F}$ is the vector of forces and torques acting on the moving object relative to its reference point, $\mathbf{v}_0$ is the commanded velocity vector, $\mathbf{v}$ is the actual velocity, and $\mathbf{B}$ is a *damping matrix*. It is generally convenient to take $\mathbf{B}$ as a diagonal or even identity matrix. One may view the vector $\mathbf{F}$ as the net force acting on the moving object due to other objects in the moving object's environment.

Equation (2.1) should be regarded as a control law specifying the ideal behavior of the world. The implementation of this ideal law is the responsibility of the underlying control system. Errors in approximating reality[1] to this control law are captured by the velocity error cone about the desired velocity $\mathbf{v}_0$. Recall how this cone represents the control error. The possible motion directions that an unobstructed object may follow when commanded to move in the direction $\mathbf{v}_0$ lie within the cone (see also Fig. 1.4). The effect of uncertainty on Eq. (2.1) is therefore to substitute for $\mathbf{v}_0$ any of the velocities in the error cone about $\mathbf{v}_0$.

[1] Reality is governed at a certain resolution by Newton.

In order to understand the manner in which a generalized damper relates sliding velocities and forces, consider Fig. 2.8. The commanded velocity is pointing into the surface and slightly to the right. Since motion into the surface is impossible, the actual velocity must be tangential to the surface. In the absence of friction, the reaction force formed by the surface is normal to the surface. Assuming an identity damping matrix, Eq. (2.1) indicates that the reaction force cancels the normal component of the commanded velocity. In general, given a commanded velocity, and a normal vector to a surface of contact, the resulting motion is tangential to the surface of contact, since the reaction force cancels the normal component of the commanded velocity.

It is possible to rewrite Eq. (2.1) in a fashion which clarifies the claim of the previous paragraph:

$$\mathbf{F}_{reaction} + \mathbf{F}_{applied} = \mathbf{B}\mathbf{v}, \tag{2.2}$$

where, using previous terms, $\mathbf{F}_{reaction} = \mathbf{F}$, and $\mathbf{F}_{applied} = \mathbf{B}\mathbf{v}_0$. In other words, one may think of the commanded velocity as specifying an applied force which is related to the reaction force by a first order differential equation.

2.2.2. Other Dynamics

The generalized damper is a special case of a system of dynamics described by the equation

$$\mathbf{F} = \mathbf{M}(\mathbf{a} - \mathbf{a}_0) + \mathbf{B}(\mathbf{v} - \mathbf{v}_0) + \mathbf{K}(\mathbf{p} - \mathbf{p}_0), \tag{2.3}$$

This system consists of an inertial term, a damping term and a stiffness term. Choice of the parameters $\mathbf{M}, \mathbf{B}$ and $\mathbf{K}$ determines the behavior of the system. With $\mathbf{M} = \mathbf{K} = \mathbf{0}$, the system reduces to a generalized damper. With $\mathbf{B} = \mathbf{K} = \mathbf{0}$, the system reduces to Newton's world. With $\mathbf{M} = \mathbf{B} = \mathbf{0}$, the system reduces to a *generalized spring*. See also Salisbury (1980).

The general system, and in particular, Newton's world is represented by a second order differential equation. The system of a generalized damper is described by a first order differential equation, while that of a generalized spring is described by a zeroth order equation.

2.2.3. Comparison of Generalized Damper to other Dynamics

The model of a generalized spring serves many useful purposes. For example, this model is often used in the implementation of hybrid controllers (see Mason (1983), and Raibert and Craig (1981)). Directions that are being force controlled are given small spring constants in the *stiffness matrix* $\mathbf{K}$, while directions that are being position controlled are given large spring constants. One of the difficulties with this model, however, is that it only indirectly captures motion directions. It is necessary to move the desired position $\mathbf{p}_0$ in order to create motion. Such a method is not completely satisfactory, as the underlying system is governed by

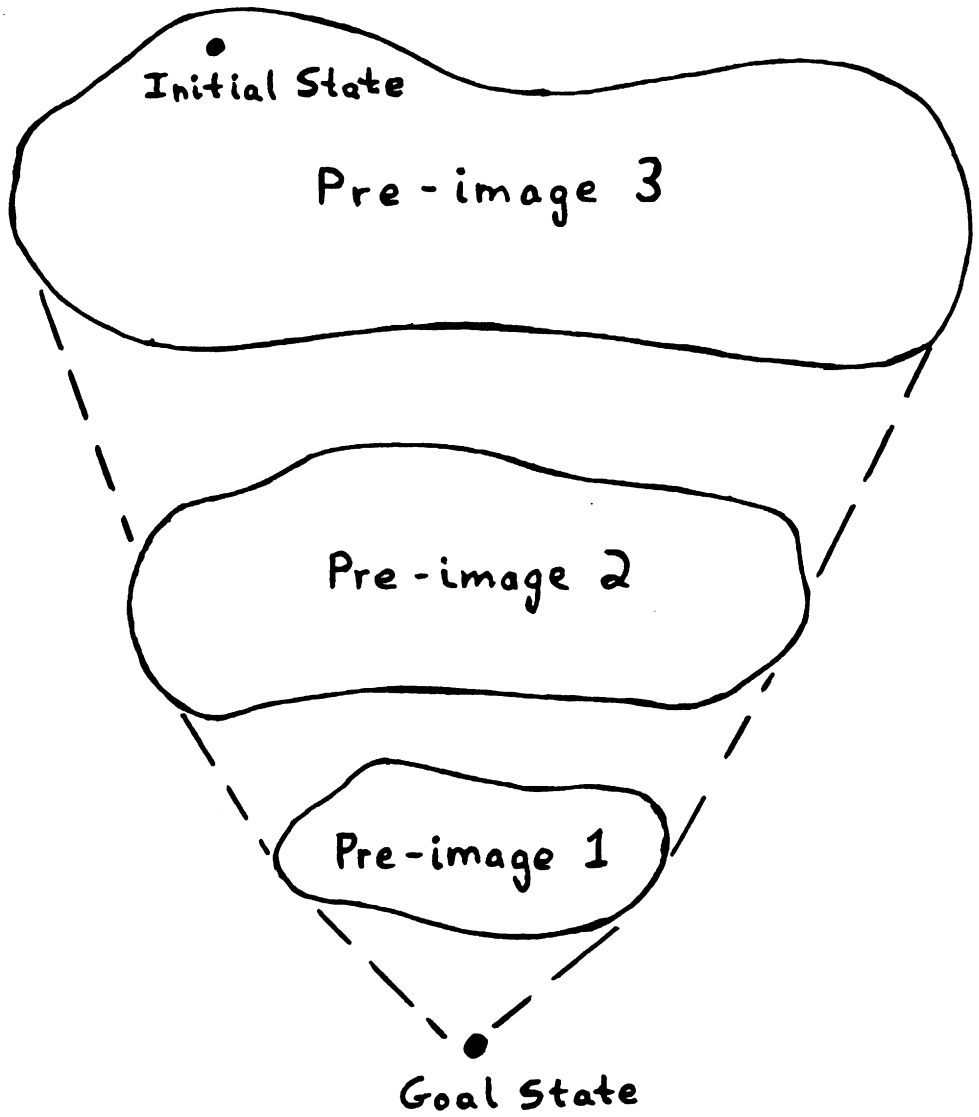


Figure 2.9. Conceptual view of backchaining. The pre-image results of one level are used as the goals for the next level. The lines abstractly represent paths between different levels of pre-images.

---

a zeroth order differential equation. Discontinuities in position are possible within the framework of the model. Unfortunately, position discontinuities complicate the representation of trajectories, which are ultimately the building blocks of a motion planner.

On the other extreme, Newton's world or the general equation are second order systems, hence are more difficult to model. In principle, of course, there is no problem. Being described by linear differential equations, the orbits of points given by these second order systems are uniquely determinable from initial conditions.

However, the geometrical tools required to model these orbits in the presence of uncertainty would require a phase space representation, rather than a simple configuration space. Such a representation would double the dimension of the underlying space. At least during a first pass, while investigating the basic problems of motion planning and uncertainty, it is desirable to avoid the complications of phase space.

The generalized damper model offers an attractive mean between the extremes of position discontinuities and phase space representations. The model is simple, hence serves as a convenient tool both for a human as well as a motion planner. The simplicity of the model manifests itself in three related areas.

- First, the generalized damper equation relates forces and velocities. Thus, thinking about directions is equivalent to thinking about forces.
- Second, the first order equation allows one to ignore second order terms such as inertial forces, centripetal forces and coriolis forces. Naturally, the control system which implements the generalized damper dynamics must account for second order terms. However, within the ideal world seen by the motion planner such forces may be ignored. The ability to ignore these forces is of particular interest within the context of modelling friction. The planner must avoid surfaces on which sticking can occur. A simple representation of friction is of benefit in recognizing such surfaces. Thus, being able to ignore second order terms considerably simplifies the planning process, without removing the central issues raised by motion planning in the presence of uncertainty.
- Third, the orbits of points in free space under generalized damper dynamics are straight lines. This also simplifies the planning process. In particular, it facilitates the determination of constraints during the backprojection phase of the planning process. This phase will be described in greater detail later.

2.3. Backprojections

The previous two sections have described two well established tools in robotics. The configuration space representation allows a formulation of tasks as motion planning problems involving a point, while the generalized damper dynamics treats these motions as straight lines through free space and tangential movements along obstacle surfaces. The rest of the chapter assumes the configuration space and generalized damper dynamics as tools. This section discusses an algorithm for determining regions from which certain motions are guaranteed to arrive at the task's goal region. The next section discusses a representation of friction assumed by the algorithm of this section.

2.3.1. The Planner's Plan of Pre-Images

Recall the objective of a planner. It should synthesize strategies specified by motions that exploit surfaces along which sliding can occur while avoiding surfaces on which sticking can occur. Additionally, the motions should terminate in unambiguous states, so that the plan executor can recognize the successful completion of a task.

Within the motion planning context, a task is specified by a desired set of geometrical relations. The input to the planner is a geometrical description of the world. Included in this description are the relationships of objects to one another, in particular, the initial and desired relationships of objects (see also Lozano-Pérez (1976)). The planner must determine a means of changing the relationship of objects from the initial to the goal state. For a single subtask, this entails moving one object. Thus, in terms of configuration space, the goal of the subtask is specified by a region in configuration space into which the point representing the moving object must move. The initial state is represented similarly, either by a region or by a point, depending on the accuracy with which the initial state is known or needed.

Clearly, the goal state is of primary concern. The initial state may be changed into intermediate steps, if required, but the goal state embodies the purpose of the task.

Therefore, the question to be asked is:

(1) "If I want to go there, where must I be, and how must I move, to get there?"

Of course, given the earlier discussion, two other questions follow immediately:

(2) "If I'm not where I should be to get where I'm going, how do I get to where I should be?"

(3) "How do I know when I've gotten to where I'm going?"

These questions are classical questions that arise in a number of artificial intelligence problems (see Nilsson (1980)). Their structure indicates an answer involving some form of subgoaling or chaining. In fact, Lozano-Pérez, Mason, and Taylor (1983) have proposed a formal system that uses backchaining to answer these questions and develop a methodology for planning motions in the presence of uncertainty.

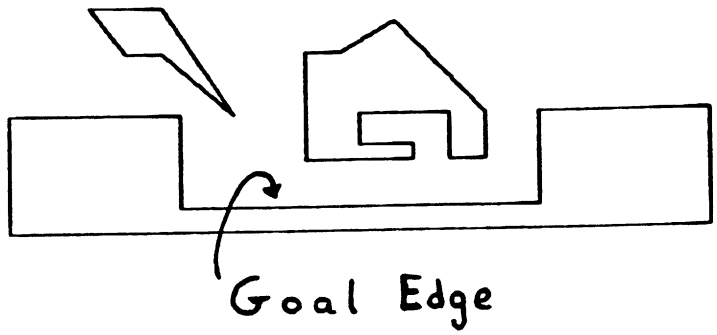


Figure 2.10. The task is to determine regions, and motion directions from those regions that will move a point to the goal edge.

The formal system envisioned by Lozano-Pérez, Mason, and Taylor answers question (1) by defining for any desired state the pre-image of that state. The pre-image is precisely the set of locations from which one can get to where one wants to go. In fact, the pre-image of a desired state is so defined that it also answers question (3). In other words, if one starts off in a pre-image location one will not only get to where one is going (question (1)), but also recognize that one has arrived (question (3)). The technical details of pre-images are not of importance to this discussion, but will be mentioned in later chapters.

Having answered the first and third questions, it remains to answer the second question. This answer is provided by backchaining. The scheme proposed by Lozano-Pérez, Mason, and Taylor may be visualized abstractly by the diagram of Fig. 2.9. In words, given a desired state, the planner constructs that desired state's pre-image. The pre-image is a collection of states, from which movement into the desired state is guaranteed. All motion directions are considered. Of course, one can view the pre-image itself as a collection of desired states. It is then possible to construct the pre-image of this collection, that is, it is possible to construct the pre-image of the pre-image. Proceeding in this fashion, the planner backchains from the first desired state, repeatedly creating new levels of pre-images until eventually one pre-image contains the initial state of the world.

Backchaining answers question (2), for it produces a sequence of pre-images, hence a sequence of motions, that leads to the goal. Thus the pre-images form a list of intermediate locations and motion directions that comprises the answer to question (2).

2.3.2. Approximating Pre-Images with Backprojections

The definition of pre-images does not provide an algorithm for their computation. In fact, it is not known whether it is possible to construct such an algorithm. In Chapter 3 of this thesis it will be shown that the general form of

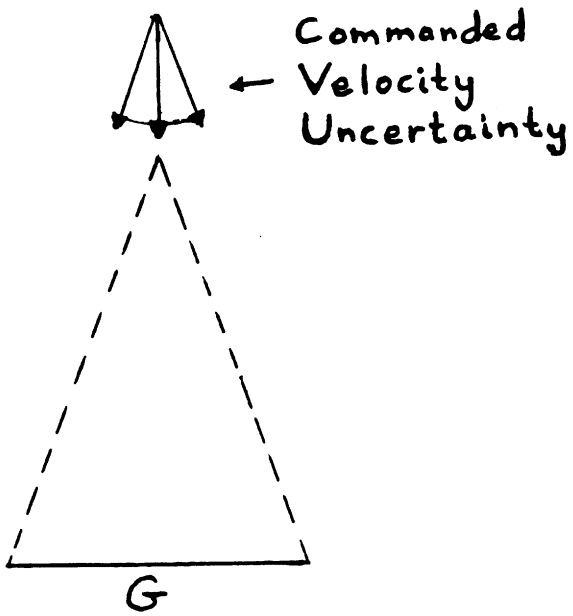


Figure 2.11. The goal is the edge. Given the commanded velocity uncertainty, the triangular region is the maximal region from which points are guaranteed to move to the edge.

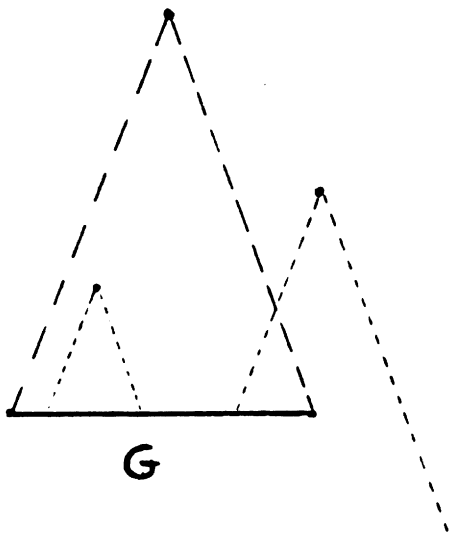


Figure 2.12. A point in the triangle is guaranteed to hit the edge. A point outside the triangle may miss the edge.

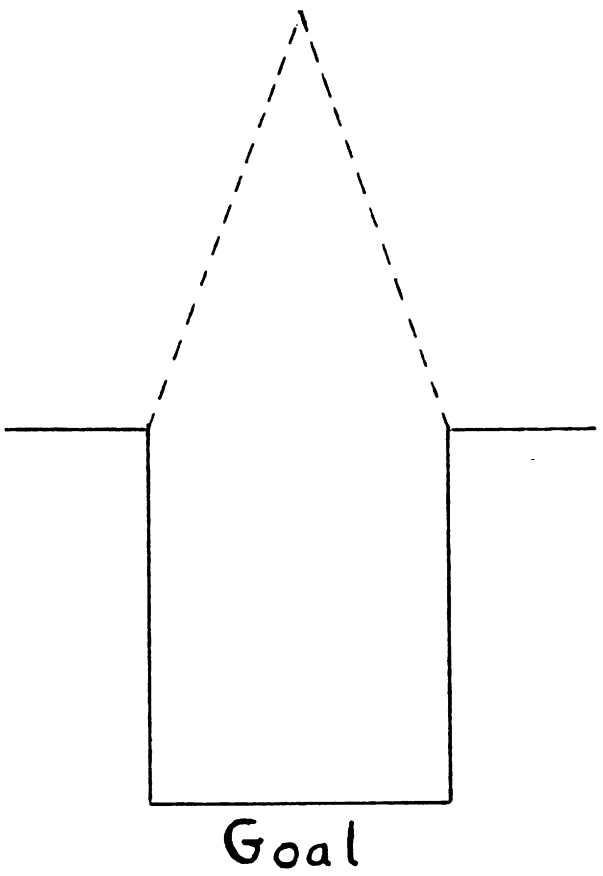


Figure 2.13. Walls that cause sliding towards the goal edge increase the backprojection region.

---

pre-images is uncomputable. Fortunately, this uncomputability is not a fundamental problem, as it applies to environments and tasks that do not generally arise in practice.

This thesis presents an algorithm for computing regions in space, called *backprojections*, from which motions in certain directions are guaranteed to enter the goal. The properties of backprojections are similar to those of pre-images. In fact it is possible to approximate certain classes of pre-images using backprojections. Thus backprojections form the primitive elements from which a planner can construct more complicated pre-images. The following subsections present an intuitive treatment of backprojections, while later chapters provide formal definitions and explore the detailed relationship of backprojections to pre-images.

2.3.3. A Two Dimensional Algorithm

Consider the case of a planner which is faced with the task of moving a point in the plane amongst polygonal obstacles. Given the tool of configuration space, this is not an artificial problem, as it corresponds to the problem of moving in the plane an object which is allowed to translate but not to rotate.

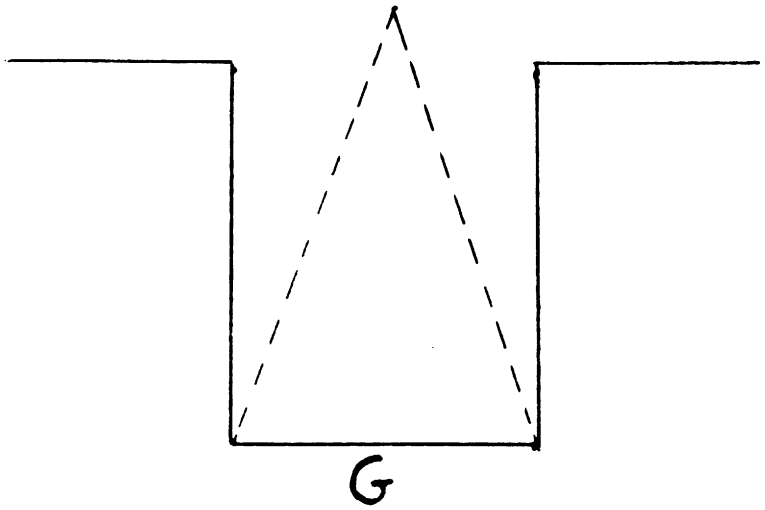


Figure 2.14. If sticking is possible on the walls, then the backprojection region must avoid the walls.

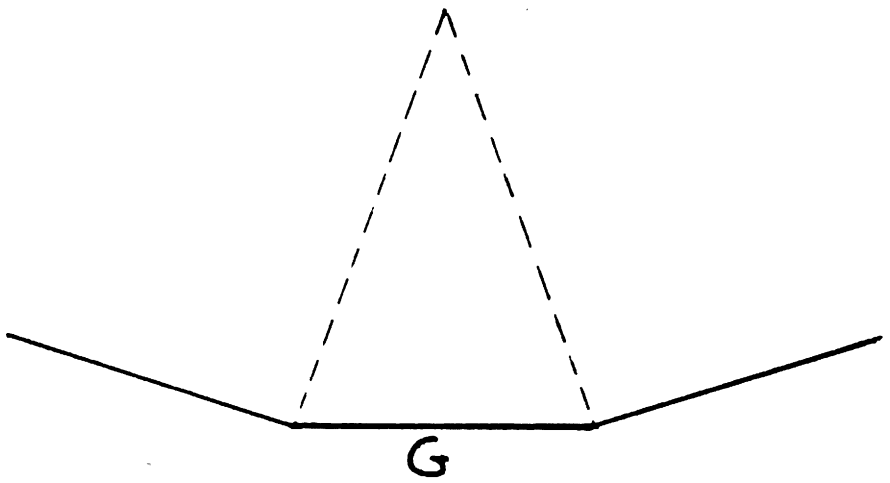


Figure 2.15. Frictionless edges may also cause sticking. This occurs if the edge normal points into the velocity cone.

In particular, consider the example of Fig. 2.10. The goal region is the edge indicated. The objective of the planner is to choose motion directions and regions in space from which those motion directions are guaranteed to enter the goal region. For every possible motion direction, the planner can construct a (possibly empty) region from which, using the given motion direction, the moving point is guaranteed to enter the goal.

2.3.3.1. Backprojection from an Edge

It is useful to consider a simpler example before attempting a solution of the original problem. Consider Fig. 2.11, which consists of a single edge in space. That edge is also the desired goal region. Suppose that the given motion direction is

straight down. Also suppose that the error cone about the desired velocity is as shown in the figure.

Recall that a backprojection region should be a region from which any motion commanded along the desired direction is guaranteed to enter the goal. However, the actual motion associated with a commanded velocity is known only to lie within the error cone about the desired velocity. Thus in order to guarantee entry into the goal region, it must be the case that all motion directions within the error cone are guaranteed to enter the goal region.

With this realization in hand, the planner can easily backproject from the goal edge, determining the triangular backprojection region shown in Fig. 2.11. It is clear that this region is correct. To verify the correctness, pick any point within the region, such as the point of Fig. 2.12. Now attach the velocity error cone to the chosen point. The resulting cone represents the possible locations of the point, having been commanded to move straight down. Since ultimately every trajectory within the cone enters the goal region, the original point is a valid backprojection point.

Also consider a point outside of the triangular region. Attaching the velocity cone to such a point shows that there are some trajectories that emanate from the chosen point that do not enter the goal region. Hence the point is not a valid backprojection point.

This is an extremely simple example. Nonetheless, it provides a clue regarding the structure of backprojection regions. Note that the boundary of the backprojection region consists of, other than the goal edge, two line segments that are parallel to the extreme rays of the velocity error cone. This observation will later be used to devise an algorithm for computing backprojections. The algorithm outlines the boundary of a backprojection region by erecting constraint rays that are parallel to the extreme rays of the velocity error cone about the desired direction of motion.

2.3.3.2. Backprojection Involving Sliding

The previous example did not consider the use of sliding surfaces. Consider the augmented example of Fig. 2.13. In this example the edge of the previous example has been surrounded by two vertical walls. The desired velocity is straight down, as in the previous example. Assume that the walls are frictionless, so that all velocities within the velocity error cone about the desired velocity cause sliding. In other words, if the moving point hits one of these walls, then the point will slide along the wall towards the goal edge.

The backprojection region for the new example is the region above the goal between the vertical walls and the dashed lines of Fig. 2.13. Notice how the walls have pushed the triangular region back from the goal, thereby enlarging the backprojection region. This example demonstrates the utility of sliding surfaces as guides to the goal.

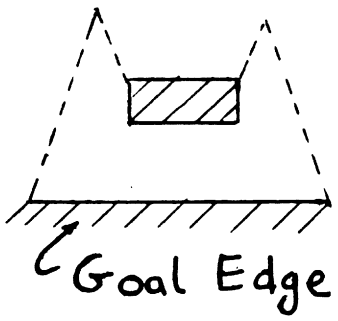


Figure 2.16. Backprojection region that is obstructed by an obstacle.

As a word of caution, recall that the planner should exploit surfaces that cause sliding towards the goal, while avoiding surfaces on which sticking might occur. Should the coefficient of friction on the walls be large enough to cause sticking for some velocity in the velocity error cone, then the planner can no longer guarantee that all motions within the error cone lead to the goal. Thus the planner would avoid the walls, producing instead the smaller backprojection region of Fig. 2.14.

Similarly, suppose the walls are not vertical, but tilted at an angle, as in Fig. 2.15. Recalling the generalized damper model of Sec. 2.2.1, it is clear that sticking can occur, even if the walls are frictionless. This is because for each of the walls, one of the velocities within the error cone is anti-parallel to the wall's normal (see Eq. (2.1)). Consequently, the wall can generate a reaction force that cancels the applied velocity. Should this happen, the point would cease to move. Thus the planner again could not guarantee entry into the goal, and so would avoid the walls.

Once again, it is noteworthy that the boundary of the backprojection region consists of, other than the goal edge, the surfaces along which sliding to the goal is guaranteed, plus line segments that are parallel to the extreme rays of the velocity cone. Notice also that the line segments emanate from the vertices of the offending walls. In essence, these boundary segments act as constraints, insuring that the moving point won't stray into forbidden territory.

2.3.3.3. Backprojection Obstructed by an Obstacle

Now consider the original edge example, but suppose that there is a rectangular obstacle above the edge, which would lie within the triangular backprojection region previously constructed. Clearly that triangular region is no longer appropriate, as it is possible to stick on the obstructing obstacle.

The planner must construct a region which avoids the top edge of the obstacle. Assuming that the vertical edges of the obstacle are frictionless, the backprojection

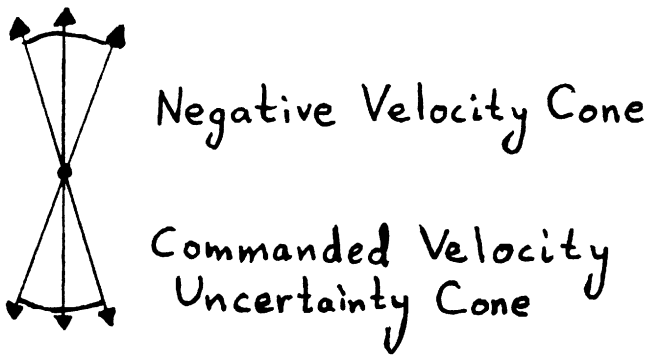


Figure 2.17. Relationship of negative velocity cone to commanded velocity uncertainty cone.

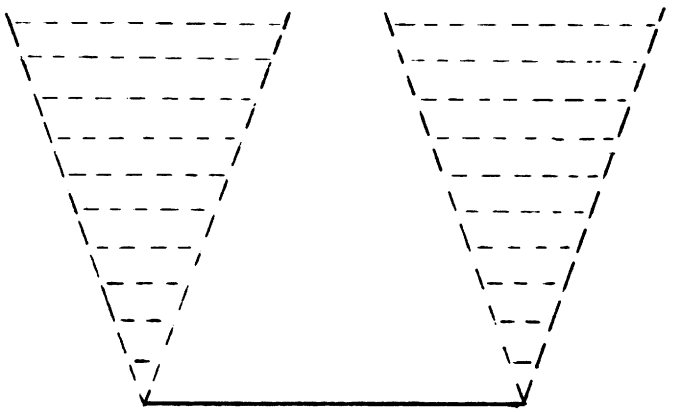


Figure 2.18. Erecting constraints at both vertices of an edge that is to be avoided.

---

of the goal edge is the region shown in Fig. 2.16. Once again, the boundary of the region consists of the goal edge, of obstacle edges along which sliding is possible, and of line segments parallel to the extreme rays of the velocity error cone. These line segments bound the edge on which sticking can occur, constraining the moving point away from the edge.

2.3.3.4. An Algorithm

The previous three simple examples have demonstrated three aspects of the backprojection algorithm. Specifically, the examples have shown how (1) the extreme rays of the velocity error cone outline portions of the backprojection regions, how (2) sliding surfaces extend the backprojection regions, and how (3) sticking surfaces, in particular, obstacles within the backprojection, can be avoided.

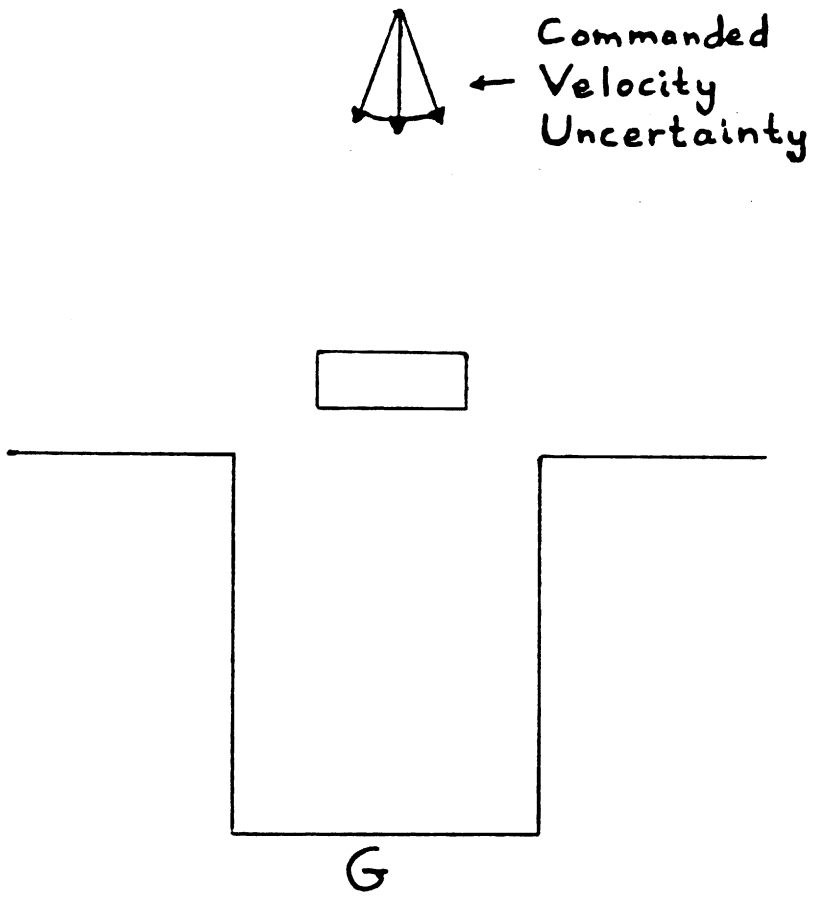


Figure 2.19. The goal is the bottom edge in the hole.

Observation (1) is of particular importance. In order to systematically make use of this observation, it is convenient to define the *negative velocity cone* as the cone which is formed from the velocity error cone by inverting all the velocity vectors. The negative velocity cone of the error cone used in the previous examples is shown in Fig. 2.17.

The velocity error cone represents the possible directions of motion. The trajectories in this cone are solutions to the first order differential equation of the generalized damper, with time increasing. In other words, the velocity error cone consists of the possible locations of a point in time, given that it starts at the apex of the cone. Inversely, the negative velocity cone represents the possible solutions to the generalized damper equation with time decreasing. Thus, the negative cone consists of all those points which could move to the apex of the cone. In other words, a point in the negative velocity cone could move to the apex of that cone given an appropriate effective velocity (or sequence of velocities) from the original velocity cone.

The negative velocity cone may be used to backproject constraints from surfaces that are to be avoided. Suppose that sticking might occur at a vertex. The planner

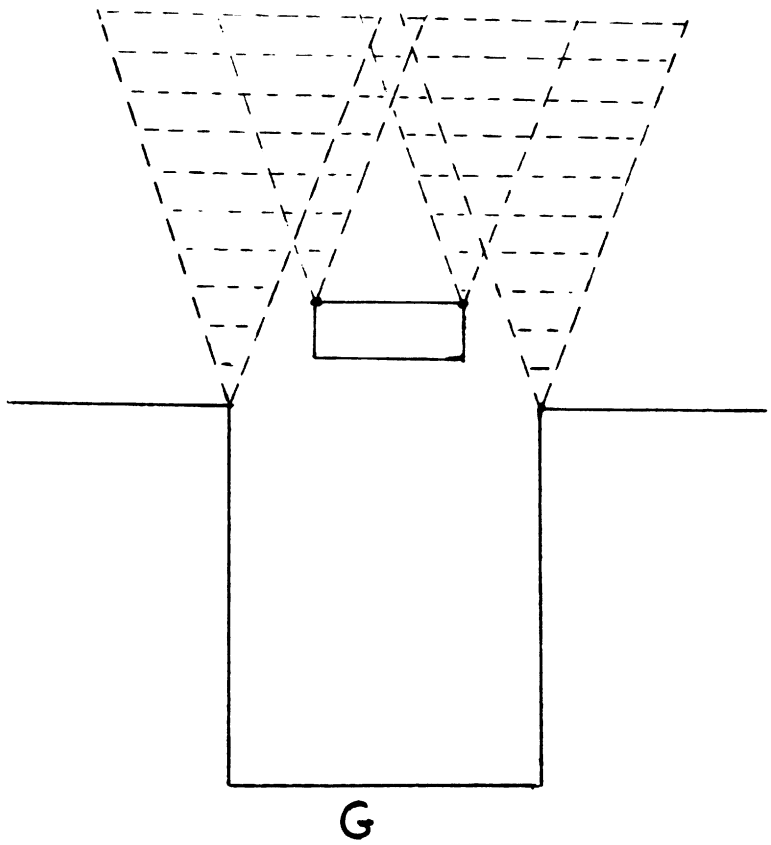


Figure 2.20. Constraints erected while backprojecting from the bottom edge.

---

should avoid that vertex. Since the negative velocity cone explicitly defines those points which could possibly hit and stick at the vertex, the planner need only erect the negative velocity cone at the offending vertex. Points in the resulting cone are to be avoided, while points outside the cone cannot possibly hit the vertex.

Similarly, suppose that sticking might occur on an edge. The planner can avoid the edge by erecting the negative velocity cone at both of the edge's vertices, as shown in Fig. 2.18. The cones and the regions between them consist of points that might move to the edge and stick, while the region to the left or right of both cones consists of points that cannot possibly hit the edge.

A backprojection algorithm is now evident:

(1) Consider every vertex in the environment. Mark every vertex at which sticking can occur, or which abuts a non-goal edge at which sticking can occur. Also mark a vertex if it abuts a goal edge and a non-goal edge, and if it is possible to slide away from the vertex on the non-goal edge.

(2) At every marked vertex erect the negative velocity cone. This involves erecting the extreme rays of the negative velocity cone at the marked vertex. Each ray is attached to the vertex at one end, and intersected with the nearest edge or ray in the environment to form the other end. Thus the rays create constraint edges.

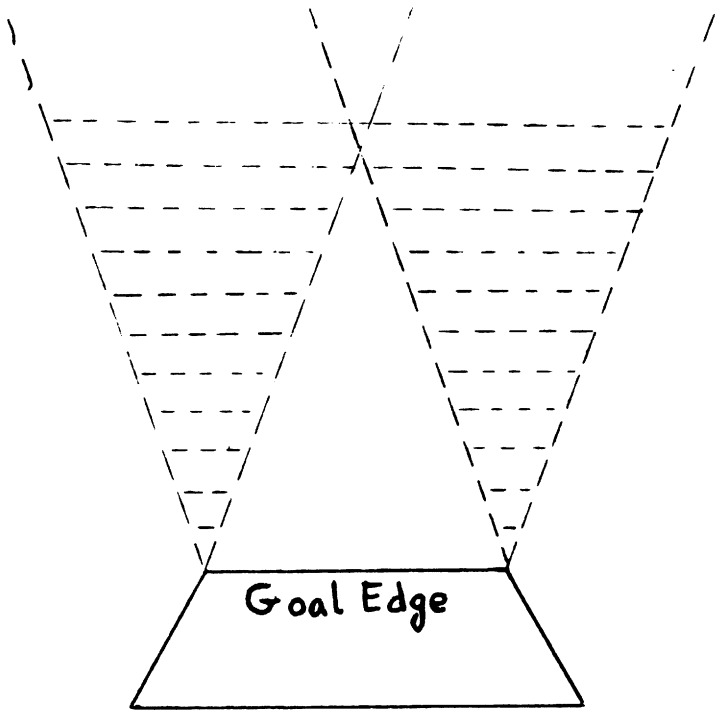


Figure 2.21. The goal is the top of the trapezoid. Since sliding is possible away from this edge at its vertices, the backprojection algorithm must erect constraints at the vertices.

---

(3) Beginning at the goal edge(s) trace out the backprojection region. This is accomplished by regarding both obstacle and constraint edges as edges with an interior and an exterior side. The interior side of a constraint edge is the side which lies interior to the negative velocity cone from which the edge was originally constructed. Tracing out the region entails tracing along an edge until encountering a vertex. Assume that all edges incident at this vertex are ordered by angle. The current edge in the trace has an exterior and an interior side. Choose the edge which is adjacent to the current edge on its exterior side. This adjacent edge is used to continue the trace.

In deciding whether sticking can occur on a vertex, assume that the vertex can produce reaction forces which are linear combinations of the reaction forces that each of the abutting edges can produce. A vertex is the intersection of two one dimensional surfaces. Thus this definition is consistent with the notions of superposition of forces and intersection of constraints.

In order to gain a complexity estimate, note that each vertex in the environment

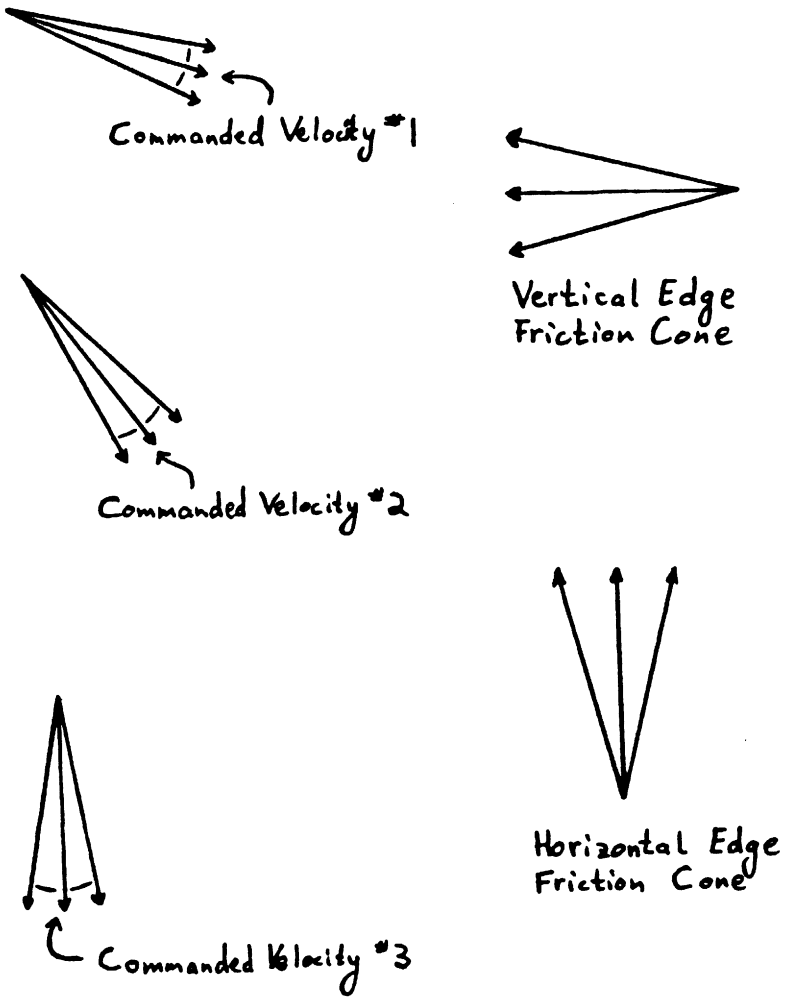


Figure 2.22. Three different commanded velocities, along with their uncertainty cones. These velocities are used in Fig. 2.23 to backproject from the goal edge of Fig. 2.10. The three velocity uncertainty cones are shown in the left column. For comparison, the friction cones associated with vertical and horizontal edges are shown in the right column (only one orientation is indicated). The first commanded uncertainty cone partially points into the friction cone associated with vertical edges. The third uncertainty cone points into the friction cone associated with horizontal edges. The second uncertainty cone does not point into either of the friction cones.

---

can contribute two constraint rays. Thus the algorithm erects $O(n)$ constraint rays, where n is the number of vertices. Given that the obstacles are polygons, the number of edges is also $O(n)$. Consequently, the intersection of rays with the environment can be computed in $O(n^2)$ time. Finally, the trace can be performed in $O(n)$ time. The overall complexity of the algorithm is therefore $O(n^2)$. Given that the ray intersection portion of the algorithm can be speeded up to $O((n+c)\log n)$ time, the final complexity of the algorithm is actually $O((n+c)\log n)$. Here c is the number of intersections.

As an example, consider how the algorithm would determine the backprojection region for the example of Fig. 2.19. The velocity error cone is the same as for the

previous examples. The edges are assumed frictionless. Note, however, that sticking can occur on any horizontal edge, since the velocity uncertainty cone contains a velocity vector anti-parallel to the edge normal.

Sticking can occur at the top edge of the obstacle and at the top edges of the walls surrounding the goal edge, since the normals for these edges point straight up, opposing the commanded velocity. Therefore, the algorithm erects the negative velocity cone along each of these edges (see Fig. 2.20).

Beginning at the goal edge, the algorithm traces out the backprojection region. First it traces up the left wall, whereupon it encounters the constraint erected at the top of the wall. The algorithm then traces along this constraint edge, until it encounters the constraint edge arising from the left vertex of the top edge of the obstacle. It traces down this constraint, encounters the left edge of the obstacle, traces down that edge, encounters the bottom edge of the obstacle, and so forth.

In order to understand the second part of step (1), recall that the planner should exploit surfaces that cause sliding towards the goal. Edges which might cause sliding away from the goal naturally should be avoided in much the same way that edges which cause sticking are avoided.

Consider the example of Fig. 2.21. The goal edge is the top edge of the trapezoid. Notice that it is possible to slide away from the goal edge when in contact with one of the edges abutting the goal edge. The desired velocity is straight down, with the usual type of error cone. Under the velocities given by this error cone, sticking can only occur at the goal. Thus there are no vertices that need to be marked on the basis of sticking. Nonetheless, clearly the backprojection of the goal edge should be the indicated triangular region of Fig. 2.21. The second part of step (1) insures that the constraints defining this region are erected.

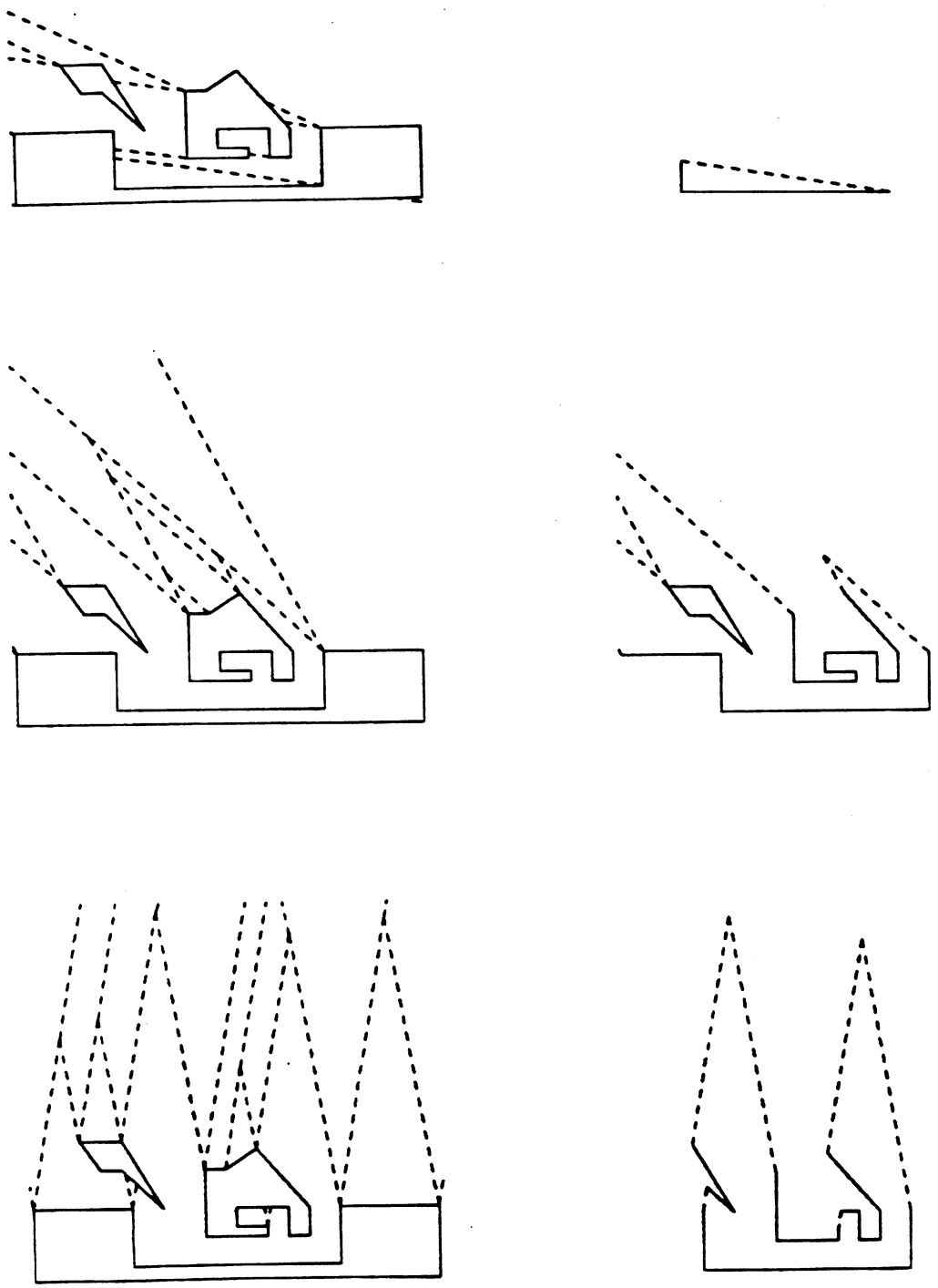


Figure 2.23. Backprojection regions for the three commanded velocities and associated uncertainty cones shown in Fig. 2.22. The constraints erected by the backprojection algorithm are shown in the left column. The resulting backprojection regions are shown in the right column.

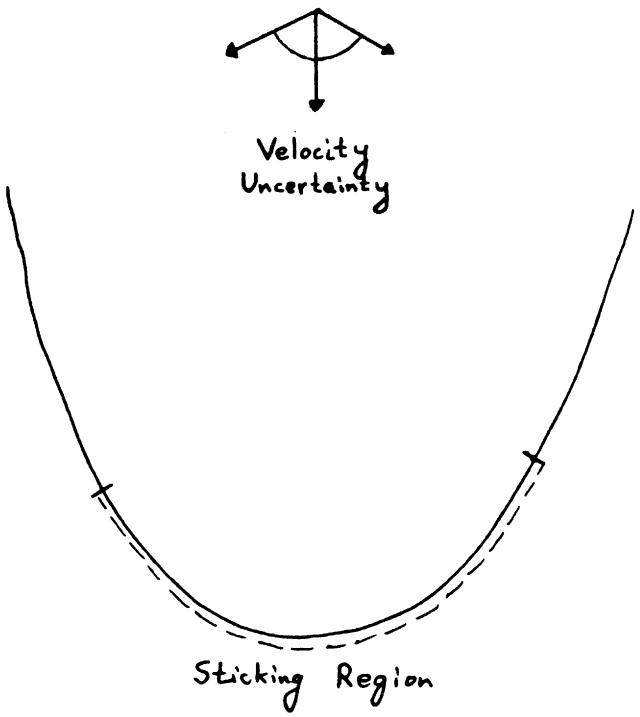


Figure 2.24. Sticking is possible on part of the parabolic arc for the velocity uncertainty shown.

It is now a simple matter to determine, for any desired motion direction, the backprojection of the goal edge in the example of Fig. 2.10. The coefficient of friction on each of the edges in this example is assumed to be one fourth. Fig. 2.22 shows three different commanded velocities, along with their uncertainty cones. Also shown in Fig. 2.22 are the friction cones for horizontal and vertical edges. 2.23 shows the constraints and regions determined by the backprojection algorithm, for each of the commanded velocities and uncertainty ranges shown in Fig. 2.22. The left column in Fig. 2.23 displays the constraints erected for each of the commanded velocity uncertainty cones. The right column displays the resulting backprojection regions.

As is indicated in Fig. 2.22, the first commanded velocity uncertainty cone partially points into the friction cone associated with vertical edges. The third commanded velocity uncertainty cone points into the friction cone associated with horizontal edges. The second uncertainty cone points into neither of these friction cones. The consequences of these intersections is evident in the backprojection regions of Fig. 2.23.

For the first commanded velocity, sliding cannot occur on vertical edges with outward normals pointing to the left. Consequently, the hole's right wall at the goal

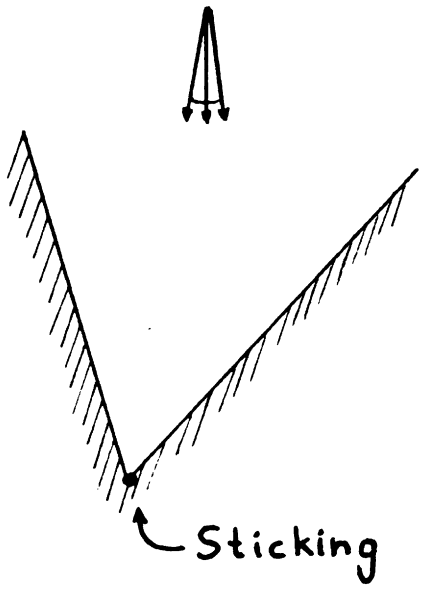


Figure 2.25. Edges can cause sliding to non-goal vertices at which motions may become trapped.

---

edge does not aid in forming a backprojection region. For the third commanded velocity, sliding cannot occur on horizontal edges with outward normals pointing up. Consequently, the horizontal edges at the top of the hole surrounding the goal edge do not aid in forming a backprojection region. In contrast, for the second commanded velocity, sliding is possible both on vertical and horizontal edges. This is evident from the backprojection region formed for this commanded velocity. The backprojection region includes both vertical and horizontal edges that facilitate sliding towards the goal edge.

2.3.3.5. Comments

The algorithm presented above tacitly assumes that obstacles in the environment are polygons. The advantage of assuming a polygonal environment lies in the discreteness and finiteness of motion states. Given a constant coefficient of friction over an edge, if sticking can occur anywhere on the edge, then it can occur everywhere. In other words, the property of sticking is invariant over the edge. Similarly, the directions of sliding determined from the generalized damper equation are invariant over the edge. Thus, it is sufficient to consider an edge's vertices both when deciding whether sticking can occur on the edge, and when deciding in which directions sliding is possible. Furthermore, having decided that an edge is to be avoided, it is sufficient to erect constraints at both endpoints. Since there are only a finite number of vertices and edges in the environment, construction of the constraints is a simple discrete and finite process.

Suppose that the surfaces of concern are not edges, but arbitrary continuous

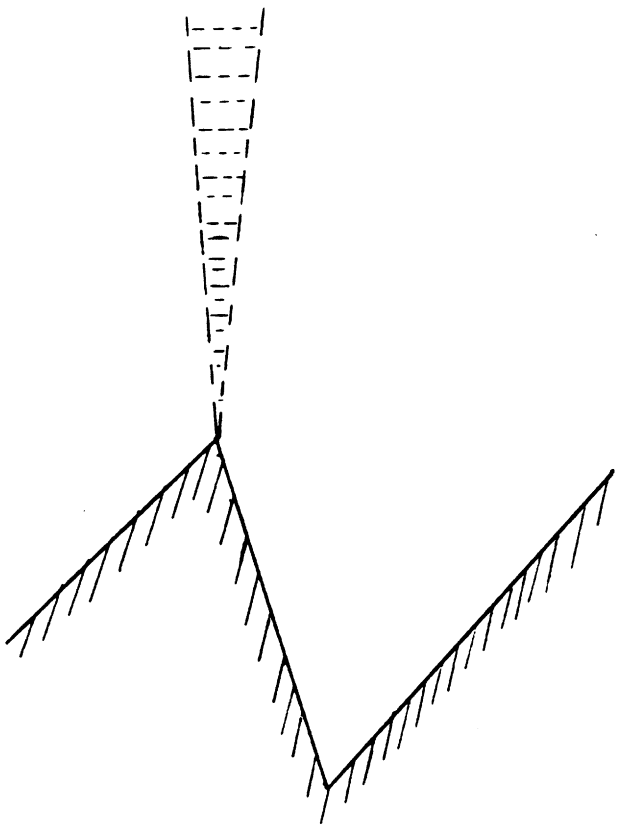


Figure 2.26. Edges that can lead to traps are automatically avoided by the backprojection algorithm.

(or perhaps smooth or analytic) curve segments in the plane. Additionally, perhaps the coefficient of friction varies in a continuous fashion along these curves. It is no longer possible to restrict the algorithm's concern to the endpoints of these curves. Instead it is necessary to explicitly examine the curves, determining subportions on which sticking can occur. The resulting examination may still lead to a finite number of points at which constraints should be erected. In fact, for any practical application, the number of constraints should be finite. However, the location of these constraints is not known a priori.

For example, for the frictionless parabolic surface of Fig. 2.24, with the specified velocity error cone, sticking can occur throughout the subsegment at the bottom of the arc, as indicated in the figure. Sticking cannot occur anywhere else on the arc.

A surface which yields an infinite number of pairwise disjoint sections at which sticking can occur is the classical curve $x \mapsto sin(\frac{1}{x})$. In practice the planner need not worry about such surfaces, since they do not represent actual objects.

It is worthwhile to mention another subtle point that hides under the the benevolent protection of the simplicity of two dimensions, but will be forced out of hiding by the more stringent demands of higher dimensions. Suppose that sticking

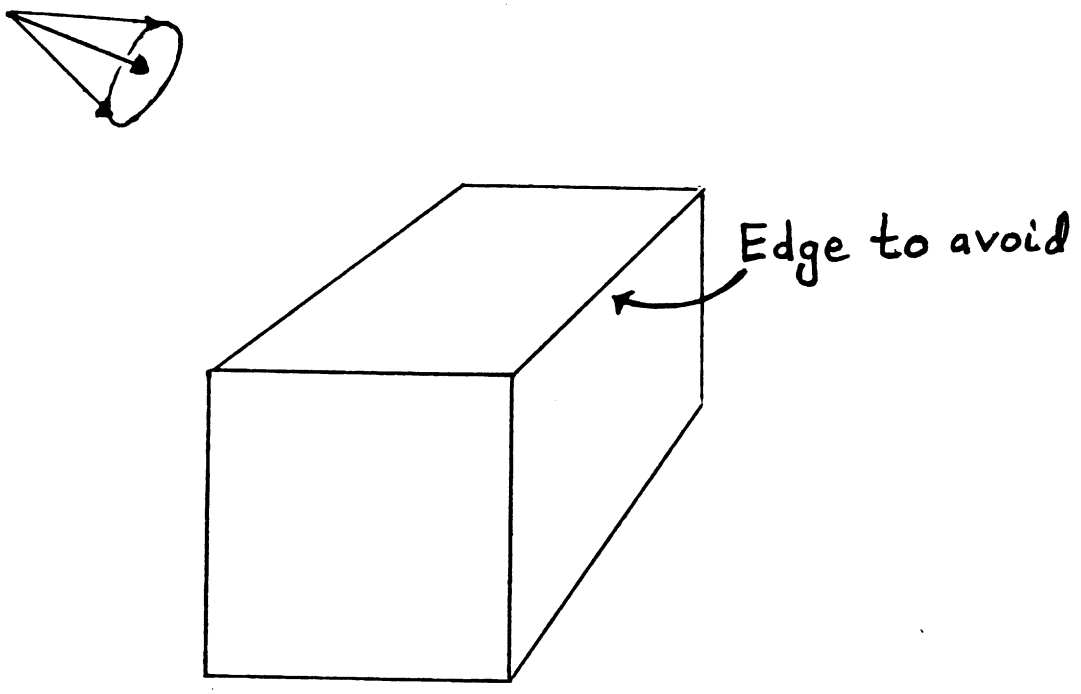


Figure 2.27. The task is to avoid the top right edge, given the specified velocity uncertainty cone.

---

can occur at a non-goal vertex, although it does not occur on the edges abutting the vertex. Such a situation could arise in the example of Fig. 2.25. Clearly, the edges abutting the vertex are to be avoided, as it is possible to slide on them towards the vertex and stick.

No explicit steps are taken in the 2D algorithm to avoid these edges. Instead, the algorithm operates indirectly. In order to believe this, consider the other vertex of one of the edges. The algorithm may erect a constraint at this vertex, if sticking is possible there. Fig. 2.26 demonstrates a case in which such a constraint would be erected. If a constraint is erected, then the edge is automatically avoided. Suppose no constraint is erected at the other vertex. In order for one of the edges to be included in the backprojection region, it would be necessary for the edge to appear in a trace of edges beginning at a goal edge. This is impossible, since one of the following conditions must hold:

(1) At some point there is a shared vertex between two edges, such that sticking occurs at the vertex. In other words, either sticking occurs on some edge, or sticking occurs at a vertex because the orientation of the edges is changing enough to cause sticking, as in Fig. 2.26.

(2) Sliding away from the goal edge can occur at an abutting edge.

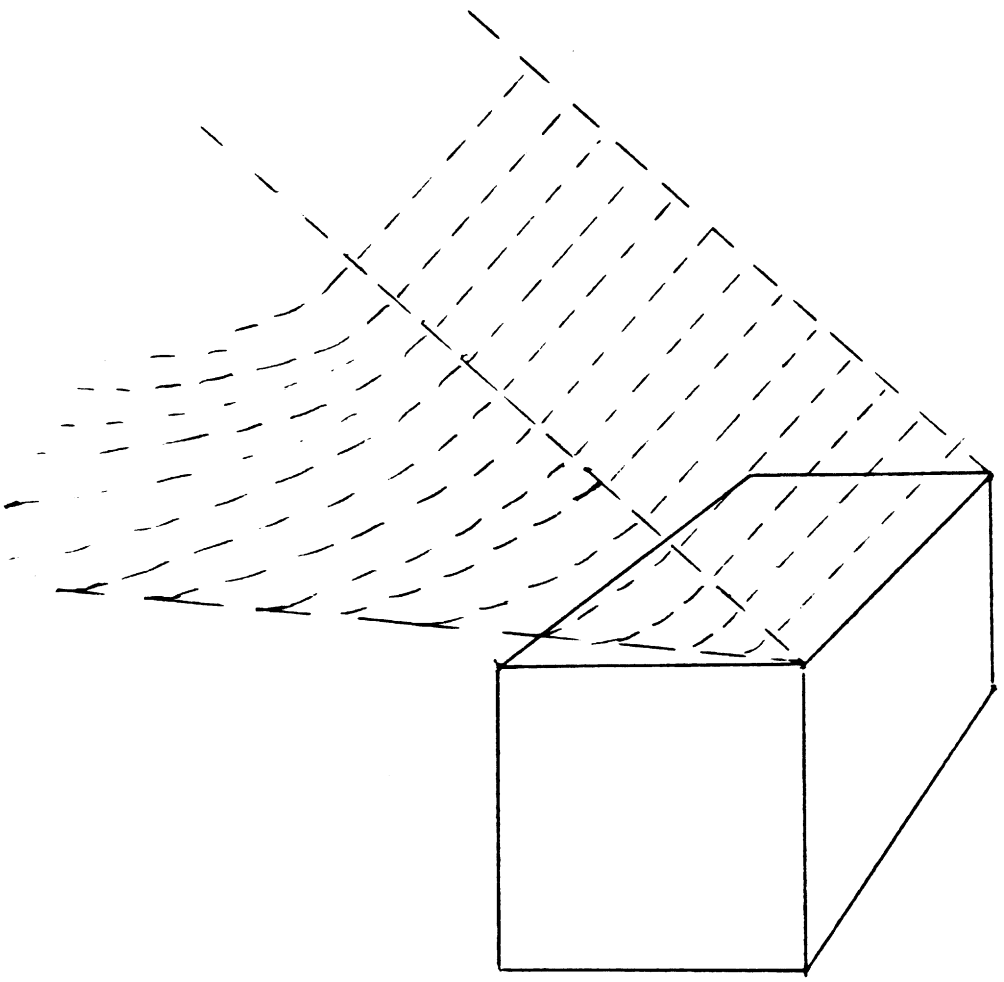


Figure 2.28. Constraints erected in free space in order to avoid the top right edge.

In either case, a trace would never reach the edge which is to be avoided. Thus the algorithm indirectly avoids edges on which sliding can lead to a vertex at which sticking can occur, although no sticking can occur on the edge itself. This fortune appears by virtue of the simplicity of surfaces in the plane. For surfaces in higher dimensional spaces it will be necessary to devise a slightly more elaborate algorithm.

2.3.4. A Three Dimensional Algorithm

The objective of a planner is to automatically derive motions that use sliding surfaces as guides, thereby circumventing uncertainty. Fig. 1.9 displayed one such strategy for inserting a peg in a hole. The commanded motion was the same

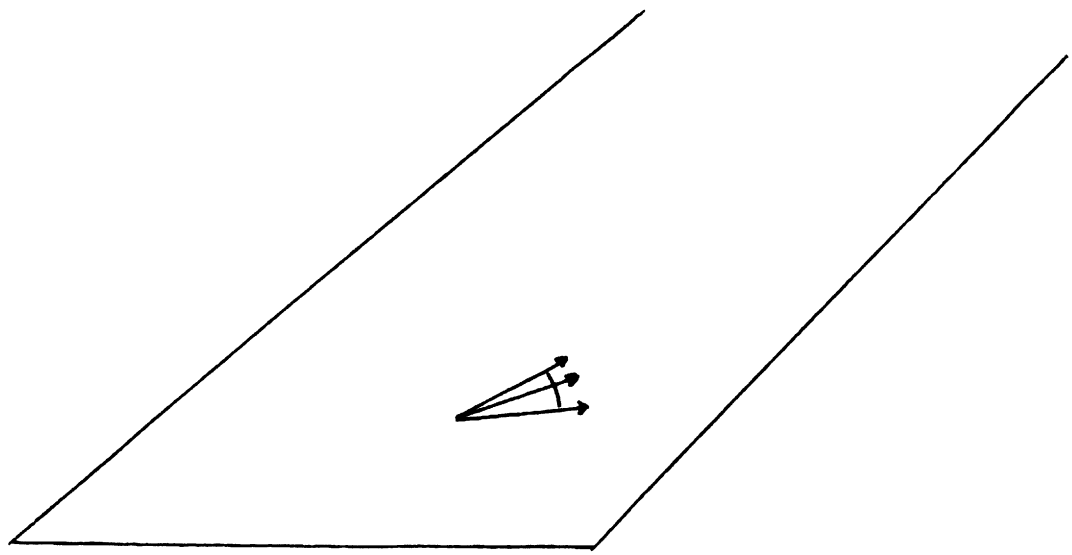


Figure 2.29. Projection of the velocity cone onto the top face of the cube.

throughout, namely down and towards the right. Thus the plan executor could command the motion once, then let the peg move until it achieved its goal. The peg started in free space, moving down and to the right until it hit the edge at the top left part of the hole. Upon hitting the edge, the peg began to slide towards the hole, then tumbled down to the goal edge at the bottom of the hole. The plan executor did not need to worry about the precise location of the guiding edge, or consider the accuracy of the peg-in-hole clearance ratio, or examine the effect of friction on the motion. The planner had already considered all these issues, then produced a strategy that was guaranteed to succeed.

The previous subsection outlined an algorithm for determining one step motion strategies. Recall that these one step motions form the building blocks of a larger strategy which is constructed from the single step motions via backchaining. The algorithm assumed a two dimensional space. This subsection extends the algorithm to higher dimensions. The algorithm will be stated for three dimensions, in particular, for the three dimensional configuration space corresponding to the parameter space of a planar object with two translational and one rotational degrees of freedom. The underlying principles of the algorithm are applicable to higher dimensional spaces. Thus the algorithm applies to the six dimensional motion planning problem arising from the task of moving in real space a rigid object with three translational and three rotational degrees of freedom.

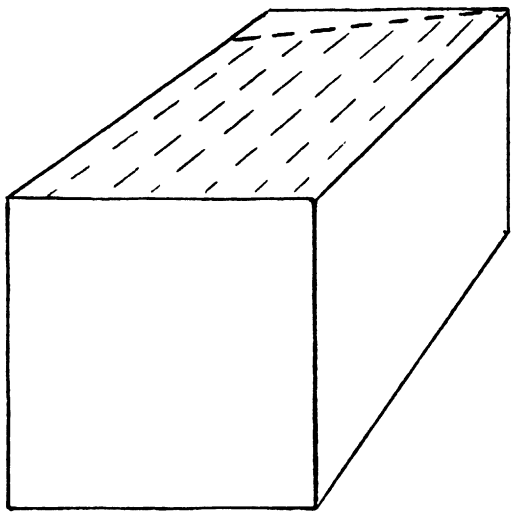


Figure 2.30. Sliding constraints erected in order to avoid sliding to the top right edge on the top face.

2.3.4.1. Basic Tenets

The guiding principle for the 2D algorithm was to backproject constraints from locations in the environment that were to be avoided. Because of the simplicity of two dimensions, the algorithm could easily backproject the constraints. It had merely to erect the negative velocity cone at the undesirable locations. This could be done by intersecting the extreme rays of the negative velocity cone with the edges in the environment.

In higher dimensions the underlying principle is exactly the same. Specifically, the algorithm should use the negative velocity cone to describe regions of space from which it is possible to move into regions where sticking is possible. Regions not so constrained, that are connected to the goal, comprise the desired backprojection. Motions from these backprojection regions are guaranteed to lead to the goal.

This principle effectively says, "Don't go somewhere, if it is possible to get from there to somewhere bad." The reason for using the negative velocity cone in determining what regions to avoid, is its usefulness in representing geometrically the solutions of the generalized damper differential equation.

Solutions to the generalized damper differential equation are trajectories. Given that the commanded velocity in this equation is subject to uncertainty, the solutions are actually classes of trajectories. Trajectories that encounter points at which sticking might occur, or which are offensive in some other fashion, define regions

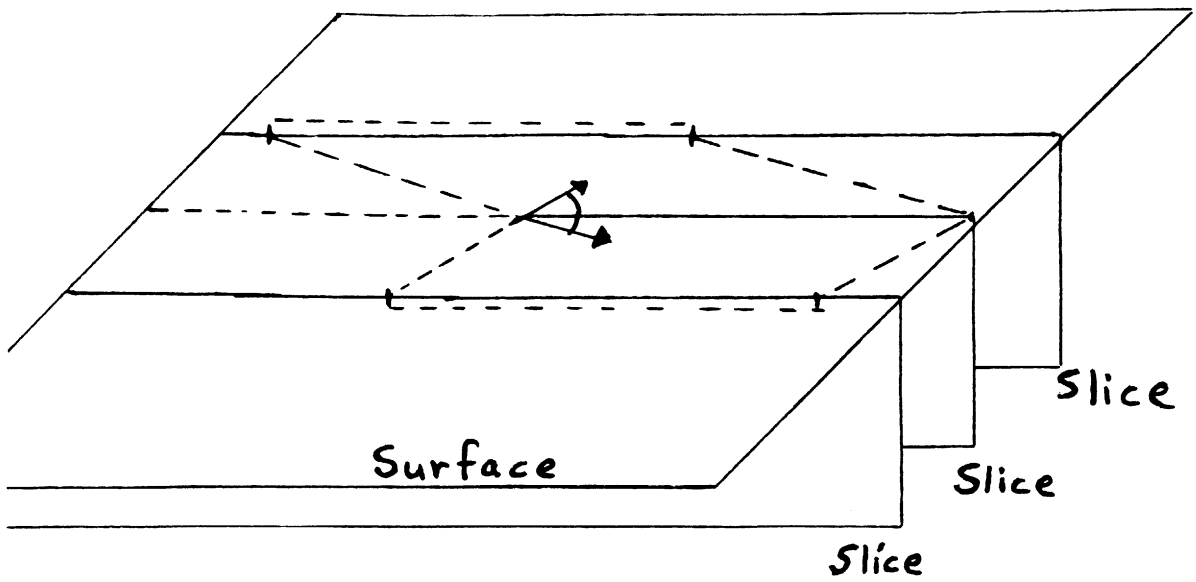


Figure 2.31. Backprojection of sliding constraints between slices. The horizontal surface is represented by the three slices shown. Assume that the right portion of the middle slice should be avoided. Also assume that when in contact with the surface, the range of sliding velocities is given by the indicated cone. Erecting the negative of this velocity cone along the portion of the slice that is to be avoided, shows that the regions outlined by the dashed lines should also be avoided.

---

of space that are to be avoided. The negative velocity cone captures the geometry of these regions, thereby providing natural constraints for use by the planner.

The negative velocity cone, however, only provides solutions to the generalized damper equation for free motion, that is, motion not in contact with obstacles. As noted in the comments to the 2D algorithm, it is in general also necessary to avoid surface regions from which it is possible to slide into undesirable regions. Thus, the planner must solve the differential equation of the generalized damper when in contact with surfaces of obstacles. In short, it is necessary to perform backprojections along surfaces as well as through free space. This surface backprojection occurred indirectly within the 2D algorithm, but must be handled explicitly for higher dimensional spaces.

2.3.4.2. Backprojection Along Surfaces —— An Example

An example should clarify the last point of the previous subsection. In this example, the problem is one of moving a point in a three dimensional frictionless space. Consider the rectangloid in Fig. 2.27. Assume that the the velocity error cone is the cone shown. Assume further that the planner has decided that the

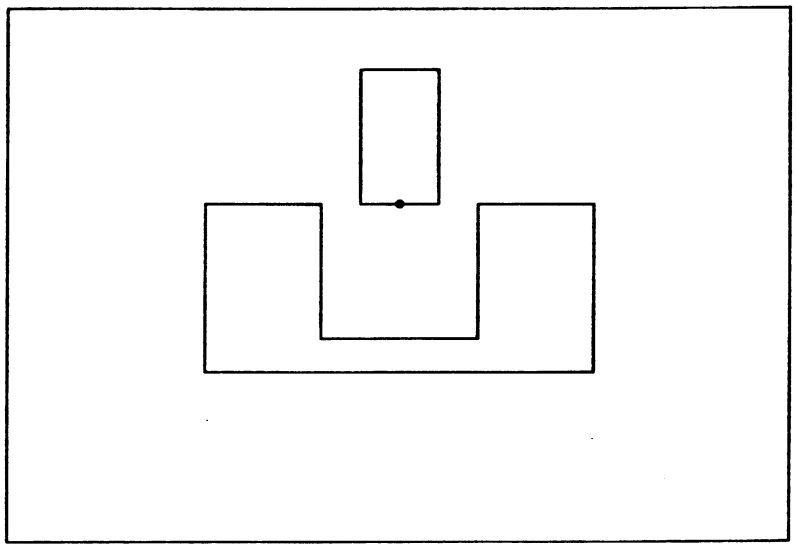


Figure 2.32. A peg-in-hole task. This task is considered in the following figures.

edge indicated is to be avoided. Erecting the negative velocity cone along the edge generates the constraints shown in Fig. 2.28.

This alone, however, is not sufficient to avoid the edge. It is still possible for the point to encounter the edge by sliding along the top face of the cube. Suppose that the point is in contact with this face. Then, given the commanded motions possible from the velocity error cone, the resulting motions along this surface, using the generalized damper equation, are those shown in Fig. 2.29. It is now a simple matter to backproject constraints along the surface. Given the effective velocity cone along the surface, the algorithm need merely construct the negative of this cone and backproject from the edge in the usual fashion. The resulting constraints on the surface are shown in Fig. 2.30. Finally, in order to avoid the region thus constructed, the algorithm must perform a 3D backprojection of these constraints.

2.3.4.3. Backprojection Along Surfaces —— Non-linear Equations

The previous example was very simple, for it involved a linear frictionless surface. Even adding a constant coefficient of friction to the surface would only slightly complicate matters. Unfortunately, allowing more general surfaces, and varying the friction along the surfaces, can drastically alter the form of the damper equation. The orbits of points on the surface are no longer easily invertible straight lines.

This problem becomes evident even when considering the configuration space of a planar object with two translational and one rotational degrees of freedom.

The configuration space surfaces in this three dimensional space are non-planar ruled surfaces. Furthermore, as will become evident in later sections and chapters, the description of friction is complicated. Friction varies in a non-linear fashion over the surfaces. Thus the damper equation during contact is non-linear. The computation of constraints is correspondingly difficult.

2.3.4.4. A Formal Procedure

In principle, the previous discussion has provided a formal procedure for computing backprojections in spaces of arbitrary dimension. The procedure is:

(1) Determine regions of space which are to be avoided. Specifically, mark all locations at which sticking can occur. Also, mark all locations abutting the goal region from which it is possible to slide away from the goal.

(2) Select a marked region. Backproject from this region the constraints determined by the negative velocity cone. This step entails erecting constraints in free space and along surfaces. The constraints for the free space backprojection are simply those of the negative velocity cone, while the constraints for the backprojection along surfaces are those of a projected velocity cone. The projected velocity cone is formed by solving the damper equation with contact for the surface in question. Note that the projected velocity cone may vary over the surface. The constraints are erected by intersecting the appropriate cone with the environment. The region between the intersection points and the anchor points of the cone determines a new region which is to be avoided. Mark this region.

(3) Repeat Step (2) until there are no more marked regions.

(4) Trace out a volume beginning at the goal region. The trace is a multi-dimensional trace along the surfaces and constraints in the transitive closure of those abutting the goal region.

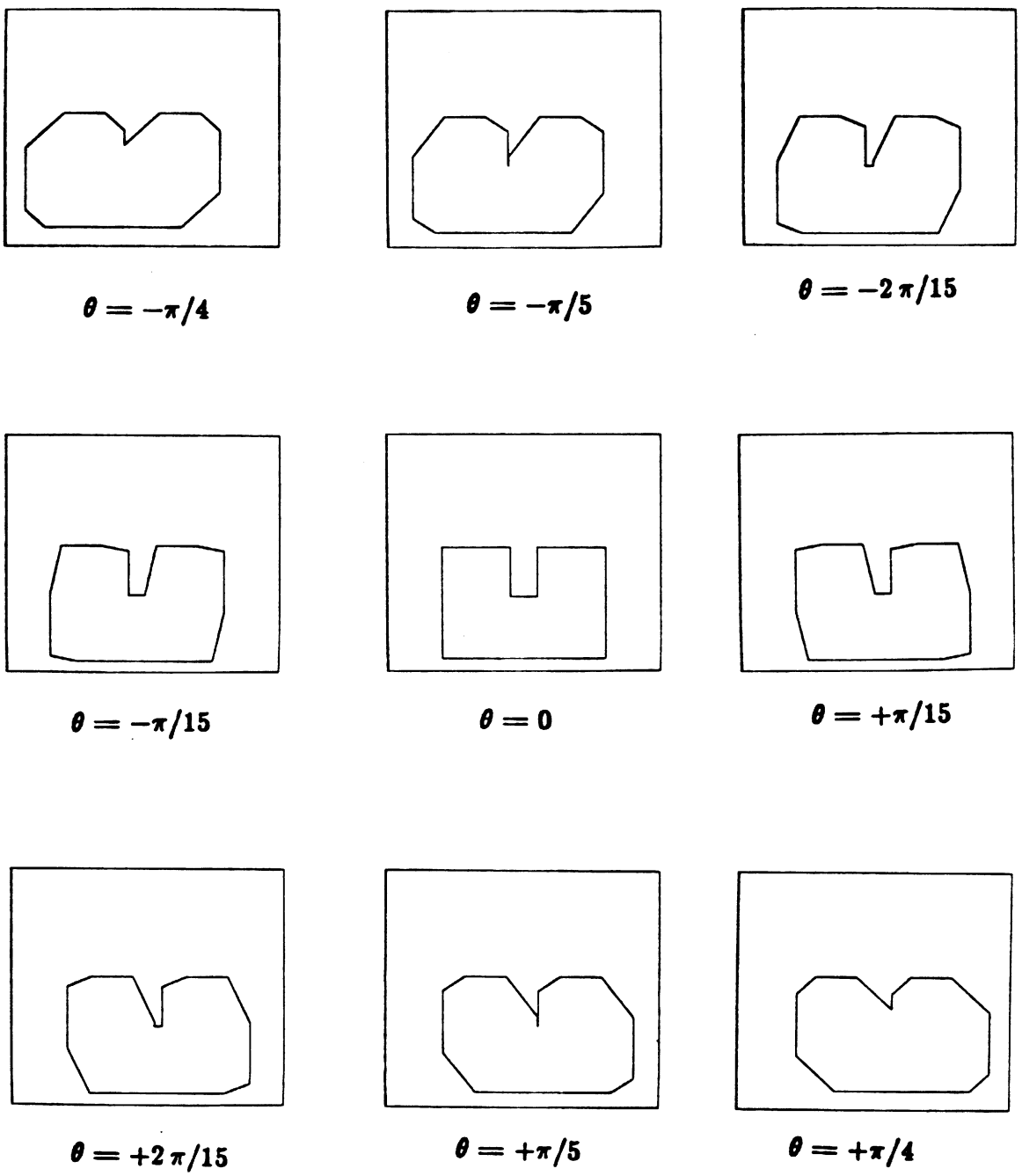


Figure 2.33. Several configuration space slices of the peg-in-hole task. The goal is the bottom of the hole for the configuration space slices corresponding to a (nearly) vertical orientation of the peg. In the figure, the goal is in the slice for $\theta = 0$

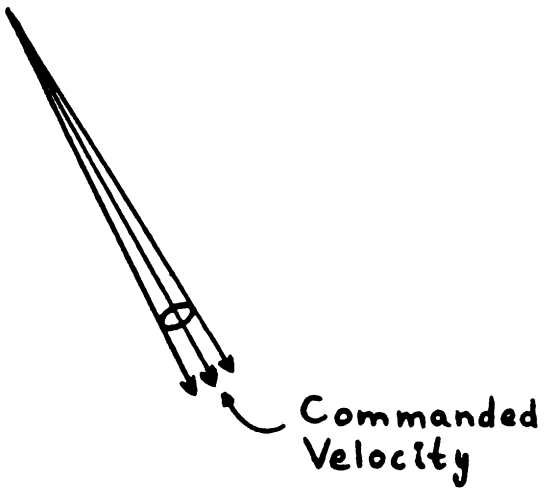


Figure 2.34. Commanded velocity and associated uncertainty cone for the peg-in-hole backprojection example. The cone is three dimensional. The commanded velocity is down and to the right, with a slight error in all three dimensions.

2.3.5. A Three Dimensional Slice Algorithm

While the previous section has presented a formal procedure for computing backprojections, that procedure is subject to the difficulties of solving a non-linear differential equation in order to compute the constraints that need to be backprojected along surfaces. The damper equation may become non-linear once the effect of friction while in contact with a configuration space surface is included. It is desirable to consider approximate algorithms that compute backprojections by locally linearizing the damper equation. Local linearization entails linearizing both the surfaces on which contact can occur, as well as linearizing the description of friction over these surfaces.

The following subsection presents an algorithm for computing backprojections in the three dimensional configuration space of a rigid polygonal object moving in the plane with two translational and one rotational degrees of freedom. The obstacles in the plane are also assumed to be polygons. The algorithm uses two dimensional slice approximations corresponding to fixed orientations of the moving object. Backprojection is performed both within a given slice as well as across slices. The algorithm generalizes immediately to the six dimensional configuration space of a translating and rotating rigid object, using three dimensional slices.

2.3.5.1. Statement of the Algorithm

The algorithm expects as input a series of slices corresponding to the (x, y) configuration spaces of the moving object for given discrete orientations. As the object and the obstacles are all polygons, each of the slices is also a collection of polygons. Since it is necessary to erect constraints across slices, it is useful to indicate the connectivity between slices. In other words, an edge or vertex should

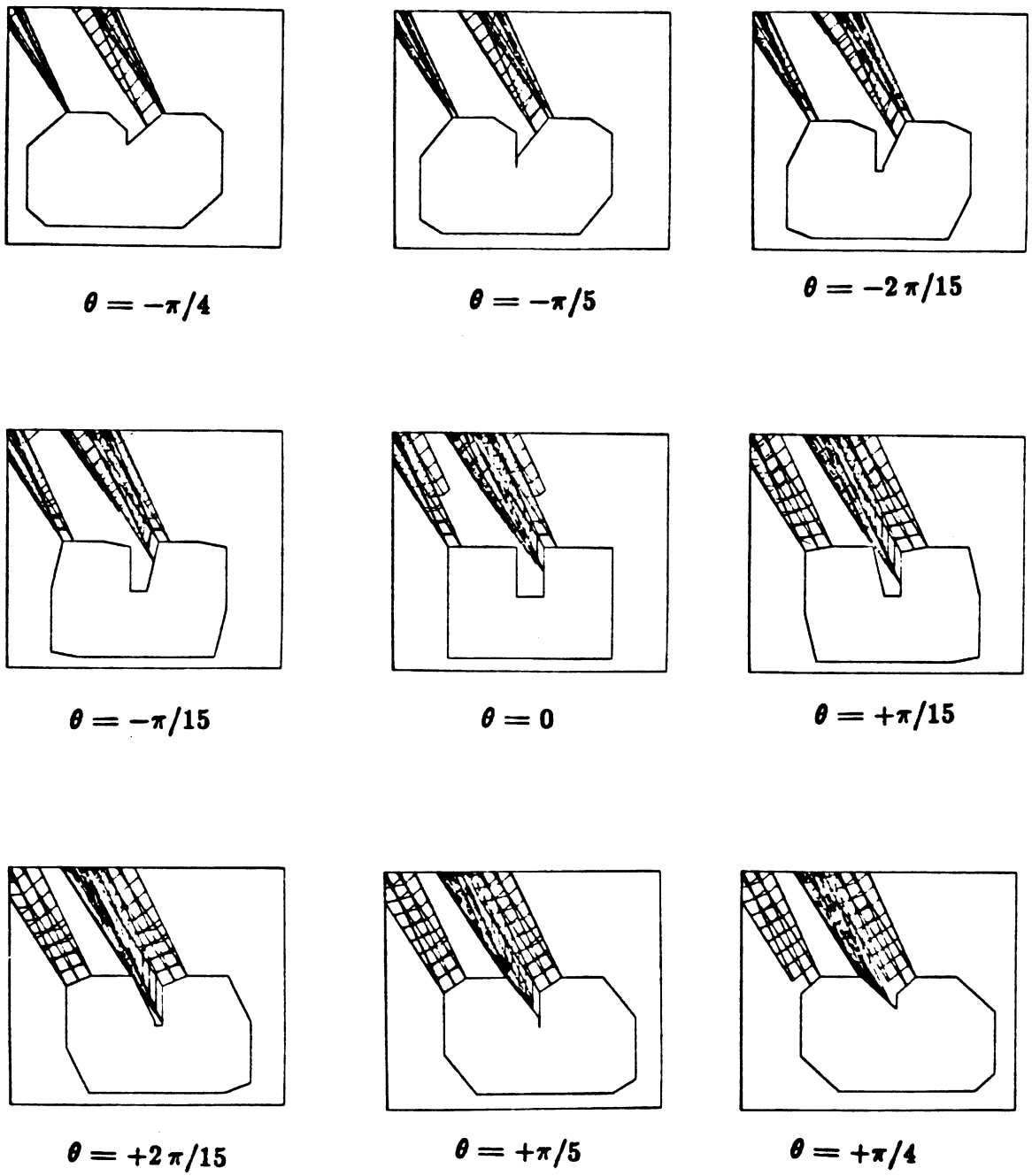


Figure 2.35. Constraints erected by backprojecting from the goal at the bottom of the hole. The commanded velocity uncertainty is shown in Fig. 2.34

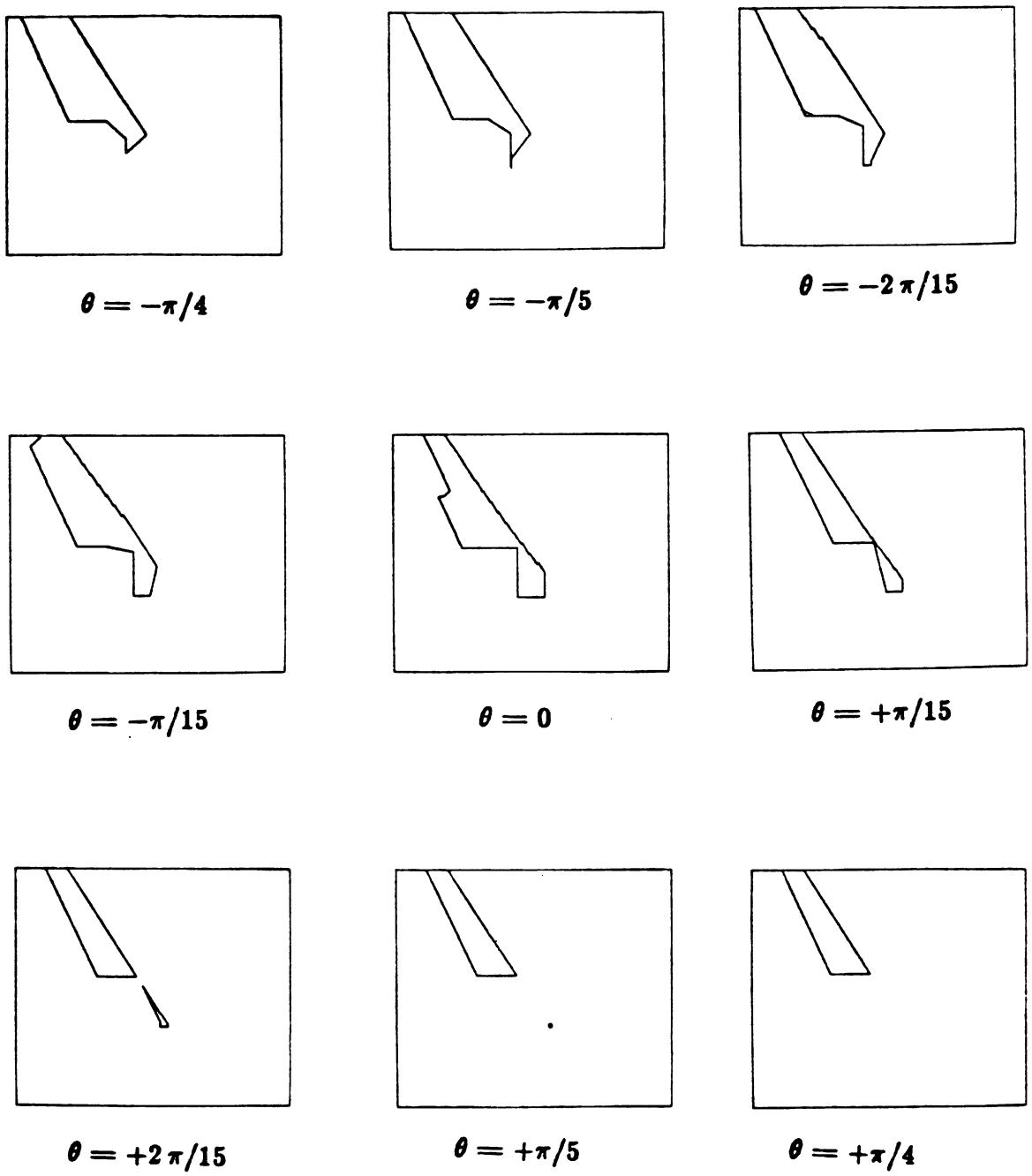


Figure 2.36. The resulting backprojection regions determined from Fig. 2.35.

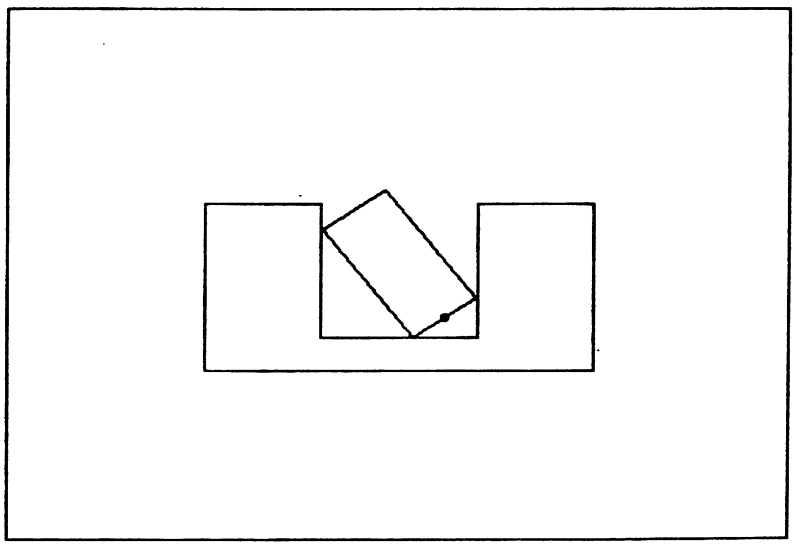


Figure 2.37. In this orientation the peg may stick in the hole. The commanded velocity is down and to the left, pushing into the corner of the hole.

---

contain pointers to edges and vertices in adjacent slices which are on the same configuration space surface as the given edge or vertex.

The algorithm assumes that configuration space surface normals and friction do not change between neighboring edges in adjacent slices. Additionally, since configuration space normals and friction may vary even over an edge in a given slice, the algorithm should cut such edges into smaller edges over which the change in friction is fairly small. Thus the algorithm can assume that friction is constant both between slices and over a given edge. This assumption comprises the linearization of the damper equation.

The algorithm is:

(1) For each slice, compute the sliding directions for the edges and vertices in the slice. In other words, solve the damper equation for contact at the edges and vertices, using the velocity error cone to specify the range of commanded velocities. The negative of the sliding directions thus computed corresponds to the projected negative velocity cones mentioned in Sec. 2.3.4.2. By assumption of local linearity, these may be used to perform the surface backprojections discussed earlier.

(2) For each slice, determine the (non-goal) vertices and edges at which sticking can occur. Mark these. Also mark any vertices and edges abutting the goal at which it is possible to slide away from the goal.

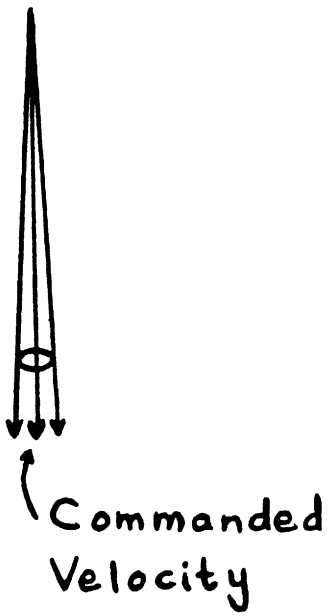


Figure 2.38. Commanded velocity and uncertainty cone for the backprojection of Figs. 2.39 and 2.40. The cone is three dimensional. The commanded velocity is straight down, with a slight error in all three dimensions.

---

(3) Select a marked vertex or edge. Perform a 3D backprojection from the vertex or edge, using the negative velocity cone. Mark the region interior to the cone portion thus constructed. Perform a 2D backprojection using the sliding information calculated in Step (1). The 2D backprojection entails determining whether it is possible to slide to the given vertex or edge from regions in both the given slice as well as the two adjacent slices. This information is readily available from the sliding information of Step (1) and the connectivity pointers of the input. Any such regions should be marked.

(4) Repeat Step (3) until there are no more marked regions.

(5) Trace out two dimensional regions in each of the slices that contains a goal edge. This trace is identical to the trace used in the 2D backprojection algorithm (Sec. 2.3.3.4). Using the connectivity information, determine edges in adjacent slices from which sliding is possible to edges in the regions already traced out. Trace out regions containing these edges. Repeat this process of tracing out across slices, until the regions traced out within a slice shrink to a point, or all slices have been considered.

In order to understand the 2D backprojection of Step (2), consider Fig. 2.31, which depicts portions of three adjacent slices. Imagine performing a 2D backprojection from the edge in the middle slice, and suppose that the range of sliding directions possible on the edge is given by the cone indicated. The algorithm would mark the slice regions delineated by the dashed lines, as regions from which sliding onto the given edge is possible.

2.3.5.2. An Example

Fig. 2.33 depicts several configuration space slices of the peg-in-hole task of Fig. 2.32. The large clearance ratio in this example was chosen for display reasons, in order to make the configuration space hole and the backprojection constraints clearly visible.

The desired velocity is down and to the right, with a velocity error cone as indicated in Fig. 2.34. The real space coefficient of friction is assumed to be one-fourth. See Sec. 2.4 and Chapter 4, for an analysis of friction in configuration space.

Fig. 2.35 shows the constraints erected by the slice algorithm, while Fig. 2.36 shows the resulting backprojection regions. Note how tracing out across slices (Step (5)) can generate regions that slowly shrink to a point.

The resolution of orientations considered by the backprojection algorithm was four times that indicated by the figures. Specifically, slices were constructed for orientations ranging between $-\pi/4$ and $+\pi/4$, in increments of $\pi/60$.

In configuration space, friction may vary over a surface. In other words, friction need not be constant over a particular edge in a configuration space slice. Consequently, for each edge in a configuration space slice, friction cones were computed at both the edge's endpoints and at the edge's midpoint. The composite of these three friction cones was used to decide whether sticking could occur on the edge. In computing sliding directions over the configuration space surface patch represented by the edge, the edge's midpoint was considered. The configuration space normal and friction cone at the midpoint were used to project the velocity uncertainty cone onto the configuration space surface. These computations were fairly crude approximations to the actual sticking regions and sliding directions.

For each vertex in a configuration space slice, the friction cone at the vertex was computed by considering all the endpoint friction cones arising from the edges meeting at the vertex. Similarly, the sliding directions at the vertex were computed by considering the intersection of all the configuration space surfaces associated with the edges meeting at the vertex. These friction cones and sliding directions were fairly exact.

The goal region was the bottom of the hole for orientations that were nearly vertical. The constraints erected take advantage of edges in the environment that facilitate sliding towards the goal edge and cause rotation to a vertical orientation. These edges are the vertical edges surrounding the hole and the horizontal edge at the top left part of the hole.

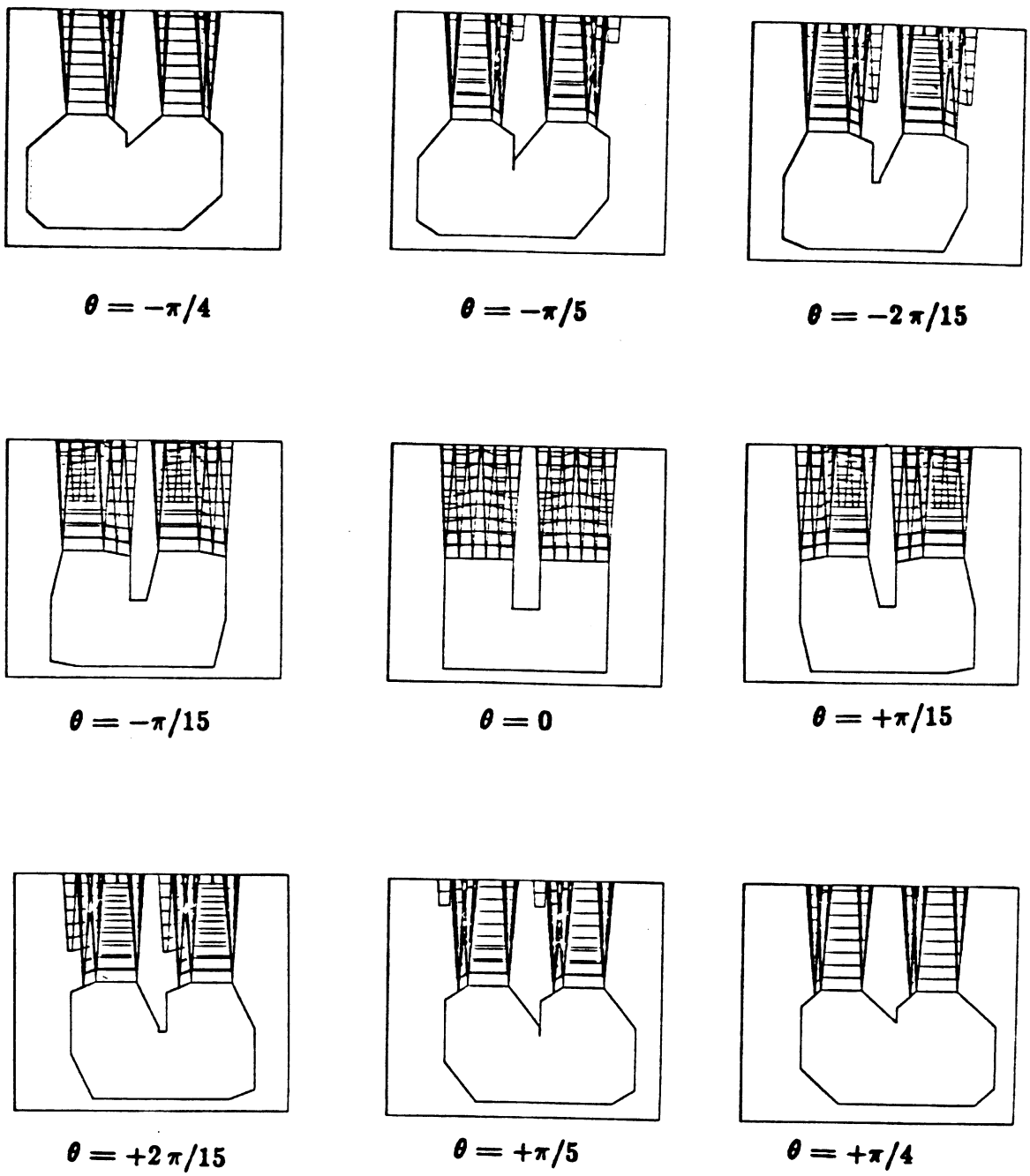
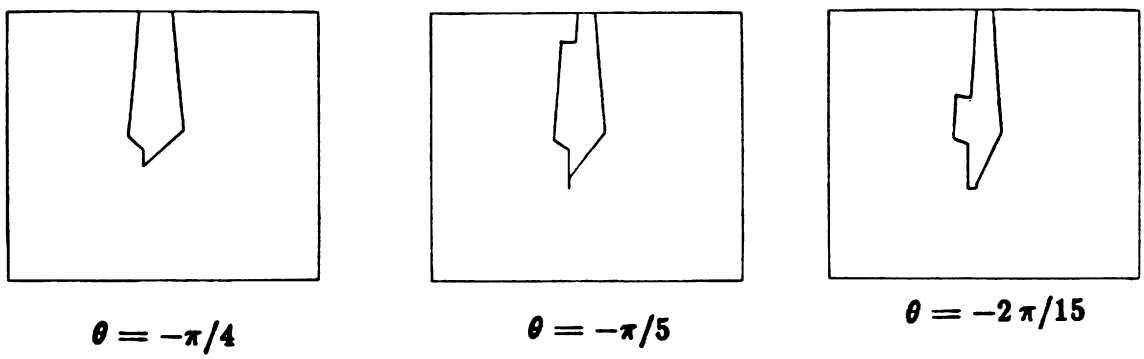
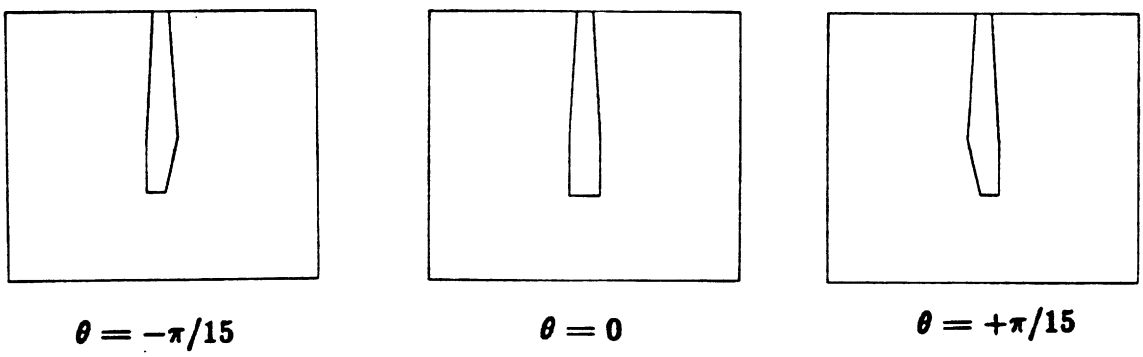


Figure 2.39. Constraints erected by backprojecting from the goal at the bottom of the hole. The commanded velocity uncertainty is shown in Fig. 2.38





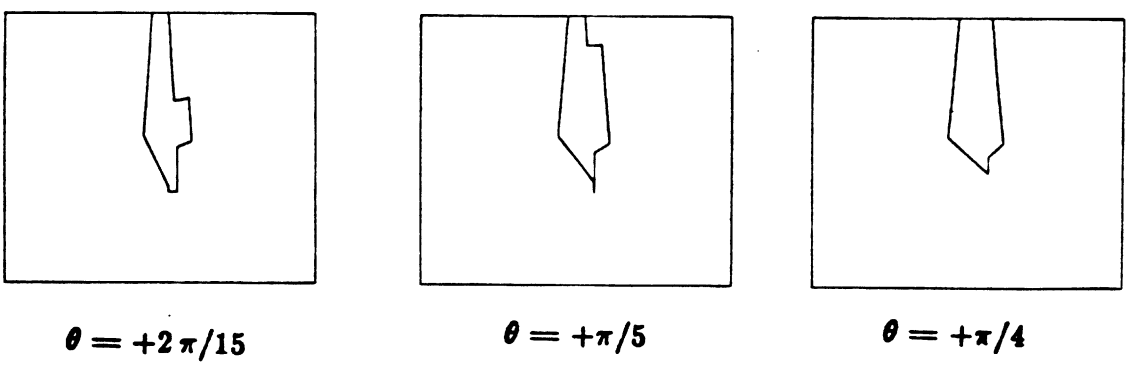


Figure 2.40. The resulting backprojection regions determined from Fig. 2.39.

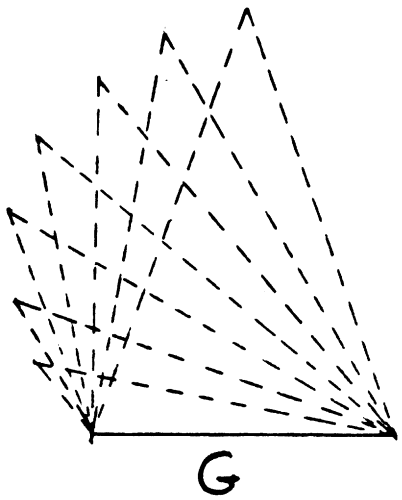


Figure 2.41. Several backprojections of the edge, for different commanded velocities. The velocity error cone has the same error angle for each commanded velocity.

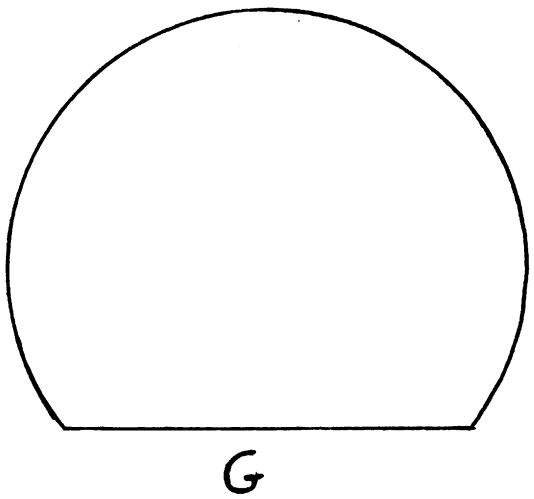


Figure 2.42. The union of all the backprojection regions of the edge is a portion of a circle.

The constraints erected suggest that the peg either be in the hole or be offset to the left of the hole. Regardless of orientation, when the peg makes contact with the horizontal edge at the top left of the hole, the peg is forced into a vertical orientation, then slides towards the hole. The basic motion is similar to the one shown previously in Fig. 1.9.

Notice that the slices of the backprojection region are larger for orientations in which the peg is tilted to the right, than for orientations in which the peg is tilted to the left. An understanding of this shape may be found be examining the corners of the hole at which the peg's motion may be halted. Tilting the peg to the left may cause it to stick on the top or bottom right corners of the hole, as shown in Fig. 2.37. When the peg makes contact with the right wall of the hole, the choice of reference point and commanded velocity cause it to rotate towards the left, away from a vertical orientation. Depending on the initial contact height

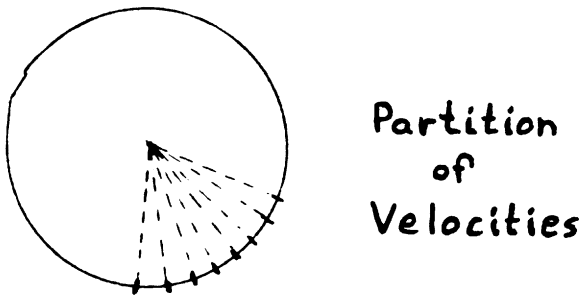


Figure 2.43. Partition of commanded directions.

on the wall, the peg may be able to rotate to an orientation from which it can slide to and stick in the bottom right corner of the hole. For this reason portions of the right wall must be avoided for particular orientations in which the peg is tilted to the left. Backprojecting constraints from these portions of the wall constricts the backprojection region for orientations in which the peg is tilted to the left.

The fairly crude resolution employed in backprojecting constraints from edges is evident in some of the discontinuous jumps between backprojection constraints. These resulted from the decision to identify the behavior of a point on a slice edge by the behavior of a point at the edge's midpoint. If the midpoint was deemed as point to be avoided, either because sticking was directly possible at the midpoint, or because it was possible to move from the midpoint to some other point to be avoided, then the whole edge was labelled as a region to be avoided.

As another example, assume that the commanded velocity and uncertainty cone are as in Fig. 2.38. The nominal commanded velocity is straight down. Then the constraints erected by the backprojection algorithm are those of Fig. 2.39, with resulting backprojection regions as in Fig. 2.40. These backprojection regions take advantage of the hole walls to slide the peg to the goal edge, while orienting it vertically.

2.3.6. Relation of Backprojection to Backchaining

Recall that the planner uses backprojections as building blocks in a backchaining process. The problem is that at each level of the backchaining process all the regions generated in the previous level for all possible motion directions should be used as goal regions. Since the number of motion directions is infinite, there are infinitely many backprojection regions to consider at each level of the backchaining.

Fortunately, in many cases it is possible to replace an infinite collection of sets by a single set. For example, consider the old example of backprojecting from an edge. Fig. 2.41 shows the constraints that are erected for a number of desired velocities. The resulting backprojection regions are always triangles. The union of all the backprojections is a portion of a circle, similar to the region shown in

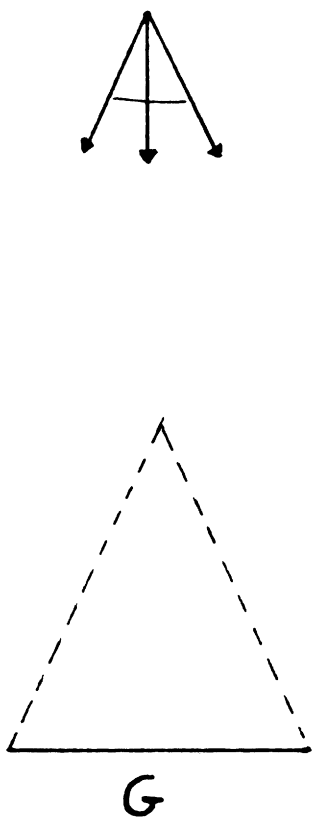


Figure 2.44. Smaller backprojection region resulting from a larger effective velocity cone.

Fig. 2.42. The size of the circle depends on the length of the edge and the angle subtended by the velocity error cone.

In some cases the analytic description of the region formed by the union of component regions may be elusive. Consequently, a planning algorithm might consider discretely and finitely partitioning the space of velocity directions. The planner could then construct a finite number of backprojection regions. This would enable a geometric union algorithm to construct the region needed by the next backchaining phase. In constructing the backprojection regions, the algorithm could use either a single representative velocity error cone for each partition region, or use a larger velocity error cone formed from the union of several individual error cones.

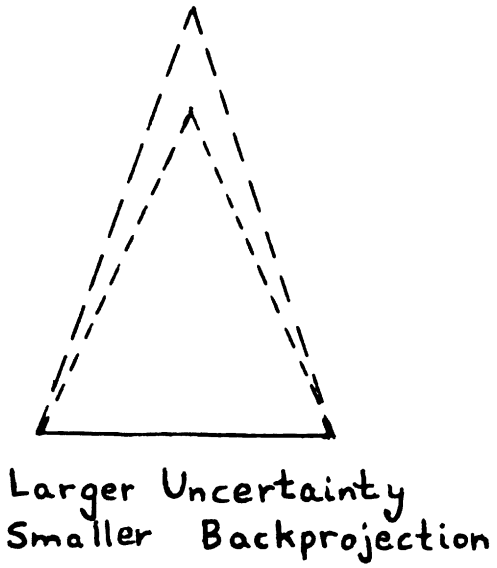


Figure 2.45. Backprojection region of Fig. 2.44 superimposed on the backprojection region of Fig. 2.41, for a commanded velocity pointing straight down. The smaller region is the region from Fig. 2.44.

As an example, for the edge backprojection, the planner could partition the range of velocity directions as in Fig. 2.43. If it chose to use the middle velocity direction as a representative for each partition region, then the resulting backprojection regions would be those of Fig. 2.41. On the other hand, if the planner chose to use all velocities in a partition region, then a typical augmented error cone would be of the form shown in Fig. 2.44. The resulting backprojection region is shown in Fig. 2.44. Fig. 2.45 shows this region superimposed on the backprojection region of Fig. 2.41, for a commanded velocity pointing straight down. Note how the region has shrunk.

Clearly an analytic solution is preferable. The approximate solutions suffer from the usual resolution problems. In the case of single representatives, possibly successful trajectories are ignored because not all motion directions are considered. In the case of augmented velocity cones, all velocity directions are considered. However, the backprojection regions are constructed so as to guarantee success for every motion direction in the augmented cone. This is an excessively harsh requirement. Consequently, the regions constructed are smaller than desired, and again successful trajectories may be ignored.

2.3.7. Relation of Backprojection to Pre-Images

The formal relationship of pre-images and backprojections is explored in Chapter 3. That chapter derives a structure equation which characterizes the use of backprojections as building blocks for pre-images. In Sec. 2.3.1 it was noted that pre-images also capture the ability to recognize that a motion has achieved its goal. In other words, pre-images incorporate the termination conditions that a plan executor must employ to decide that a motion should be terminated.

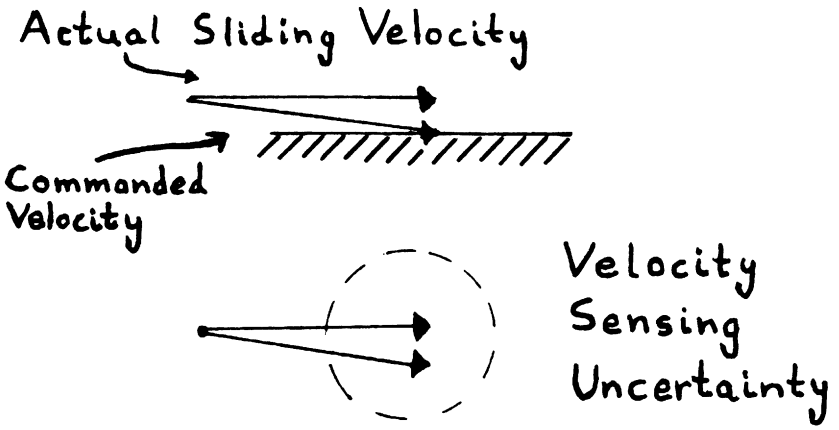


Figure 2.46. Deflection caused by hitting a surface is not discernible due to velocity sensing error.

Backprojections do not consider termination conditions. That is the difference between pre-images and backprojections.

The planner, in constructing pre-image regions with associated motion directions, must decide how a plan executor can decide that a motion has successfully arrived at the goal. It is of no use to plan and execute a motion which is guaranteed to reach the goal, if the plan executor cannot recognize entry into the goal.

The ability to recognize proper termination conditions for a motion depends directly on the sensors available. For example, consider again the example of a goal consisting of a single edge. As noted earlier, there are two basic sensors for detecting entry into the goal, namely position and force sensors. With a perfect position sensor it is an easy matter to detect entry into the goal edge. On the other hand, if there is some uncertainty ball associated with the position sensor's value, then that sensor is useless for terminating the motion. Instead a force sensor must be used.

A force sensor can detect a reaction force from the contact surface, thereby indicating entry into the goal. Effectively, given the generalized damper relationship between forces and velocities, a force sensor may be converted into a velocity sensor.[2] Thus the velocity sensor detects collisions by indicating a deflection in the actual velocity from the commanded velocity. When the motion hits the edge, the vertical component of motion becomes zero. The velocity sensor detects pure horizontal motion (or no motion), indicating success.

Unfortunately, even this method is not failsafe. Suppose that an error ball is associated with the velocity sensor, as with the position sensor. If the commanded

[2]A force sensor may be turned into a velocity sensor, and a velocity sensor may be turned into a force sensor.

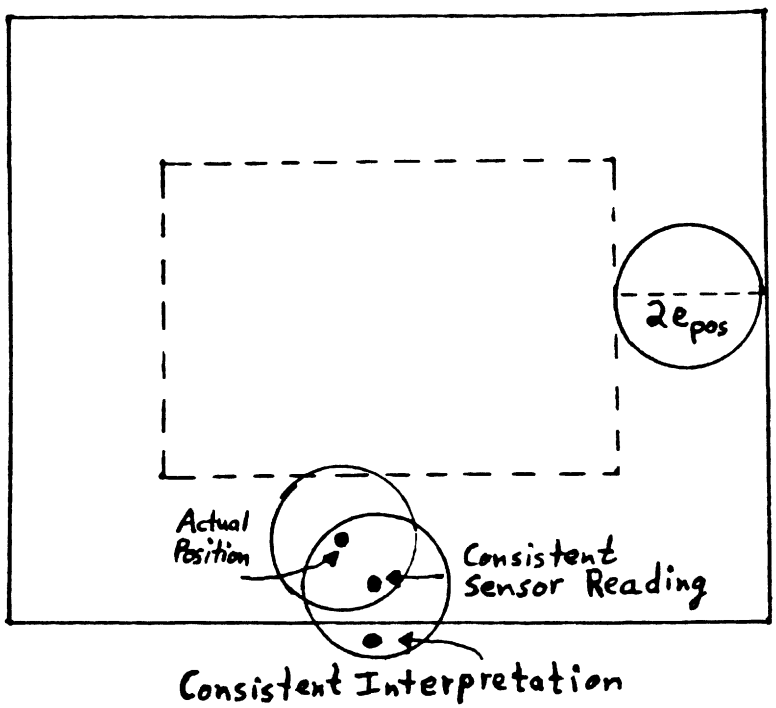


Figure 2.47. In order to decide that the point corresponding to a measured position is inside the larger rectangle, the point must actually be inside the smaller rectangle.

---

velocity is almost horizontal, then, even with perfect control, the deflection caused by hitting the edge may be too small for the velocity sensor to reliably detect. This situation is diagrammed in Fig. 2.46. As a result, only certain motion directions generate backprojection regions from which the motion is guaranteed to both hit the goal as well as hit the goal in a recognizably successful condition.

In deciding whether a planned motion will yield a recognizably successful termination, the planner must consider both the position and velocity at the time of entry into the goal. All interpretations of all possible measured position and velocity values must be clearly distinguishable from the position and velocity values that could hold if the point were not in the goal.

It follows from this, that to decide whether a point is inside a region using position information alone, it must be the case that the actual position lie at least $2\,e_{position}$ within the region, where $e_{position}$ is the radius of the position sensor's error ball (see Fig. 2.47). Otherwise, there is some measured value possible which is consistent with the actual position, such that an interpretation of the measured value lies outside of the region.

A similar argument applies to termination conditions involving velocity sensors only. A slightly more complicated argument along the same lines holds for

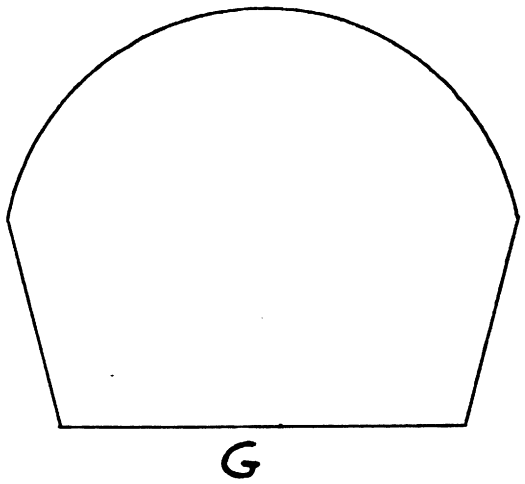


Figure 2.48. Smaller subregion of the circle of Fig. 2.42. Backprojection regions have been eliminated that correspond to velocities for which a collision with the goal edge cannot be reliably detected.

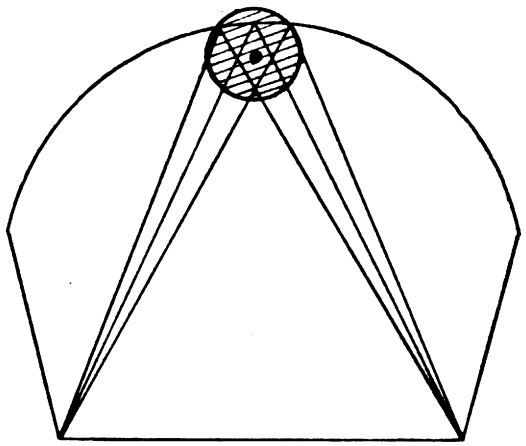


Figure 2.49. The given point is known to be in the circular region. It is impossible to decide in which of the three triangles the point lies, due to position uncertainty.

---

termination conditions involving both position and velocity sensors.

The planner's task in constructing backprojection regions thus involves not only computing the backprojection, but also restricting the set of commanded motions to those that can be terminated. In this fashion, backprojections can usefully approximate pre-images. For further details regarding the dependency of pre-images on termination predicates see Lozano-Pérez, Mason, and Taylor (1983), and Mason (October 1983).

As a final example, it is instructive to re-examine the construction of the single circular region from the infinite collection of backprojections determined in Sec. 2.3.6. This circular region supposedly represents the goal region of the planner

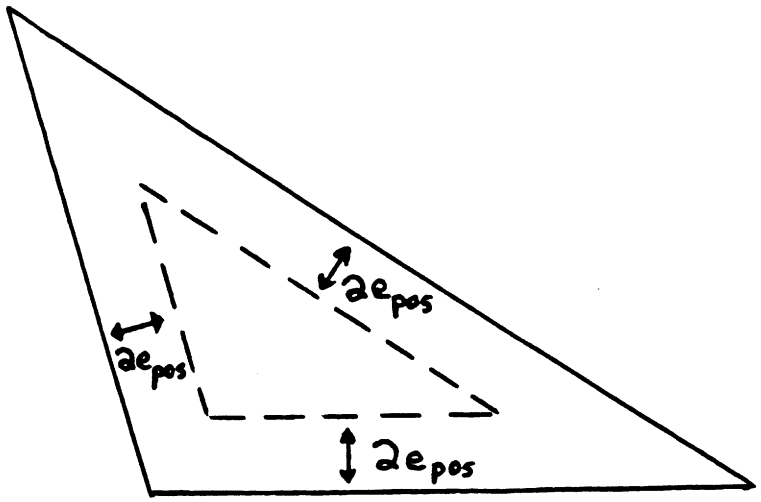


Figure 2.50. Points in the shrunken triangle are easily recognizable as being in the original triangle. The position uncertainty ball is assumed to have a radius $e_{position}$.

---

at the second level of backchaining. Given that a point has attained the circular region, it can then choose a motion to the goal edge, since the circular region is the union of all possible backprojection regions from the edge goal.

Two comments are in order. First, not all of the first level backprojection regions are valid, as not all result in motions that are recognizably successful. This comment was the gist of Fig. 2.46. Thus, the circular region should be trimmed slightly, as indicated by Fig. 2.48.

The second comment is more subtle. It is true that the circular region is the union of all possible backprojections, and it is therefore true that every point in the circular region lies on a trajectory that is guaranteed to hit the goal. Unfortunately, it is not true that the plan executor can necessarily decide which trajectory is the correct trajectory. Suppose the plan executor knows that a point is in the circular region. Given that only position information is of use in free space, it may be impossible for the plan executor to decide in which particular backprojection region the point lies (see Fig. 2.49).

By earlier observations, in order to decide that a point lies in a region, it must be the case that the point is completely distinguishable from points outside of the region. Thus, in order to decide that a point lies in one of the backprojection regions, it must be the case that it cannot possibly be confused with a point outside of the region. This is only possible, on the basis of position information alone, if the point lies within the $2\,e_{position}$ barrier. It follows that the effective goal regions passed to the second level of backchaining are the original triangular backprojection regions shrunk by $2\,e_{position}$, as shown in Fig. 2.50. The union of these regions is a

portion of a limaçon.[3]

The second level of backchaining uses this limaçon rather than the circle as its desired goal region. Given that a point is in the limaçon, not only is the point in a first level backprojection region, but the plan executor can also decide precisely in which backprojection region the point lies. Thus the plan executor can choose a motion direction which successfully reaches the goal edge.

In fact, the shrinking of the backprojection regions is a safe method which is stronger than necessary. The definition of pre-images uses the local history of a motion to disambiguate sensor readings. Doing so permits constructing larger goal and pre-image regions than is possible with position and velocity sensors alone. Backprojections do not generally make full use of this local history. The details of the difference are beyond the scope of this chapter's discussion, but see Lozano-Pérez, Mason, and Taylor (1983), and the next chapter for further information.

[3]A limaçon is given in polar coordinates by equations of the form $r(\theta) = a - b\cos\theta$.

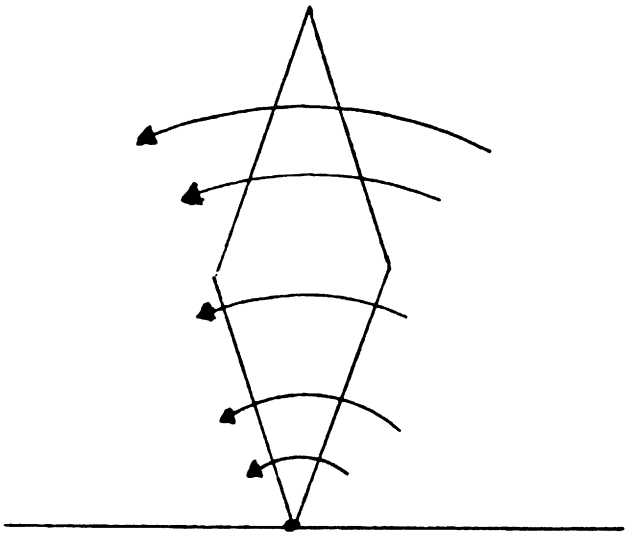


Figure 2.51. Pure rotation about the contact point does not generate frictional reaction forces.

2.4. Friction

The planner must construct motions that avoid surfaces on which sticking can occur. Sticking occurs if an applied force is balanced by an equal and opposite reaction force. In the absence of friction, the reaction forces generated by a surface lie along the normal to the surface. In the presence of friction, reaction forces contain both tangential and normal components. The presence of friction increases the range of reaction forces that a surface can produce, thereby increasing the range of applied forces under which sticking can occur. In order to effectively synthesize strategies that avoid sticking surfaces, the planner must possess an understanding of friction.

Recall from Sec. 2.1.4 that configuration space surfaces possess the same properties as real space surfaces, with respect to the application of and reaction to forces. In particular, a configuration space surface produces a reaction force in the generalized force space[4] of configuration space that lies along the normal to the surface. In this sense, pushing on a configuration space surface is exactly the same as pushing on a real space surface.

The representation of friction in configuration space is not immediate. This is because only some of the degrees of freedom represented by the configuration space may be subject to friction. For example, suppose a moving object that can both translate and rotate is in point contact with another object. The moving object experiences friction only in sliding directions. Pure rotation about the contact point does not create frictional reaction forces (see Fig. 2.51).

[4] Forces and torques.

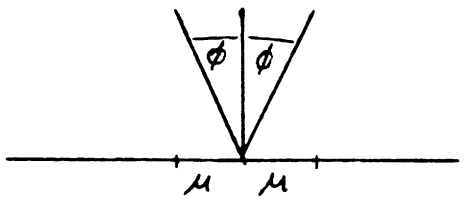


Figure 2.52. Real space friction cone. The coefficient of friction is μ.

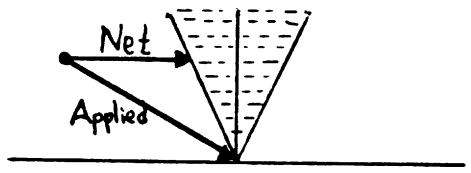


Figure 2.53. Computation of net force in the presence of friction.

2.4.1. A Real Space Friction Cone

A convenient representation of friction in real space is the friction cone. This cone specifies the range of reaction forces that a surface can generate. The cone's axis is parallel to the surface's normal, and its sides make an angle $\phi = \tan^{-1}\mu$ with this axis, where μ is the coefficient of friction (see Fig. 2.52). An applied force that points into the cone is cancelled completely by a reaction force from within the cone, whereas a force that lies outside of the cone, is only partially cancelled by a reaction force from the sides of the cone. In the latter case, the resulting net force is tangent to the surface (see Fig. 2.53).

Given a generalized damper model with identity damping matrix, the planner can easily use the friction cone to geometrically decide whether sticking is possible. By the equivalence of forces and velocities, the planner need merely decide whether a desired velocity points into the friction cone. If so, then sticking is possible. Thus, given a commanded velocity, should the velocity error cone point into the friction cone, then sticking is possible. Equivalently, should the negative velocity cone overlap the friction cone, then sticking is possible. The statics decision regarding sticking has been reduced to a purely geometrical question involving the intersection of two cones.

A slight word about damper dynamics is in order here. Suppose that Newton's

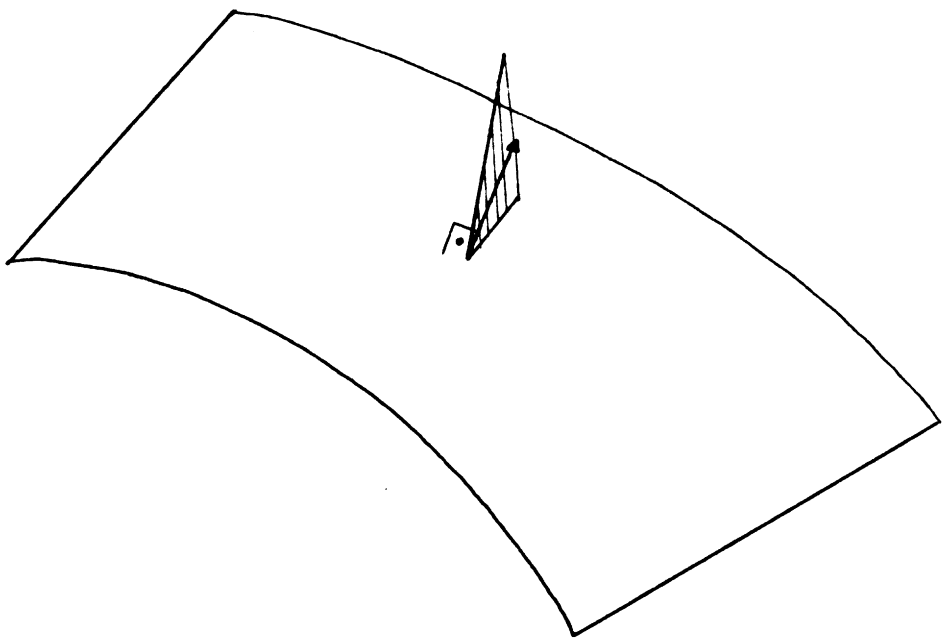


Figure 2.54. The configuration space friction cone is a two dimensional subset of the three dimensional force space.

---

law $\mathbf{F} = m\,\mathbf{a}$ formed the underlying conceptual dynamics. Now suppose that a force is applied to an object in contact with another object, so that the force lies on the edge of the friction cone. The friction cone will generate a reaction force resulting in a zero net force. If the objects are in relative motion, then they remain in relative motion. If they are stationary relative to each other, then they remain stationary.

In contrast, under generalized damper dynamics, the objects must be stationary relative to each other, if the applied force is on the edge of the friction cone. To see this, suppose that the objects are in relative motion. Then, by Coulomb's law, the reaction force is on the edge of the friction cone, and the net force is zero. Using Eq. (2.2), this says that the relative velocity of the two objects must be zero.

The other two cases, the first, in which the applied force lies outside of the friction cone, and the second, in which the applied force lies inside the friction cone, are fairly similar for Newton's world and the damper's world. When the applied force lies outside of the friction cone in Newton's world, the object accelerates with constant acceleration, whereas, in the damper's world, it slides with constant velocity. When the force lies inside the friction cone, the object decelerates and stops in Newton's world, whereas it instantly stops in the damper's world.

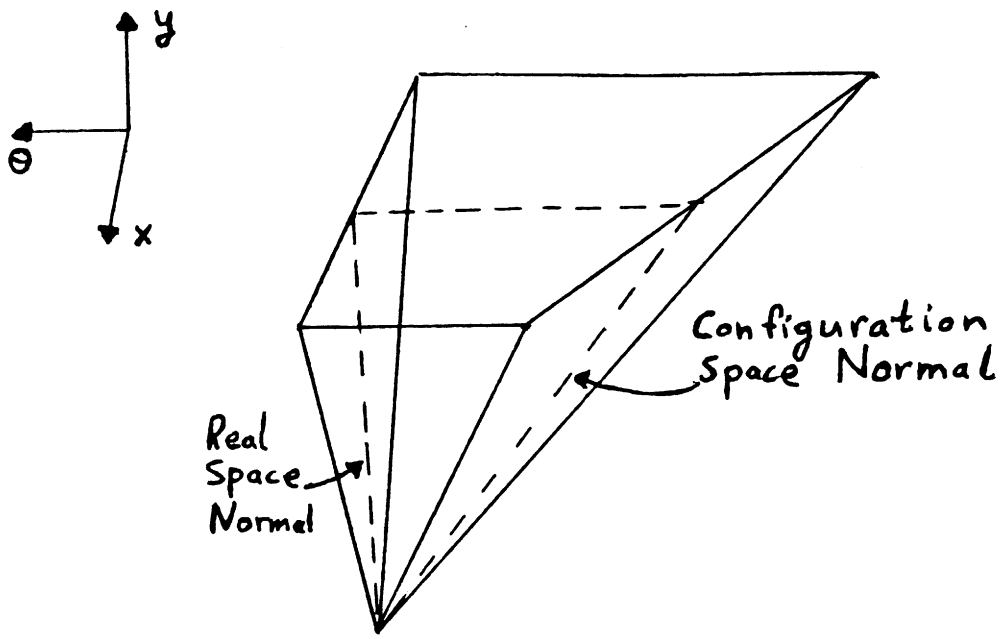


Figure 2.55. Relationship of real space and configuration space friction cones.

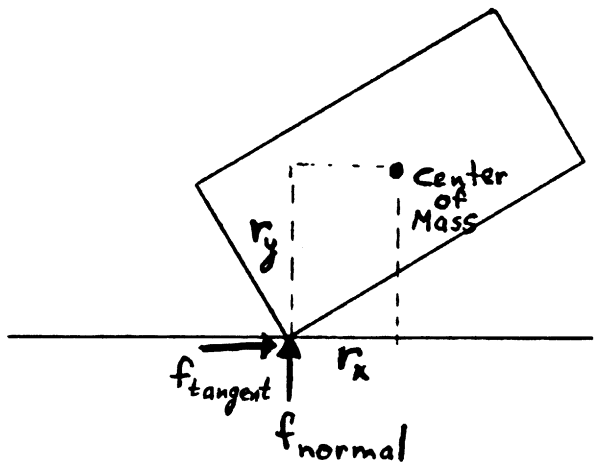


Figure 2.56. An object in contact with an horizontal edge in the presence of friction.

2.4.2. A Configuration Space Friction Cone

The friction cone of real space is a simple representation useful in predicting reaction forces and therefore in deciding whether sticking is possible. Under the damper model, the decision is purely geometric. It is desirable to extend this geometrical representation to configuration space, in the hopes of availing oneself of its simplicity during the backprojection process.

For the two dimensional configuration space of a translating planar object, the friction cone is identical to the one employed in real space. Similarly, the friction cone for the three dimensional configuration space of a translating three dimensional object is identical to the real space friction cone. These claims are

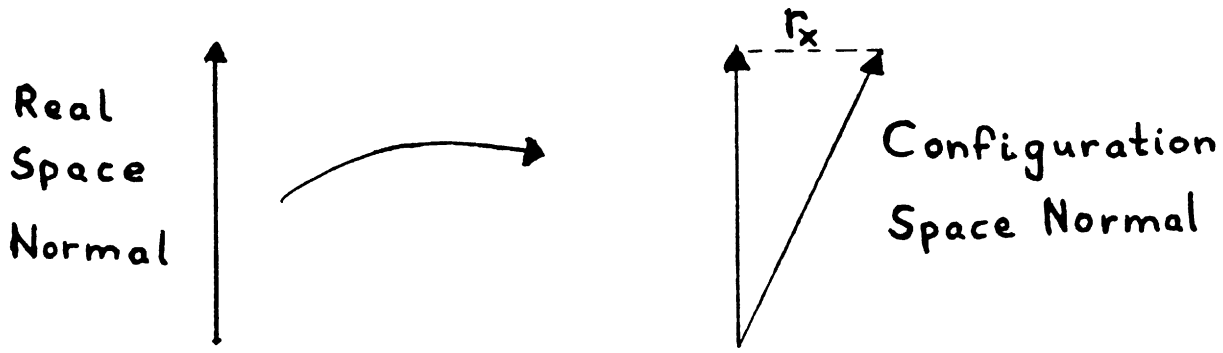


Figure 2.57. The configuration space normal is formed from the real space normal by the addition of a reaction torque induced by the real space normal reaction force. (The configuration space normal has not been scaled to unit length.)

---

clear, since the only object interactions are sliding ones.

Permitting an object to rotate changes the friction cone. As noted previously, pure rotations about point contacts do not generate frictional reaction forces. The friction cone represents the possible range of reaction forces. Thus, for configuration spaces involving rotations, the friction cones cannot possibly be symmetric like their real space counterparts.

Recognizing that the friction cone should represent the possible range of reaction forces, it is clear that the cone must include the surface's normal vector. Furthermore, the dimensionality of the cone must be exactly the same as the number of degrees of freedom subject to friction. Thus, for a real space object with three translational and three rotational degrees of freedom, the friction cone must form a three dimensional region in the six dimensional generalized force space. Similarly, for a planar object with two translational and one rotational degrees of freedom, the friction cone must form a two dimensional region in the three dimensional generalized force space. Fig. 2.54 depicts the latter situation.

The configuration space friction cone should embody the real space frictional constraints. In fact, it is possible to derive the configuration space cone from the real space constraints, once the effects of moments are included. The details of this derivation may be found in Chapter 4. Thus the projection of the configuration space cone into real space should yield the real space cone. Conversely, the configuration space cone may be viewed as a tilting and twisting of the real space cone. The relationship between the two cones is depicted in Fig. 2.55.

Before exploring the relationship between real and configuration space friction cones, it is useful to examine the relationship between the real space normal to a real space object, and the configuration space normal to the corresponding

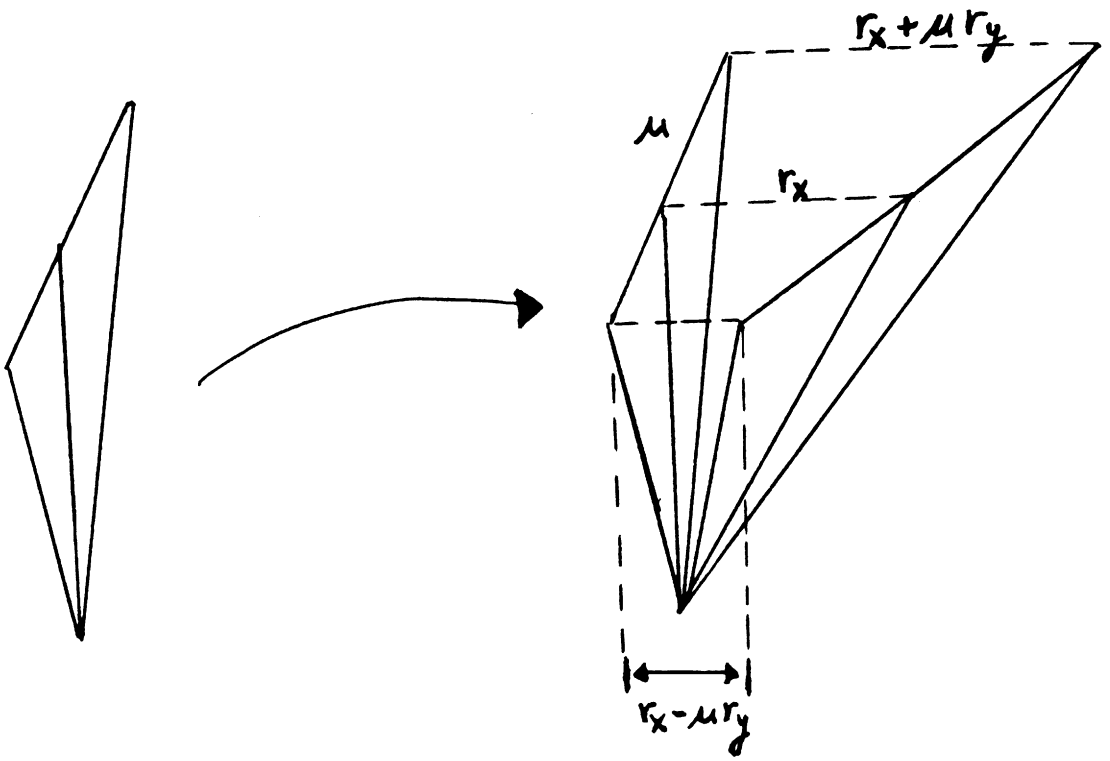


Figure 2.58. The configuration space friction cone is related to the real space friction cone by a twisting of the frictional reaction torque. The twisting occurs because of the sign change present in the range of frictional reaction forces.

---

configuration space surface. Fig. 2.55 also indicates this relationship. Recall that the normals, both to real and configuration space surfaces, may be viewed as the directions of reaction forces generated by these surfaces. The real space normal only embodies translational directions, while the configuration space normal includes both translations and rotations. Consequently, the projection of a configuration space normal into real space yields the real space normal.

The difference between the two normals is due to the moment component of the reaction force. For example, for the contact of Fig. 2.56 the translational normal reaction force also induces a reaction torque about the center of mass. The magnitude of this normal torque is captured by the configuration space normal (see Fig. 2.57).

The derivation of the configuration space friction cone is now a simple matter. One need merely add to the real space friction cone the effect of induced torques. The real space reaction force contains both a normal and tangential component. The normal component is the usual real space normal reaction force. This induces

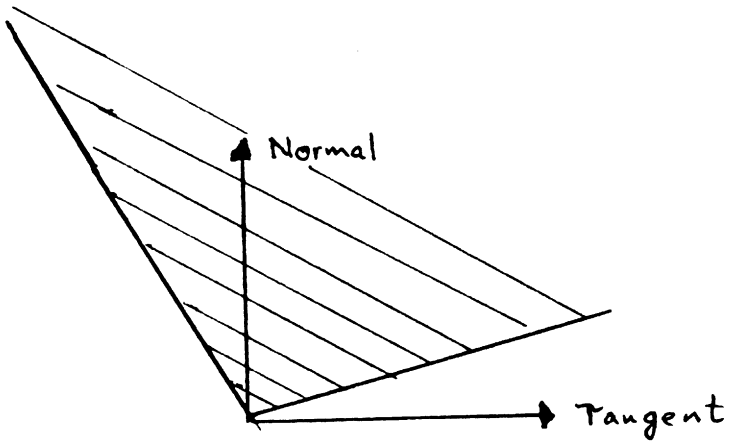


Figure 2.59. A configuration space friction cone depicted in the plane that contains it.

---

the usual reaction torque, as captured by the configuration space normal. Adding this torque to the real space friction cone tilts the cone, as in Fig. 2.57.

Now consider the effect of the tangential component of the real space reaction force due to friction. It too induces a torque, call it a frictional torque. The real frictional reaction force varies in sign and magnitude. The extremes of this variance define the boundary of the real space cone. Commensurately, the frictional torque varies in sign and magnitude. The extremes of this variance define the boundary of the configuration space cone. The resultant effect is a twisting of the real space cone by the frictional torque, as shown in Fig. 2.58.

2.4.3. Properties of the Configuration Space Friction Cone

2.4.3.1. General Comments

The properties of the configuration space friction cone are fairly similar to those of the real space friction cone. In particular, the planner can perform a geometric intersection of the friction cone with the negative velocity error cone, in order to decide whether sticking is possible on a configuration space surface.

In the damper world, velocities and forces are equivalent, since the damper equation is a first order equation. In Newton's world, second order effects arising from the introduction of moments, such as coriolis and centripetal forces, affect the range of reaction forces. This lessens the utility of a geometrical representation, although standard tricks, such as the introduction of fictitious forces, may be used to accurately predict reaction forces using the friction cone. A more complete representation would probably require deriving constraints in phase space. For a first order system, such as the damper's world, this extra work is not needed. The simplicity of the friction representation in the damper's world provides another argument for using a damper as the underlying ideal world dynamics.

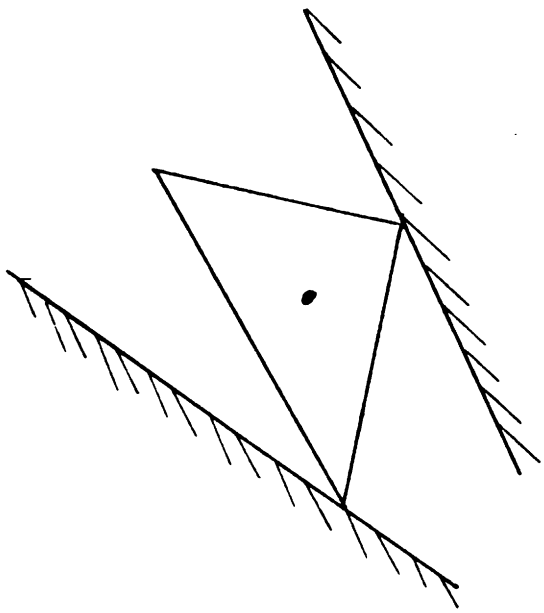


Figure 2.60. An object in two-point contact with other obstacles.

2.4.3.2. Motion Ambiguities

The introduction of friction and moments creates a bizarre effect that leads to motion ambiguities. It is well known that the behavior of a system with friction in classical mechanics is not always deterministic. Generally, this ambiguity arises with multiple contacts, where the distribution of reaction forces among the points is not sufficiently constrained to be uniquely determined. However, the configuration space representation shows that ambiguities can arise even for one-point contact.

Consider a friction cone for the standard three dimensional configuration space of a planar object. Fig. 2.59 depicts such a cone in its own plane. The vertical axis corresponds to the configuration space normal, while the horizontal axis is tangential to the surface. The cone need not be symmetrical about the configuration space normal. This is because in one direction the frictional torque adds to the tilting caused by the normal torque, while in the other direction it subtracts from the tilting. Only if the normal torque is zero, that is, if the real and configuration space normals are identical, is the friction cone symmetric.

Interestingly, it is possible for one side of the friction cone to dip below the tangent plane to the surface. An applied force pointing into this region of the cone is actually pointing away from the configuration space surface. In the absence of friction, the surface could provide no reaction force to balance the applied force, as it points away from the surface. Any resulting motion would move away from the surface. In the presence of friction, however, the surface can provide a reaction

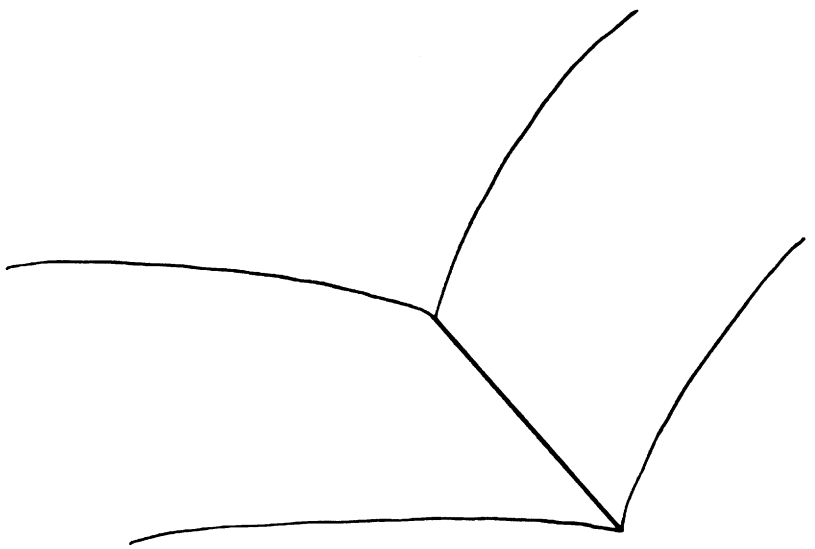


Figure 2.61. Two constraints are satisfied in configuration space by moving along the intersection of two surfaces.

---

force, arising primarily from the frictional torque, that can cancel the applied force.

The projection into real space of the applied force still points into the real space obstacle. In configuration space the applied force points away from the surface, because the applied torque component of the force is large enough to rotate the moving object away from the contact surface. Since the real space portion of the applied force points into the real space surface, it is still possible for that surface to exert a reaction force. In order to so, however, contact must be maintained. This means that the frictional torque induced by the frictional reaction force must be large enough to overcome the effect of applied torque. This is possible since the friction cone dips below the tangent plane. Dipping below the tangent plane implies that the twisting of the frictional torque is greater than the tilting of the normal force and torque, which is precisely the desired condition.

The previous paragraph shows that sticking is possible for an applied force that lies in the friction cone below the tangent plane. However, suppose that the surface offers no reaction force. Then, as in the case of a frictionless surface, the resulting motion will leave the surface. Both sticking and breaking of contact are legitimate solutions. It is impossible to predict which will occur, based solely on the equations of classical mechanics and Coulomb's description of friction.

Sec. 4.4.5 derives the exact conditions under which the previous motion ambiguity can arise. As far as the planner is concerned, however, since sticking is possible under the previous conditions, it should plan motions that avoid a contact which generates those conditions. As before, the recognition of potential sticking

is accomplished by intersecting the friction cone with the negative velocity cone. Thus the potential ambiguity does not affect the structure of the planner.

2.4.3.3. Multiple Points of Contact

Suppose that a moving object is in multiple contact with obstacles in its environment, as in Fig. 2.60. Each point of contact defines a constraint. In configuration space, the point representing the moving object lies on the intersection of multiple surfaces, one surface for each point of contact, as in Fig. 2.61. The object can move tangentially but not normally to these surfaces, unless it breaks contact with one or more of the surfaces. This is equivalent to saying that the range of reaction forces is spanned by the reaction forces possible at each point of contact. In fact, the complete range of reaction forces is just the vector sum of the ranges of reaction forces arising from the various single points of contact.

The previous statement is just an affirmation of the superposition principle. Specifically, the effect of a collection of forces acting in unison is just the sum of the effects of the forces acting individually. This principle applies both to the frictionless normal reaction forces, as well as to the frictional reaction forces. Thus the effective friction cone for multiple contacts is just the vector sum of the individual friction cones. Hence, for multiple contact situations, the planner can again simply intersect the negative velocity cone with the composite friction cone to determine whether sticking is possible.

A word of caution. During multiple contact, the prediction of reaction forces is complicated. For one-point contact, a reaction force may be found merely by projecting the applied force normally onto the friction cone associated with the point of contact. For multiple contact this is no longer possible. In fact, motion ambiguities may arise. This is because the distribution of reaction forces among multiple contacts need not be uniquely determined. In fact, some points of contact may provide no reaction force. In effect, for multiple contacts, it is necessary to consider all subcollections of contacts. For example, a two-point contact should be treated both as a single two-point contact and as two separate one-point contacts. The last two views are equivalent to breaking one of the points of contact, while maintaining the other. For each of the three views, it is possible to determine a resulting motion. Some motions may violate the surface constraints, hence be invalid. It is possible, however, that there exist several valid motions. In this fashion multiple point contact may give rise to ambiguity. See Chapter 4 for details.

2.5. Summary

This chapter has outlined an algorithm for computing backprojections and has developed a representation of friction in configuration space. Both the algorithm and the friction representation are geometric tools that should be used by a motion planner to compute motions in the presence of uncertainty.

The underlying dynamics were assumed to be generalized damper dynamics in configuration space. Thus the specified control commands were velocities. Given a desired commanded velocity, the actual commanded velocity was assumed to lie in some uncertainty cone about the desired velocity. Consequently, the range of trajectories emanating from a given point formed a cone with apex at the point.

The backprojection of a desired goal consists of all those points that are guaranteed to reach the goal. In order to compute the backprojection of a goal, the algorithm considered all points that might not reach the goal. In particular, all points which could cause motions away from the goal, or at which motions could prematurely terminate, were marked as undesirable points. In addition, all points from which a trajectory could reach an undesirable point were also marked as undesirable. Effectively, the backprojection algorithm computed the transitive closure of all undesirable points. This was accomplished by erecting at any undesirable point the negative velocity cone in order to determine all trajectories that might encounter that point.

In deciding whether a motion could stick at a point, it was necessary to develop a representation of friction in configuration space. This representation modelled both the reaction force and induced reaction torque generated by the normal and frictional forces at a point of contact. The range of these forces formed a cone about the normal to the configuration space surface corresponding to the point of contact. For multiple points of contact the composite friction cone was seen to be the vector sum of the individual one-point friction cones.

Within the generalized damper framework, particularly with an identity damping matrix, velocities are equivalent to forces. This equivalence facilitates the determination of points at which sticking might occur. In order to decide whether a nominal commanded velocity might cause sticking at a particular point, it is sufficient to intersect the negative commanded velocity uncertainty cone with the friction cone corresponding to the point. Sticking is possible whenever this intersection is not empty.

This chapter has developed geometrical tools for planning motions in the presence of uncertainty. The friction cone model provided a geometric test for determining points at which sticking could occur. The backprojection algorithm used this test to construct regions from which motions were guaranteed to achieve desired goal states. The algorithm operated by geometrically solving the generalized damper equation.

The next two chapters consider the analytical details and theoretical framework that underlie these tools. In particular, the role and power of backprojections within

the planning process are investigated. Additionally, the physical validity of the friction cone representation is established.

[redacted]

[redacted]

Symbol Table

The following is a list of symbols that appear frequently in the current chapter. Each symbol is followed by a very brief description. The numbers in parentheses refer to the pages on which the symbols are first defined.

p	(110)	Position.
p^*	(110)	Measured position.
v	(110)	Velocity.
v^*	(110)	Measured velocity.
v_θ^*	(110)	Commanded velocity in the direction θ.
$B_r(z)$	(110)	Ball of radius r about z.
$\{ G_\alpha \}$		Collection of position goal sets.
$\mathcal{G}$	(132)	Collection of goals that might be returned by the termination predicate.
$\{ H_\alpha \}$	(121)	Collection of position and velocity goal sets.
$P_R(\{ G_\alpha \})$	(102)	Pre-image of the goals $\{ G_\alpha \}$, with starting position known to be in the set R.
$P_{\theta,R}(\{ G_\alpha \})$	(104)	Directional pre-image of the goals $\{ G_\alpha \}$, given a commanded velocity v_θ^*.
$S(p_0^*, R, \{ G_\alpha \})$	(102)	Set of velocities guaranteed to move a point recognizably into one of the goals $\{ G_\alpha \}$. The point is known to be in the set R, with measured position p_0^*.
$\overline{S}(p, \{ G_\alpha \})$	(122)	Set of velocities guaranteed to move the point p into one of the goals $\{ G_\alpha \}$.
$B_\theta(\{ G_\alpha \})$	(114)	Backprojection of the goals $\{ G_\alpha \}$, given a commanded velocity v_θ^*.
$F_\theta(R)$	(127)	Forward projection of the set R, given a commanded velocity v_θ^*.
$\pi F_\theta(R)$	(129)	The position component of the forward projection of R.
$(T_p(t), T_v(t))$	(116)	Position and velocity of the trajectory T at time t.
$\mathcal{T}(v_0^*)$	(117)	Collection of trajectories with commanded velocity v_0^*.
$\mathcal{T}(p_0^*, R, v_0^*)$	(117)	Trajectories with commanded velocity v_0^* whose initial points lie in R and have measured value p_0^*.
$A(R)$	(143)	The almost simple pre-image containing the pre-image R.
$E(R)$	(141)	The first entry set of the pre-image R.

3. Backprojections

In order to gain a perspective on the issues that confront this thesis, consider the motion planning problem in some generality. The domain of interest is an abstract manifold corresponding to the configuration space of some real (or imaginary) object. Associated with the manifold is a system of dynamics and a system of constraints. Built on top of the dynamics is a control system. The control system permits specification of parameters appearing in the dynamics. Given a specification of these parameters, the evolution of object coordinates occurs subject to the dynamics and the system of constraints. Uncertainty adds spice to this mixture. The control parameters which the control system transmits to the manifold are not necessarily the same as those which are presented to the control system. The relationship between the suggested and actual specifications is governed by some probability distribution. Thus the actual evolution of coordinates is subject to uncertainty, as determined by the probability distribution. Further spice is added by the unreliability of information obtained from the manifold. Sensors which interrogate the state of the system are also subject to error, as determined by some set of probability distributions.

The motion planning problem consists of automatically generating a sequence of control specifications which will cause the system's state to evolve from a given initial state into one of a desired set of recognizable goal states, when such a sequence exists, and to accurately recognize situations in which such a sequence does not exist. This chapter begins with a brief review of a formal planning system suggested by Lozano-Pérez, Mason, and Taylor (1983). Next the chapter addresses the question of effectively computing the pre-image mapping P_R proposed by this formal system. A means of computing approximations to pre-images with backprojections (see also Ch. 2) is presented.

Interleaved with the computation of the mapping P_R is the nature of the termination predicate used by the planner. Thus the form of the termination predicate must be investigated in parallel with the problem of computing P_R. Recall from Chapter 2, that pre-images implicitly take account of the termination predicate, whereas backprojections ignore termination conditions. In other words, motions from pre-images are not only guaranteed to reach the desired goal, but also are guaranteed to recognizably reach the goal. In contrast, motions from backprojections are only guaranteed to reach the goal, without necessarily doing so in a manner that allows the termination predicate to signal success. In order to insure recognizability, the planner must explicitly refine the goals from which it wishes to backproject.

It is desirable to establish precisely the power of backprojections relative to the power of pre-images. Since the difference between the two lies in the use of termination predicates, this chapter considers the restrictions on termination predicates that permit a good approximation of pre-images by backprojections. This investigation leads to a structure equation which defines pre-images that use the restricted form of termination predicate, as backprojections of certain special

sets. The planner should therefore refine its goal sets into these special sets before backprojecting.

3.1. Chapter Goal and Overview

The main goal of this chapter is to characterize the nature of backprojections relative to pre-images. The motivation behind seeking such a characterization lies in the hope that pre-images may be computed by backprojecting from particular types of goal sets.

To this end, the chapter considers a termination predicate that does not possess a clock. Pre-images that are defined using this termination predicate are structurally simple. In particular, the property that a trajectory may be recognized as being in a goal at a given point, is independent of the path by which the trajectory reaches that point. Goal recognizability is a local property independent of time.

The chapter shows that pre-images may be computed by backprojections from distinguished sets. These sets are the first entry sets of trajectories into goals. Every trajectory passes through a point at which it is recognizably in a goal for the first time. The collection of all trajectories' first entry points forms a first entry set. By the time restriction on the termination predicate, goal recognizability at a first entry point is independent of the trajectory that generated that point. Any trajectory beginning at a first entry point is guaranteed to recognizably enter a goal. Thus, any trajectory beginning in the backprojection of a first entry set is also guaranteed to recognizably enter a goal.

Finally, the chapter considers the internal structure that goal sets must possess in order to be used as bases from which to backproject. The issue is whether a trajectory that enters such a goal set is guaranteed to be recognized by the termination predicate as being in the goal. Certainly, if the termination predicate can decide that a trajectory is in a goal at a given point, independent of the trajectory's velocity at that point, then the point may be used as a backprojection base. The difficulty lies with points for which the termination predicate can recognize entry into a goal only for some velocities. Intuitively, one would expect that trajectories, which pass through points with velocities for which the termination predicate cannot recognize entry into a goal, must eventually pass through a point for which the termination predicate can recognize entry into a goal for all velocities. The chapter formally validates this intuition.

These results provide a means for computing pre-images. Given a goal set whose pre-image is to be computed, the planner restructures the goal set so that all trajectories beginning in the goal set are guaranteed eventually to be recognized by the termination predicate as being in the goal set. A pre-image can then be computed by backprojecting from the restructured goal set.

3.2. Pre-Images

Lozano-Pérez, Mason, and Taylor (1983) proposed a formal system for automatically synthesizing fine motion strategies. Given a goal state, the system

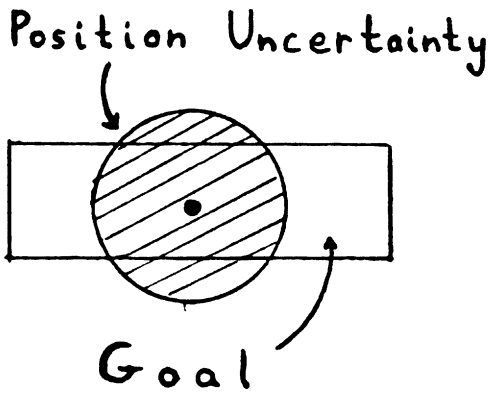


Figure 3.1. The position uncertainty is greater than the goal.

---

determines regions from which certain motions are guaranteed to successfully attain the goal in the presence of sensing and control errors. A planner employs this construction as the basic step in a backchaining process, which yields a sequence of motion commands guaranteed to achieve the goal. The introductory chapters have already presented an overview of this system and its role in planning motions. The current section provides a short review of this work, as it forms the point of departure for this thesis.

The LMT (Lozano-Pérez, Mason, and Taylor) approach is formulated in terms of the configuration space of the moving object. The configuration space of an object is the parameter space representing the degrees of freedom of the object. Obstacles in real space constitute constraints on the object's degrees of freedom. Hence, they may be represented as surfaces in the configuration space. LMT assumes a generalized damper dynamics for the moving object. In other words, in terms of configuration space parameters, the dynamics are described by $F = B(v - v_0)$.[1]

The desired velocity v_0 is a control parameter. In general, there are errors between the effective control parameter v_0 and the commanded control parameter v_0^*, between the actual velocity v and the measured velocity v^*, and between the actual position p and the measured position p^*.

A termination predicate is employed during a motion to a collection of goal sets, which signals success once one of the goal sets is known to have been attained. When signalling success, the termination predicate returns the identity of the goal attained. During the motion, the predicate monitors the available sensors. In particular, the termination predicate is aware of the measured position, velocity

[1]This chapter drops the vector notation employed in Ch. 2. This is done to avoid the impression that the results of this chapter are confined to vector spaces. The results apply to more general manifolds, and to more general motion specifications.

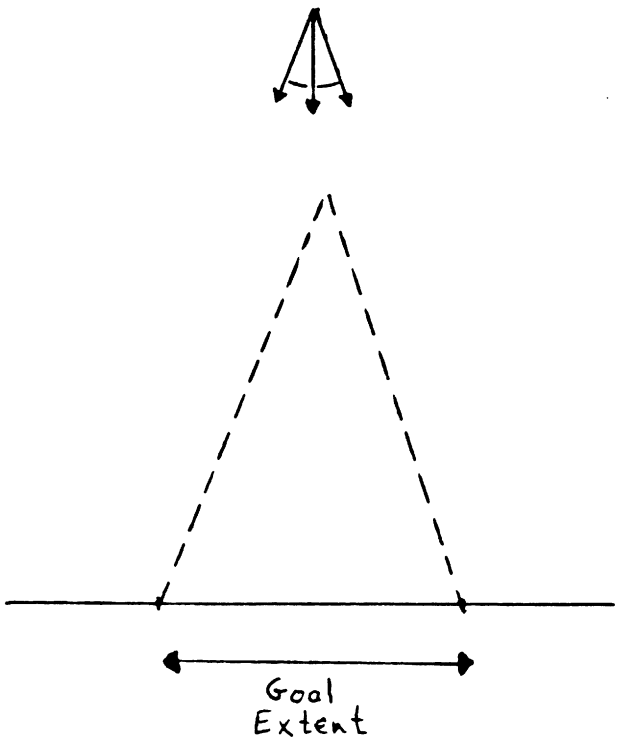


Figure 3.2. The goal is a subportion of the edge. The commanded velocity uncertainty is the cone shown.

---

and time. Additionally, the predicate is given the commanded control velocity v_0^*, and a set R, containing the actual position of the moving object at the beginning of the motion. The predicate may also have state, that is, it may be able to record sensor data for later use. See Mason (October 1983) for details. Depending on the available sensors and the desired goal, a predicate may not be able to signal success even when the actual position is inside a goal. Such a situation occurs, for example, with a predicate whose only sensor is a position sensor with an error ball greater than the extent of the desired goals (see Fig. 3.1).

By including the set R in the list of items available to a termination predicate, the predicate is given some local history. Knowing where a motion began allows the predicate to restrict the possible targets of a motion. The termination predicate can thus rule out entry into certain regions in space that it might otherwise confuse with the goal. Consider the example of Fig. 3.2. The goal is the indicated subregion of the edge. The desired velocity and error cone are as shown. Entry into the goal is detected by observing force and position sensors. The force sensors detect a collision with the edge, while the position sensors decide that the collision has occurred in the goal portion of the edge.

Now suppose that entry into the goal of an actual motion occurs close to the

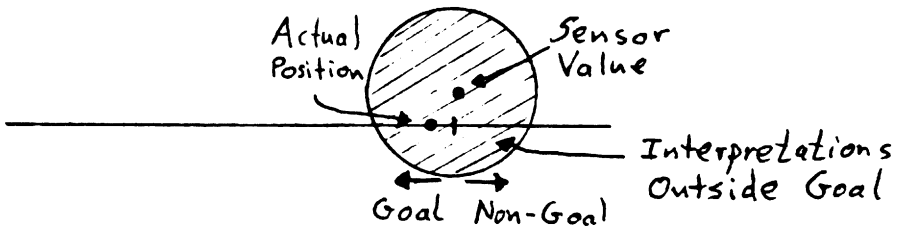


Figure 3.3. An actual position on the edge that is close to the goal boundary, may be confused with points on the edge that are outside of the goal.

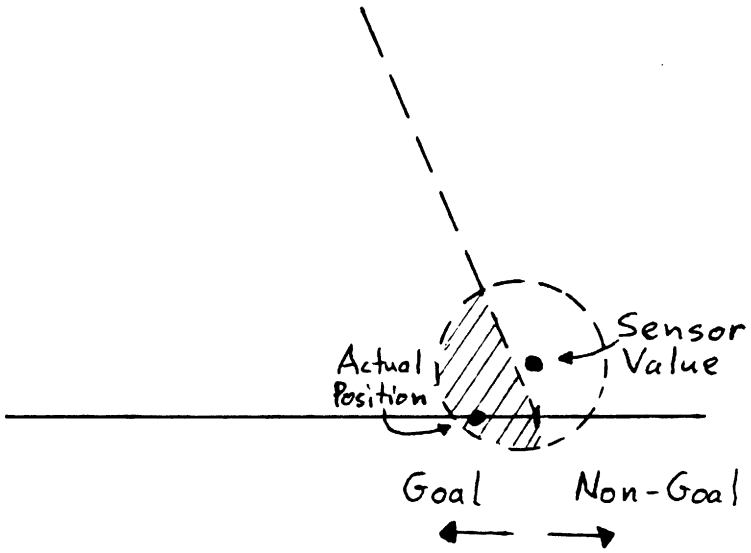


Figure 3.4. Knowing that all trajectories must lie to the left of the dashed line, allows the termination predicate to decide that a point is in the goal. The termination predicate first uses force sensors to recognize that the point is on the edge. Then it uses the trajectory history to decide that the point is on the goal portion of the edge.

---

boundary between the goal and the rest of the edge, so that the position sensors cannot determine that the collision is actually within the goal, due to position uncertainty (see Fig. 3.3). If the termination predicate had no further information, then the planner would have to restrict the starting positions of motions in order to avoid hitting the edge so close to the goal boundary.

However, suppose that the termination predicate knows that the starting point of the motion lies in the triangle formed from the velocity cone, as shown in Fig.

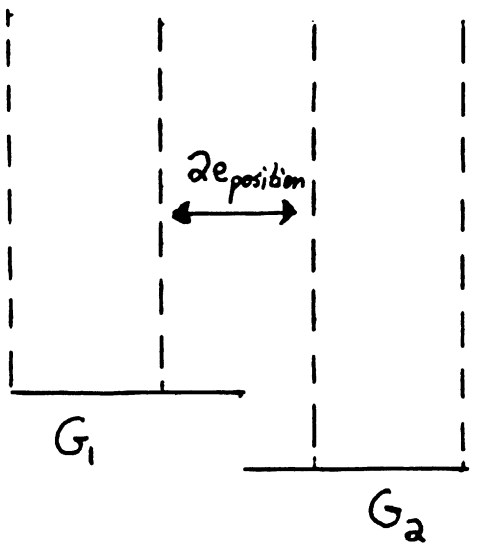


Figure 3.5. $G = G_1 \bigcup G_2$. **Assume perfect velocity control, perfect velocity sensing, and perfect vertical position sensing. Assume that the horizontal position sensor has an error radius** $e_{position}$. **Termination is detected by vertical position sensing. The commanded control velocity is** $v_\theta^* = (0, -1)$. **Then a maximal pre-image of** G **under** v_θ^* **is formed by any pair of vertical cylinders which are separated by distance** $2\,e_{position}$, **such that there is one cylinder over each of the line segments** G_1 **and** G_2.

3.2. (This is just a backprojection of the goal as described in Chapter 2). The motion can never stray out of this triangle, since all motion directions must lie within the bounds of the velocity error cone. Thus the termination predicate can signal success upon detecting a collision with the edge even if that collision is close to the goal's boundary. In effect, the termination predicate intersects the possible interpretations of the force sensors, with the possible interpretations of the position sensors, with the restriction imposed by knowing the starting set R, to determine the effective set of interpretations of the sensors. If this set lies inside a goal, then the termination predicate can signal success. Fig. 3.4 shows this process for a point close to the goal's boundary.

3.2.1. Basic Definitions

Given a measured position p_0^*, a set R, and a collection of goal sets $\{ G_\alpha \}$, LMT defines $S(p_0^*, R, \{ G_\alpha \})$ to be the set of all commanded control velocities v_0^*, such that the termination predicate, knowing that the initial measured position p_0^* corresponds to an actual position p in R, is guaranteed to signal success. A pre-image of a collection of goals $\{ G_\alpha \}$ relative to a set R is the set

$$P_R(\{ G_\alpha \}) = \left\{ p \in R \,\middle|\, \forall p_0^* \in B_{ep}(p),\ S(p_0^*, R, \{ G_\alpha \}) \neq \emptyset \right\}, \qquad (3.1)$$

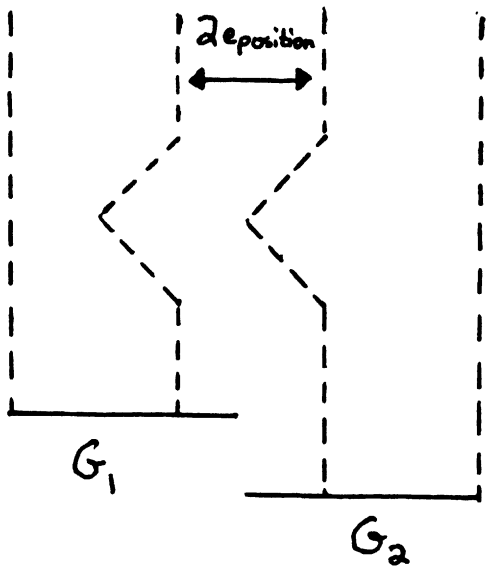


Figure 3.6. Another type of maximal pre-image of G under v_θ^* (see Fig. 3.5).

---

that is, the set of all positions p, such that, for all measured positions p_0^* consistent with p, there is some commanded control velocity v_0^* such that the termination predicate is guaranteed to signal success. The inclusion of the initial measured position p_0^* is another means of retaining local history. The termination predicate uses p_0^* much as it uses the set R. Allowing the predicate to have state, as suggested by Mason (October 1983), further increases the local history available to the termination predicate .

In order that $P_R(\{ G_\alpha \})$ be a suitable subgoal for the next level of backchaining, Lozano-Pérez, Mason and Taylor (1983) showed that the equation

$$R = P_R(\{ G_\alpha \}) \tag{3.2}$$

must hold. The definition of $P_R(\{ G_\alpha \})$ insures that $P_R(\{ G_\alpha \}) \subseteq R$. This inclusion is necessary, as otherwise a point $p \in P_R(\{ G_\alpha \})$ might be a good pre-image point only if the termination predicate thinks that the point is elsewhere, namely in R. Such a definition would be absurd. Lozano-Pérez, Mason and Taylor established the reverse inclusion $R \subseteq P_R(\{ G_\alpha \})$. In order for the termination predicate to know that a motion begins in R, it must be the case that the previous motion terminated in R. The previous motion can be successfully continued only if it terminated in the pre-image of the current goal, by definition of pre-image. In other words, the previous motion must have terminated in $P_R(\{ G_\alpha \})$. The desired inclusion follows.

Thus the collection of suitable pre-images $\{ R_\beta \}$ of a collection of goals $\{ G_\alpha \}$ is the collection of all sets R that satisfy $R = P_R(\{ G_\alpha \})$.

The planner envisioned by **LMT** is given an initial goal set G and an initial position set I. The planner recursively constructs pre-image collections, beginning

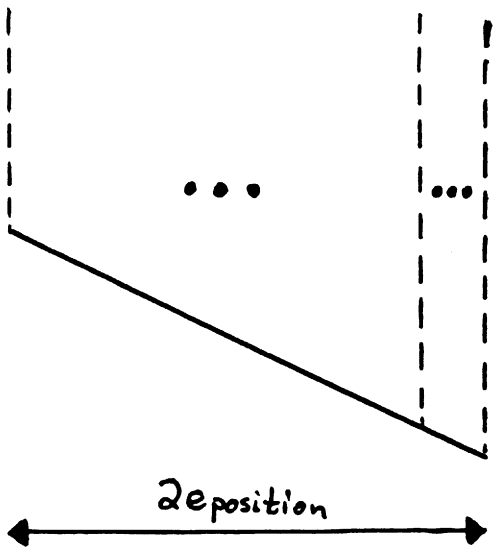


Figure 3.7. This example is a generalization of the examples in Figs. 3.5 and 3.6. The maximal pre-images in this case are vertical lines with lower endpoint in G.

---

with the collection consisting of G and all of its subsets, until at some stage $I \subseteq R_\beta$ for some β. The sets $S(p_0^*, R, \{G_\alpha\})$ can then be used to collapse the recursion and actually move the object. As previously indicated, Mason (October 1983) has shown that the planner, employing a termination predicate with state, is guaranteed to find a solution if a solution using a bounded number of steps exists.

3.2.2. Directional Pre-Images

Consider the set

$$P_{\theta,R}(\{G_\alpha\}) = \left\{p \in R \,\middle|\, \forall p_0^* \in B_{\epsilon p}(p),\ v_\theta^* \in S(p_0^*, R, \{G_\alpha\})\right\}. \tag{3.3}$$

Here v_θ^* is a velocity vector that, for instance, in the planar case, makes angle θ with the x-axis. Mason (October 1983) states that it is sufficient for the planner to consider subgoal sets of the form R, with $R = P_{\theta,R}(\{G_\alpha\})$. In other words, the performance of the planner is not diminished by considering only the mappings $P_{\theta,R}$. This result is fortunate, for it allows one to consider fewer and simpler pre-images. Specifically, it is sufficient to consider pre-images relative to particular commanded velocities. Happily, the result does not depend on whether the termination predicate has state.

This completes the summary of **LMT** .

3.3. Comments on Computability

In the appendix to this chapter (Sec. 3.12) it is shown that pre-images in arbitrary environments are uncomputable. The result applies as well to the computability of backprojections. The proof proceeds by a reduction from the halting problem. Specifically, given an instance of a Turing machine, the reduction transformation creates a maze of constraints and goal edges. The maze corresponds to the Turing machine's transitions, while the goal edges correspond to the Turing machine's halt configurations.

The scenario of that section is somewhat artificial, in that the surfaces and goal edges recursively defined in the reduction do not arise in practice. Nonetheless, the result is suggestive of the difficulty involved in computing the mapping P_R. By restricting oneself to polygonal environments, it is possible to reduce this difficulty. For example, if the goal G_g in the construction of Claim 3.35 is taken to be a single horizontal edge, then the problem is easily solvable. This corresponds, in the language of Turing machines, to the question of deciding whether a machine will halt in a particular configuration after a specified number of steps. That problem is clearly computable.

At present the complexity of computing pre-images and backprojections in the simplified polygonal environment is unknown. A reasonable conjecture is that backprojections are computable in algebraic environments with generalized damper dynamics and algebraic descriptions of friction. Specifically, given a finite number of algebraic constraints, one expects the backprojection problem to be computable. The intuition behind this expectation lies in the finiteness of the constraints. See also Sec. 3.10 and the work by Schwartz and Sharir (1982). By the results of this chapter, which relate backprojections to pre-images, it is therefore also reasonable to expect that the pre-image problem is computable. More theoretical work in the area of pre-image computability is required.

3.4. Solving the Pre-Image Equation

Recall that one of the issues in computing pre-images lies in solving the equation $R = P_R(G)$. This section investigates two properties which could be of use in describing solutions to the pre-image equation. In particular, idempotency and maximality are considered. The pre-image mapping is idempotent. This fact simplifies the construction of pre-images. In search of further simplicity, it is natural to ask whether pre-images may be described by a class of maximal pre-images. The hope is that consideration of a particular class of solutions to the pre-image equation is sufficient to fully describe all pre-images. Maximal pre-images form a natural candidate for this class of sets. Unfortunately, arbitrary pre-images need not be contained in unique maximal pre-images. In fact, maximal pre-images may not even exist. Consequently, the pre-image problem cannot be reduced to solving the pre-image equation for maximal sets. Nonetheless, the notion of maximality provides a guideline to follow in the the later definition of backprojections.

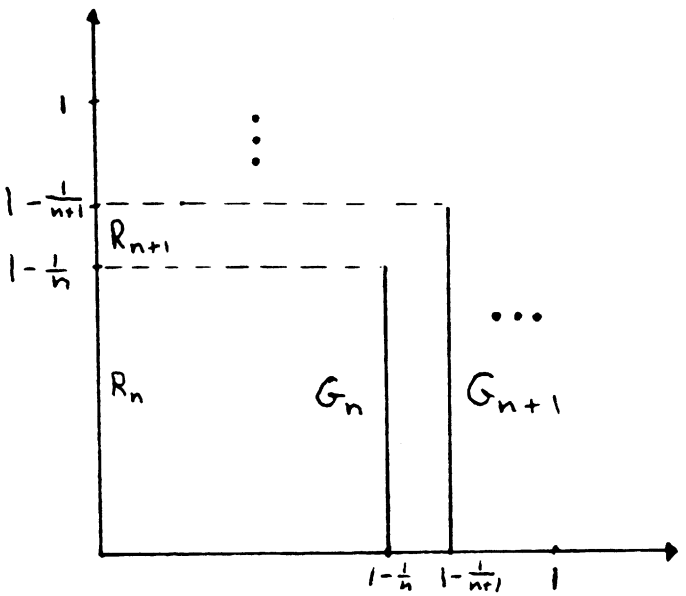


Figure 3.8. A countable collection of goals and pre-image regions. The pre-image regions R_n are all on the y-axis.

3.4.1. Idempotency

Note that the mapping P_R is idempotent, when viewed as a function of R, for a fixed collection of $\{ G_\alpha \}$. In other words,

$$\text{If } R' = P_R(\{ G_\alpha \}), \text{ then } R' = P_{R'}(\{ G_\alpha \}). \tag{3.4}$$

The claim is intuitively clear, and its formal proof is straightforward. This result does not permit solution of the pre-image equation, but it does describe the recursive structure of P_R in a fashion that, in principle, allows one to "list" all suitable pre-images (the list is, however, highly uncountable). Furthermore, this observation permits one to deal with suitable pre-images whenever convenient, as one can always generate a suitable pre-image R' from a set R by a single application of the mapping P_R. The idempotency also holds for the mapping $P_{\theta,R}(\{ G_\alpha \})$ when viewed as a function of R, for fixed θ and $\{ G_\alpha \}$.

3.4.2. Maximal Pre-Images

It is instructive to note that there exist in general an infinite number of sets R satisfying $R = P_R(G)$, for a fixed set G. The truth of the last statement is obvious since any subset of a suitable pre-image is itself a suitable pre-image. More important is the realization that there may exist an infinite number of maximal suitable pre-images for a specified goal G. See Fig. 3.5, Fig. 3.6, and Fig. 3.7, for three examples. These examples do not assume a termination predicate with state, although clearly the values of p_0^* and R must be remembered by the termination predicate in all three cases.

An infinite collection of maximal suitable pre-images does not a priori exclude the possibility of a finite representation of the collection itself. For example, in Fig. 3.6 the collection of maximal pre-images, expressed in English, consists of all pairs of sets S_1 and S_2 separated by a distance $2\,e_{position}$, such that S_i is a subset of the cylinder erected over G_i $(i = 1, 2)$, differing from the cylinder only at the right (for $i = 1$) or left (for $i = 2$) boundary segment. The search for finite representations of infinite collections of pre-images is a motivating factor for this thesis.

As a final comment, maximal pre-images may not even exist. Mason's (October 1983) point on a hill example describes one problem for which maximal pre-images need not exist. The failure of maximality in his example stems from the termination predicate's ability to monitor time. As another example, consider Fig. 3.8. The failure of maximality in this case stems from the non-compactness of the union of goals.

The example of Fig. 3.8 consists of moving a point in the plane under generalized damper dynamics. There are no obstacles in the environment. The velocity sensors have no associated uncertainty, and the velocity control is perfect. The position sensors have perfect horizontal position sensing, but are subject to infinite vertical position uncertainty. In other words, a point can tell where it is horizontally, but not vertically. The example could easily be modified, so that all sensing and control errors are nonzero and finite. For simplicity of presentation, this was not done.

The commanded velocity v_0 is directed along the positive x-axis. The goal sets in the example are the vertical lines

$$G_n = \left\{ (x, y) \;\middle|\; x = 1 - \frac{1}{n}, \text{ and } 0 \leq y \leq 1 - \frac{1}{n} \right\}, \qquad n = 1, 2, \ldots \tag{3.5}$$

The sets R_i which satisfy the pre-image equation are given by

$$R_i = \left\{ (0, y) \;\middle|\; 0 \leq y \leq 1 - \frac{1}{i} \right\}, \qquad i = 1, 2, \ldots \tag{3.6}$$

It is clear that each of the R_i satisfies the pre-image equation relative to the commanded velocity v_0, that is, $R_i = P_{0,R_i}(\{ G_n \})$. Also, since R_i is properly

contained in R_{i+1}, no finite union of the R_i is maximal. So, consider the infinite union $R = \bigcup_{i=1}^{\infty} R_i$. Note that $R = \{ (0, y) \mid 0 \leq y < 1 \}$.

R does not satisfy the pre-image equation, that is, $R \neq P_{0,R}(\{ G_n \})$. To see this, suppose that a motion is known to start in the set R. The termination predicate cannot decide where the motion starts vertically within R. Therefore, the termination predicate cannot predict when the motion will enter one of the goal sets G_n, since no goal set contains all possible trajectories emanating from R.

Furthermore, if R' is any set containing R, then R' cannot satisfy the pre-image equation. This is because any subset of a pre-image is itself a pre-image. Thus there does not exist any maximal pre-image which contains any of the sets R_n.

It is unfortunate that maximal pre-images do not always exist, and it is unfortunate that, even when they do exist, they may exist in abundance. The hope, generally, in seeking out maximality conditions, is to make an otherwise infinite problem tractable. In the current case, maximal pre-images would have served as basic building blocks from which all other pre-images could have been constructed by subsetting. This section shattered that hope. The next section partially rebuilds the hope, by defining an approximation to pre-images that does exhibit maximality.

3.5. Formal Definition of Backprojection

This section formalizes the intuitive description of backprojections presented thus far. The motivation for introducing backprojections lies in their simplicity. The basic difference between pre-images and backprojections is the use of termination predicates. By removing knowledge of termination predicates from backprojections, the construction of backprojections is simplified. Additionally, backprojections possess a maximality property. This property simplifies their effective computation. Of course, by simplifying backprojections relative to pre-images, the power of backprojections has been commensurately diminished. The extent of this power loss is the focus of later sections.

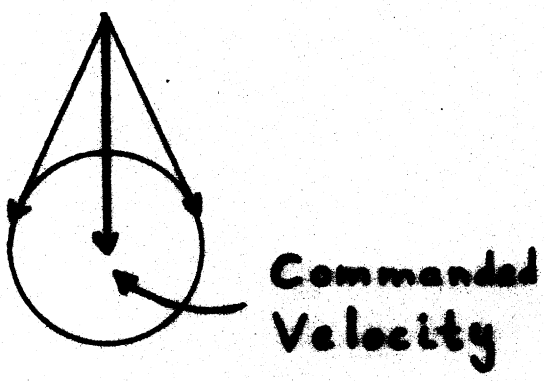


Figure 3.9. The velocity error ball may be described by a cone, if directions are of primary interest.

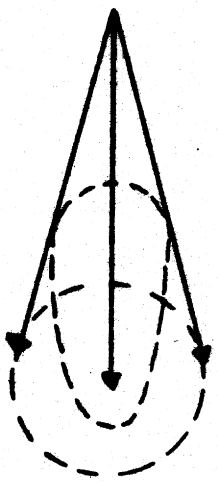


Figure 3.10. Several different error balls can give rise to the same error cone.

3.5.1. Preliminary Definitions

This subsection defines the type of uncertainty assumed throughout the remainder of this chapter.

Definition 3.1

p	Actual position.
p^*	Measured position. The measured position lies in the position error ball about the actual position.
$B_{ep}(p)$	Position error ball about the position p.
p_0^*	Initial measured position.
v	Actual velocity.
v^*	Measured velocity. The measured velocity lies in the velocity error ball about the actual velocity.
$B_{ev}(v)$	Velocity error ball about the velocity v.
v_0^*	Commanded velocity.
v_0	Effective commanded velocity. This term appears in the damper equation. It lies within the error ball about the commanded velocity.
$B_{ec}(v_0^*)$	Error ball about the commanded velocity v_0^*.
v_θ^*	Commanded velocity in a particular direction specified by the parameter θ (this may be multi-dimensional).

The error balls are all assumed to be open balls.

The two measurement balls B_{ep} and B_{ev} are assumed to satisfy the symmetry condition

$$x \in B_{error}(y) \quad \text{iff} \quad y \in B_{error}(x), \tag{3.7}$$

which is motivated by the desire to have

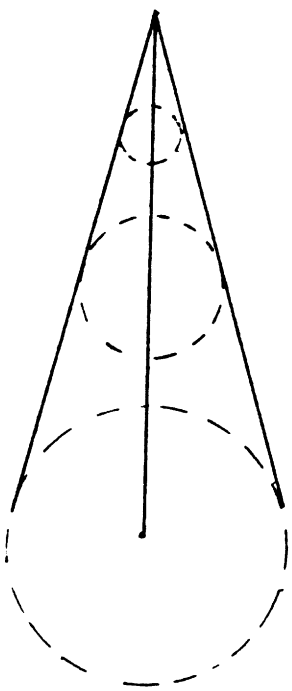


Figure 3.11. The size of the error ball is directly proportional to the magnitude of the commanded velocity. This preserves direction, and the shape of the error cone.

$$\text{For every } x, \quad B_{error}(x) = \{ y \mid x \in B_{error}(y) \}. \tag{3.8}$$

The error ball about the commanded velocity is merely assumed to be open. No particular structure is assumed. It is often convenient to describe the error between the effective commanded velocity and the actual commanded velocity by a cone about the commanded velocity, as in Fig. 3.9. The cone's extreme rays are given by the velocities in the error ball that have the greatest angular misalignment with the actual commanded velocity. Note that there are many error balls that could give rise to the same cone, as indicated by Fig. 3.10.

The error cone view effectively considers only direction, ignoring magnitude. Alternatively, one can imagine an error ball which depends on the magnitude of the actual commanded velocity. Specifically, the diameter of the ball varies directly with the magnitude of its center, so that direction about the actual commanded velocity is always preserved (see Fig. 3.11).

For the most part, the sequel will ignore the precise shape of the error cone or ball. The terms 'error' and 'uncertainty' are used interchangeably. Note that the current definitions ignore any probabilistic distribution of effective values about nominal values.

3.5.2. Ignoring Termination Predicates

In the definition of pre-images, in particular, in the definition of $S(p_0^*, R, \{G_\alpha\})$, consider dropping the requirement that motions recognizably enter goal regions. In other words, ignore the influence of termination predicates. This modification would define $S(p_0^*, R, \{G_\alpha\})$ to be the set of all commanded control velocities v_0^*, such that, given an initial measured position p_0^* corresponding to some unknown actual position p in R, the actual position is guaranteed to enter one of the goal sets G_α. One could then define a new version of pre-images, analogous to Eqs. (3.1) and (3.3), using the new definition of $S(p_0^*, R, \{G_\alpha\})$. For the moment, refer to these sets as *simple backprojections.*

The new definition says essentially that, in order for a point p to be in a simple backprojection, it must be the case that all points, which are consistent interpretations of any initial sensor reading p_0^* corresponding to p, are guaranteed to move to a goal. This definition seems too complicated. Neither the set R, nor the fact that p_0^* is a sensor reading, are needed in deciding whether a motion will eventually enter a goal.

A simpler definition is possible. The class of simple backprojection sets generated by the simpler definition is the same as the class of simple backprojections generated by the previous definition. This follows from a bit of symbol juggling. Thus the simpler definition about to be presented should be preferred.

The simpler definition throws away the initial measured position and the local history. Specifically, define $\overline{S}(p, \{G_\alpha\})$ to be the set of all commanded control velocities v_0^* that are guaranteed to move the point p into one of the goal sets G_α. The difference between the set $\overline{S}$ and the set S defined by LMT lies in the use of predicates. The set $\overline{S}$ consists of all commanded velocities guaranteed to move the point p into one of the goals, although it may be impossible for the termination predicate to recognize entry into the goal. One can then define $\overline{P}_R$ as the set of all points in R which are guaranteed to move into the goal regions under some commanded velocity, and $\overline{P}_{\theta,R}$ as the set of points which are guaranteed to move into the goal regions under a particular commanded velocity. Specifically,

Definition 3.2 The *simple non-directional backprojection* of a collection of goal sets $\{G_\alpha\}$ is given by

$$\overline{P}_R(\{G_\alpha\}) = \left\{ p \in R \,\middle|\, \overline{S}(p, \{G_\alpha\}) \neq \emptyset \right\}. \tag{3.9}$$

The *simple backprojection* of a collection of goal sets $\{G_\alpha\}$ under the commanded velocity v_θ^* is given by

$$\overline{P}_{\theta,R}(\{G_\alpha\}) = \left\{ p \in R \,\middle|\, v_\theta^* \in \overline{S}(p, \{G_\alpha\}) \right\}. \tag{3.10}$$

These definitions capture the notion that a backprojection should consist of those points which are guaranteed to hit a goal region. Sensor values, local history, and termination predicates do not play a role.

It is fairly clear, and will be proved in Sec. 3.5.4, that, with $p \in R$,

$$S(p, R, \{ G_\alpha \}) \subseteq \overline{S}(p, \{ G_\alpha \}) \tag{3.11}$$

$$P_R(\{ G_\alpha \}) \subseteq \overline{P}_R(\{ G_\alpha \}) \tag{3.12}$$

$$P_{\theta,R}(\{ G_\alpha \}) \subseteq \overline{P}_{\theta,R}(\{ G_\alpha \}) \tag{3.13}$$

These relations are interesting, for they state that every pre-image is a backprojection. This makes sense, since a pre-image point must fulfill stronger conditions than a backprojection point. The former must recognizably enter a goal, the latter need only enter, without being necessarily recognizable. While the result is not surprising, it is certainly a necessary condition for investigating backprojections as approximations to pre-images. By considering the class of all backprojections, one can consider the class of all pre-images. Later sections address the question of how to narrow down the class of backprojections that need to be considered.

With these definitions in hand, consider the union

$$\overline{M}_\theta = \bigcup_R \overline{P}_{\theta,R}(\{ G_\alpha \}). \tag{3.14}$$

(by idempotency one need not worry about considering only those R for which $R = \overline{P}_{\theta,R}(\{ G_\alpha \})$.)

Whether a point is in a simple backprojection, depends in some sense solely on properties of that point, not on properties of neighboring points. Clearly therefore, the union of any collection of simple backprojections is itself a simple backprojection. In particular, the union of all simple backprojections is a backprojection.

$$\begin{aligned} \overline{M}_\theta &= \overline{P}_{\theta,\overline{M}_\theta}(\{ G_\alpha \}) \\ &= \left\{ p \,\middle|\, v_\theta^* \in \overline{S}(p, \{ G_\alpha \}) \right\}. \end{aligned} \tag{3.15}$$

In other words, in the absence of predicates that can recognize entry into goals, it is possible to define a unique maximal set $\overline{M}_\theta$, such that commanding the velocity v_θ^* guarantees that every point in $\overline{M}_\theta$ will enter some goal G_α. Unfortunately, it may not always be possible to detect entry into the goal G_α, or even predict which goal will be entered. Nonetheless, by Eqs. (3.13) and (3.14), any suitable pre-image R must be a subset of $\overline{M}_\theta$. Thus the set $\overline{M}_\theta$ provides a constraint on the possible collections of suitable pre-images.

The mapping $\theta \times \{ G_\alpha \} \mapsto \overline{M}_\theta$ will be called a maximal backprojection, denoted by $\overline{M}_\theta = B_\theta(\{ G_\alpha \})$. Formally,

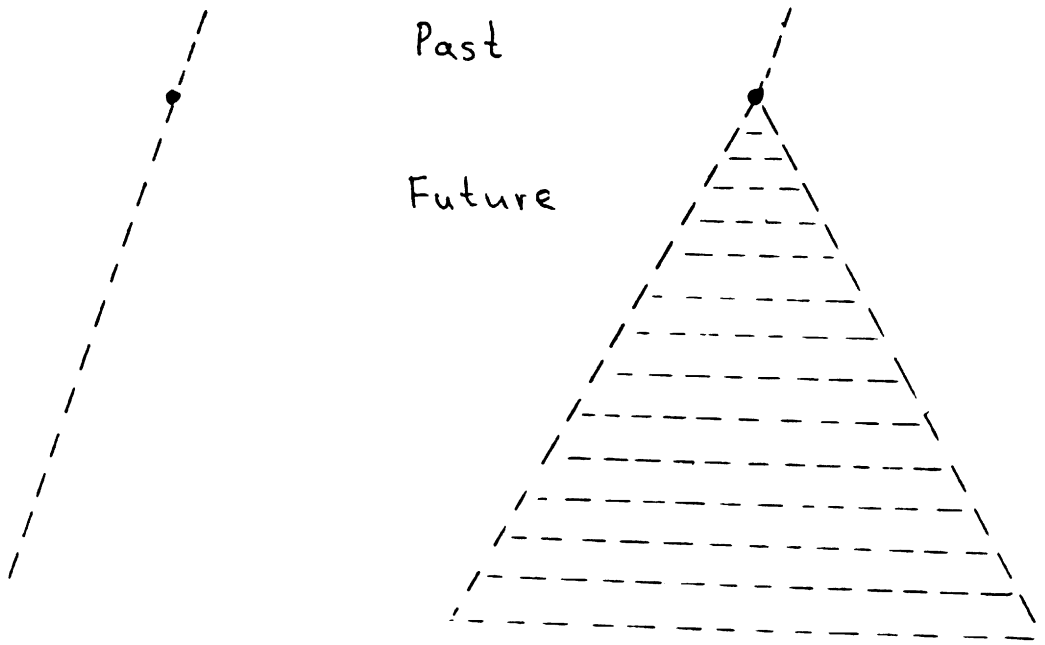


Figure 3.12. The future path of a point is more easily predicted if the commanded velocity error is constant. If the error can vary over time, then the future is determined only up to a cone of uncertainty.

Definition 3.3 The *maximal backprojection* of a collection of goal sets $\{ G_\alpha \}$ under a commanded velocity v_θ^* is just the union of the simple backprojections. It is given by

$$B_\theta(\{ G_\alpha \}) = \bigcup_R \overline{P}_{\theta,R}(\{ G_\alpha \}). \tag{3.16}$$

The constraint defined by backprojections is simply stated as

$$\text{For every } R \text{ and } \theta, \quad P_{\theta,R}(\{ G_\alpha \}) \subseteq \overline{P}_{\theta,R}(\{ G_\alpha \}) \subseteq B_\theta(\{ G_\alpha \}). \tag{3.17}$$

3.5.3. Basic Assumptions and Definitions

In the definitions of pre-images and backprojections above, the successful velocity sets $S(p_0^*, R, \{ G_\alpha \})$ and $\overline{S}(p, \{ G_\alpha \})$ were only intuitively defined. This subsection provides a formal characterization of $S(p_0^*, R, \{ G_\alpha \})$. The characterization may be thought of as an axiom, if so desired. If fact, an analogous axiom is then presented to define $\overline{S}(p, \{ G_\alpha \})$. Most of the subsection consists of building tools needed for the characterization. The tools are used in the sequel.

Additionally, the subsection provides some intuition about the nature of pre-images and termination predicates while constructing the tools.

3.5.3.1. History

Throughout, the termination predicate is assumed to have knowledge only of the current sensors, plus the local history provided by the initial measured position p_0^* and the set R, which contains the actual initial position. In particular, the termination predicate does not possess continuous past state information.

3.5.3.2. Errors

The sensor and control errors are assumed given. They do not explicitly appear as parameters in what follows, although their precise values clearly affect the success of motion strategies.

3.5.3.3. Trajectories

A few notational definitions are in order. Recall that under a damper model, trajectories are continuous, but velocities may vary discontinuously. Define a trajectory, relative to a commanded velocity v_0 to be a continuous mapping from positive time into free space which satisfies the generalized damper equation (Eq. (2.1)) with commanded velocity v_0.

More generally, one can define a collection of trajectories relative to a velocity error ball. The only difficulty with such a definition lies in deciding how the effective commanded velocity v_0 varies over time relative to a constant commanded velocity v_0^*. There are several possibilities. The simplest is to assume that the effective commanded velocity v_0 remains constant. At the other extreme, one can assume that v_0 can vary arbitrarily within the error ball about the actual commanded velocity v_0^*.

The difference between these two world views manifests itself in the regions reachable from a given trajectory point. In the first case, once motion is in progress, the effective commanded velocity remains constant, so the reachable region lies along a trajectory. In the second case, the direction of motion can vary, so the reachable region is a cone. The difference is shown in Fig. 3.12. Said in another way, for constant error, the set of possible places at which a moving point may find itself is more constrained, hence more predictable, than for varying error.

Constant error is too simple a model. Arbitrarily varying error is probably physically unrealistic. Nonetheless, it provides a model that more closely approximates uncertainties arising from uncontrolled and unpredictable disturbances in a first order system.

The rest of the chapter assumes that the effective commanded velocity can vary discontinuously and fairly arbitrarily. The only restriction imposed on the effective commanded velocity is that the damper equation be integrable. In other words, it should be that case that

$$p(t_0) = p(0) + \int_0^{t_0} v(t)\,dt, \tag{3.18}$$

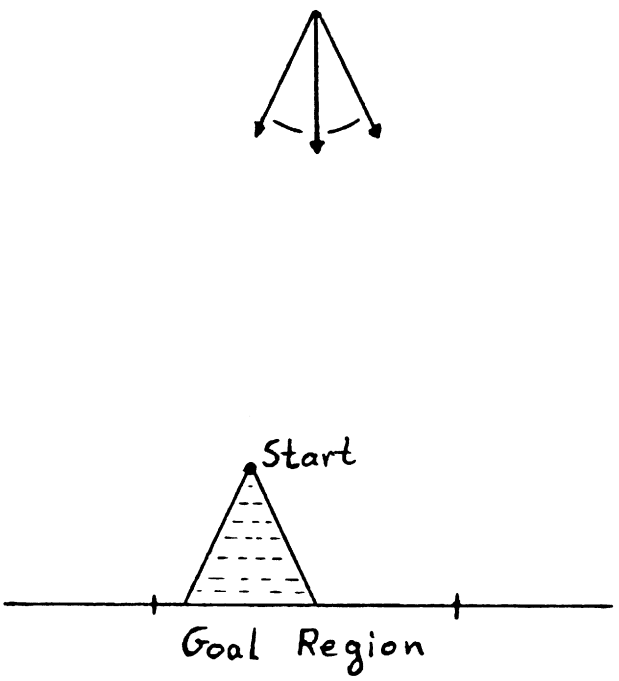


Figure 3.13. The goal region is a subportion of the edge. A point starting at "Start" is guaranteed to hit the edge.

---

where $v(t)$ is obtained from the damper equation.

It is probably more realistic to assume that changes in the effective commanded velocity be at least piecewise continuous. For simple environments, such as those containing a finite number of algebraic constraints, such a continuity restriction does not change the results obtained below, although it changes some of the proofs. In particular, all proofs that assume a discontinuous change of commanded velocity should be modified so that the commanded velocity change occurs continuously over an arbitrarily small region about the trajectory.

Given the integrability condition (3.18), it is more appropriate to view a trajectory as a mapping from time into velocity space. Position can then be determined by integrating velocity. For simplicity, the following definition combines both position and velocity.

Definition 3.4 A *trajectory that satisfies the damper equation with uncertainty relative to a commanded velocity* v_0^* is a mapping

$$\begin{aligned} T: \quad [\,0,\, \infty\,[\quad &\rightarrow \quad \text{Free-space} \times \text{Velocity-space} \\ T(t) \quad &\mapsto \quad (T_p(t),\, T_v(t)), \end{aligned} \tag{3.19}$$

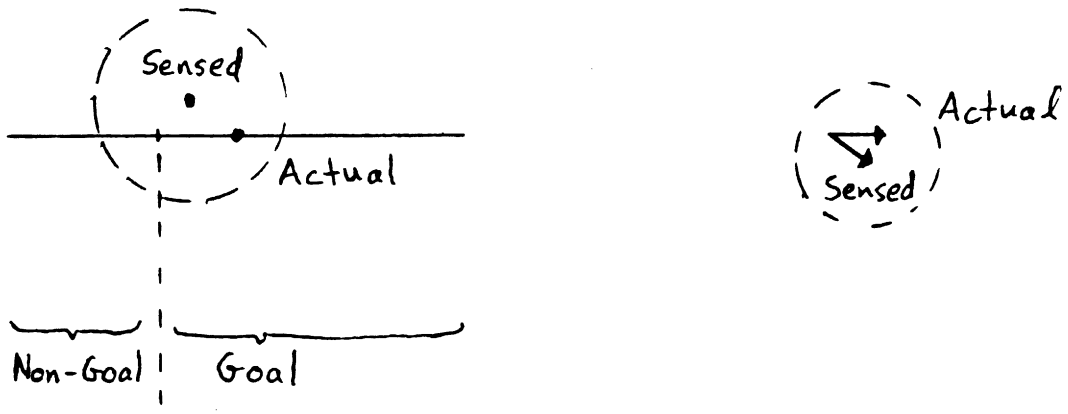


Figure 3.14. Sensed position and velocity values corresponding to actual position and velocity values. Also shown are the uncertainty balls about each of the sensed values.

---

such that at all times t, $T_v(t)$ satisfies the damper equation (Eq. (2.1)) with effective commanded velocity $v_0(t) \in B_{ec}(v_0^*)$. Furthermore, T_p is related to T_v by the integral equation (3.18).

Definition 3.5 Let $\mathcal{T}(v_0^*)$ be the set of all trajectories that satisfy the damper equation with uncertainty relative to v_0^*.

It is often useful to speak of a trajectory as being consistent with a sensor value, or of a sensor value as being consistent with a trajectory. Formally, a trajectory T is *sensor consistent at time* t with a sensor reading (p^*, v^*) if and only if $(T_p(t), T_v(t)) \in B_{ep}(p^*) \times B_{ev}(v^*)$. Similarly, a sensor reading (p^*, v^*) is *consistent with a trajectory* T *at time* t if and only if $(p^*, v^*) \in B_{ep}(T_p(t)) \times B_{ev}(T_v(t))$. The terminology makes sense because of the symmetry conditions on error balls. Effectively, the definition states the intuitive belief that two things are consistent if they lie within the uncertainty range of each other.

Definition 3.6 The set of trajectories consistent with an initial measured position p_0^* which is known to correspond to an actual position $p \in R$ is

$$\mathcal{T}(p_0^*, R, v_0^*) = \left\{ T \in \mathcal{T}(v_0^*) \,\middle|\, T_p(0) \in B_{ep}(p_0^*) \bigcap R \right\}. \tag{3.20}$$

The set $\mathcal{T}(p_0^*, R, v_0^*)$ represents all trajectories whose starting points are consistent with a given initial measured position. This set is useful for it allows the

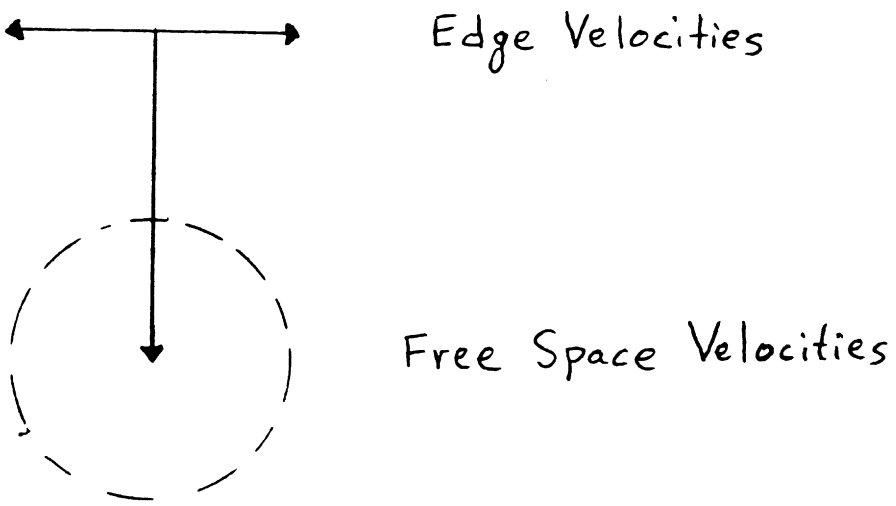


Figure 3.15. The circular region comprises the range of velocities possible in free space. The horizontal region comprises the velocities possible while in contact with the edge of Fig. 3.13.

---

planner and therefore the termination predicate to predict the possible locations that a point could attain. The information is used below to restrict the possible interpretations of sensor values.

3.5.3.4. Termination Predicates and Forward Projections

This subsection describes the process by which a termination predicate decides that it has attained a goal. The basic process was described by Lozano-Pérez, Mason, and Taylor (1983), and has been intimated at numerous times.

In essence, the termination predicate considers the current sensor values, determining all position and velocity interpretations consistent with the sensor values. The termination predicate then uses any local history or state which it possesses, and knowledge of its environment, to further restrict the set of position and velocity interpretations. If the restricted set is wholly included in a goal set, then the termination predicate signals success.

Consider the example of Fig. 3.13. The commanded velocity is straight down, the velocity error cone is as shown, and the goal is the subportion of the edge indicated. Suppose that the actual motion started at the point indicated. Thus, the termination predicate knows that the possible positions of the motion must lie within the cone of Fig. 3.13. Suppose that the actual motion has collided with the

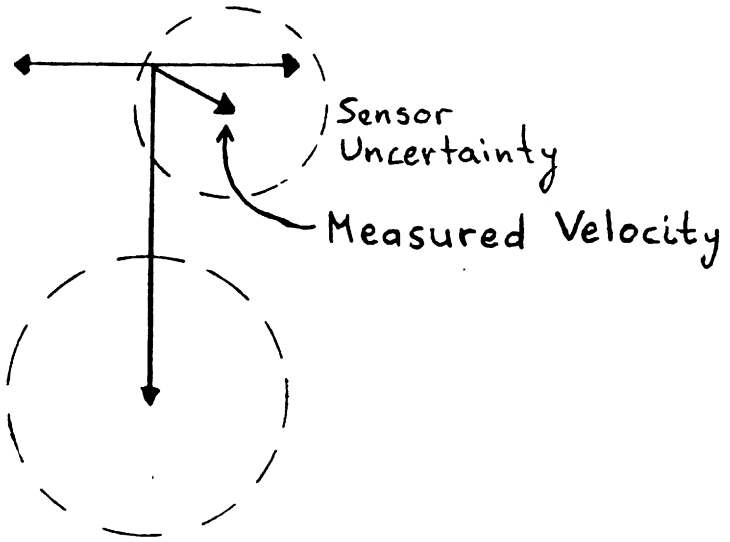


Figure 3.16. The measured velocity and its uncertainty ball. Note that the uncertainty ball overlaps the edge velocities, but not the free space velocities.

---

edge inside the goal. The position and velocity sensor values at the time of collision are shown in Fig. 3.14, along with their uncertainty balls.

Also shown in Fig. 3.15 is the commanded velocity and its uncertainty ball. This ball represents the range of velocities that are possible in free space. The second "ball" in Fig. 3.15 corresponds to the velocities that are possible when in contact with the edge. This ball is one dimensional, as it represents the range of velocities found by solving the damper equation for all possible commanded velocities in the commanded velocity uncertainty ball.

Assume that the measured velocity ball does not intersect the free space velocity ball, but that it does intersect the contact velocity ball. This is shown in Fig. 3.16.

Using the position and velocity sensors alone, the termination predicate cannot decide that the actual position lies inside the goal. Taking the commanded velocity and its uncertainty ball into account, the termination predicate can decide that the current motion cannot be in free space. Instead, a collision must have occurred. This allows the termination predicate to restrict the position interpretations to the edge. Finally, using the cone of possible positions defined by the starting position, the termination predicate can actually recognize entry into the goal region (see also Fig. 3.4).

The process just outlined is easily described by an intersection of sets in the tangent bundle. This intersection is the essence of the characterization of the set S.

The rest of this subsection presents a formal description of the intuition presented above.

The following definition involving positions and velocities parallels that given for trajectories at the end Sec. 3.5.3.3. The goal collection $\{ G_\alpha \}$ is assumed given.

Definition 3.7 The *forward projection at time t* of a measured position p_0^* known to correspond to an actual position in the set R, under the commanded velocity v_θ^*, is given by

$$F_{p_0^*,\theta,R}(t) = \left\{ (T_p(t),\, T_v(t)) \;\middle|\; T \in \mathcal{T}(p_0^*,\, R,\, v_\theta^*) \right\}. \tag{3.21}$$

Definition 3.8 The *timeless forward projection* is just the union of the forward projections with time, over all values of time,

$$F_{p_0^*,\theta,R} = \bigcup_{t \geq 0} F_{p_0^*,\theta,R}(t). \tag{3.22}$$

The forward projection at time t lists the positions and velocities that are possibly reachable at time t from the initial measured position p_0^*, while the timeless forward projection lists the positions and velocities that are possibly reachable at some time from p_0^*. Both types of forward projection assume that p_0^* corresponds to an actual position in the set R. Thus the forward projections capture the local history available to the termination predicate.

Referring to the earlier example, note that the position component of the timeless forward projection spans the cone of possibly reachable positions. Additionally, the velocity component of the forward projection expresses the difference between the uncertainty ball of velocities in free space, and the uncertainty ball of velocities when in contact with the edge. For each position p that is in free space, the forward projection includes points of the form (p, v), where v spans all of the free space velocity uncertainty ball. Similarly, for each position p in contact with the edge, the forward projection includes points (p, v), with v spanning the contact velocity ball.

In short, the forward projection at time t captures the information available to the termination predicate from its local history, and from the environment. The timeless forward projection captures less of the local history, as it ignores the relationship between time and the positions of a moving point. This difference will be of interest later.

Recall, that in the earlier example, the termination predicate intersected its local history with its interpretation of the sensor values, in order to obtain an effective interpretation of the possible states of the motion. In terms of the tools just developed, this intersection is given by

$$F_{p_0^*,\theta,It}(t) \bigcap B_{ep}(p^*) \times B_{ev}(v^*) \tag{3.23}$$

3.5.3.5. Goals in Velocity Space

The following definition is made solely for notational convenience.

Definition 3.9 Given a subset G of position space, define its *cylinder* to be the set $G \times$ Velocity-space.

More generally, given a collection of goal sets $\{ G_\alpha \}$ define the collection $\{ H_\alpha \}$ to be the set of cylinders constructed from $\{ G_\alpha \}$.

Thus the collection $\{ H_\alpha \}$ is just a way of representing the goals $\{ G_\alpha \}$ in phase space. In general, instead of constructing cylinders, one can imagine constructing complex goal sets in phase space. Such goals would not only specify desired position, but also desired velocity. Such a scheme would require a second order system as the underlying motion description. It would be interesting to investigate this problem, but it is not of direct concern here.

3.5.3.6. Characterization of $S(p_0^*, R, \{ G_\alpha \})$

The discussion in the previous pages has provided a framework for the formal characterization of the set $S(p_0^*, R, \{ G_\alpha \})$. Intuitively, a commanded velocity v_θ^* is in the set $S(p_0^*, R, \{ G_\alpha \})$ exactly when the termination predicate can eventually signal success of any motion whose initial position is measured to be p_0^*. The termination predicate knows that the actual position corresponding to p_0^* must be in the set R. In deciding whether the motion has attained a goal, the termination predicate considers this local history, plus the position and velocity sensor readings (p^*, v^*). It does not remember the sensor readings.

The key to the characterization lies in realizing that the termination predicate must be able to signal success for every trajectory that is initially consistent with p_0^*. Furthermore, since the termination predicate can't predict the sensor readings at the time of success, it must be the case that every trajectory eventually enters a goal region in a fashion that allows the termination predicate to signal success for every possible sensor reading that is consistent with the trajectory. Conversely, if for every trajectory initially consistent with p_0^*, there is some time at which all sensor readings consistent with the trajectory only have interpretations inside of goal regions, then the termination predicate can signal success for every trajectory.

These observations prove the following

Claim 3.10 $v_\theta^* \in S(p_0^*, R, \{ G_\alpha \})$ if and only if for every trajectory $T \in \mathcal{T}(p_0^*, R, v_\theta^*)$, there is some time $t \geq 0$, such that, for any sensor value (p^*, v^*) consistent with T at time t, there is some goal set $H \in \{ H_\alpha \}$ which contains the set of sensor interpretations, that is, $F_{p_0^*,\theta,R}(t) \cap B_{ep}(p^*) \times B_{ev}(v^*) \subseteq H$.

In words, the commanded velocity v_θ^* is a good velocity if, and only if, for every trajectory consistent with the initial information, there is some time at which,

given any sensor readings, there is some goal which contains all interpretations of the given sensor readings. The sensor readings must, of course, be consistent with the trajectory.

3.5.3.7. Definition of $\overline{S}(p, \{ G_\alpha \})$

The claim of the previous section provides a guideline for defining the successful velocity set $\overline{S}(p, \{ G_\alpha \})$. Recall that the difference between pre-images and backprojections is that the latter do not use termination predicates. The only issue is whether a motion is guaranteed to reach a goal, not whether a termination predicate can recognize that it has reached the goal. Sensor values are not relevant. This leads to

Definition 3.11 $v_\theta^* \in \overline{S}(p, \{ G_\alpha \})$ if and only if for every trajectory $T \in \mathcal{T}(v_\theta^*)$, with $T_p(0) = p$, there is some time $t \geq 0$, and there is some goal set $H \in \{ H_\alpha \}$, such that the trajectory is in the goal, that is, $T(t) \in H$.

3.5.4. Simple Comparison of Pre-Images and Backprojections

This subsection establishes Eqs. (3.11), (3.12), and (3.13). Recall, that these equations essentially state that any pre-image is also a simple backprojection. The first result establishes the relationship of the successful velocity sets S and $\overline{S}$. The second two results establish the inclusion of pre-images in backprojections. Since any subset of a simple backprojection is itself a simple backprojection, this says that every pre-image is a simple backprojection.

Claim 3.12 Assume $p \in R$. Then $S(p, R, \{ G_\alpha \}) \subseteq \overline{S}(p, \{ G_\alpha \})$.

Proof: Let $v \in S(p, R, \{ G_\alpha \})$. Then for every trajectory $T \in \mathcal{T}(p, R, v)$, there is some time $t \geq 0$, such that, for every sensor value (p^*, v^*) consistent with T at time t, there is some goal set $H \in \{ H_\alpha \}$, such that the set of interpretations $F_{p,\theta,R}(t) \cap B_{ep}(p^*) \times B_{ev}(v^*) \subseteq H$.

In particular, for every trajectory $T \in \mathcal{T}(v)$ with $T_p(0) = p$, there is some time $t \geq 0$ and some goal set $H \in \{ H_\alpha \}$, such that $T(t) \in H$. So $v \in \overline{S}(p, \{ G_\alpha \})$. ∎

Claim 3.13 $P_R(\{ G_\alpha \}) \subseteq \overline{P}_R(\{ G_\alpha \})$.

Proof: Let $p \in P_R(\{ G_\alpha \})$. Then $p \in R$, and for every initial measured position $p_0^* \in B_{ep}(p)$, $S(p_0^*, R, \{ G_\alpha \}) \neq \emptyset$.

In particular, $S(p, R, \{ G_\alpha \}) \neq \emptyset$. So, by Claim 3.12, $\overline{S}(p, \{ G_\alpha \}) \neq \emptyset$. This says that $p \in \overline{P}_R(\{ G_\alpha \})$. ∎

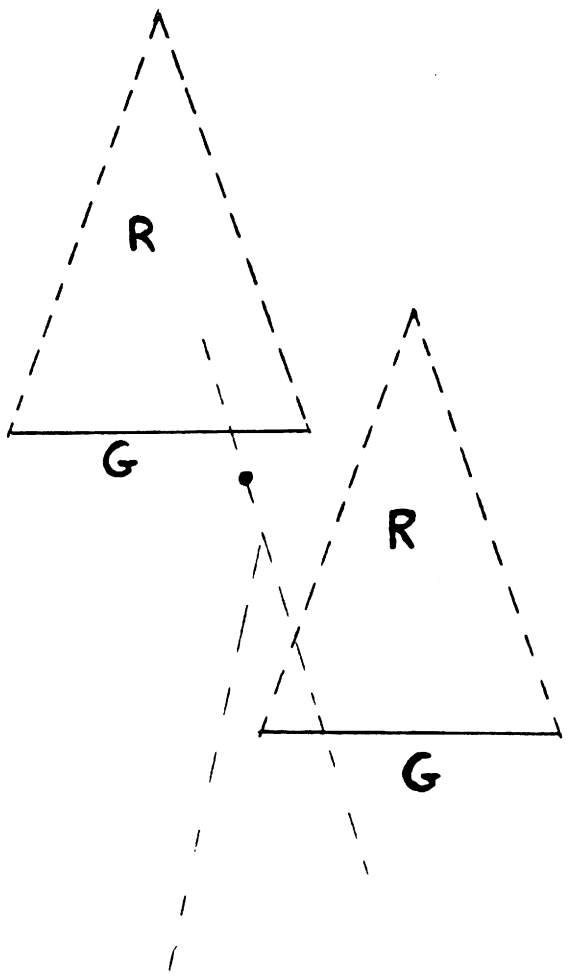


Figure 3.17. Points in the triangles are guaranteed to pass through at least one of the two horizontal lines. Some points may pass through both lines, but there is no guarantee.

Claim 3.14 $P_{\theta,R}(\{G_\alpha\}) \subseteq \overline{P}_{\theta,R}(\{G_\alpha\})$.

Proof: Similar to that of Claim 3.13. ∎

3.6. The Property of Being Self-Contained

The most striking property of backprojections, which in fact gives them their maximality, is that they are self-contained. Specifically, suppose that p is in the maximal backprojection of some collection of sets S. Then every point that is on any feasible trajectory emanating from p must also be in the maximal backprojection. Thus there are no gaps between a backprojection point p and its target in S.

Actually, to be precise, one should note that only those points that lie on a trajectory beginning at p before the trajectory's first entry into S are in the backprojection. To see this, consider Fig. 3.17. The backprojection of the two lines, under the given velocity uncertainty, yields two triangles. Consider any trajectory that begins inside one of these triangles. The trajectory eventually hits one of the two lines. Additionally, any point on the trajectory before it hits the line, is also in the backprojection. Note, however, that there are trajectories that hit both lines. Points on the portion of these trajectories between the two lines are not in the backprojection, because these points cannot be guaranteed to hit one of the two lines.

Finally, since a goal set may be open, there need not be a first time at which a trajectory enters a target set in S. However, there is a time after which the trajectory is in the open set. In general, for arbitrary sets, the time to consider is the infimum[2] of the set of all times at which the trajectory is in a target set in S.

This notion is worthy of a definition.

Definition 3.15 A set R is *semi self-contained* relative to a collection of sets S and a commanded velocity v_θ^* if and only if for any trajectory $T \in \mathcal{T}(v_\theta^*)$ with $T_p(0) \in R$ it is true that $T_p(t) \in R$ for all values of time $0 \leq t \leq t_0$, where t_0 is the infimum of the set $\{ t \geq 0 \mid T_p(t) \in A \text{ for some } A \in S \}$.

In particular, if a trajectory never enters a set in S, then it must always remain in R.

The following three claims follow from the definition. The commanded velocity's uncertainty cone is the same throughout the claims.

Claim 3.16 A maximal backprojection $B_\theta(S)$ of a collection of sets S is semi self-contained relative to that collection.

Claim 3.17 Suppose R is semi self-contained relative to S. Suppose further that S' is a collection of sets that *subsumes* S, that is, for every $A \in S$ there is a set $A' \in S'$ such that $A \subseteq A'$. Then R is semi self-contained relative to S'.

Claim 3.18 Suppose that R_1 and R_2 are both semi self-contained relative to a collection of sets S. Then $R_1 \cap R_2$ is also semi self-contained relative to S.

The first claim merely reiterates the motivation for defining the property of being semi self-contained. The second claim says that increasing the target sets, while holding R fixed, only makes it easier for R to be semi self-contained. The third claim says that a trajectory which starts in the overlap of two semi self-contained sets, remains in the overlap of those two sets at least until hitting a target set. The proofs of these claims are purely definitional. The results will be used later.

[2] The infimum $inf(A)$ is the greatest lower bound for A. If A consists of non-negative numbers then $inf(A)$ exists. It is taken to be $+\infty$ if $A = \emptyset$.

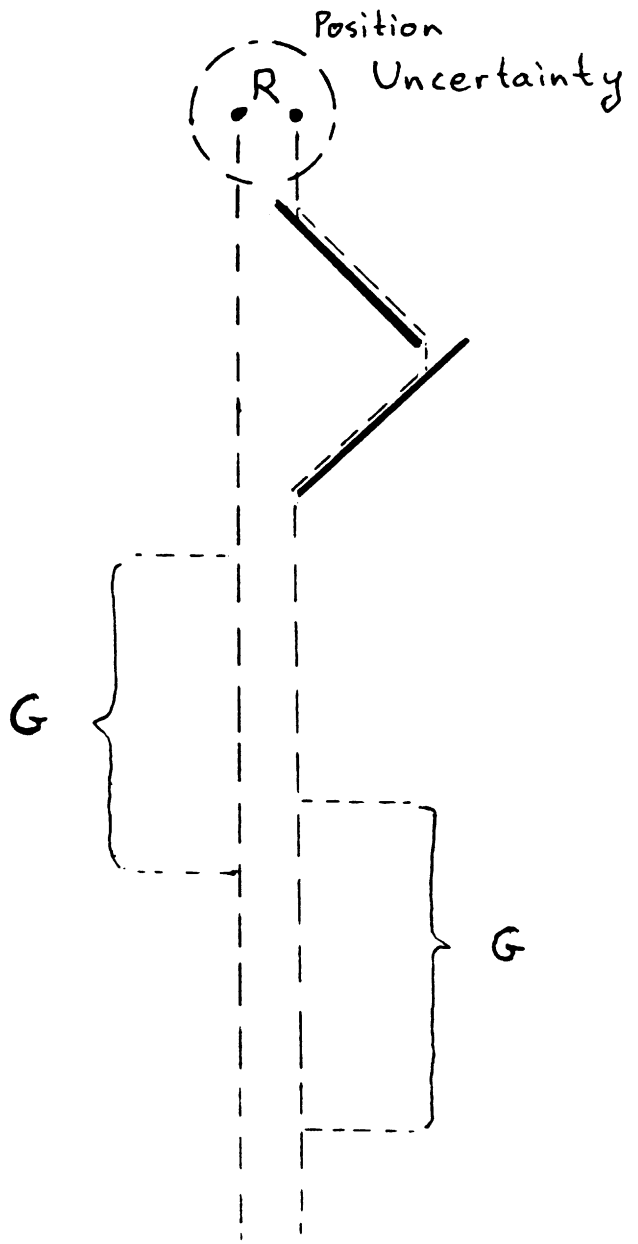


Figure 3.18. The goals and the starting points lie within the position uncertainty. The trajectories may be distinguished as they temporarily leave the uncertainty range. This observation is useful to a termination predicate that can remember sensor values.

3.7. The Power of Termination Predicates

The appendix to this chapter (Sec. 3.12) shows that pre-images are uncomputable in full generality. The uncomputability proof of that section applies as easily to backprojections as it does to pre-images. Both types of regions are uncomputable, given complex surfaces and goal regions. Backprojections do not therefore derive

their computational desirability from an increase in the class of solvable problems. The motivation for defining backprojections lies in their simplicity. It is hoped that backprojections can usefully approximate pre-images.

In order to understand the quality of this approximation it is necessary to examine the relative power of the two methods. The difference between pre-images and backprojections lies in their use of termination predicates. This section begins examining the relationship between pre-images and backprojections by considering the power of termination predicates within the definition of pre-images. A hierarchy of termination predicates is developed, which later will be used to relate pre-images and backprojections. The hierarchy makes explicit the power that is lost from pre-images by backprojection approximations.

3.7.1. The Standard Termination Predicate

Throughout the previous development the termination predicate employed is best described by the characterization of the successful velocity set $S(p_0^*, R, \{G_\alpha\})$. In other words, the termination predicate can monitor the position and velocity sensors, it has a sense of time, and it remembers the measured initial position p_0^* and the set R known to contain the actual initial position.[3]

3.7.2. The Termination Predicate with State

Mason (October 1983) described a termination predicate that remembers not only the measured initial position p_0^* and the set R, but also the sensor readings (p^*, v^*) for all previous values of time t. The advantage of this approach over the standard termination predicate is that the predicate can disambiguate interpretations of sensors using past sensor values, in addition to current sensor values and the forward projection.

Consider the example in Fig. 3.18, which is a variant of an example due to Mason (October 1983). The set R consists of the two points indicated, while the goal set G consists of the two regions indicated. The commanded velocity is straight down.

Assume that the position sensing error is a ball, as shown. The velocity control error is zero, and the velocity sensing error is irrelevant. Since the two points that comprise R lie within the position sensing uncertainty, a termination predicate cannot decide where the actual position starts. Thus the forward projection includes at all times points from both trajectories that emanate from R.

Each trajectory eventually passes through a goal region. Unfortunately, every point in the goal region of one trajectory can be confused with non-goal points either on the same or on the other trajectory, because of the position uncertainty. Thus the standard termination predicate cannot signal success. However, Mason's termination predicate can signal success, since there is a point in time at which the two trajectories are far enough apart to be distinguishable. Remembering the sensor values, the termination predicate can decide on which trajectory a particular motion lies. The termination predicate can therefore rule out non-goal interpretations when in the middle of the trajectory's goal region.

[3]The set R is a previous subgoal that was attained by a previous motion.

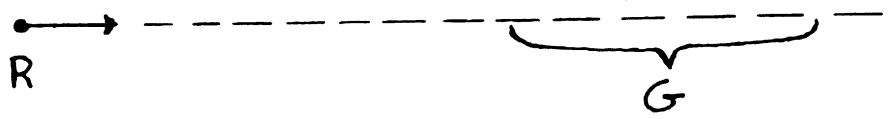


Figure 3.19. With infinite position uncertainty and perfect control, entry into the goal region is detected by observing the time sensor.

3.7.3. The Termination Predicate with no Sense of Time

The termination predicate with no sense of time is a standard termination predicate which cannot consider time in making decisions. Thus, in disambiguating sensor interpretations, the termination predicate must use the timeless forward projection. In other words, the set of interpretations is given by $F_{p_0^*,\theta,R} \cap B_{ep}(p^*) \times B_{ev}(v^*)$, rather than by $F_{p_0^*,\theta,R}(t) \cap B_{ep}(p^*) \times B_{ev}(v^*)$, as with the standard termination predicate.

Consider the example of Fig. 3.19. The region R is a single point, as shown. The commanded velocity is towards the right. Assume zero control error, but infinite position sensing error. The standard termination predicate can detect entry into the goal, by keeping track of the distance covered. The termination predicate with no sense of time cannot detect entry into the goal, since it requires knowledge of the elapsed time in order to calculate the distance travelled. A similar example applies for non-zero finite error in both control and sensing.

3.7.4. The Termination Predicate without History or Time

The termination predicate without history is a termination predicate with no sense of time that does not remember the initial measured position p_0^*. Intuitively, a termination predicate with no history is one that bases all its decisions on current data, with no clock or recollection of the past, other than the previously accomplished subgoal. Consequently, in disambiguating sensor values, a termination predicate without history can only use knowledge of the previously attained subgoal R. Thus the set of interpretations is given by $F_\theta(R) \cap B_{ep}(p^*) \times B_{ev}(v^*)$, where $F_\theta(R)$ is given by

Definition 3.19 The *forward projection (without history)* of a set R under the commanded velocity v_θ^* is given by

$$F_\theta(R) = \bigcup_{t \geq 0} \left\{ (T_p(t),\, T_v(t)) \;\middle|\; T \in \mathcal{T}(v_\theta^*) \text{ and } T_p(0) \in R \right\}.$$

This forward projection defines all positions and velocities reachable by a trajectory that starts in R, subject to the given velocity uncertainty.

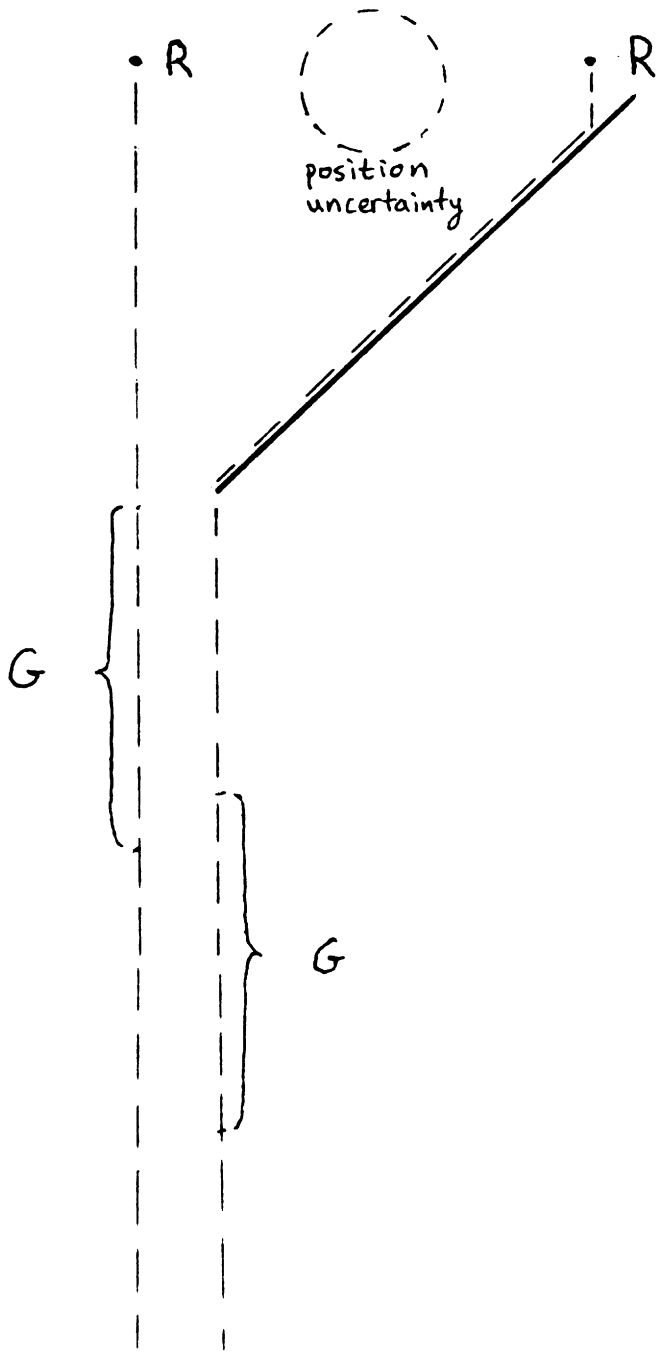


Figure 3.20. The goal regions lie within each others position uncertainty. The initial trajectory positions are distinguishable. This of no use to the termination predicate without history, as it cannot remember the starting position. Thus the termination predicate without history cannot recognize entry into the goal regions.

It is useful also to define the position component of the forward projection. Let π denote the projection function into the first coordinate. Specifically, $\pi(p, v) = p$. Now let πF_θ represent the position component of the forward projection F_θ.

3.7.5. No Sense of Time vs. No History

The difference between the termination predicate with no sense of time and the termination predicate without history is fairly clear. Consider Fig. 3.20, with the specified position uncertainty. The initial set R and the goal regions G are shown. The commanded motion is straight down. Assume perfect velocity control. Thus the forward projection of the set R consists of two trajectories emanating from R. Notice that the two points comprising R are distinguishable, based on position sensing, while each trajectory's goal region may be confused with non-goal points either on the same or on the other trajectory.

The termination predicate with no sense of time remembers the initial measured position p_0^*, so it can decide on which trajectory emanating from R an actual motion lies. Thus it knows which goal region to expect. The termination predicate without history, on the other hand, cannot signal success. It cannot distinguish the two trajectories, as it is only permitted to remember starting in R.

In order to formally characterize the difference between the two termination predicates, let S_{time} and $S_{history}$ be the successful velocity sets defined for the two termination predicates. These sets are defined analogously to the standard termination predicate's successful velocity set $S(p_0^*, R, \{ G_\alpha \})$ (see Claim 3.10). The only real difference lies in the disambiguating sets. In place of $F_{p_0^*,\theta,R}(t)$, there appears $F_{p_0^*,\theta,R}$ for the termination predicate with no sense of time, and $F_\theta(R)$ for the termination predicate without history. Of course, p_0^* does not appear in the definition of $S_{history}$.

For convenience, the precise definitions of these two successful velocity sets are given here.

Definition 3.20 $v_\theta^* \in S_{time}(p_0^*, R, \{ G_\alpha \})$ if and only if for every trajectory $T \in \mathcal{T}(p_0^*, R, v_\theta^*)$, there is some time $t \geq 0$, such that, for any sensor value (p^*, v^*) consistent with T at time t, there is some goal set $H \in \{ H_\alpha \}$ for which $F_{p_0^*,\theta,R} \cap B_{ep}(p^*) \times B_{ev}(v^*) \subseteq H$. The corresponding definition of pre-image is as before

$$P_{\theta,R}(\{ G_\alpha \}) = \left\{ p \in R \;\middle|\; \forall p_0^* \in B_{ep}(p),\ v_\theta^* \in S_{time}(p_0^*, R, \{ G_\alpha \}) \right\}. \qquad (3.24)$$

Definition 3.21 $v_\theta^* \in S_{history}(R, \{ G_\alpha \})$ if and only if for every trajectory $T \in \mathcal{T}(v_\theta^*)$, with $T_p(0) \in R$, there is some time $t \geq 0$, such that, for any sensor value (p^*, v^*) consistent with T at time t, there is some goal set $H \in \{ H_\alpha \}$ for which $F_\theta(R) \cap B_{ep}(p^*) \times B_{ev}(v^*) \subseteq H$. The corresponding definition of pre-image is

slightly changed, specifically

$$P_{\theta,R}(\{ G_\alpha \}) = \begin{cases} R, & \text{if } v_\theta^* \in S_{history}(R, \{ G_\alpha \}) \\ \emptyset, & \text{otherwise} \end{cases} \tag{3.25}$$

Given these observations, the following claim follows immediately. Recall that a suitable pre-image is a set R that satisfies the pre-image equation $R = P_{\theta,R}(\{ G_\alpha \})$.

Claim 3.22 R is a suitable pre-image with respect to the termination predicate that has no sense of time, if and only if, for every $p_0^* \in B_{ep}(R)$, the set $B_{ep}(p_0^*) \cap R$ is a suitable pre-image with respect to the termination predicate that has no history.

The claim says simply, that the only advantage that history gives the termination predicate with no sense of time over the termination predicate without history, is in narrowing down the starting set from R to $B_{ep}(p_0^*) \cap R$.

3.7.6. Comments on the Loss of Power

Both the standard termination predicate and Mason's termination predicate with continuous state are very powerful. In computing pre-images using these termination predicates, it appears necessary to precompute forward projections indexed by time. The current definition of backprojections does not incorporate time. In fact, it is clear that backprojections are just timeless forward projections in reverse. Consequently, by studying pre-images that employ termination predicates with no sense of time, it should be possible to ascertain the relationship of backprojections to pre-images.

These remarks make clear the loss of power taken by approximating pre-images with backprojections. Tasks whose plans involve motions that must monitor time cannot be solved using backprojections.

Finally, the relationship expressed by Claim 3.22 suggests that one study pre-images generated by the termination predicate without history. One can always patch together distinguishable collections of such pre-images to create a pre-image generated by the termination predicate with no sense of time. Conversely, any pre-image with respect to the termination predicate with no sense of time must be separable into a distinguishable collection of simpler pre-images.

By a *distinguishable collection of sets* is meant a collection of sets from which it is always possible to select a set containing the actual, albeit unknown, position p corresponding to a measured value p_0^*. The example of Sec. 2.3.7 and Fig. 2.50 created a distinguishable collection of triangles interior to the triangles generated by backprojection.

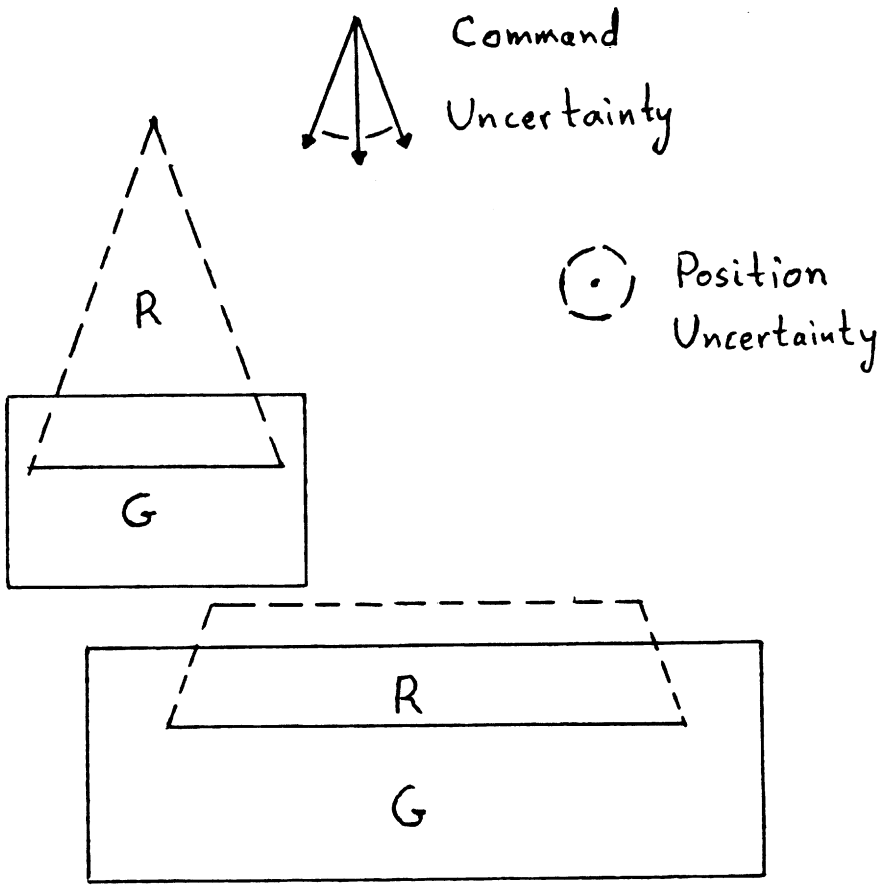


Figure 3.21. A goal region consisting of two rectangular areas, along with two pre-images.

3.8. The Relationship Between Backprojections and Pre-Images — Basic Issues

The search for maximal pre-images tried to discover a collection of simple pre-images from which all other pre-images could be easily constructed. That search failed. In its place, backprojections were developed. Given any collection of sets, it was possible to define a unique maximal backprojection of those sets. Furthermore, it was observed in the last section, that, having removed time from the termination predicate used in defining pre-images, the decision to signal success during a motion is based on a forward projection that is essentially just the inverse of a backprojection. This suggests a relationship between maximal backprojections and pre-images computed using the termination predicate without history or time. This section examines that relationship. It is assumed throughout that the termination predicate is without history or time.

3.8.1. Reachable Goals

By way of introduction, recall that all pre-images of a collection of sets $\{ G_\alpha \}$ are subsets of the maximal backprojection of the same collection, that is, $P_{\theta,R}(\{ G_\alpha \}) \subseteq B_\theta(\{ G_\alpha \})$ (see Eq. (3.17)). Thus the maximal backprojection $B_\theta(\{ G_\alpha \})$ is itself not a very useful tool. Consider, however, all subcollections of the given collection of goals, that is, all $\{ G_\beta \} \subseteq \{ G_\alpha \}$. It is convenient, though not necessary, to assume that $\{ G_\alpha \}$ is closed under subsets, specifically, that $G \in \{ G_\alpha \}$ whenever $G \subseteq G_i$ for some $G_i \in \{ G_\alpha \}$. By considering all subcollections $\{ G_\beta \}$ and their maximal backprojections $B_\theta(\{ G_\beta \})$, one hopes that it should be possible to determine a relationship between backprojections and pre-images.

Suppose that the initial set R and the commanded velocity v_θ^* are given. There is a distinguished subcollection of the goal sets $\{ G_\alpha \}$ which appears to be a good candidate for approximating $P_{\theta,R}(\{ G_\alpha \})$. Specifically, let $\mathcal{G} = \mathcal{G}(R, v_\theta^*, \{ G_\alpha \})$ be the collection of all goals that might be returned by the termination predicate. Recall the definition of $S_{history}$. If T is a trajectory starting in R, then there exists at least one point in time, t, at which the set of all consistent interpretations $F_\theta(R) \cap B_{ep}(p^*) \times B_{ev}(v^*)$ of any sensor reading (p^*, v^*) is the subset of some cylindrical goal set $H \in \{ H_\alpha \}$. H depends on (p^*, v^*). Assuming that $\{ G_\alpha \}$ is closed under subsets, the position component of the set of interpretations, that is, $\pi(F_\theta(R) \cap B_{ep}(p^*) \times B_{ev}(v^*))$ is equal to some goal $G \in \{ G_\alpha \}$. The termination predicate returns this set G as the attained subgoal.

Note that for each possible sensor reading (p^*, v^*) that is consistent with T at time t, there is some goal set G which the termination predicate returns. Also, there may be several points in time at which the termination predicate could signal success for the given trajectory T. The collection $\mathcal{G}$ is simply the union of all such goal sets. The union is taken over all possible trajectories starting in R, over all possible points in time at which the termination predicate could signal success for these trajectories, and over all possible consistent sensor readings at these times.

The collection $\mathcal{G}$ is a subcollection of $\{ G_\alpha \}$. It comprises all the goals that could possibly ever be returned by the termination predicate. The goal sets that are in $\{ G_\alpha \}$ but not in $\mathcal{G}$ play no role in the termination predicate's decisions. In other words, $P_{\theta,R}(\{ G_\alpha \}) = P_{\theta,R}(\mathcal{G})$. Thus $\mathcal{G}$ consists of precisely those goals needed to define the pre-image R.

3.8.2. Backprojecting from the Reachable Goals

3.8.2.1. Straight Backprojection

Now consider $B_\theta(\mathcal{G})$, the maximal backprojection of the collection of goals reachable from R. The question is whether $B_\theta(\mathcal{G})$ can be used to approximate $P_{\theta,R}(\mathcal{G})$. Suppose in fact that R is a suitable pre-image, that is, that $R = P_{\theta,R}(\{ G_\alpha \})$. Then it follows that $R \subseteq B_\theta(\mathcal{G})$. The inclusion may be a proper inclusion. This is not surprising, since different sets can give rise to the same collection of reachable

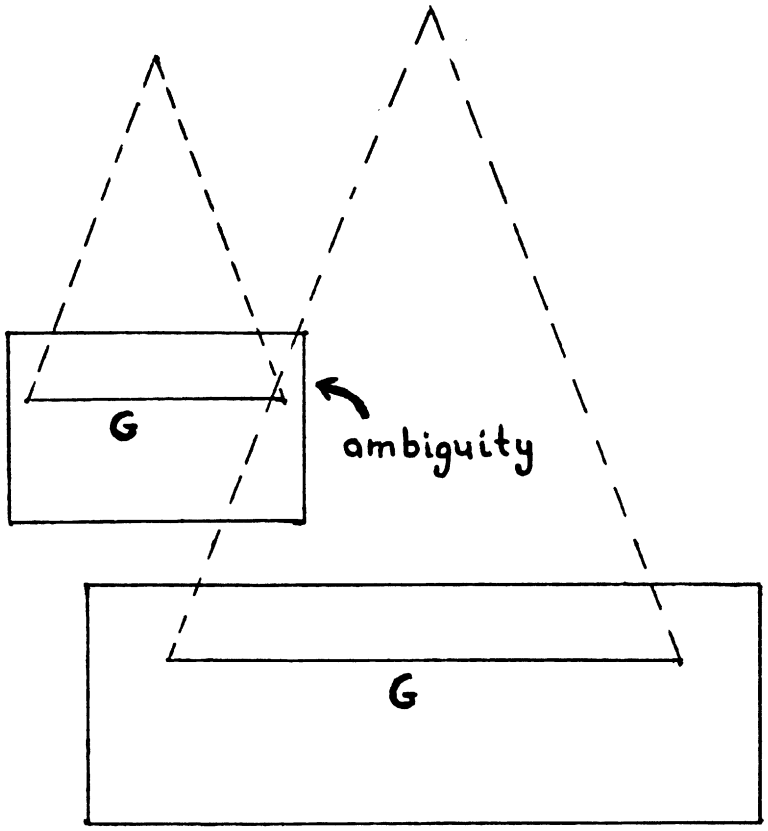


Figure 3.22. Backprojection of the reachable goals of Fig. 3.21. This backprojection introduces extra non-goal interpretations for the top left goal region.

---

goals. However, perhaps $B_\theta(\mathcal{G})$ can serve as a useful pre-image building block, from which other pre-images can be constructed by subsetting. The discovery of such building blocks is precisely the relationship being sought.

While it is certainly true that every pre-image whose reachable goal set is $\mathcal{G}$, must be a subset of $B_\theta(\mathcal{G})$, this is not a very useful relationship, since $B_\theta(\mathcal{G})$ may itself not be a pre-image. Basically, there are two properties that prevent $B_\theta(\mathcal{G})$ from being a suitable pre-image. First, note that the termination predicate consults the set of interpretations given by the forward projection $F_\theta(R)$ in deciding to halt a motion. There is absolutely no reason to believe that $B_\theta(\mathcal{G})$ is a subset of this forward projection. Thus the termination predicate might not be able to halt previously successful motions, because its set of interpretations has grown from $F_\theta(R)$ to $F_\theta(B_\theta(\mathcal{G}))$. The point is that $B_\theta(\mathcal{G})$ might include points outside of all suitable pre-images, that is, points that can reach $\mathcal{G}$, but can't do so in a manner that the termination predicate can recognize.

For a specific example, consider Fig. 3.21. The goal region consists of the two rectangles specified. Attached to these are a triangle and a trapezoid, which

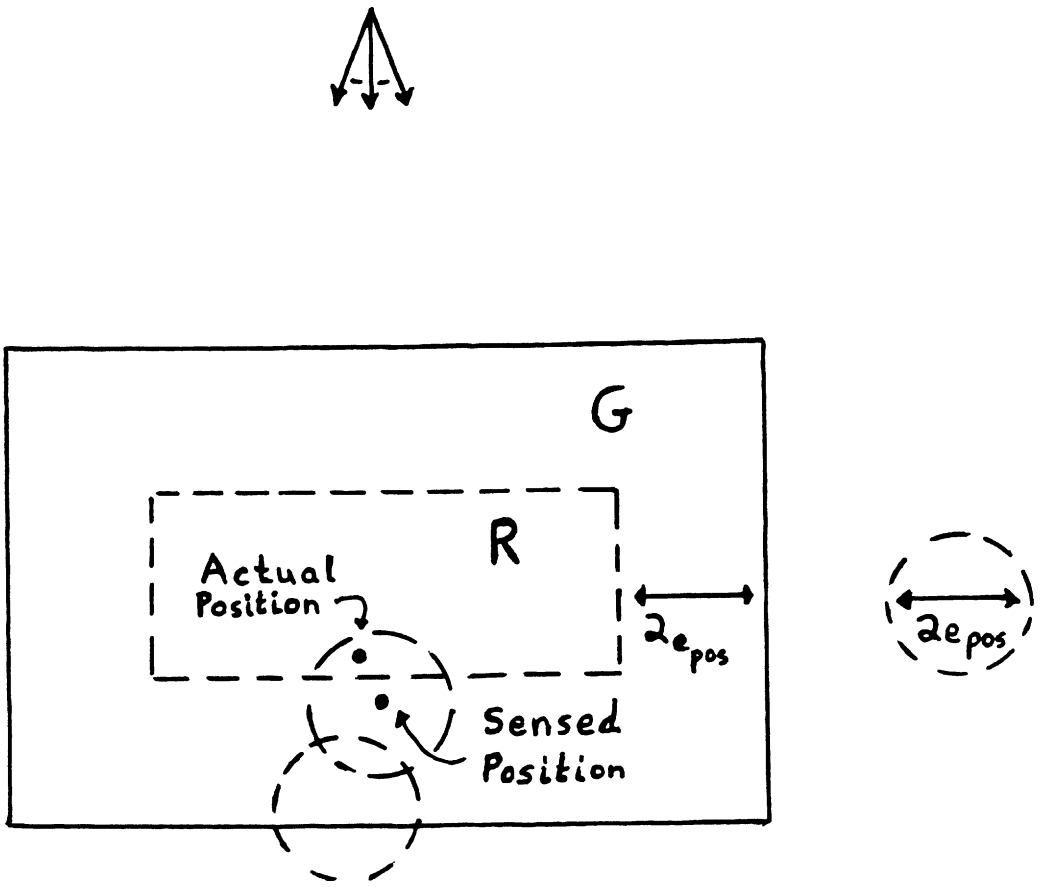


Figure 3.23. The center circle depicts the possible sensor interpretations corresponding to an actual point in the inner rectangle. The interpretations all lie within the goal, but are not themselves valid pre-image points. This is because sensor interpretations of points in the center circle may lie outside of the goal region. This is indicated by the bottom circle.

---

constitute the set R. The commanded velocity is straight down with the indicated error cone. Motion success is detected by position sensing alone. Now suppose that the backprojection of the reachable goals is considered in place of the set R. The backprojection is sketched in Fig. 3.22. It is no longer possible for the termination predicate to successfully terminate all motions. This is because some of the trajectories that used to terminate in the upper left goal region, now have interpretations that lie outside of that goal region, namely in the backprojection of the lower right goal region.

3.8.2.2. Backprojection Intersected with the Forward Projection

This first problem is easily solved by considering $F_0(R) \cap B_0(\mathcal{G})$ instead of the larger $B_0(\mathcal{G})$. By intersecting with the forward projection, the backprojection is guaranteed to remain inside the set of interpretations known to be recognizably successful. Now the set $F_0(R) \cap B_0(\mathcal{G})$ would appear to be a good pre-image building block. It contains the set R, and appears to be a suitable pre-image. Certainly any point in $F_0(R) \cap B_0(\mathcal{G})$ is guaranteed to move to one of the goal sets which are known to be recognizable by the termination predicate.

Unfortunately, $F_0(R) \cap B_0(\mathcal{G})$ is not a suitable pre-image either. To see this,

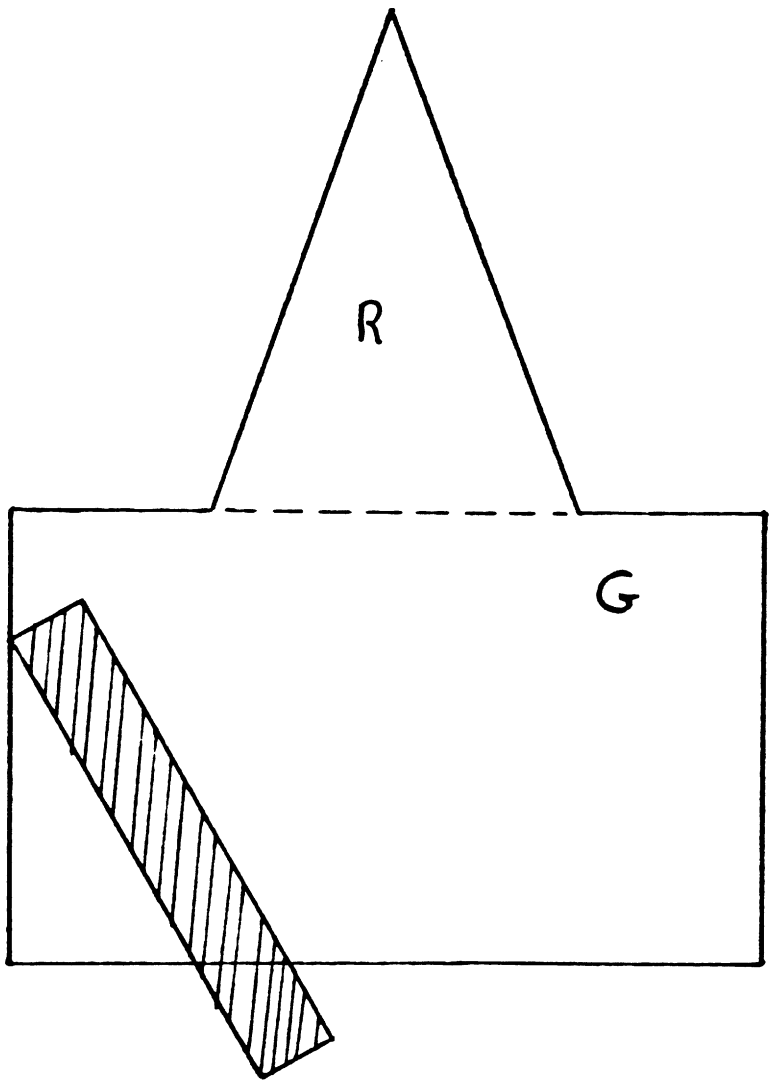


Figure 3.24. The goal is the entire region except for the right edge of the obstacle. The pre-image R is the triangle. The commanded velocity is straight down with the usual error cone. All points in the triangle are recognizably in the goal.

---

consider the second reason that prevents $B_0(\mathcal{G})$ from being a suitable pre-image. The difficulty is the inclusion of the goal sets $\mathcal{G}$. This may seem peculiar, since these goals are precisely those which the termination predicate might return as successfully attained goals. However, merely because a set constitutes the successful interpretations of a terminated motion, does not imply that every point in that set is itself on a successful trajectory.

This point is clarified by the example of Fig. 3.23. The commanded velocity is straight down, with the usual cone. The goal region is delineated by the solid line rectangle. Velocity sensing does not enter into the picture. Instead, success is achieved by entering the rectangular region indicated by the dashed line. Inside

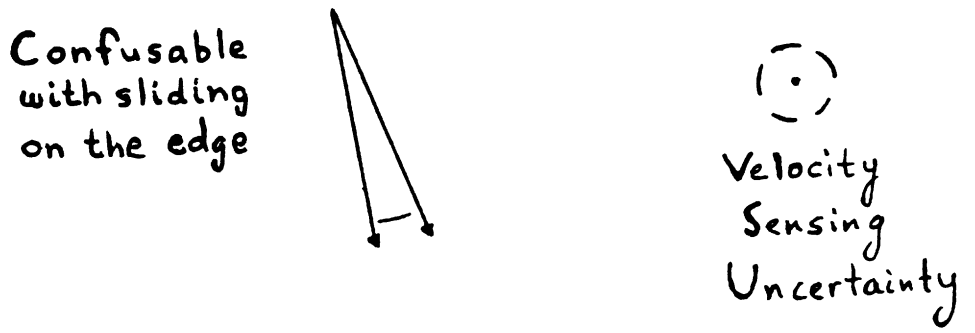


Figure 3.25. The small cone corresponds to free space velocity directions that are confusable with sliding velocities on the edge, due to velocity sensing uncertainty.

---

this region, the position sensors unambiguously indicate that the actual position is inside the goal. All interpretations of any position sensor reading consistent with an actual position inside the dashed line region lies inside the goal. Now consider one possible set of interpretations of a sensor value, as shown in Fig. 3.23. Such a set would be part of the goal sets described by $\mathcal{G}$. Note, however, that a portion of this set lies outside of the dotted line region. These points are not starting points of any successful trajectory, as all trajectories starting at these points always have interpretations of consistent sensor readings that lie outside of the goal. Thus the set of interpretations contains points that cannot be in any pre-image which also contains R.

3.8.2.3. Backprojection from Successful Termination Points

The second problem described above apparently arises because there are points in the reachable goals which are simply interpretations of termination positions without themselves being valid termination points. This is only half the story. Were it all, then one might consider backprojecting from a distinguished subset of the points in $\mathcal{G}$. Specifically, recall that the collection $\mathcal{G}$ is constructed from the set of interpretations of trajectories at successful termination times. Instead of considering all possible consistent interpretations of a trajectory T at time t, suppose one only considered the actual trajectory position $T_p(t)$. Backprojecting from the set of all such actual trajectory positions at times of success avoids the problem of including in the backprojection points that arise merely as interpretations. Any point in the backprojection is guaranteed to move to a point that is actually the successful termination point of some trajectory at some time.

Once again, this optimism quickly gives way to reality. Backprojecting from the set of successful termination positions is fine if the termination predicate only employs position sensors. With velocity sensors, the story is different. A point may be a successful termination point for one trajectory with a particular velocity at

that point, yet not be a successful termination point for another trajectory with a different velocity.

Consider the example of Fig. 3.24. The goal set is the entire region outlined by the solid line, except for the right edge of the obstacle. The initial set R is the top triangular portion of the region. The commanded velocity is straight down, with the usual error cone. The position uncertainty is chosen so that all points in R are automatically decidable to be in G, using the forward projection of R and position sensors alone. Thus R is a suitable pre-image. Since the edge causes sliding it is possible to decide that a point is not on the edge by observing the velocity sensors. This is possible for some effective commanded velocities, but not for all, because of the uncertainty in the velocity sensors (see Fig. 3.25). In fact, given the indicated velocity sensing error, if the effective commanded velocity lies in the range indicated by the cone of Fig. 3.25, then it is impossible to distinguish between sliding on the edge and moving in free space (see also Fig. 3.38).

Thus, if a point is within the position uncertainty of the edge, then for some effective commanded velocities it is impossible to decide that the point is not on the edge, while for others it is possible to decide that the point is not on the edge. The bad velocity can be so chosen, that the point remains close enough to the edge to be undecidable all the while it is in the goal region $G - R$. In short, there are points that are successful termination points of trajectories, but are not themselves pre-image points.

The basis for this counterexample stems from the inclusion of points that are recognizable as being in a goal only for certain velocities. Notice, however, that these points are preceded by other successful termination points. Were they not so preceded, then it would be possible to find trajectories emanating from R which never terminated successfully. This would contradict the observation that R is a suitable pre-image.

3.8.3. Backprojection in Phase Space

The form of the set of interpretations suggests, and the counterexample of the last subsection underscores the point that goal sets are both position and velocity goal sets. Goal sets lie in the tangent bundle of a manifold, not just in the manifold itself. Consequently, what is needed is a means of performing backprojections in phase space. Given a goal in the form of a set of desired positions and velocities, the phase space backprojection computes all positions and velocities from which motions are guaranteed to reach the goal.

Phase space backprojection provides the tool necessary to relate backprojections to pre-images. Specifically, the phase space backprojection of a set of successful termination points in phase space must be a pre-image relative to a termination predicate without history, much in the same way that a regular backprojection of a set of successful termination points is a pre-image relative to a termination predicate that only consults position sensors.

Lest one become drunk with success, one should note that phase space backprojections do not make much sense for a first order system. For a first order system one cannot specify velocities at resolutions better than the uncertainty in the commanded velocity. One can certainly insure that velocity uncertainties shrink in dimension by forcing sliding on some surface or on the intersection of some surfaces. However, one cannot specify the tangential sliding velocity at accuracies better than the tangential uncertainty arising from the uncertainty in the commanded velocity. This is similar to being unable to specify, in a pure position control system, position accuracies on surfaces of any given dimension at resolutions better than the inherent positioning uncertainty for motions of that dimension.

The only subgoals that can reliably be achieved are goals that are subsets of position space, although, of course, both position and velocity information should be used in achieving these goals. This is apparent from the definition of pre-image, which only considers previously attained position subgoals R. Assuming previously attained velocity subgoals would be absurd if those subgoals constrained velocity more than was inherently possible.[4] On the other hand, backprojections in phase space that never overconstrain velocity are equivalent to regular backprojections in position space.

The only method by which one could perform true phase space backprojection would be to consider an underlying second order model. However, in such a model accelerations would presumably be both control commands and sensed qualities, both subject to uncertainty. Consequently, any system that used position, velocity, and acceleration sensing in making motion termination decisions, while commanding accelerations, would be faced with the desire but inability to backproject in $position \times velocity \times acceleration$ space. Thus the problem of overconstraining a control command subject to uncertainty would not be solved, but only pushed back a level.

[4] This is precisely why the velocity subgoals that are used are just cylinders. Note that the set of interpretations defined by the forward projection $F_\theta(R)$ never overconstrains velocity.

The simplicity of backprojections from a computational viewpoint, both in position and in phase space, is their historical insensitivity. Whether a point reaches a goal is independent of knowing whether any neighboring or preceding point reaches a goal. However, in order to compute pre-images, some local history is required. Ultimately, in constructing good goal sets, one must take this local history into account. This amounts to computing backprojections that consider the velocity dependencies between motions. For example, knowing that a motion cannot be recognized for some velocity in a certain goal may imply that the motion must next enter a different goal, in which it will be recognized. A later section addresses this issue briefly. For the time being however, the main task is to find a direct relationship between regular maximal backprojections and pre-images.

3.8.4. Almost Simple Pre-Images

The difficulty with $F_\theta(R) \cap B_\theta(\mathcal{G})$ and its variant consisting only of successful termination points, is that they include extra points arising from goal portions that are not themselves pre-images. Nonetheless, these sets appear attractive due to their simplicity. Furthermore, they differ only slightly from what is really required, as the previous pages have shown. The problem therefore, is to prune the set from which backprojection is being performed. The set $\mathcal{G}$ and its variant consisting only of successful termination points, are simply too large. The next section investigates a replacement set for $\mathcal{G}$. This subsection considers the properties that should be retained while trimming the set $F_\theta(R) \cap B_\theta(\mathcal{G})$.

Recall that the maximal backprojection $B_\theta(\mathcal{S})$ of any collection of sets $\mathcal{S}$ is semi self-contained relative to that collection (see Sec. 3.6). Also note that the forward projection $\pi F_\theta(R)$ of any set R is semi self-contained relative to the collection $\{\emptyset\}$ consisting solely of the empty set. This is because any motion that starts in $\pi F_\theta(R)$ remains in $\pi F_\theta(R)$, by definition. Therefore, by Claim 3.17, $\pi F_\theta(R)$ is semi self-contained relative to any collection of sets $\mathcal{S}$. Furthermore, the intersection $\pi F_\theta(R) \cap B_\theta(\mathcal{S})$ is also semi self-contained relative to $\mathcal{S}$, by Claim 3.18.

The property of being semi self-contained is analogous to the property of being maximal. In a sense, a semi self-contained set is a complete set. It contains all points it can possibly contain within a certain range. Specifically, one can add no points that lie on trajectories between points in the set and points in some distinguished collection of sets. Being semi self-contained is not quite like being maximal. It is still possible to add points before the set, or after the targets $\mathcal{S}$, but being semi self-contained is the best that is achievable for backprojections that are also to be pre-images.

Under the belief that pre-image building blocks should be pre-images that are also semi self-contained backprojections, the following definition is made.

Definition 3.23 A pre-image R, with respect to a collection of goals $\{ G_\alpha \}$ and a commanded velocity v_θ^* is said to be *almost simple* if and only if there is some set E_R such that the following four conditions hold.[5]

$$(i) \quad R = P_{\theta,R}(\{ G_\alpha \})$$

$$(ii) \quad E_R \subseteq R$$

$$(iii) \quad E_R \subseteq \overline{\bigcup_\alpha G_\alpha}$$

$$(iv) \quad R = \pi F_\theta(R) \bigcap B_\theta(\{E_R\})$$

The first condition simply says that R has to be a suitable pre-image. The fourth condition says that R can be written as a semi self-contained backprojection. The choice of the form of this backprojection, namely as the intersection of a forward projection and a maximal backprojection, stems from the intuition gained in the previous subsections on reachable goals. The second condition simply says that the generating set should itself be part of the pre-image. This requirement avoids cluttering the set E_R with unnecessary points that lie outside of $\pi F_\theta(R)$. The third condition insures that the set from which the backprojection is constructed is contained in the closure of the union of all the goals. It should be clear, given the previous discussion of reachable goals, that $\{ G_\alpha \}$ could be replaced by $\mathcal{G}$. The reason for choosing the closure of the goals, rather than just the goals themselves, is due to a technical point. This will become clear later. Essentially, the issue is that, as in the definition of semi self-contained, for open sets there may not exist distinguished points or times, only limits and infima.

The next sections are concerned with showing that almost simple pre-images actually exist. Additionally, these sections show that almost simple pre-images form the basic building blocks from which all other pre-images may be constructed as subsets. Thus almost simple pre-images provide the desired relationship between backprojections and pre-images.

3.9. First Entry Sets and the Structure Equation

The previous discussion has shown that the set of all reachable goals and its variant, the set of all successful termination points, are too large to serve as the generating sets of an almost simple pre-image. Both sets contain points that need not themselves be pre-image points. One can, of course, insist that these points also be pre-image points. In fact, this is precisely what any practical application should require. Later, in discussing good goal sets, this assumption will be made. However, in order to understand the relationship between backprojections and pre-images, one must seek a general type of set that acts as a generating set for almost simple

[5]The closure of a set A relative to some topology is denoted by $\overline{A}$.

pre-images. This section exhibits such a type of set, thereby also demonstrating the existence of almost simple pre-images.

3.9.1. First Entry Points

One wonders precisely in what sense reachable goals and sets of termination points are too large. The examples of the previous section provide a clue. Recall, in the example of Fig. 3.24, the successful termination points that were not themselves pre-image points. It was noted that these points had to lie on trajectories that passed through preceding successful termination points. In short, the fact that these non-pre-image points were successful termination points was almost coincidental and certainly irrelevant. Trajectories passing through these points could be recognized as being in the goal at earlier times.

The previous observation suggests that perhaps there are points in a goal that are distinguished termination points. Specifically, there should be points that are the first points at which a trajectory could successfully terminate. Whether there are any points after these first termination points is irrelevant, as the trajectory has at least passed through one termination point.

It is clear that there need not, in general, exist such first termination points. For example, if the goal is an open set, then, even with perfect sensing and control, there is never a first point at which a trajectory can be terminated.

Furthermore, notice that knowledge of whether a trajectory can be terminated at a single point is not very informative. One can always change a trajectory's velocity at a single point[6] without affecting the path of the trajectory. Consequently, unless successful termination can be signalled for all possible effective commanded velocities within the velocity uncertainty cone, knowing that the trajectory can be terminated for some effective commanded velocity says absolutely nothing about whether the termination point is itself a pre-image point.

A notion closely allied to first termination points is suggested by the open set counterexample. Rather than consider first termination points, consider earliest times of termination. Specifically, consider the time which is the infimum of all possible termination times of a trajectory. Denote the position of the trajectory at that time as the *first entry point* of the trajectory. It may or may not be possible to successfully terminate the trajectory at the first entry point, but it certainly is not possible to terminate the trajectory at any point preceding the first entry point. Thus first entry points provide the proper generalization of first termination points. This leads to the following definition.

Definition 3.24 Suppose that R is a pre-image satisfying the pre-image equation relative to a commanded velocity v_θ^* and a collection of goals $\{ G_\alpha \}$. Define the *first entry set* $E(R)$ of R to be the set of all first entry points of trajectories that start in R. Specifically,

$$E(R) = \left\{ T_p(t_0) \,\middle|\, T \in \mathcal{T}(v_\theta^*),\ T_p(0) \in R,\ \text{and}\ t_0 = inf(S_T) \right\}, \tag{3.26}$$

[6] In fact, one can do so on any set of measure zero, given the integrability condition (3.18).

where S_T is the set of all successful termination times of the trajectory T, that is

$$S_T = \{\, t \mid t \text{ satisfies the condition of Def. 3.21. for the trajectory } T \,\}. \quad (3.27)$$

The first entry set is well defined, since R is a suitable pre-image. In other words, when R is not void, the first entry times are non-negative and finite, since the sets of successful times S_T are not empty. When R is void, so is $E(R)$.

3.9.2. Properties of First Entry Sets

The following subsection establishes the simple properties of first entry sets that are the basis for using these sets to generate almost simple pre-images.

The first claim proves that first entry points possess the property which motivated their definition. Specifically, the claim implies that first entry points are themselves valid pre-image points. Any trajectory that begins at a first entry point is guaranteed to terminate recognizably in a goal. Recall that the lack of this property for reachable goals and successful termination points prompted the search for almost simple pre-images.

The claim actually says that any trajectory which passes through any first entry point is guaranteed to recognizably enter a goal some time after passing through the first entry point. The statement of the previous paragraph follows immediately from this claim. Note that the claim does not assume that the trajectory passes through its own first entry point, merely any point in the first entry set.

Claim 3.25 Suppose that R satisfies the pre-image equation $R = P_{\theta,R}(\{\, G_\alpha \,\})$ with respect to the termination predicate without history. Let T be a trajectory in $\mathcal{T}(v_\theta^*)$ with initial point in R. Suppose that at time t_0, $T_p(t_0) \in E(R)$, that is, at time t_0 the trajectory is in the first entry set of R. Then there is some time $t \geq t_0$ at which the condition of Def. 3.21 is satisfied.

Proof: First, one can assume without loss of generality that $t_0 > 0$, as otherwise the claim follows by definition of suitable pre-image. Let $p_0 = T_p(t_0)$. Since p_0 is a first entry point, there is some trajectory $\hat{T} \in \mathcal{T}(v_\theta^*)$, with $\hat{T}_p(0) \in R$, such that $p_0 = \hat{T}_p(\hat{t})$, where $\hat{t} = inf(S_{\hat{T}})$, as in Def. 3.24.

Now suppose the claim is false. Then for every $t \geq t_0$ there is some sensor reading (p^*, v^*) consistent with T at time t, such that the set of consistent interpretations $F_\theta(R) \cap B_{ep}(p^*) \times B_{ev}(v^*)$ is not the subset of any goal cylinder $H \in \{\, H_\alpha \,\}$. Define the trajectory $\tilde{T}$ by

$$\tilde{T}(t) = \begin{cases} \hat{T}(t), & 0 \leq t < \hat{t} \\ T(t + t_0 - \hat{t}), & \hat{t} \leq t \end{cases} \quad (3.28)$$

In other words, up to, but not including, p_0, $\tilde{T}$ is same as the trajectory $\hat{T}$ which gives rise to the first entry point p_0, after which $\tilde{T}$ is the same as the original

trajectory T. Note that

$$\begin{aligned} \tilde{T}_p(\hat{t}) &= T_p(t_0) = \hat{T}_p(\hat{t}) \\ \tilde{T}_v(\hat{t}) &= T_v(t_0). \end{aligned} \tag{3.29}$$

By definition of T and $\hat{T}$, $\tilde{T} \in \mathcal{T}(v_\theta^*)$ and $\tilde{T}_p(0) \in R$. Furthermore, $\tilde{T}$ cannot be terminated successfully at any time t with $0 \leq t < \hat{t}$, by definition of $\hat{t}$ as the first entry time for $\hat{T}$. Additionally, by the assumption that the claim is false, $\tilde{T}$ cannot be terminated for any time t with $t \geq \hat{t}$. But this contradicts the fact that R is a suitable pre-image. ∎

Note how the fact that the termination predicate has no history was used in the proof. The fact that the termination predicate has no sense of time was used to glue the trajectories T and $\hat{T}$ together. This, in combination with the fact that the termination predicate has no history, was used to retain the same forward projection $F_\theta(R)$ throughout.

The second claim of this subsection states the obvious fact, that a suitable pre-image lies inside the backprojection of its first entry set. In other words, any motion emanating from a pre-image is guaranteed to pass through a first entry point. Were this not the case, then the first entry set would be of little use.

Claim 3.26 Suppose that R is a suitable pre-image relative to some commanded velocity v_θ^* and some collection of goals $\{ G_\alpha \}$. Then $R \subseteq B_\theta(\{E(R)\})$.

Proof: Definitional. ∎

3.9.3. A Structure Equation

This section shows that every suitable pre-image is the subset of an almost simple pre-image. Thus the class of almost simple pre-images constitutes a collection of basic building blocks from which all other pre-images can be constructed as subsets. The almost simple pre-images are the natural analogues of the maximal backprojections.

Denote by $A(R)$ the desired, almost simple pre-image which contains R, R being some suitable pre-image. The relationship between backprojections and pre-images is contained in the following structure equation, proved by the claim below.

$$A(R) = \pi F_\theta(R) \bigcap B_\theta(\{E(R)\}) \tag{3.30}$$

Claim 3.27 Suppose that R is a suitable pre-image. Then R may be extended to an almost simple pre-image $A(R)$, given by equation (3.30).

Proof: By Claim 3.26, R is a subset of $A(R)$. Therefore, one need only show that $A(R)$ is almost simple relative to $E(R)$.

Applying the forward projection twice to a set gains nothing over applying it just once. It follows that

$$A(R) = \pi F_\theta(A(R)) \bigcap B_\theta(\{E(R)\}). \tag{3.31}$$

This shows that $A(R)$ satisfies condition (iv) of the definition for almost simple (Def. 3.23). Since a first entry point is either in a goal set, or is the limit of points in goal sets, $A(R)$ satisfies condition (iii) with respect to $E(R)$. Again, using claim 3.26, it follows that $E(R) \subseteq A(R)$, so A(R) satisfies condition (ii). It remains therefore only to show condition (i), namely that $A(R)$ is itself a suitable pre-image. To see this, note that any trajectory which starts in $A(R)$ must eventually enter a first entry point, by construction. Claim 3.25 then guarantees that the trajectory recognizably enters a goal some time later. ∎

3.9.4. Maximality of the Extension $A(R)$

This subsection shows that the extension $A(R)$ of R is maximal in the sense that repeating the extension gains nothing. While this result is not needed to show that almost simple pre-images form a class of basic building blocks in the construction of pre-images, it further underscores the naturality of this class.

Before the actual claim, there are a few lemmas that are needed in the proof of the claim. The first lemma says that the backprojection of a subset of a backprojection cannot leave the original backprojection.

Lemma 3.28 Let S be a collection of sets, and suppose that $A \subseteq B_\theta(S)$. Then $B_\theta(\{A\}) \subseteq B_\theta(S)$.

Proof: Definitional. ∎

The second lemma says that the forward projection of a forward projection is just the original forward projection. As a corollary to this lemma it follows that the forward projection of $A(R)$ is the same as the forward projection of R. This result was used in the proof of the structure equation.

Lemma 3.29 $F_\theta(\pi F_\theta(R)) = F_\theta(R)$.

Proof: Definitional. ∎

Corollary 3.30 $F_\theta(A(R)) = F_\theta(R)$.

Proof:

Certainly $R \subseteq A(R) \subseteq \pi F_\theta(R)$.

Therefore $F_\theta(R) \subseteq F_\theta(A(R)) \subseteq F_\theta(\pi F_\theta(R)) = F_\theta(R)$. ∎

The third lemma says that the first entry set of the extension $A(R)$ of R must be in the backprojection of the first entry set of R itself. Note, by the way, that $E(R) \subseteq E(A(R))$.

Lemma 3.31 $E(A(R)) \subseteq B_\theta(\{E(R)\})$.

Proof: Suppose $p \in E(A(R))$. Then there is a trajectory T beginning in $A(R)$ whose first entry point is p. There are two cases.

First, suppose that the trajectory T passes through a point $\hat{p} \in E(R)$ before or while passing through p. Then there is a second trajectory $\hat{T}$ beginning in R whose first entry point is $\hat{p}$. One can therefore construct a third trajectory which is simply $\hat{T}$ up to, but not including, $\hat{p}$, and which, beginning with $\hat{p}$, is the original trajectory T. The key to the third trajectory is that its velocity at $\hat{p}$ is the same as the velocity of T at $\hat{p}$. By construction this third trajectory has p as a first entry point. In other words, $p \in E(R)$, so $p \in B_\theta(\{E(R)\})$.

In the second case, the trajectory T does not pass through $E(R)$ before passing through its first entry point p. Suppose that $p \notin B_\theta(\{E(R)\})$. Then there is some trajectory $\tilde{T}$ which begins at p, but never passes through $E(R)$. Clearly, therefore, the concatenation of T, up to p, with $\tilde{T}$, as of p, never passes through $E(R)$. This contradicts the fact that the starting point of this trajectory is in $A(R)$, which is a subset of $B_\theta(\{E(R)\})$. ∎

Finally, the main claim of this subsection is almost immediate.

Claim 3.32 $A^2 = A$

That is, for any suitable pre-image R, $A(A(R)) = A(R)$.

Proof: Note that

$$\begin{aligned} A(R) &= \pi F_\theta(R) \bigcap B_\theta(\{E(R)\}) \\ A(A(R)) &= \pi F_\theta(A(R)) \bigcap B_\theta(\{E(A(R))\}). \end{aligned} \tag{3.32}$$

By Cor. 3.30, the first two terms in each of the two intersections are the same. By Lemmas 3.28 and 3.31, and the comments preceding the second of these lemmas, the second terms in each of the two intersections are the same. ∎

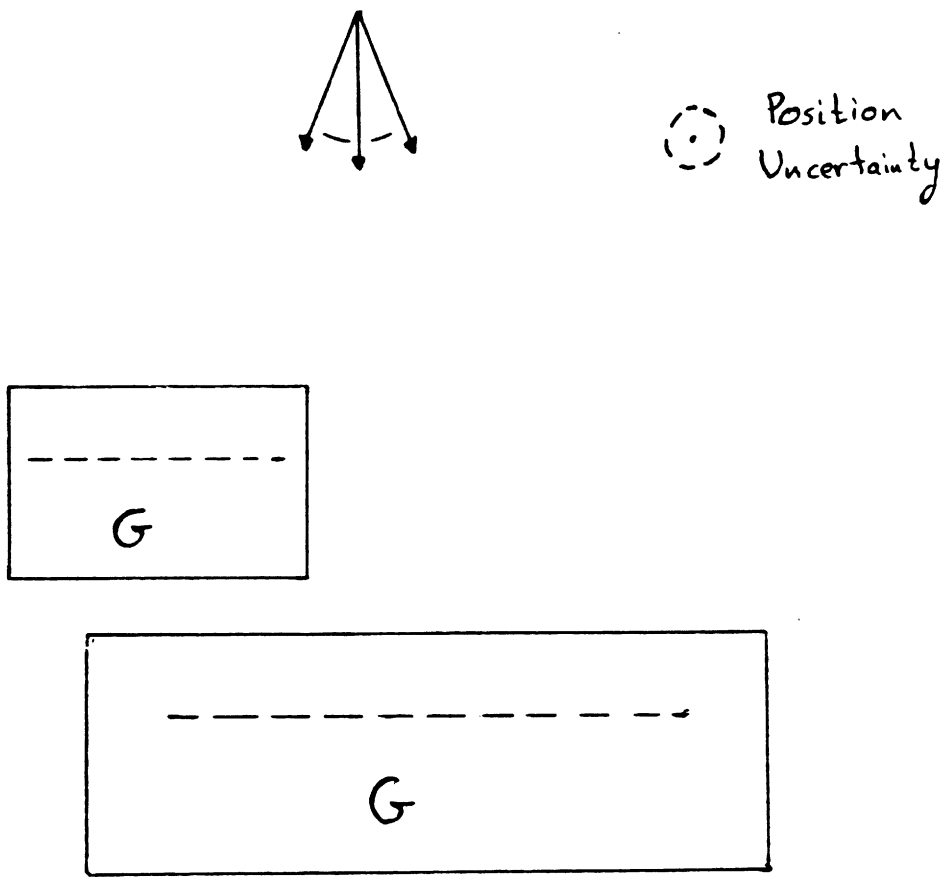


Figure 3.26. The dashed lines represent points that are unambiguously contained in the goal regions, relative to their forward projection and the indicated uncertainties.

3.10. Selection of Goal Sets

The previous section established a relationship between backprojections and pre-images, based on first entry sets. The task now is to use this equation in order to approximate pre-images by backprojections. This section briefly attacks this task.

3.10.1. Review of the Grand Scheme

Chapter 2 described a planning scheme (Lozano-Pérez, Mason, and Taylor (1983)) which used backchaining of pre-images to generate a sequence of motion commands guaranteed to achieve some desired task. The task was specified as a geometrical goal in configuration space. The planner formed all possible pre-images of this goal. Motions from these pre-images to the goal comprised the possible last motions in any sequence of motions ultimately decided on by the planner. All motion directions were considered. The planner then gathered all the pre-images thus determined into a new collection of goals. A second level of pre-images was formed. Motions from these pre-images constituted the possible next to last motions

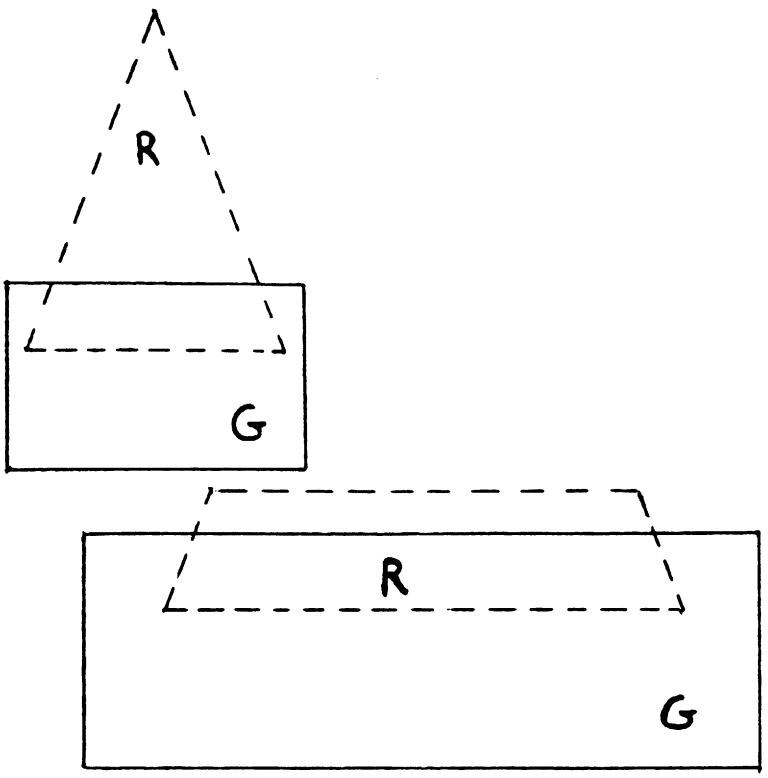


Figure 3.27. Partial backprojection of the dashed lines of Fig. 3.26 that avoids the ambiguity demonstrated in Fig. 3.22.

---

of any sequence of motions found by the planner. The planner then considered all second level pre-images as goals for another level of backchaining. This process continued until one of the levels generated a pre-image containing the initial configuration of the task.

Backprojections are a means of approximating pre-images. Chapter 2 also intuitively described an algorithm for computing backprojections in certain configuration spaces. Backprojections do not take explicit account of termination conditions. This implies that the regions generated by one level of backchaining may not be suitable goals for the next level of backchaining. The goals are certainly suitable goals for computing pre-images. However, since backprojection is being used as an approximation to computing pre-images, it may be necessary to modify the regions generated by one level of backchaining before passing them as goals to the next level. The reason for this is simply that backprojection from unmodified goals may result in trajectories that do not recognizably enter the goal regions.

The issue therefore is one of understanding the nature of good goal sets. By characterizing that nature it should be possible to mold goals into regions

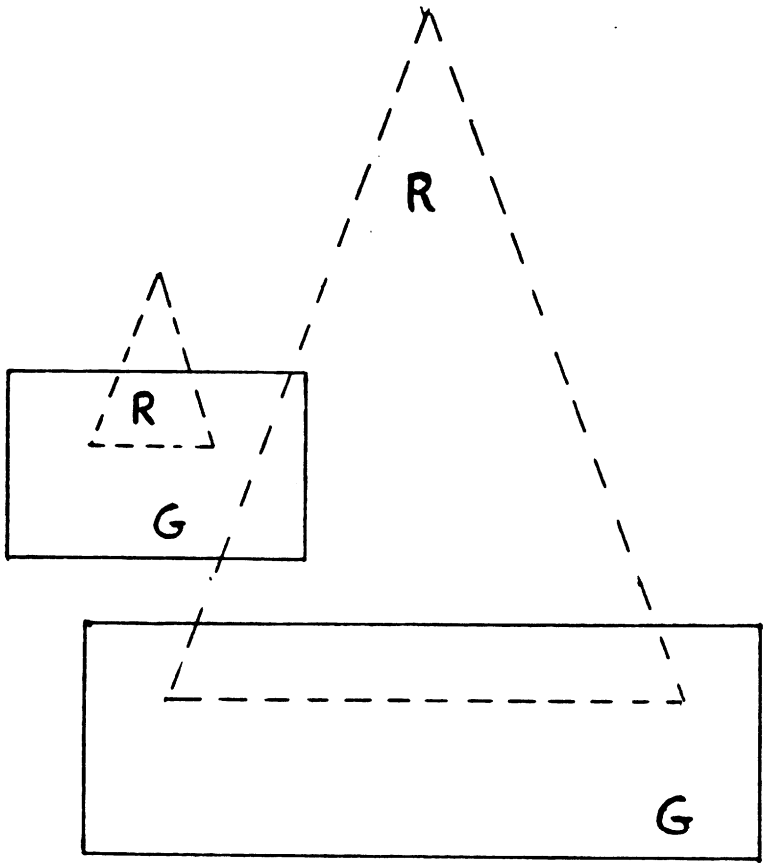


Figure 3.28. Backprojection from a smaller subset of the dashed goal lines of Fig. 3.26.

---

from which backprojection is guaranteed to yield motions that successfully and recognizably attain the desired goals.

3.10.2. The Burden on First Entry Sets and Forward Projections

The structure equation provides a relationship between backprojections and pre-images through first entry sets and forward projections. Thus the key to choosing good goal sets lies in forward projections and first entry sets.

The role of first entry sets is simple. Recall that every first entry point is itself a valid pre-image point. In other words, any motion that starts at a first entry point is guaranteed to terminate successfully. That is the reason that backprojecting from first entry points is guaranteed to yield a pre-image, since all points in the backprojection must pass through a first entry point, hence are guaranteed to terminate successfully.

It appears then, that any set which possesses properties similar to those of first entry sets is an appropriate set from which to backproject. This is not quite correct. The problem is that the decision whether a point is a good pre-image point

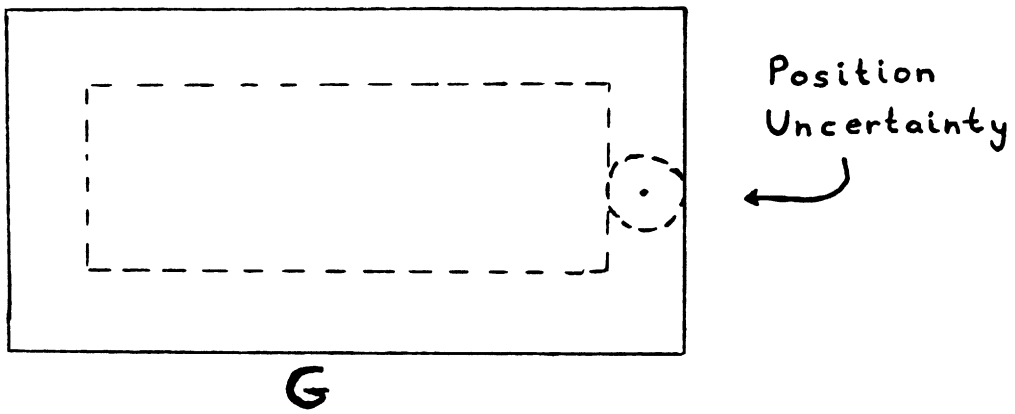


Figure 3.29. All points inside the dashed region are recognizably in the goal. The dashed region is so chosen that the possible sensor interpretations of all points lie within the larger goal region.

---

depends on whether the termination predicate can restrict the set of interpretations $F_\theta(R) \cap B_{ep}(p^*) \times B_{ev}(v^*)$ to lie within a goal. This set of interpretations depends on the forward projection $F_\theta(R)$. Although a set may be a good pre-image set, its backprojection may not be a pre-image. That is why the structure equation was formulated in terms of an intersection with $\pi F_\theta(R)$.

Therefore, in selecting good first entry sets, the planner has two parameters to play with, namely the set itself, and the forward projection used in the set of interpretations. Rather than backproject completely from the selected first entry set, the planner only backprojects within the selected forward projection.

The example of Fig. 3.26 should clarify these remarks. The example is a variant of that presented earlier in Fig. 3.21. There are two goal regions, as indicated. The commanded velocity is straight down. Both the velocity error cone and the position sensing uncertainty are shown. Velocity sensing does not play a role in recognizing entry into the goals. A very conservative modified goal collection is indicated by the dashed lines. Any point in the regions delineated by these lines is unambiguously contained in the goal sets, relative to their forward projection and the given position uncertainty. Thus these regions possess the properties shared by first entry sets.

However, not every backprojection of the two dashed-line regions is itself a suitable pre-image. In particular, the maximal backprojection is not a suitable pre-image, as was noted previously for Fig. 3.22. There are many ways to shrink the backprojections so that they become suitable pre-images. Two such solutions are shown in Fig. 3.27 and Fig. 3.28.

The trouble spot lies in the ambiguity introduced between points in the upper left goal and points in the backprojection of the lower right goal. If the goal points

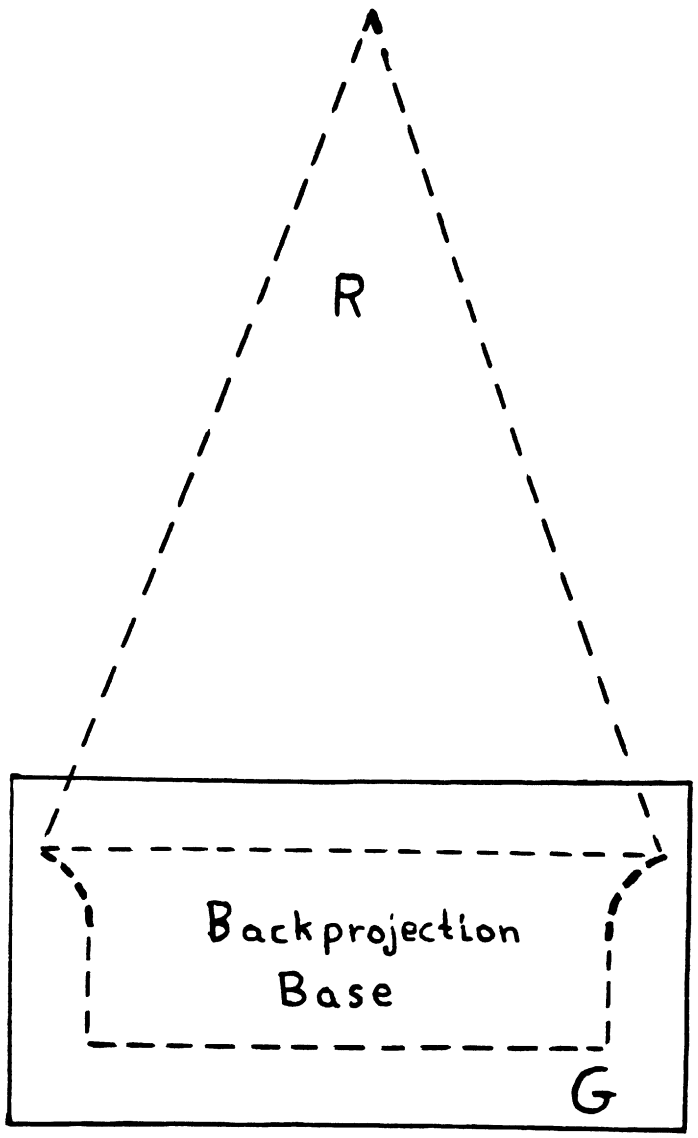


Figure 3.30. Knowing that all positions must lie within the forward projection of the cone, permits a larger region of immediately recognizable goal points.

---

are to be retained as suitable pre-image points, then the backprojection must be constrained. If, on the other hand, the backprojection points are to be retained, then the goal region must be constrained. Both solutions considered the interplay of forward projection interpretations and suitable goal points, removing the ambiguity present in the interplay.

As another example, consider the region of Fig. 3.29, under the same assumptions as for the previous example. Not constraining the set of interpretations, forces one to use the conservative goal region indicated by the dashed line as a base for backprojection. Knowing that the set of interpretations is contained in the

Figure 3.31. Pre-image R of a goal edge. R does not satisfy the condition $A(R) = R$. Trajectories from R to the goal must be contained in R in order that $A(R) = R$. In this example all trajectories emanating from R leave R before hitting the goal edge. See also Fig. 3.32.

forward projection of the cone of Fig. 3.30 allows one to use a larger goal set as a backprojection base. This phenomenon is, of course, just the effect of local history, as discussed previously and by Lozano-Pérez, Mason, and Taylor (1983).

The current subsection may be summarized by the following claim.

Claim 3.33 Let X be a subset of position space and let F be a subset of phase space. Assume the commanded velocity v_θ^* is given. Suppose that every point in X satisfies the termination condition for $S_{history}$ with respect to a set of interpretations given by $F \cap B_{ep}(p^*) \times B_{ev}(v^*)$ (see Def. 3.21). Now let R be any subset of $B_\theta(X)$ such that $F_\theta(R) \subseteq F$. Then R is a suitable pre-image.

Proof: Definitional. ∎

Claim 3.33 provides a partial converse to the structure equation. In constructing goal sets, the planner must choose the sets X and F so as to satisfy the conditions of the claim. Suitable pre-images are then easily generated by subsetting. Note, of course, that any reasonable set F should satisfy $F_\theta(F) \subseteq F$.

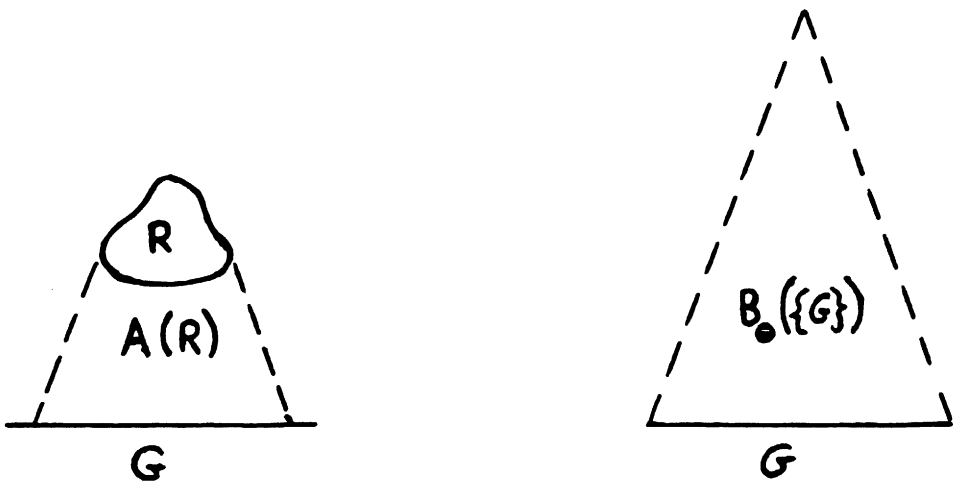


Figure 3.32. The region on the left is the extension $A(R)$ of the pre-image R from Fig. 3.31. The region on the right is the backprojection of the goal edge. Notice that $A(R)$ is a backprojection of a subset of the goal edge, intersected with the forward projection of the set R. Also notice that $A(R)$ is a subset of the backprojection $B_\theta(\{G\})$. This backprojection is a basic pre-image building block.

3.10.3. Moral

The moral to the story is simple. All suitable pre-images are subsets of pre-images R that satisfy the maximality condition $A(R) = R$. It may be difficult to generate all sets that satisfy this maximality condition. However, any such pre-image is the backprojection of a first entry set, restricted to lie inside a forward projection. Fortunately, given any set which is a suitable pre-image relative to a forward projection, it is possible to construct another pre-image by forming the backprojection of the given set, while restricting it to lie inside the forward projection. Thus the basic pre-image building blocks are pre-images which are restricted backprojections of fairly simple sets. These sets are subsets of the closure of the union of all goals.

Fig. 3.31 depicts a pre-image R of a goal edge. Entry into the goal edge is detected by a collision with the edge. The pre-image R is not almost simple, and does not satisfy the relationship $A(R) = R$. This is because there are trajectories that leave R before hitting the goal. Fig. 3.32 shows the extension $A(R)$ of R, as well as the backprojection of the goal edge. The backprojection is itself an almost simple pre-image, and forms a basic building block for constructing pre-images. In particular, the set $A(R)$ is a subset of this backprojection. Fig. 3.32 also indicates that $A(R)$ may be formed by intersecting the forward projection of R with the backprojection of a subset of the goal edge.

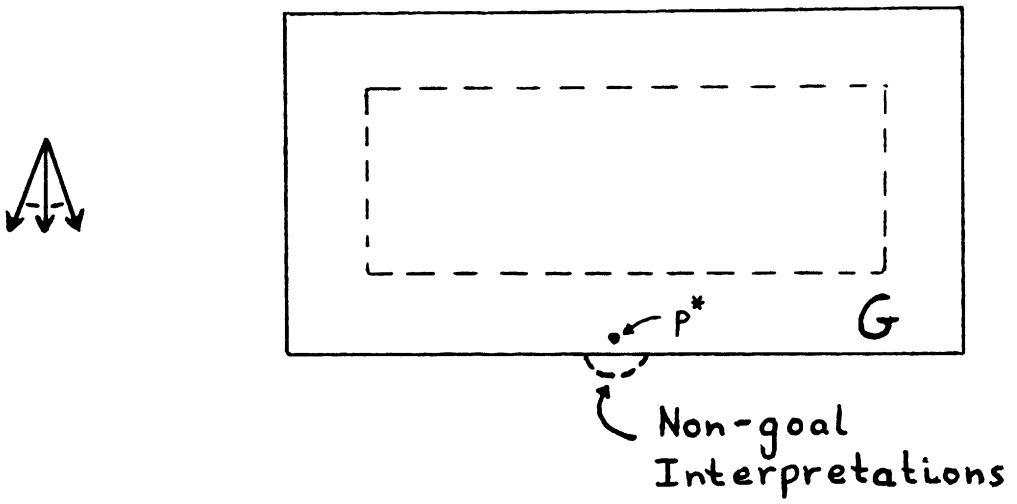


Figure 3.33. The sensor value p^* has interpretations outside of the goal. A forward projection that includes actual points consistent with p^* must avoid this circular region.

3.10.4. Compatible Goals and Interpretations

The previous subsections demonstrated the relationship between the first entry set of a pre-image, and the set of interpretations, as determined by the forward projection. The difficulty with this relationship lies in the mutual dependency of first entry sets and forward projections. Whether a set is a suitable first entry set depends on the forward projection, but the forward projection depends on the set. Dissecting this recursive dependency is, of course, just the old problem of solving the pre-image equation $R = P_R(\{ G_\alpha \})$. The structure equation and its partial converse have changed this problem from listing all suitable pre-images R, to finding all pairs of compatible goal subsets and forward projections.

3.10.4.1. Constraints on the Forward Projection

The task then is to remove the dependency between the goal subsets and the forward projections. This may be done fairly easily. Suppose $p \in \bigcup_\alpha G_\alpha$, and assume that it should be possible to recognize entry into some goal at p. Specifically, suppose that any trajectory T with velocity v at point p can be recognized to have entered a goal relative to some forward projection F. The objective is to derive the constraints imposed on the forward projection by this assumption.

In order for (p, v) to be a recognizable point, it must be the case that the condition defining $S_{history}$ be true, relative to the assumed forward projection F. In other words, given any sensor value (p^*, v^*) consistent with (p, v) it must be true that there is some goal cylinder $H \in \{ H_\alpha \}$ such that $F \cap B_{ep}(p^*) \times B_{ev}(v^*) \subseteq H$. In general there may be several such satisfying cylinders. Given a point and velocity (p, v), a consistent sensor reading (p^*, v^*) and a desired goal H, the constraint on the forward projection F is[7]

[7]The complement of a set A in its ambient space is denoted by A^c.

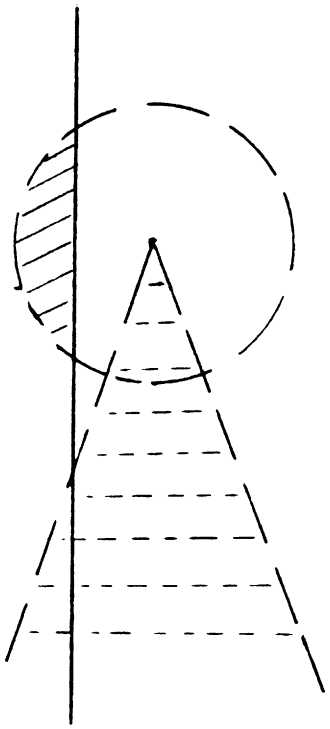


Figure 3.34. The forward projection of the point does not overlap the set of position interpretations that lie outside of the goal. The point can be moved closer to the goal boundary.

$$F \bigcap C(p, v, p^*, v^*, H) = \emptyset, \tag{3.33}$$

where

$$C(p, v, p^*, v^*, H) = H^c \bigcap B_{ep}(p^*) \times B_{ev}(v^*). \tag{3.34}$$

From this it follows that a useful constraint on the shape of F is

$$F \subseteq C'(p, v, p^*, v^*, H), \tag{3.35}$$

where

$$C'(p, v, p^*, v^*, H) = H \bigcup (B_{cp}(p^*) \times B_{ev}(v^*))^c. \tag{3.36}$$

In other words, the forward projection lies either inside the goal, or outside the set of possible interpretations.

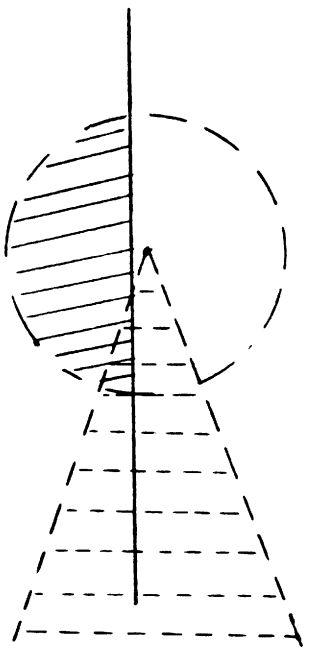


Figure 3.35. The forward projection of the point overlaps the set of position interpretations that lie outside of the goal. The point is too close to the goal boundary.

Therefore, suppose that one is constructing the set X, so that every point in X is recognizably in the collection of goals. For each point in X, one selects one or more desired velocities, all possible sensor values, and a host of goal cylinders H. The result is a collection of constraints of the form of Eq. (3.36). Taken together this collection yields a complete specification of the conditions on the forward projection F as

$$F_\theta(F) \subseteq F \subseteq \bigcap_{\{(p,v,p^*,v^*,H)|p\in X\}} C'(p, v, p^*, v^*, H). \tag{3.37}$$

The first part of the constraint merely insures that the forward projection is being considered in full. The second part of the constraint states that the forward projection should not contain any points that both are possible interpretations and lie outside of the desired goals.

3.10.4.2. A Practical Note

The constraint given by Eq. (3.37) is, in general, the intersection of an infinite number of sets. Even for finite sets X, the number of velocities v, and the possible sensor readings (p^*, v^*), are infinite. Additionally, the collection $\{ G_\alpha \}$, even before

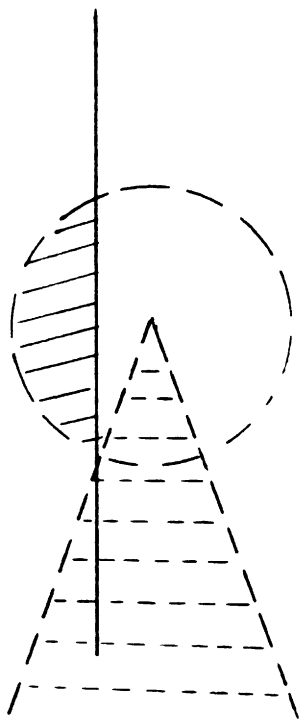


Figure 3.36. The point is horizontally as close to the goal boundary as possible.

---

being closed under subsets, may contain an infinite number of goals. Generally, the number of regions generated at any level of backchaining will be infinite.

A practical planner must deal with finite representations of sets. The collection of goal sets should be taken to be finite. This involves some type of reduction operation, such as the one described in the triangle example of Sec. 2.3.7. The gist of that example was to replace an infinite collection of sets with another infinite collection of mutated sets, then take the union over the collection of mutated sets. The mutated sets were so chosen as to be distinguishable relative to each other. In other words, knowing that a point was in the union, it was always possible to identify at least one of the mutated sets containing the point. The reduction of the termination predicate with no sense of time to the termination predicate without history was another application of the method of distinguishable sets.

Also, rather than deal with individual points in the set X, a practical planner must deal with a finite collection of regions that are subsets of X. While it cannot consider all subsets of X, it can parameterize a few selected types of subsets. Similarly, instead of being recognizable for single velocities, points should be recognizable for a range of velocities. Finally, instead of considering all sensor values individually, the range of sensor values should be split into a finite number of subranges.

For example, in finite polyhedral environments, there exist but a finite number of constraints. Thus it is possible to split space into a finite number of regions, over each of which recognizability is constant. For finite algebraic environments, it is possible to split space into a finite number of regions over which recognizability varies in a continuous fashion. Thus, the sets $\{C'(p, v, p^*, v^*, H)\}$ can be reduced either to a finite collection of constant sets, or to a finite collection of sets that vary piecewise smoothly over a finite number of regions.

Consider, as a simple example, the old problem of recognizing entry into a region based solely on position information, as in Fig. 3.29. The position uncertainty is as indicated. For points within the dotted line region, the sets $C'(p, v, p^*, v^*, H)$ are simply all of Position-space $\times$ Velocity-space. This is because the possible sensor interpretations all lie inside the goal, so that the union of the goal and the complement of the interpretations is all of phase space. As one moves towards the boundaries of the goal, the sets $C'(p, v, p^*, v^*, H)$ become pockmarked with holes. For example, for the sensor point in Fig. 3.33, the position component of the constraint region is all of the plane except for the circular portion indicated. The size and shape of this circular portion varies continuously as a point moves from the dotted lines towards the goal boundary.

The circular holes below the goal indicate clearly that points below the dotted region cannot be pre-image points. This is because any forward projection of these points must pass through the forbidden holes, thereby violating the constraint of Eq. (3.37). Points above the dotted line region may be pre-image points, but only if the forward projection does not reach above the goal. This therefore constrains any backprojection, which insists that points above the dotted line be recognizable points, to lie below the top of the goal.

Note that one can increase the dotted line region to a larger region of recognizable points in the horizontal directions. Consider moving the top left corner of the dotted line region to the left. Construct the constraint set that is the intersection of the $C'(p, v, p^*, v^*, H)$, taken over all values p^* consistent with the point. Initially, the hole in the constraint set is guaranteed to be disjoint from the forward projection, as shown in Fig. 3.34. As one continues to move the point left, the hole eventually does overlap the forward projection, as shown in Fig. 3.35. The happy medium is to be found when the left part of the forward projection just intersects the boundary of the hole, as in Fig. 3.36. Repeating this process on the right side of the dotted line region, yields a larger set of recognizable points, as in Fig. 3.30. Any backprojection from this set is guaranteed to be a pre-image.

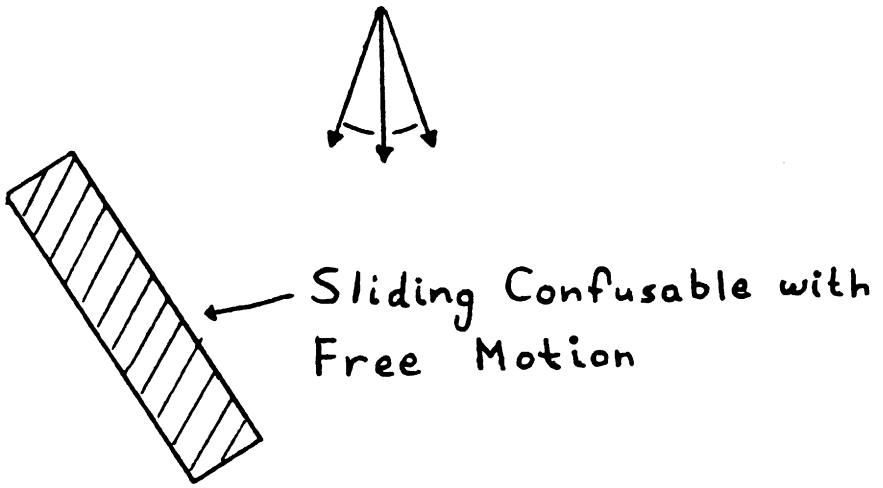


Figure 3.37. Obstacle and commanded velocity uncertainty cone. The top right edge is of primary interest in the following figures.

3.10.4.3. Constraints on the Goal Sets

The previous development indicates a means of generating recognizable goal points. Limits of these may be taken to form the more general form of first entry points. There remains a final problem to consider. While Eq. (3.37) places constraints on the relationship between forward projections and goal points, this is not sufficient to ensure that the set of goal points form a pre-image. Certainly each of the points is recognizable for some velocity, but it need not be the case that trajectories emanating from these points are always guaranteed to terminate successfully. This issue was first raised in Sec. 3.8.2.3. Further constraints are needed that deal with the relationship between goal points, in order to satisfy the hypotheses of Claim 3.33.

Suppose X is a set consisting of recognizable goal points and limits of recognizable points, relative to some forward projection F. Recall that goal points may be recognizably in a goal for some but not all effective commanded velocities in the velocity uncertainty cone. The current objective is to find conditions on the points of X that ensure that these points are also pre-image points. Given these conditions, X will behave much like a first entry set. For simplicity, assume that only points in X are permitted as recognizable termination points of trajectories beginning in X. In other words, using the terminology of Sec. 3.8.2.3, X contains all of its successful termination points.

Consider a variant of the forward projection, call it $F_{bad,\theta}(X)$, which consists of all points and velocities that are reachable from the set X using only unrecognizable velocities. In other words, every point $p \in \pi F_{bad,\theta}(X)$ is reachable from a point

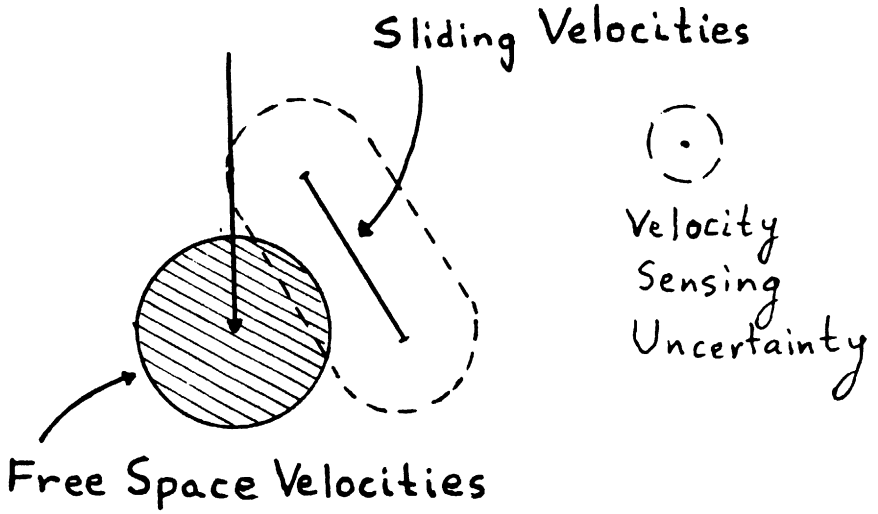


Figure 3.38. The circle comprises the range of free space velocities. The tilted line comprises the range of sliding velocities possible on the edge of Fig. 3.37. The dashed line region represents the possible set of interpretations of the sliding velocities.

---

in X along a trajectory that does not recognizably enter a goal before or at p. Now consider points on the boundary of X that are reachable from X using only unrecognizable trajectories, that is, consider the set $\partial X \cap \pi F_{bad,\theta}(X)$.[8] Consider a trajectory that begins at a point p of $\partial X \cap \pi F_{bad,\theta}(X)$. If the trajectory is not recognizable as being in a goal at p, then the trajectory must eventually re-enter X. Furthermore, unless it leaves X forever after a finite time, the trajectory must repeatedly be recognizable as being in a goal. Formally,

Claim 3.34 Let X be a set consisting of recognizable goal points and limits of recognizable goal points (relative to some fixed forward projection F). Assume X contains all of its successful termination points.

Then X is a suitable pre-image (relative to F) if and only if the following two conditions hold:

(i) If $p \in \partial X \cap \pi F_{bad,\theta}(X)$ and T is a trajectory beginning at p, then either T sticks at p and is recognizably in a goal for all effective commanded velocities that cause sticking, or T eventually enters a point in $X - \{p\}$.

(ii) If T is a trajectory beginning in X, then either T forever leaves the set X after a finite amount of time, or the set of times at which T is recognizably in a goal is unbounded.

Proof: **I.** Suppose that X is a suitable pre-image. Let $p \in \partial X \cap \pi F_{bad,\theta}(X)$, and let T be a trajectory with initial point p. If T sticks at p, then p is recognizably

[8] The boundary of a set is just the intersection of its closure and the closure of its complement, that is, $\partial X = \overline{X} \cap \overline{X^c}$.

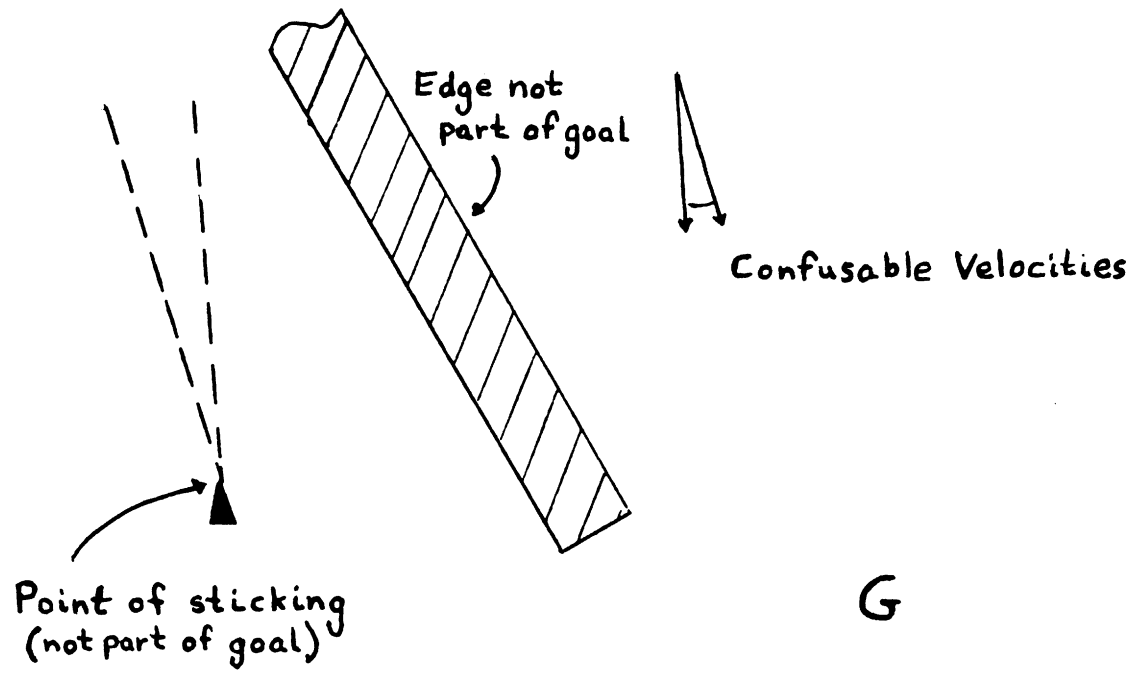


Figure 3.39. **Assume that the top vertex of the triangular obstacle lies within the position uncertainty of the obstacle edge. The vertex and the edge are not parts of the goal. The cone corresponds to velocities for which the termination predicate cannot recognize that a point is not on the edge. For velocities in this range, a point can move to and stick on the vertex without being recognized as not being on the edge. In order to avoid this state, any pre-image must avoid the region defined by the cone.**

---

in a goal for all commanded velocities that cause sticking. Otherwise there would be a trajectory beginning in X which terminated at p without ever recognizably entering a goal.

Suppose that T does not stick at p. Since $p \in \pi F_{bad,\theta}(X)$ we can assume without loss of generality that T is not recognizably in a goal whenever T is at p. Consequently, T must return to X, as X contains all of its successful termination points. If T returns to p without first entering a point in $X - \{p\}$, then there is a trajectory beginning in X that eventually cycles from p to p, and that is never recognizably in a goal. Since X is a suitable pre-image this can't happen, so T must enter a point in $X - \{p\}$. This establishes condition (i).

Now let T be any trajectory starting in X. Suppose there is some time after which T is never recognizably in a goal. Then T remains outside the set X, for otherwise there is a trajectory beginning in X which never recognizably enters a goal. This establishes condition (ii).

II. Suppose that the two conditions are satisfied. Let $p_x \in X$, but suppose that there is a trajectory beginning at p_x which never recognizably enters a a goal. By condition (ii), there is some time after which the trajectory never re-enters the set X. Thus the trajectory must eventually pass through a boundary point p of X, after which it never returns to X. Clearly $p \in \partial X \cap \pi F_{bad,\theta}(X)$, which violates condition (i). ∎

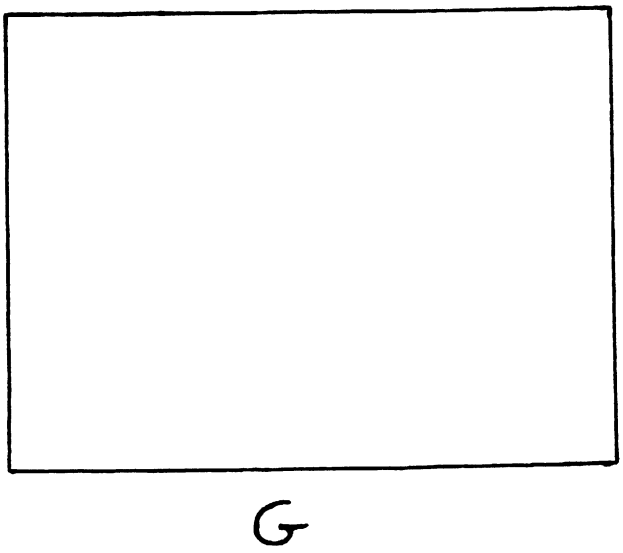


Figure 3.40. The nominal goal is the rectangle. The velocity uncertainty cone is the cone shown. The objective is to modify the goal so that the backprojection of the modified goal is a pre-image.

As an immediate corollary, note that any point in X from which there exists a trajectory that never re-enters X, must recognizably be in a goal for all possible effective commanded velocities, if X is to be a suitable pre-image. Thus these boundary points of X form a simple foundation for X. Any trajectory that passes through one of these points is recognizably in a goal, independent of its velocity. Furthermore, all trajectories that are otherwise unrecognizable, must eventually pass through one of the special, fully recognizable, boundary points.

3.10.4.4. Another Practical Note

First, a brief intuitive comment on how velocities influence the recognizability

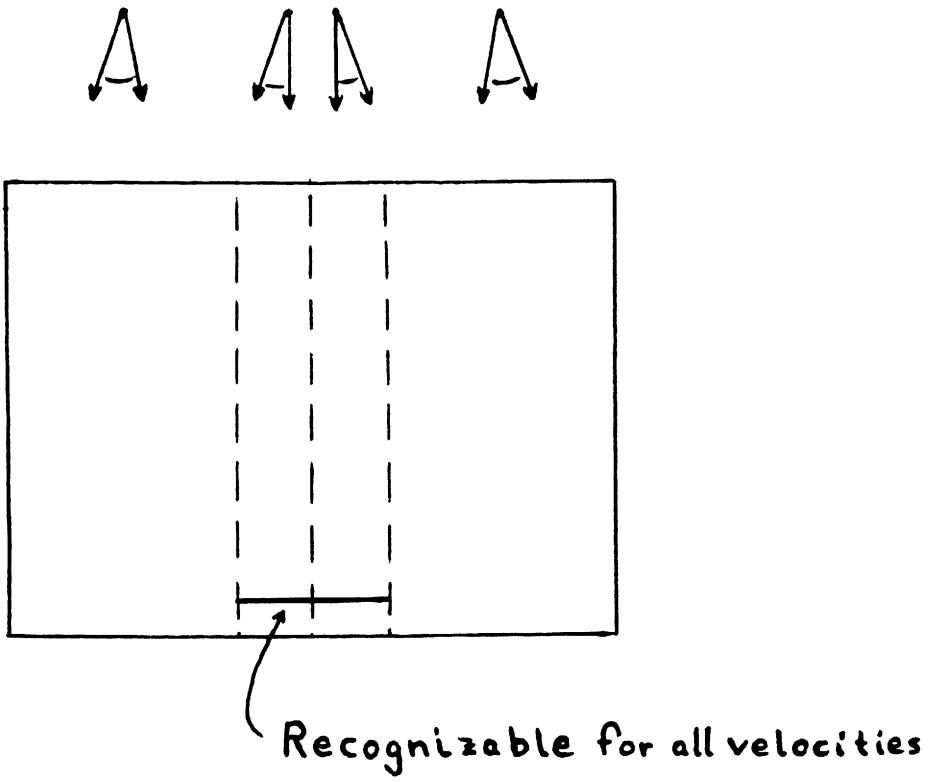


Figure 3.41. The rectangle is split into a line segment and four regions over which the range of separating velocities is constant. The line segment in the lower middle of the rectangle is assumed to be recognizable as being in the goal for all velocities in the velocity uncertainty cone. The ranges of velocities for which each of the regions is recognizably in the goal are shown above the regions.

---

of termination conditions. Consider Fig. 3.37. Assume there is no friction. The commanded velocity is straight down, with the indicated error cone. This range of commanded velocities causes sliding on the specified edge, with an uncertainty range in the tangential component. The relation between free space and sliding velocities is shown in Fig. 3.38.

Suppose that the termination predicate wishes to decide that a point is in free space, rather than on the edge, based on velocity measurement alone. Such is the decision faced by a termination predicate whenever a point lies within the position uncertainty of the edge. The velocity sensing uncertainty is also shown in Fig. 3.38. The problem is identical to position sensing with uncertainty. The velocity component of the forward projection is simply the union of the possible free or sliding velocities. In order to decide that a particular free space velocity is actually a free space velocity it must be the case that twice the error ball about that velocity does not overlap the range of sliding velocities. This is possible for

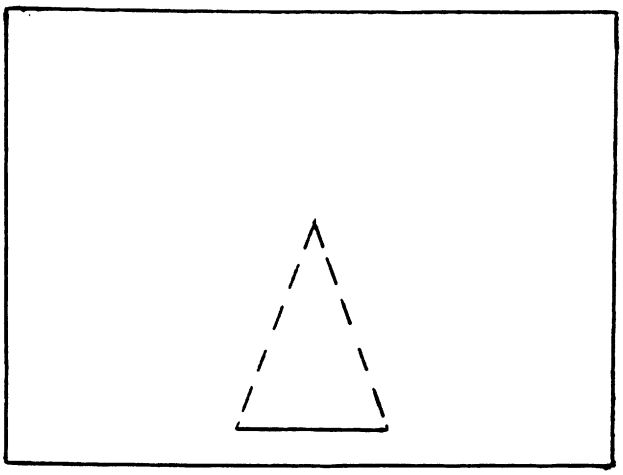


Figure 3.42. Backprojection of the line segment.

some, but not all of the free space velocities, as shown in Fig. 3.38.

Thus, if a point is close to the edge, then it is recognizably away from the edge only for a subset of the free space velocities. Of course, if it is far enough away, then it is impossible to confuse the edge with free space.

Now, suppose that one can partition a set X as described in Claim 3.34 into a finite number of regions over which velocity separability varies continuously. This is possible, for instance, if the surfaces and goals are described by a finite number of algebraic constraints. Let this decomposition be given by the collection $\{X_1, \ldots, X_n\}$. For expositional simplicity suppose that the velocity ranges which permit the termination predicate to recognize entry into a goal are constant over the X_i. Let the separating velocity range for X_i be V_i.

In the previous example, the edge and free space are uniquely separable for all effective commanded velocities in one region, while they are separable only for a proper subset of the effective commanded velocities in another region. Thus free space can be divided into two regions over which the separating velocities vary continuously. In fact, they are constant over each of these regions.

The essence of Claim 3.34 says that any trajectory beginning at a point of X must either encounter some other point of X, or be recognizably in a goal independent of velocity at the beginning of the motion. Furthermore, if the trajectory is effectively commanded by a velocity that lies outside the separating

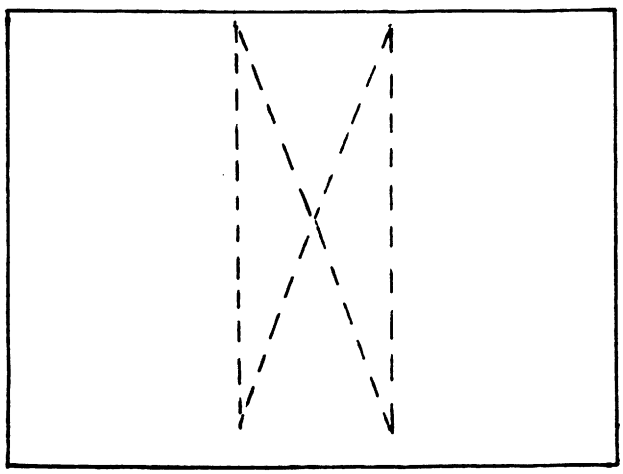


Figure 3.43. The two triangular regions are valid goal regions from which to backproject. This is because any trajectory beginning in these regions is either recognizably in the goal or eventually enters the backprojection of Fig. 3.42.

---

range V_i of X_i, then the trajectory must eventually enter a subregion of X other than X_i.

The corollary to the claim says that any point of X, from which there is a trajectory that never re-enters X, must have a separating velocity range that consists of all possible effective commanded velocities. Thus the associated subregion has a separating velocity range which is the entire commanded uncertainty ball.

As an extension of the edge identification example, consider Fig. 3.39. Added to the region near the edge is a vertex on which sticking can occur. The termination predicate cannot distinguish the vertex from the edge, based on pure position information. It can therefore distinguish the vertex from the edge only for a subset of the possible commanded velocities, as discussed earlier. Consider backprojecting from the offending vertex, using the velocity range that does not separate the vertex from the edge. The resulting cone is also shown in Fig. 3.39. Now let the goal be the entire space, except for the edge and the vertex. Let R be the entire space, except for the vertex and its attached cone. Then R is a pre-image. To see this, note that any velocity which cannot separate a point of the goal from the non-goal edge, must cause the point to eventually move far away from the edge. No point can move and get stuck on the non-goal vertex, without first having a velocity that recognizably separates the point from the edge. In short, if a point is recognizably in the goal for some proper subset of the commanded velocities,

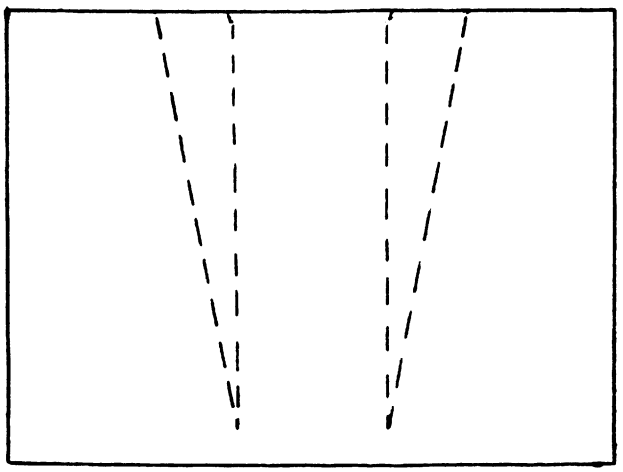


Figure 3.44. The triangular regions are also valid goal regions from which to backproject. This is because any trajectory beginning in these regions is either recognizably in the goal or eventually enters one of the triangular regions of Fig. 3.43.

---

then the non-recognizable velocities must move the point into a region that is recognizably in the goal independent of velocity.

3.10.4.5. An Example Illustrating Separating Velocities

Consider the example of Figs. 3.40 - 3.45, which demonstrates the method by which a planner might generate a goal set from which to backproject. Fig. 3.40 displays the nominal goal and the commanded velocity uncertainty cone. The goal is the solid rectangle. Backprojection from this rectangle will not generate a pre-image, since not all points in the rectangle can actually be recognized as being in the goal for all velocities in the uncertainty cone.

Fig. 3.41 indicates the velocity ranges for which different subregions of the goal are recognizably in the goal. The example does not necessarily correspond to any realistic situation. The point is merely to demonstrate goal construction. Assume that the short line segment at the bottom of the rectangle is recognizably in the goal for all velocities in the uncertainty cone. The rest of the rectangle is divided into four regions. The recognizable velocities for each of these regions are indicated above the regions. Basically, the middle two regions each are recognizable for velocities in half of the velocity uncertainty cone, while each of the outer two regions is recognizable for three-fourths of the velocity uncertainty cone. For simplicity, ignore the recognizability of points on the boundaries between the four regions.

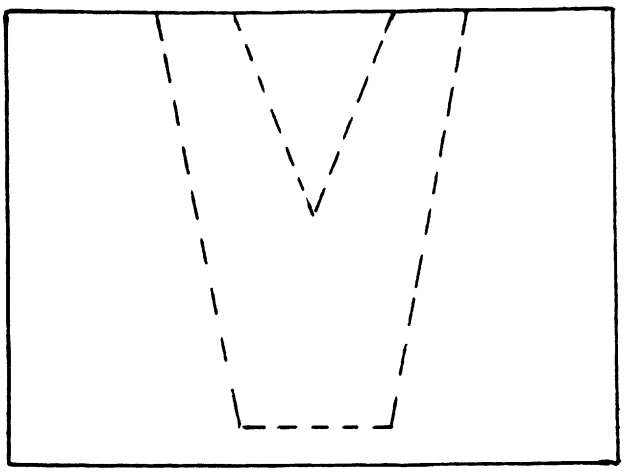


Figure 3.45. The final modified goal region which serves as a base for backprojection.

The short line segment forms the foundation of the goal. The planner should backproject from the foundation to determine all points which are guaranteed to fall onto the foundation. This backprojection is shown in Fig. 3.42. The planner can then extend the goal region by adding all regions in the nominal goal for which unrecognizable trajectories are guaranteed to fall onto the foundation.

A first pass adds to the goal the two triangular regions shown in Fig. 3.43. Any trajectory that starts in either of these regions either uses velocities that allow the termination predicate to recognize presence in the nominal goal, or uses velocities by which the trajectory enters the backprojection of the foundation. Thus these triangular regions are valid goal regions from which to backproject.

A second pass adds the two regions of Fig. 3.44. Any trajectory starting in these regions is either recognizably in the goal, or must move to the triangular regions of Fig. 3.43. Thus these regions are also valid goal regions from which to backproject. The final goal region from which the planner should backproject is shown in Fig. 3.45.

These examples show that a planner can, at least in a polyhedral environment, verify the intra-goal constraints using simple variations on backprojection and forward projection. More importantly, the planner can construct suitable first entry sets, by first constructing a foundation of goal points that are recognizable for all effective commanded velocities, then erecting higher levels of points which are guaranteed to fall onto the foundation for effective commanded velocities that are

non-separating.

3.11. Conclusion

This chapter has shown that a special subclass of pre-images may be described by backprojections. This class of pre-images is formed relative to a termination predicate that may only consult current sensor values and forward projections. The first entry set of a pre-image is defined as the set of points at which trajectories originating in the pre-image first recognizably enter a goal. The structural equation relating pre-images and backprojections states that the backprojection of a pre-image's first entry set, when intersected with the pre-image's forward projection, is itself a pre-image. The original pre-image is contained in the backprojection of its first entry set. Thus, in computing pre-images, it is sufficient to backproject from sets with properties analogous to those of first entry sets. All pre-images may be obtained by taking subsets of these backprojections.

In pursuit of base sets similar to first entry sets, this chapter explored the relationship between forward projections and goal recognizability. This relationship provides a method for deciding whether a particular point can be included in the base from which to backproject. Specifically, at any point, a trajectory is recognizably in a goal exactly when the sensor interpretations of the trajectory's position and velocity lie inside the union of the the goal and the complement of the forward projection. This observation constrains the size of the forward projection relative to the size of the goal set. Whenever a given point's contribution to the forward projection overlaps another point's set of non-goal interpretations, conflict arises. One of the two points must be removed from the goal set in order to guarantee recognizability.

Continuing the pursuit, the chapter also examined the constraints imposed on the internal structure of goal sets. Of interest was the goal recognizability of points for different subsets of the commanded velocity uncertainty range. At some points, entry into a goal may be recognizable only for trajectories with particular velocities. Trajectories which pass through these points with different velocities cannot be recognized as being in the goal. However, in order for the goal set to itself be a valid pre-image, these trajectories must eventually pass through a point with a velocity for which they are recognizable as being in the goal. This suggests that the goal set must contain some points for which the termination predicate can decide entry into the goal independent of the velocity with which a trajectory passes through the points. All trajectories which do not otherwise recognizably enter a goal, must pass through these fully recognizable points.

In constructing goal sets, a planner should therefore first consider points which are recognizable as being in the goal for all possible effective commanded velocities. These points form a foundation for the goal. The goal may be extended to other points which are only recognizable as being in the goal for some velocities, by backprojecting from the foundation. Specifically, the region of the goal in which points are recognizably in the goal only for some velocities must border the backprojection of the foundation in a constrained fashion. All trajectories whose velocities prevent the termination predicate from recognizing entry into the goal throughout this region must eventually enter the backprojection of the foundation.

These results may be used to compute pre-images using backprojections. Given a desired goal, a planner should modify the goal so as to satisfy the forward projection and structural constraints. All possible modifications that satisfy these constraints may be used as bases from which to backproject. The resulting backprojections form pre-images relative to the termination predicate with no sense of time. Any motion beginning in these backprojection regions is guaranteed to reach the original goal in a manner recognizable by the termination predicate.

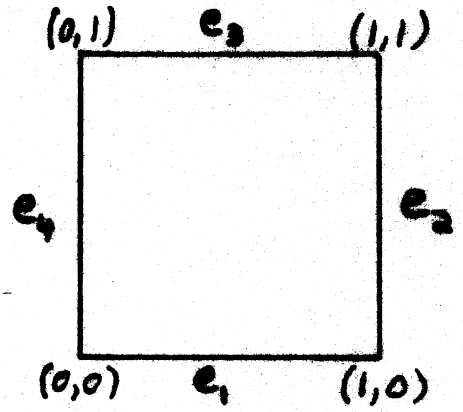


Figure 3.46. A square.

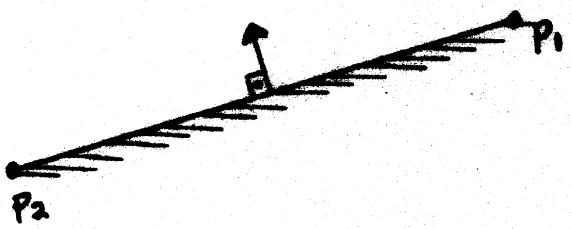


Figure 3.47. An edge is represented by the directed pair $[p_1, p_2]$. The interior of the edge is on one's left, as one traverses the edge from p_1 to p_2.

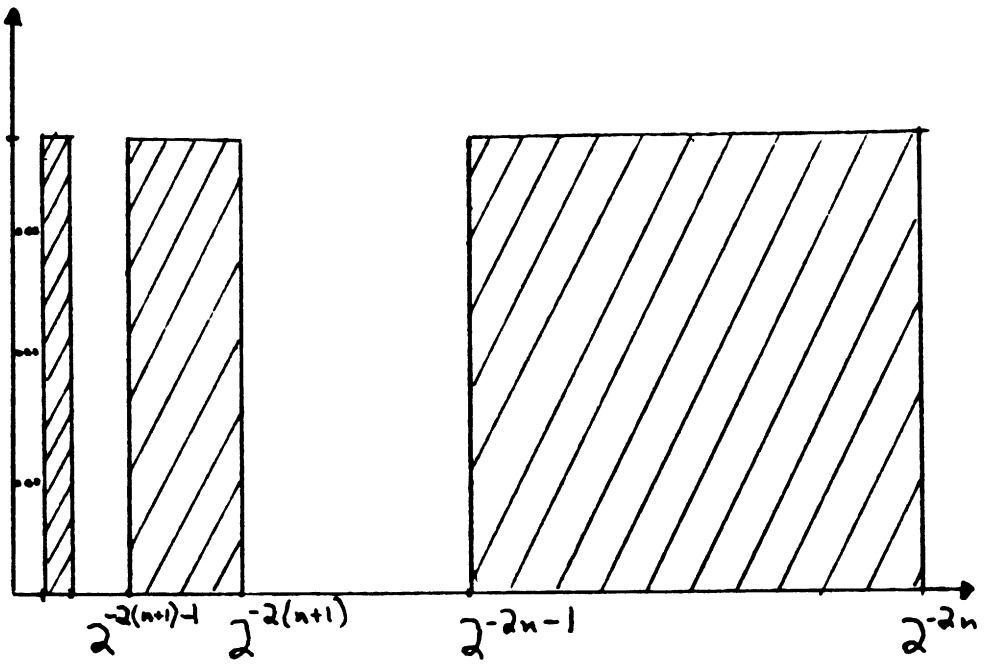


Figure 3.48. A countable collection of rectangles. The rectangles shrink in width as they approach the y-axis.

3.12. Appendix to Chapter 3 —— Computability

This section shows that the pre-image question, in full generality, is uncomputable. This result is not at all surprising, as general goal sets can be quite complex. In particular, there exist an uncountable number of sets, but only a countable number of solvable problems. Additionally, the pre-image question is a second order question. The arguments to the question are themselves complicated recursive sets. Thus, non-computability is expected.

This section may be omitted at a first reading, and its result accepted on faith. The result is fairly clear. It is not explicitly used in the thesis, but serves primarily as motivation for exploring computable subclasses of pre-images.

Consider the problem of moving a point in the plane. The constraints in the environment are restricted to be expressible by recursive functions. For further simplicity, assume that the constraints in the environment form a countable collection of line segments, whose interior portions are pairwise disjoint. The line segments may intersect at their endpoints. However, only a finite number of segments should intersect at any point. The collection of line segments should be expressible by a recursive function. Specifically, there should be a recursive function f, mapping encodings of points in the plane to encodings of lists of edges. Given a point in the plane, f returns the list of line segments that intersect at that point. The list is empty if the point is in free space. An edge is oriented, so that it possesses an outward normal. No particular significance is attached to the edges

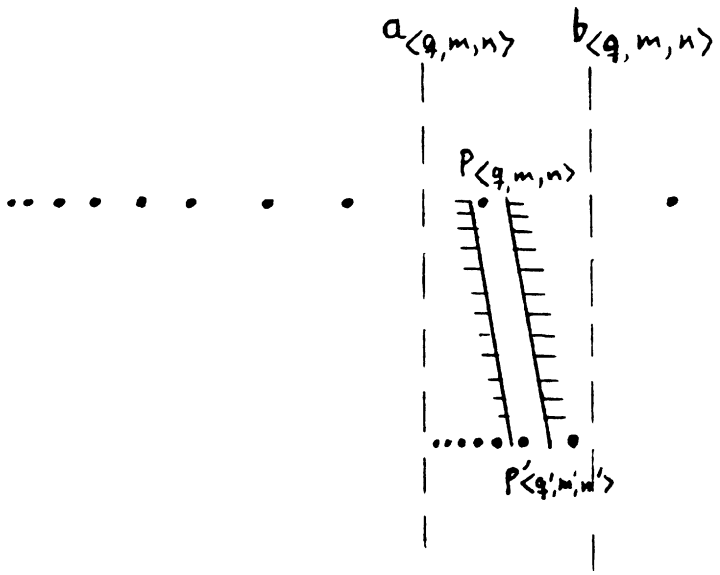


Figure 3.49. A channel between an encoding point at one level and an encoding point at the next level. The channel represents the transition of the Turing machine between the configurations encoded by the points.

---

or their normals, other than to define the direction of a reaction force felt when in contact with the edge. There is no friction.

The dynamics are assumed to be generalized damper dynamics with identity damping matrix. In other words, given a commanded velocity, a point will try to move in the commanded direction, unless obstructed by an edge. If obstructed, the point will move tangentially to the edge, using the tangential component of the commanded velocity (see Fig. 2.8). At edge intersection points, the range of reaction forces is taken to be the vector sum of the individual edge reaction forces. Sliding is possible only if the applied force lies outside of this cone. If so, the reaction force is computed by projecting normally onto the edge of the cone.

Given that the pre-image problem is unsolvable using these simplified constraints, it is certainly unsolvable using more complex constraints. The claim at the end of this section establishes unsolvability under the simplified constraints.

In order to gain some intuition about the representation of recursive constraints, consider the constraints defined by the square of Fig. 3.46. Assume that an edge is represented by an ordered pair of points $[p_1, p_2]$. The direction of the edge is so chosen that the edge's interior is on the left, that is, the edge's normal points towards the right (see Fig. 3.47).

For this example, the recursive function f which maps points in the plane to lists of edges is easily seen to be:

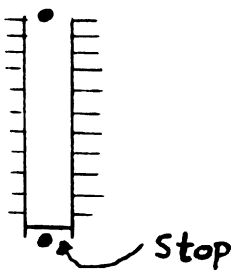


Figure 3.50. A horizontal stop in a channel corresponds to a halted configuration.

$$f(x,y) = \begin{cases} \{e_4, e_1\}, & \text{if } (x,y) = (0,0) \\ \{e_1, e_2\}, & \text{if } (x,y) = (1,0) \\ \{e_2, e_3\}, & \text{if } (x,y) = (1,1) \\ \{e_3, e_4\}, & \text{if } (x,y) = (0,1) \\ \{e_1\}, & \text{if } 0 < x < 1 \text{ and } y = 0 \\ \{e_2\}, & \text{if } x = 1 \text{ and } 0 < y < 1 \\ \{e_3\}, & \text{if } 0 < x < 1 \text{ and } y = 1 \\ \{e_4\}, & \text{if } x = 0 \text{ and } 0 < y < 1 \\ \emptyset, & \text{otherwise} \end{cases} \tag{3.38}$$

where

$$\begin{aligned} &e_1 \text{ is the edge } [(0,0),\ (1,0)], \\ &e_2 \text{ is the edge } [(1,0),\ (1,1)], \\ &e_3 \text{ is the edge } [(1,1),\ (0,1)], \\ &e_4 \text{ is the edge } [(0,1),\ (0,0)]. \end{aligned} \tag{3.39}$$

Now consider the example of Fig. 3.48. It consists of a countable collection of rectangles. All are of the same height, but their widths shrink as they approach the the y-axis. In order to define a recursive function expressing the rectangles' constraints, it is first necessary to define two helping functions. The first helping function is essentially the same as the function defined by Eqs. (3.38) and (3.39). The difference is that the following function also adjusts the lower left-hand vertex and the width of the rectangles.

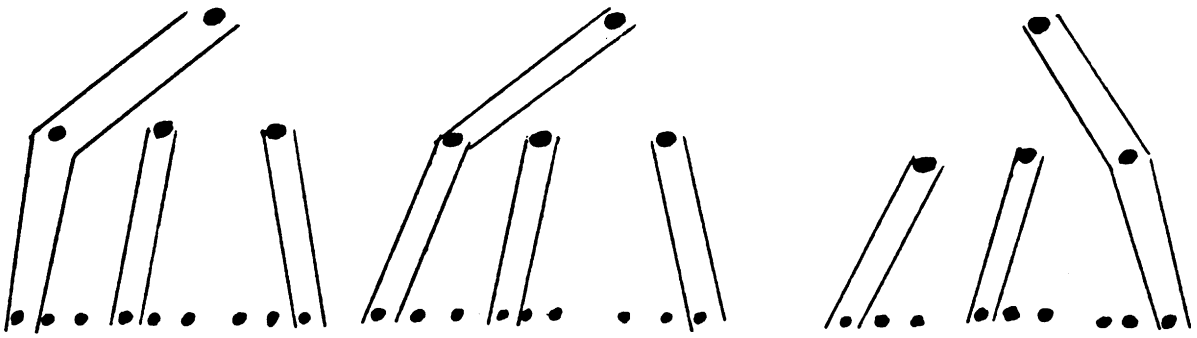


Figure 3.51. Three levels of encodings, with connecting channels.

$$g(x,y,n) = \begin{cases} \{e_4(n), e_1(n)\}, & \text{if } (x,y) = (2^{-2n-1}, 0) \\ \{e_1(n), e_2(n)\}, & \text{if } (x,y) = (2^{-2n}, 0) \\ \{e_2(n), e_3(n)\}, & \text{if } (x,y) = (2^{-2n}, 1) \\ \{e_3(n), e_4(n)\}, & \text{if } (x,y) = (2^{-2n-1}, 1) \\ \{e_1(n)\}, & \text{if } 2^{-2n-1} < x < 2^{-2n} \text{ and } y = 0 \\ \{e_2(n)\}, & \text{if } x = 2^{-2n} \text{ and } 0 < y < 1 \\ \{e_3(n)\}, & \text{if } 2^{-2n-1} < x < 2^{-2n} \text{ and } y = 1 \\ \{e_4(n)\}, & \text{if } x = 2^{-2n-1} \text{ and } 0 < y < 1 \\ \emptyset, & \text{otherwise} \end{cases} \tag{3.40}$$

where

$$\begin{aligned} &e_1(n) \text{ is the edge } [(2^{-2n-1}, 0),\ (2^{-2n}, 0)], \\ &e_2(n) \text{ is the edge } [(2^{-2n}, 0),\ (2^{-2n}, 1)], \\ &e_3(n) \text{ is the edge } [(2^{-2n}, 1),\ (2^{-2n-1}, 1)], \\ &e_4(n) \text{ is the edge } [(2^{-2n-1}, 1),\ (2^{-2n-1}, 0)]. \end{aligned} \tag{3.41}$$

The second helping function does most of the work. It recursively determines which rectangle is closest to the point (x, y), then calls the helping function g.

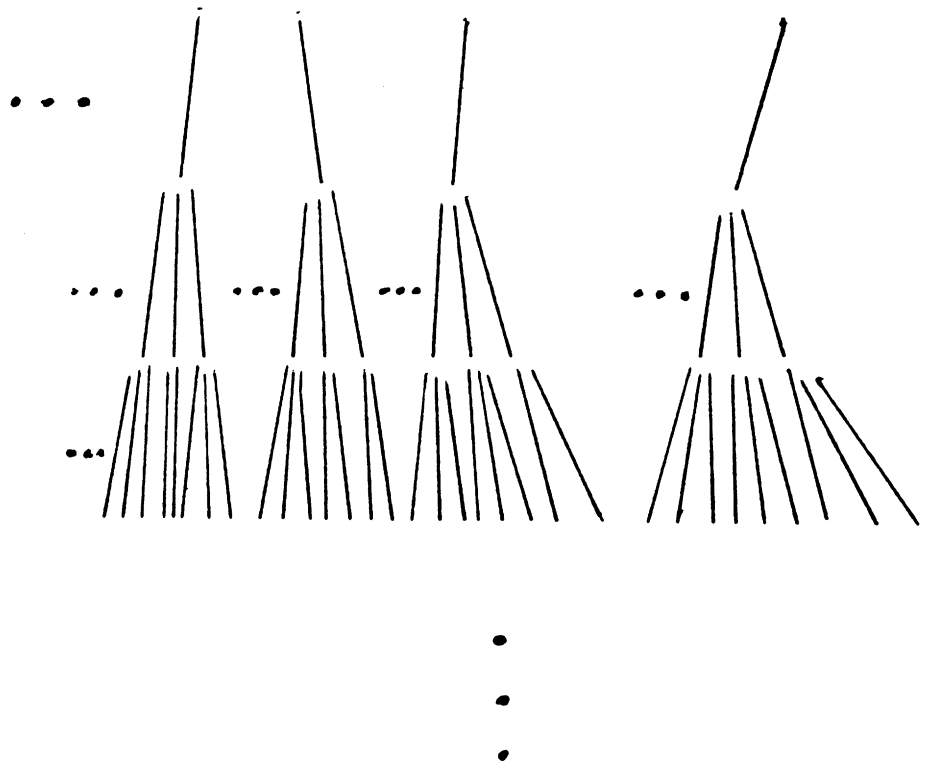


Figure 3.52. A collection of channels represented abstractly between encoding points. There are a countable number of levels. Each branch point corresponds to an entire encoding of a Turing machine. Thus there are a countable number of channels emerging from each branch point.

$$h(x,y,n) \quad = \quad \begin{cases} \emptyset, & \text{if } n < 0 \\ \emptyset, & \text{if } x \leq 0 \\ \emptyset, & \text{if } x > 1 \\ h(x,y,n+1), & \text{if } x < 2^{-2n-1} \\ h(x,y,n-1), & \text{if } x \geq 2^{-2n+1} \\ g(x,y,n), & \text{otherwise} \end{cases} \tag{3.42}$$

The first three conditions insure that h actually converges. The function f is now simply

$$f(x,y) \; = \; h(x,y,0).$$

If the rectangles were moved, so that they touched, the previous function f would be similarly constructed. Some of the lists returned by f would, of course,

contain more edges. Having established some intuition, it is now possible to abandon formally writing out the descriptive functions f. The remainder of the section describes countable collections of edges by suggestive diagrams, rather than by recursive functions. It should be clear that one can write out a recursive function for these diagrams much in the same way that one could, in the previous example, write out a recursive function for the sequence of squares in terms of the trivially recursive function for the single square.

Before turning to the claim it is necessary to consider an encoding of Turing machines by points in the plane. It is well known[9] that a Turing machine is equivalent to a finite state machine with two unsigned unbounded counters. The operations permitted on the counters are "INCREMENT," "DECREMENT," and "TEST FOR ZERO." The configuration of the machine is given by q, n, m, where q is a state, and n and m are two non-negative integers corresponding to the two counter values. There are thus only countably many configurations.

It is clear that the configurations of the counter machine can be uniquely encoded over the integers by the mapping $\langle q, m, n\rangle \mapsto 2^q 3^n 5^m$. Here q is taken to be an integer representing the state of the machine. In a similar fashion, the configuration of the counter machine can be encoded over the interval [0, 1] by $\langle q, m, n\rangle \mapsto 2^{-q} 3^{-n} 5^{-m}$, and hence, over the section $[a, b] \times \{y\}$ of the plane by

$$\langle q, m, n\rangle \quad \mapsto \quad \left(a + \frac{2^{-q} 3^{-n} 5^{-m}}{b - a},\ y\right) \tag{3.43}$$

Now consider an encoding of the counter machine over the section $[a, b] \times \{y\}$, as in the last paragraph. For each point $p_{\langle q,m,n\rangle}$ in the encoding, it is possible to erect a conceptual cylinder about the point which does not include any other point of the encoding. Suppose that the left and right vertical lines of this cylinder are given by $x = a_{\langle q,m,n\rangle}$ and $b_{\langle q,m,n\rangle}$, respectively. Clearly it is possible to encode the counter machine again, this time over the section $[a_{\langle q,m,n\rangle}, b_{\langle q,m,n\rangle}] \times \{y - \Delta y\}$, as shown in Fig. 3.49.

Suppose that when the counter machine is in configuration $\langle q, m, n\rangle$ it next moves to configuration $\langle q', n', m'\rangle$. Let $p'_{\langle q',n',m'\rangle}$ be the point in the encoding below $p_{\langle q,m,n\rangle}$ which represents the configuration $\langle q', n', m'\rangle$. The action of the counter machine may be graphically represented by two edges which form a channel from $p_{\langle q,m,n\rangle}$ to $p'_{\langle q',n',m'\rangle}$ as in Fig. 3.49. If $\langle q, m, n\rangle$ is a halted configuration, $\langle q', n', m'\rangle$ is the same configuration at the next level of encoding. In addition to forming a channel between these two configurations, it is convenient to explicitly indicate the halt state by a horizontal stop in the channel, as shown in Fig. 3.50. This stop reflects the intuition that computation cannot proceed through the channel from the halt state.

Now imagine constructing, in the same fashion, a second level encoding and channel for every point in the first level encoding. The result is partially indicated

[9] See, e.g., Lewis and Papadimitriou, Ex. 6.4.2.

in Fig. 3.51. Now repeat this whole procedure for every second level encoding (there will be a countable number of these), then for every third level encoding, and so on. Adjacent channels may be connected, so that the interiors of the channel edges are actually interior. This is a minor point.

By choosing the nth level of encodings at $y = 2^{-n+1}$, and by taking the single first level encoding to be along the section $[0, 1] \times \{1\}$, it is possible to encode all possible computation paths of the counter machine in a tree-like fashion within the square $[0, 1] \times [0, 1]$, as shown in Fig. 3.52. The encoding consists of a countable collection of oriented line segments. There are an infinite number of levels and an infinite number of branches at each level. It is clear from the construction process that there exists a recursive function f which describes this collection in the manner outlined previously. In other words, there exists a recursive function f which maps a point (x, y) in the plane to a list of edges that pass through that point. This is true because f has much the same form as the function in the sequence of squares example.

Specifically, f uses recursion to narrow in on a particular channel. Given an argument (x, y), f uses the value of y to recursively determine between what two levels the point (x, y) lies. This process is identical to the recursion used in the sequence of squares example. Similarly, f can find the nearest encoding points at any given level, using the value of x. f then interrogates the counter machine's transition function to determine the channel which is closest to (x, y). Having established the channel's identity, it is an easy matter to decide whether the given point (x, y) lies on the edges of the channel.

As a slight modification to the function f, it is possible to predict collisions of a point moving along a trajectory specified by the ray

$$\mathbf{p}_0 + t\mathbf{v}, \qquad \text{with } t \geq 0. \tag{3.44}$$

Specifically, f returns the closest point (and associated constraint edges) to $\mathbf{p}_0$ which lies on the specified ray and which lies in the topological closure of the collection of channels. In order to make sense of such an intersection, it is necessary to define what is meant by a point colliding with the limit of a a sequence of edges. Such a definition is not needed to establish the claim below, so for the purposes of this exposition, it is sufficient to rule out such collisions by explicitly placing constraints in the environment at the limit points of edge sequences. Thus, it is in principle possible to predict damper motions.

Finally, it is important to note that the function which maps counter machines to their encoding functions is itself computable. Specifically, let CM be a counter machine, and let f_{CM} be constructed as above. Then the mapping $CM \mapsto f_{CM}$ is recursive, that is, computable. The process involved is only slightly more complicated than that involved in writing out f in the sequence of squares example.

In much the same way that the collection of channels is representable by a recursive function, so too is the the collection of horizontal edges inserted as stops

inside the channels at halted configurations. Determination of this function is itself also a computable process.

Suppose that the sensor and control errors are zero, and that there is no friction. Let the commanded velocity be straight down. Given these assumptions, a point moving in the maze of channels moves in free space or slides along the channels, unless obstructed by a horizontal strip. If the point starts at a configuration point in the top level encoding, then the only horizontal edges that it can encounter are the channel stops at the halted configurations.

With the previous definitions and discussion in hand, the uncomputability claim is immediate.

Claim 3.35 Let p be a point in the plane, and let G_g be recursive subset of the plane consisting of a countable collection of line segments described by a recursive function g, as defined above. Suppose the constraints in the environment form a countable collection of line segments described by a recursive function f.

The **PRE-IMAGE** problem is unsolvable, where

$$\textbf{PRE-IMAGE} = \left\{ \langle p, f, g \rangle \;\middle|\; \{p\} = P_{\{p\}}(\{G_g\}) \right\} \tag{3.45}$$

Proof: The proof is a reduction from the halting problem. Specifically let

$$\textbf{BLANK} = \{\langle M \rangle \mid M \text{ is a Turing machine that halts on blank tape.}\}. \tag{3.46}$$

The strategy of the proof is to show that **BLANK** can be reduced to **PRE-IMAGE**. Since **BLANK** is uncomputable, this proves that **PRE-IMAGE** is also uncomputable.

To exhibit the reduction, suppose that $\langle M \rangle$ is given. For simplicity, assume that M is represented by a counter machine. The discussion prior to the statement of the claim shows how to effectively compute an encoding of M, in terms of a sequence of channels. Let f be the recursive function describing that encoding, and let g be the recursive function describing the collection of horizontal edges at halted encodings. These edges form the goal set G_g. Finally, let p be the starting configuration at the top encoding level.

This shows how to compute $\langle p, f, g \rangle$ from $\langle M \rangle$. Clearly $\{p\} = P_{\{p\}}(\{G_g\})$ if and only if M halts on blank tape. ∎

It can also be shown that the PRE-IMAGE problem is not even recursively enumerable. By letting the goal state in the above construction be the interval [0, 1] on the x-axis, it is possible to reduce the problem of whether a Turing machine does not halt on blank tape to the pre-image question. This is because a point is in the pre-image of the goal if and only if there is a computation path which does not pass through any halt state. Since the problem of deciding whether a Turing machine does not halt on blank tape is not recursively enumerable, neither is the pre-image question.[10]

[10]This stronger version of the uncomputability result was pointed out to me by B. Donald (August 1984).

Chapter Four

Epistles

4. Friction

The previous chapters have described a motion planning scheme that employs sliding on surfaces to overcome uncertainty. The motions and their starting positions are computed by backprojecting from the desired goal regions. A backprojection region for a specified motion direction consists of precisely all those points which are guaranteed to eventually reach the goal under the commanded velocity. In order to compute backprojections it is necessary for the planner to recognize all points at which a motion could prematurely terminate by sticking on a configuration space surface. The planner must therefore employ a representation of friction in configuration space that makes explicit points at which the commanded velocity could result in sticking.

4.1. Introduction

Chapter 2 described a representation of friction by a configuration space friction cone, with properties analogous to those of a real space friction cone. The representation was motivated by the fact that configuration space surfaces share many characteristics with their real space counterparts. In particular, they push back when pushed upon. Configuration space reaction forces lie along configuration space surface normals, just as real space reaction forces lie along real space surface normals.

This chapter provides the mathematical underpinnings for the intuitive treatment given in Ch. 2. The task is to provide a representation of translational real space friction in configuration space. Beyond its application to the motion planing problem, this representation is useful in simplifying the understanding of friction for tasks involving rotations and moments. Previous work modelling the effect of friction during parts mating has provided general methods for analyzing assembly tasks in real space. In particular, see the work by Simunovic (1975, 1979), Whitney (1982), Ohwovoriole and Roth (1981), and Ohwovoriole, Hill, and Roth (1980). These authors have considered assembly operations, such as the peg-in-hole problem, involving both single and multiple points of contact. Burridge, Rajan, and Schwartz (1983) have also examined the statics and dynamics of the peg-in-hole problem. In particular, they have considered the classes of motions possible for multiple points of contact, and have described possible motion ambiguities. Mason (1982) has considered friction in robot pushing and grasping operations. His work provides a method for planning grasping operations in the presence of uncertainty.

The kinematic and force analyses comprising these methods require explicit investigation of the geometric constraints on the assembly task. In other words, all possible points of contact must be explicitly considered. Such geometric constraints are directly manifest as submanifolds of configuration space. Multiple points of contact are as easily analyzed in configuration space as single points of contact. In both cases, the analysis must only deal with the motion of a point, instead of an entire object. The type of contact is evident from the dimension of the surface on which the point is moving. Thus it is desirable to develop a configuration space generalization of the classical friction cone.

The first part of the chapter restricts itself to the three dimensional configuration space arising from two translational and one rotational degrees of freedom. The ideas are developed by examining the planar version of peg-in-hole insertion. Multiple point contact is considered in the middle of the chapter, while extensions to higher dimensions are discussed in the last part of the chapter.

4.2. Planning and Simulation

The motivation for studying friction is twofold. A representation of friction is required both for planning and for simulating motion strategies. A planner uses the representation of friction to avoid surfaces on which a motion could stick and thereby fail to achieve the desired task's goal. A simulator uses the representation of friction to predict reaction forces that arise in response to control commands.

The planner throughout this thesis assumes generalized damper dynamics. In reality, Newton's laws govern the behavior of objects (at least within a resolution). A control system must try to give the appearance of a generalized damper world. Some of the control uncertainty arises as a consequences of the control system's inability to perfectly create a damper world. Simulation is useful in modelling this uncertainty.

The planner may assume the generalized damper world, with the specified control uncertainty. The simulator may then be used to verify plans computed by the planner. The simulator operates in Newton's world. It simulates both the control system's approximate implementation of the damper dynamics, as well as the motion of objects in response to a plan's suggested control commands. The simulator thus requires a representation of friction obeying Newton's laws. In particular, the simulator must be able to predict reaction forces and effective motions resulting from arbitrary applied forces and initial motions.

The analysis of this chapter assumes Newton's laws. The friction cone developed from this analysis provides a geometrical tool for predicting reaction forces under arbitrary initial conditions, subject to the laws of classical mechanics. The friction cone is useful therefore in simulating motions.

Additionally, this friction cone may be used as a geometrical tool for planning motions within the generalized damper framework. The friction cone specifies the range of reaction forces that can be generated at a point of contact. Therefore, by the equivalence of forces and velocities for a generalized damper with identity damping matrix, the negative friction cone describes the range of commanded velocities that can cause a motion to stick at a point of contact. The planner can use the friction cone to avoid surfaces on which motions could terminate prematurely.

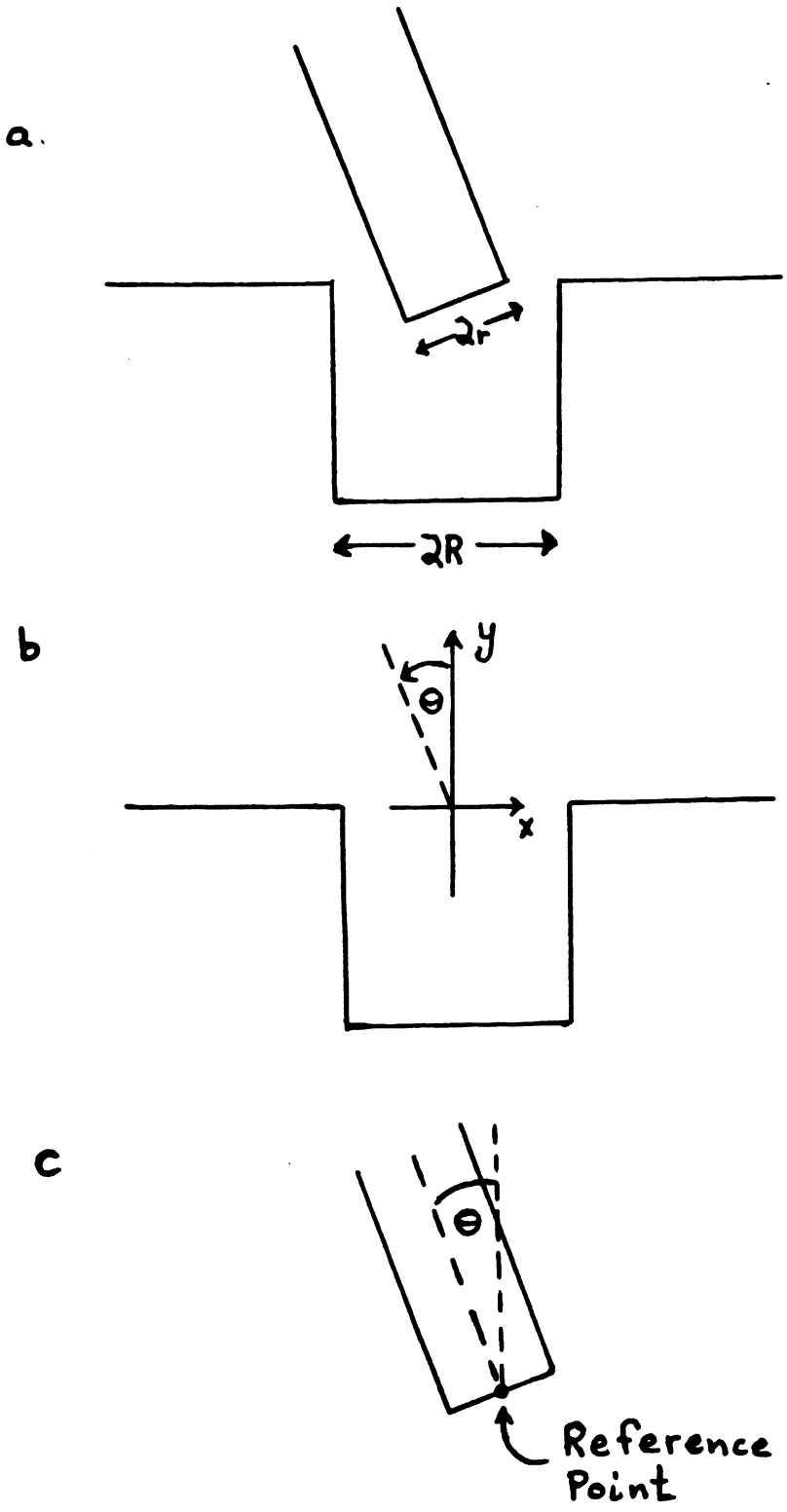


Figure 4.1. The peg-in-hole task consists of moving the peg into the hole. The coordinate system (x, y, θ) is shown.

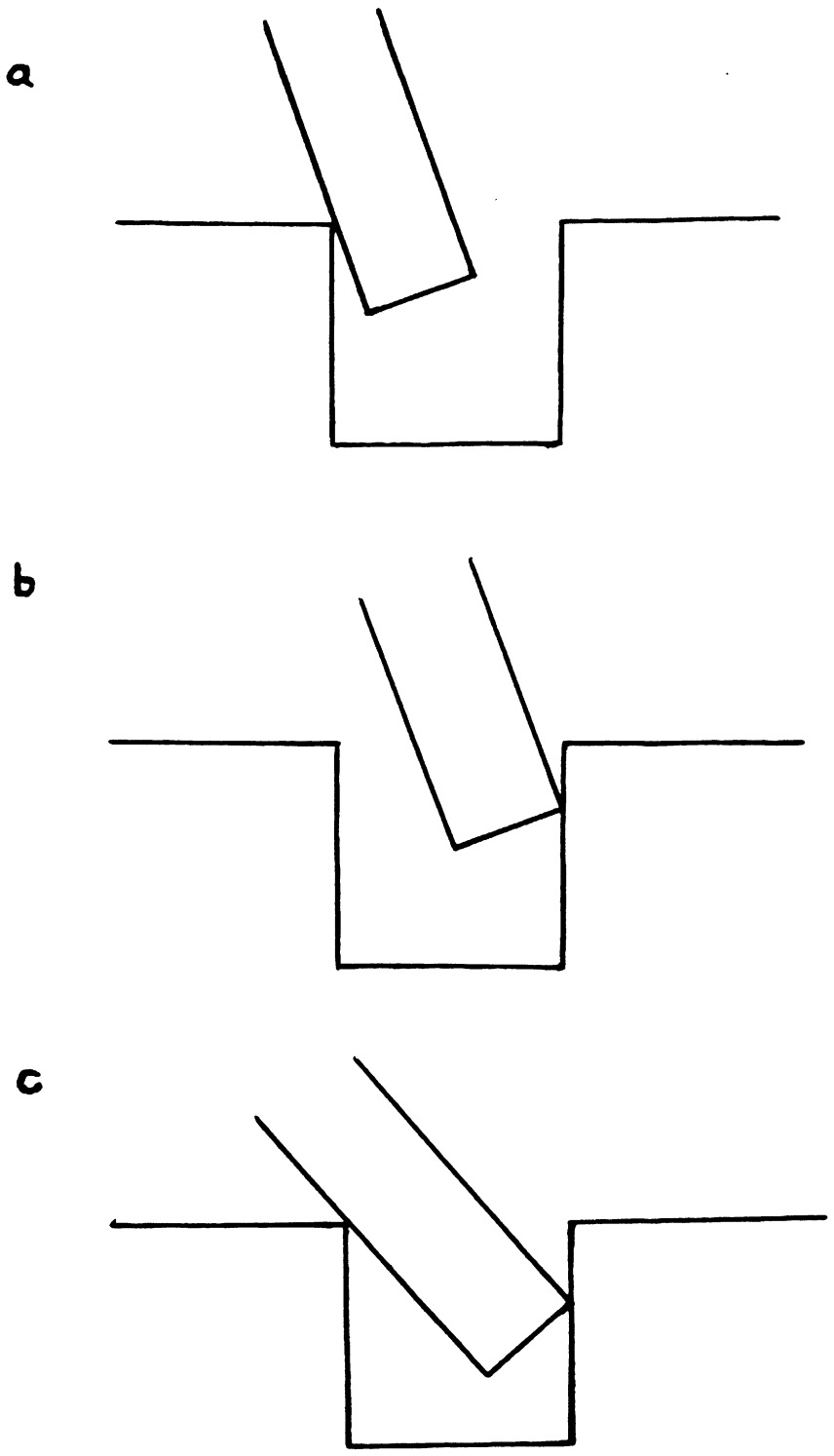


Figure 4.2. Two one-point contacts and one two-point contact. These are the typical contacts that occur for the peg-in-hole tasks.

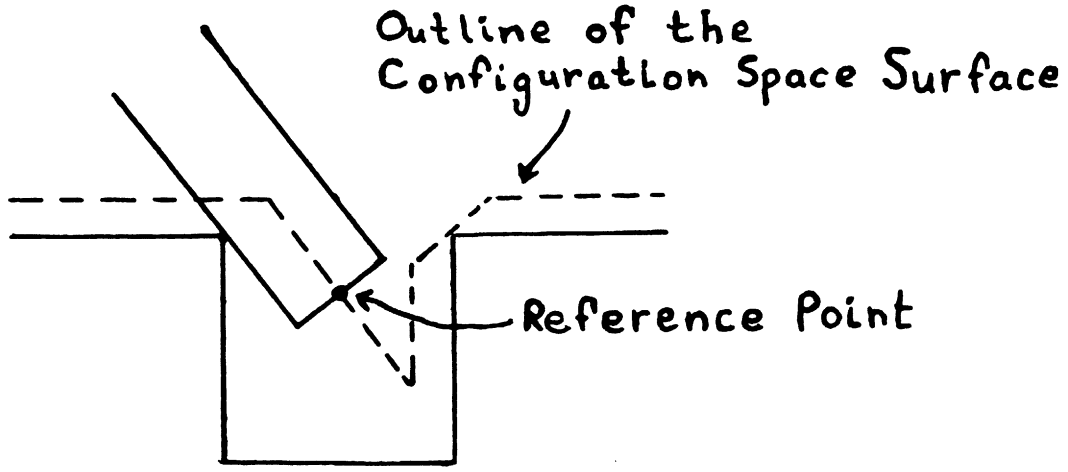


Figure 4.3. Configuration space surface that represents the constraints on the peg's reference point. A section of the surface for a single orientation of the peg is shown.

4.3. One-Point Contact of a Planar Peg-In-Hole Problem

This section develops a representation of friction for rigid objects translating and rotating in the plane. The representation is developed by analyzing one-point contact for the classical peg-in-hole problem.

4.3.1. Review, Assumptions, and Notation

The configuration space of an object with two translational and one rotational degrees of freedom is the manifold $\Re^2 \times S^1_\rho$, where S^1_ρ is the one dimensional sphere of radius ρ. By placing the usual inner product on S^1_ρ, configuration space becomes a Riemannian manifold. Varying ρ allows one to adjust the metric on this manifold. The physical significance of ρ will become apparent later.

Following a traditional abuse of notation, one may think of the representation of this configuration space as $\Re^3$. A point in this space is specified by the vector (x, y, q), where q is identified with a point θ on the sphere S^1_ρ by the relation $q = \rho\,\theta$. It is convenient to write θ, rather than q/ρ. In a similar fashion force is represented by the vector (F_x, F_y, F_q), where $F_q = \tau/\rho$, that is, F_q is torque divided by ρ.

Finally, for the sake of simplicity, it is convenient to ignore the difference between static friction and kinetic friction, assuming instead a constant coefficient of friction μ.

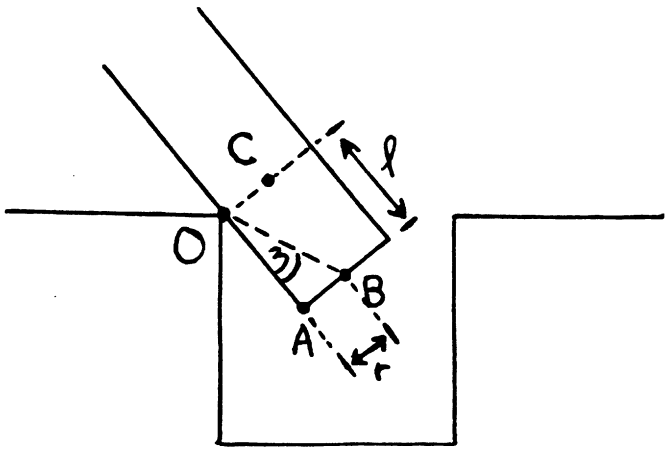


Figure 4.4. Two possible choices for the reference point are points A and B.

4.3.2. Peg-In-Hole Insertion

This section examines friction in the setting of planar peg-in-hole insertion, as indicated in Fig. 4.1.a. The hole has width $2R$, while the peg has width $2r$. A global reference frame is centered in the middle of the top of the hole, as indicated by Fig. 4.1.b. The configuration space of the peg is described with respect to a reference point in the middle of the bottom of the peg, as indicated by Fig. 4.1.c.

There are three particularly interesting types of configurations which the peg may assume. They are the two one-point contact configurations arising from either the interaction of an edge of the peg with a vertex of the hole or the interaction of a vertex of the peg with an edge of the hole, plus the one two-point contact configuration which results from the simultaneous occurrence of the one-point contacts. See Figs. 4.2.a-4.2.c. For each of the contacts shown there exists a symmetrical contact with the peg tilted in the opposite direction.

This examination will focus on the first of the one-point contacts. The legal positions of the peg in this type of a configuration give rise to a surface in configuration space. A cross section of this surface for a given value of θ is depicted in Fig. 4.3. The equation of a portion of this surface is given by

$$y \sin\theta + x \cos\theta = r - R\cos\theta,$$

or

$$y \sin\frac{q}{\rho} + x \cos\frac{q}{\rho} = r - R\cos\frac{q}{\rho}. \tag{4.1}$$

The outward pointing normal at any point on this surface is

$$\mathbf{n} = \frac{1}{\sqrt{1+\frac{\ell^2}{\rho^2}}}\left(\cos\theta,\ \sin\theta,\ -\frac{\ell}{\rho}\right), \tag{4.2}$$

where

$$\ell = \frac{x + R - r\cos\theta}{\sin\theta} \tag{4.3}$$

is the depth of the peg insertion.

4.3.3. Choice of Reference Point

Suppose one had chosen the peg's reference point to be the left vertex at the bottom of the peg, rather than the point in the middle of the base. It is interesting to see how this would change the corresponding configuration space surface. The two reference points are separated by a distance r (see Fig. 4.4). However, the surface normal does not depend on r, but merely on the depth and angle of insertion. The depth of insertion appears to depend on r, but this is illusory. To recognize the independence, parameterize the surface in terms of θ and ℓ:

$$\begin{aligned} x(\theta,\ell) &= \ell\sin\theta + r\cos\theta - R \\ y(\theta,\ell) &= -\ell\cos\theta + r\sin\theta \\ q(\theta,\ell) &= \rho\,\theta \end{aligned} \tag{4.4}$$

Then the surface normal is given by $\mathbf{n}$, as above, where ℓ is now an independent variable (see Eq. (4.2)).

In other words, the normal and tangent spaces of the configuration space surface are independent of the choice of the reference point along the bottom of the peg, depending solely on the insertion depth and orientation. The difference between surfaces generated from different reference points lies in their curvatures. Consider Fig. 4.4. A portion of the configuration space surface corresponding to the choice of A as the reference point is generated by the line segment $\overline{OA}$ as the peg rotates about the point O. Specifically, this portion of the configuration space surface is a helicoid corresponding to a helix of radius ℓ. In comparison, the configuration space surface generated by the choice of B as a reference point is determined by the movement of the line segment $\overline{CB}$ as the peg rotates about the contact point O. The portion of the configuration space surface thus generated corresponds to a particular surface lying between two helices, one of radius r, the other of radius $\sqrt{\ell^2 + r^2}$.

The normal and tangent spaces are independent of the choice of reference point along the bottom of the peg because the change in reference point along the bottom of the peg is parallel to the real space normal. Thus, moving the reference point does not change the moment arm through which a reaction force at the point of contact induces a reaction torque about the reference point. Recall from Sec.

2.4.2, that the configuration space normal is just the real space normal with an additional component corresponding to the torque induced by a real space reaction force at the point of contact. Since moving the reference point along the bottom of the peg does not change the induced reaction torque, it also does not change the configuration space normal.

A change of reference point along the length of the peg does change the direction of the configuration space normal and the configuration space tangent space. The effective moment arm about the reference point, through which a reaction force at the point of contact acts, changes. The parameter ℓ describes the length of this moment arm. Thus the torque induced by a reaction force at the point of contact changes relative to the reference point, implying a change in the configuration space normal. The dependence of the configuration space normal on ℓ is explicit in Eq. (4.2).

An arbitrary choice of reference point can be described by ℓ and r. ℓ is the moment arm through which a reaction force at the point of contact induces a reaction torque about the reference point. r is the perpendicular distance of the reference point from the point of contact, measured in terms of distance parallel to the real space normal. The configuration space normal depends only on the value of ℓ.

4.3.4. Tangents

There exist two fundamental modes of movement along the configuration space surface. One corresponds to sliding of the peg, that is, a change in insertion depth, while the other corresponds to rotation of the peg about the contact point, that is, a change in orientation. One would expect the tangents associated with each of these movements to provide useful information.

Regardless of the choice of reference point, it is clear that the tangent arising from sliding movement is given by

$$\mathbf{t}_1 = (\sin\theta,\ -\cos\theta,\ 0). \tag{4.5}$$

In order to determine the tangents arising from rotational movement, one need merely calculate the tangents to the helices which represent the movement of the reference point during rotation, as previously described.

For the reference point A, the helix is given by

$$\begin{aligned} x(\theta) &= \ell\sin\theta - R \\ y(\theta) &= -\ell\cos\theta \\ q(\theta) &= \rho\,\theta \end{aligned} \tag{4.6}$$

which has a tangent given by

$$\mathbf{t}_2 = \frac{1}{\sqrt{1+\frac{\ell^2}{\rho^2}}}\left(\frac{\ell}{\rho}\cos\theta,\ \frac{\ell}{\rho}\sin\theta,\ 1\right). \tag{4.7}$$

For the reference point B, using the notation of Fig. 4.4, the helix is given by

$$\begin{aligned} x(\theta) &= \sqrt{\ell^2+r^2}\,\sin(\theta+\eta) - R \\ y(\theta) &= -\sqrt{\ell^2+r^2}\,\cos(\theta+\eta) \\ q(\theta) &= \rho\,\theta \end{aligned} \tag{4.8}$$

Noting that $\cos\eta = \ell/\sqrt{\ell^2+r^2}$, and $\sin\eta = r/\sqrt{\ell^2+r^2}$, it is seen that the tangent to the helix is

$$\mathbf{t}_3 = \frac{1}{\sqrt{1+\frac{\ell^2+r^2}{\rho^2}}}\left(\frac{\ell}{\rho}\cos\theta - \frac{r}{\rho}\sin\theta,\ \frac{\ell}{\rho}\sin\theta + \frac{r}{\rho}\cos\theta,\ 1\right) \tag{4.9}$$

All this information may be obtained directly from the surface parameterization given earlier in Eq. (4.4).

Notice that for the reference point A, the tangent vectors arising from sliding and rotation are orthogonal, whereas for the reference point B this is not the case. The tangent vector orthogonal to $\mathbf{t}_3$ is

$$\mathbf{t}_4 = \frac{1}{k}\left(\frac{r\ell}{\rho^2}\frac{1}{1+\frac{\ell^2}{\rho^2}}\cos\theta + \sin\theta,\ \frac{r\ell}{\rho^2}\frac{1}{1+\frac{\ell^2}{\rho^2}}\sin\theta - \cos\theta,\ \frac{r}{\rho}\frac{1}{1+\frac{\ell^2}{\rho^2}}\right), \tag{4.10}$$

where

$$k = \sqrt{1+\frac{r^2}{\rho^2+\ell^2}} \tag{4.11}$$

is the appropriate normalizing factor.

In general, for any given reference point, it is possible to construct two pairs of orthogonal tangent vectors. One pair is constructed from the tangent vector for sliding, the other from the tangent vector for rotation. The pairs need not be distinct, as the reference point A shows. Each pair is useful for answering questions concerning its associated mode of movement. This will become clear while constructing the analogue to the friction cone.

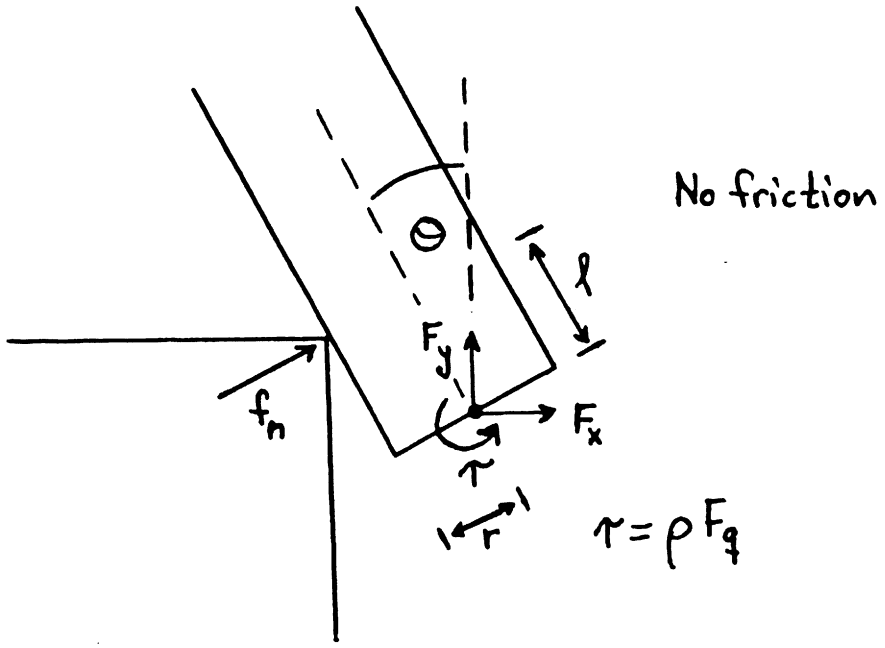


Figure 4.5. Force diagram without friction.

4.3.5. Equations of Motion without Friction

This section examines the configuration space formulation of the equations of motion of the peg during frictionless one-point contact with the hole at O. Reference point B is used throughout the analysis (see Fig. 4.4).

The configuration space surface generated by the reference point is the locus of possible positions of the reference point while in one-point contact with the hole. In particular, it provides geometric constraints on the possible movements of the reference point, restricting it to movements tangential to the surface. In other words, tangential movements are legal, whereas normal movements are prohibited, if contact is to be maintained. In fact, normal movements are only physically possible in one direction, namely away from the surface. This suggests that, (1) in order to perform a static analysis of the peg, one need merely restrict applied forces to lie in the normal space of the configuration space surface. Furthermore, (2) for any applied force, one would expect the configuration space surface to provide a reaction force equal and opposite to the normal component of the applied force (see Mason (1981) for a more detailed development of these ideas). The remainder of the section is a verification of these two remarks.

4.3.5.1. Statics

This subsection verifies remark (1), concerning the static analysis of the peg in one-point contact with the hole, as shown in Fig. 4.5. Restricting applied forces to the normal space of the configuration space surface is equivalent to constraining their tangential components to be zero. If $\mathbf{F} = (F_x, F_y, F_q)$, then this means that

$$\begin{aligned} \mathbf{F} \cdot \mathbf{t}_1 &= 0 \\ \mathbf{F} \cdot \mathbf{t}_3 &= 0 \end{aligned} \tag{4.12}$$

By Sec. 4.3.3, the tangent spaces for reference point A and reference point B are identical for a given insertion depth ℓ. In particular, $span\,\{t_1, t_2\}$ is the same as $span\,\{t_1, t_3\}$. Therefore, Eq. (4.12) is equivalent to

$$\begin{aligned} \mathbf{F} \cdot \mathbf{t}_1 &= 0 \\ \mathbf{F} \cdot \mathbf{t}_2 &= 0 \end{aligned} \tag{4.13}$$

Substituting for $\mathbf{t}_1$ and $\mathbf{t}_2$, it follows that

$$\begin{aligned} F_x \sin\theta - F_y \cos\theta &= 0 \\ F_x \frac{\ell}{\rho} \cos\theta + F_y \frac{\ell}{\rho} \sin\theta + F_q &= 0 \end{aligned} \tag{4.14}$$

Referring to Fig. 4.5, the standard static analysis yields

$$\begin{aligned} f_n + F_x \cos\theta + F_y \sin\theta &= 0 \\ F_x \sin\theta - F_y \cos\theta &= 0 \\ -f_n\,\ell + \tau &= 0, \end{aligned} \tag{4.15}$$

where $\tau = \rho F_q$.

Substituting for f_n and τ in the third equation reveals the equivalence of this system of constraints with the configuration space surface formulation (Eq. (4.14)). This verifies the first remark.

4.3.5.2. Normal Reaction Force

This subsection verifies remark (2), concerning the computation of reaction forces given arbitrary applied forces. It is claimed that, given an arbitrary applied force pointing into the configuration space surface, the configuration space reaction force lies along the configuration space normal, cancelling the normal component of the applied force.

In order that this claim be true, it is necessary to make the following two assumptions:

1. The peg's center of mass is coincident with the reference point.
2. The radius of gyration about the center of mass is ρ.

The first assumption is required to decouple the effects of forces and torques about the reference point. This assumption ensures that the acceleration of a point in configuration space in response to a net configuration space force is along the direction of the force. More generally, the reference point may be taken to be the center of compliance of the moving object. A pure force through the reference point should cause pure translation of the object, while a pure torque about the reference point should cause pure rotation of the object around the reference point.

The second assumption is motivated by the realization that rotation in configuration space corresponds to movement of a point around a circle of radius ρ. If the point has mass m, then the moment of inertia about the center of the circle is $m\rho^2$.

The normal component of an arbitrary applied force is given by $(\mathbf{F} \cdot \mathbf{n})\mathbf{n}$, in terms of configuration space notation.

Now

$$\mathbf{F} \cdot \mathbf{n} = \frac{1}{\sqrt{1 + \frac{\ell^2}{\rho^2}}} \left(F_x \cos\theta + F_y \sin\theta - F_q \frac{\ell}{\rho} \right) \tag{4.16}$$

So,

$$\begin{aligned} (\mathbf{F} \cdot \mathbf{n})\mathbf{n} = \frac{1}{1 + \frac{\ell^2}{\rho^2}} \Bigg(& F_x \cos^2\theta + F_y \sin\theta \cos\theta - F_q \frac{\ell}{\rho} \cos\theta, \\ & F_y \sin^2\theta + F_x \sin\theta \cos\theta - F_q \frac{\ell}{\rho} \sin\theta, \\ & -F_x \frac{\ell}{\rho} \cos\theta - F_y \frac{\ell}{\rho} \sin\theta + F_q \frac{\ell^2}{\rho^2} \Bigg) \end{aligned} \tag{4.17}$$

Thus the configuration space reaction force should be the negative of this vector.

In order to verify this prediction in terms of standard force analysis, assume the notation of Fig. 4.5. Additionally, introduce the following symbols:

a_n = acceleration of the center of mass in the direction $(\cos\theta, \sin\theta)$.

a_t = acceleration of the center of mass in the direction $(\sin\theta, -\cos\theta)$.

α = angular acceleration in counterclockwise direction about the center of mass.

m = mass of peg.

I = moment of inertia about the center of mass.

The reaction force consists of a pure force and a torque. The pure force has magnitude f_n, pointing in the direction $(\cos\theta, \sin\theta)$. The torque has signed magnitude $-f_n\,\ell$. Therefore, in terms of configuration space notation, the reaction force is $(f_n\cos\theta, f_n\sin\theta, -f_n\frac{\ell}{\rho})$. Comparing this vector to the predicted reaction force, one sees that it is only necessary to verify the identity

$$f_n = \frac{1}{1+\frac{\ell^2}{\rho^2}}\left(-F_x\cos\theta - F_y\sin\theta + F_q\frac{\ell}{\rho}\right). \tag{4.18}$$

The equations of motion are:

$$\begin{aligned} ma_n &= f_n + F_x\cos\theta + F_y\sin\theta \\ ma_t &= F_x\sin\theta - F_y\cos\theta \\ I\,\alpha &= \tau - f_n\,\ell \end{aligned} \tag{4.19}$$

In order to maintain contact at point O, under quasi-static assumptions, the following constraint must hold:

$$a_n = \ell\,\alpha. \tag{4.20}$$

Therefore,

$$
\begin{aligned}
f_n &= ma_n - F_x\cos\theta - F_y\sin\theta \\
&= m\ell\alpha - F_x\cos\theta - F_y\sin\theta \\
&= \frac{m\ell}{I}(\tau - f_n\ell) - F_x\cos\theta - F_y\sin\theta \\
&= \frac{m\ell}{m\rho^2}(\tau - f_n\ell) - F_x\cos\theta - F_y\sin\theta
\end{aligned}
\tag{4.21}
$$

Hence,

$$
\left(1 + \frac{\ell^2}{\rho^2}\right) f_n = \frac{\ell}{\rho^2}\tau - F_x\cos\theta - F_y\sin\theta, \tag{4.22}
$$

$$
\begin{aligned}
f_n &= \frac{1}{1 + \frac{\ell^2}{\rho^2}}\left(-F_x\cos\theta - F_y\sin\theta + \frac{\ell}{\rho^2}\tau\right) \\
&= \frac{1}{1 + \frac{\ell^2}{\rho^2}}\left(-F_x\cos\theta - F_y\sin\theta + F_q\frac{\ell}{\rho}\right)
\end{aligned}
\tag{4.23}
$$

This proves the second remark, and completes the analysis of this subsection.

4.3.6. A Configuration Space Friction Cone

This subsection develops the configuration space analogue of the classical friction cone. One-point contact is assumed throughout. As in the frictionless case previously discussed, the analysis will yield a method for computing static constraints and reaction forces in configuration space. This subsection continues to assume one-point contact.

In real space the friction cone is a method of geometrically capturing equilibrium constraints on forces. Furthermore, the friction cone may be viewed as a specification of the range of reaction forces that can arise in response to applied forces. The generalized configuration space friction cone should possess these same properties.

Consider the general case of a configuration space surface generated by the reference point of an object in one-point contact with another object. Recall that there exist two pairs of fundamental tangent vectors at each point on the surface. Let $\mathbf{t}_s$ denote the tangent vector corresponding to sliding movement, and $\mathbf{t}_r$ denote the tangent vector corresponding to rotation about the contact point. Also, let $\mathbf{t}_s^\perp$ be orthogonal to $\mathbf{t}_s$, and $\mathbf{t}_r^\perp$ be orthogonal to $\mathbf{t}_r$. Then the two fundamental pairs are $(\mathbf{t}_s, \mathbf{t}_s^\perp)$ and $(\mathbf{t}_r, \mathbf{t}_r^\perp)$. For later convenience assume that $\mathbf{t}_r^\perp \cdot \mathbf{t}_s \geq 0$. As usual, let $\mathbf{n}$ be the normal vector to the configuration space surface. Notice that in the case of one-point contact, the tangent $\mathbf{t}_s$ lies in the x-y plane, that is, has a zero q

component. Let $\mathbf{n}_0$ be another vector lying in the x-y plane, such that $\mathbf{n}_0 \cdot \mathbf{n} > 0$ and $\mathbf{n}_0 \cdot \mathbf{t}_s = 0$. Then $\mathbf{n}_0$ is the outward pointing normal of the real space obstacle from which the configuration space surface is constructed. To be precise, in the case of one-point contact of the form depicted in Fig. 4.2.b, $\mathbf{n}_0$ is the outward pointing normal of the real space object, whereas in the case of one-point contact of the form depicted in Fig. 4.2.a, $\mathbf{n}_0$ is the inward pointing normal of the peg. All vectors are of unit length.

One may now construct the generalized friction cone by examining the constraints required on applied forces in order to maintain static equilibrium. Notice that for one-point contact frictional forces arise in response to applied forces that act in directions of movement involving sliding, whereas no such forces arise in response to purely rotationally directed forces. This observation suggests the following constraints:

(i) No restriction on the normal component of an applied force, except perhaps for a sign restriction. This restriction is made solely to maintain contact.

(ii) The tangential component of an applied force in the direction of pure rotation must be zero.

(iii) The tangential component of an applied force in the direction of pure sliding must be constrained in terms of the normal component.

Conditions (i) and (ii) specify constraints on applied forces in the orthogonal directions $\mathbf{n}$ and $\mathbf{t}_r$, respectively. Condition (iii) specifies a constraint in terms of the direction $\mathbf{t}_s$, which is not necessarily orthogonal to the other two directions. It is desirable to modify the third condition so that it constrains force in the $\mathbf{t}_r^{\perp}$ direction.

Notice that the second constraint may be formulated as

$$\text{(ii)} \quad \mathbf{F} \cdot \mathbf{t}_r = 0, \tag{4.24}$$

which specifies a plane in configuration space. One would expect that the modified third constraint should be of the form

$$\text{(iii)} \quad c_1 \, \mathbf{F} \cdot \mathbf{n} \leq \mathbf{F} \cdot \mathbf{t}_r^{\perp} \leq c_2 \, \mathbf{F} \cdot \mathbf{n}, \qquad \text{with } \mathbf{F} \cdot \mathbf{n} \leq 0, \tag{4.25}$$

which selects a convex subset from this plane. This subset represents the generalized friction cone. It remains merely to determine the constants c_1 and c_2.

One can use the classical friction cone constraints to derive values for the two constants. Notice that one can assume without loss of generality that $\mathbf{F} \cdot \mathbf{n}_0 \leq 0$ in static equilibrium. This assumption is required in order to physically maintain the one-point contact in static equilibrium. Note, however that $\mathbf{F} \cdot \mathbf{n}_0 \leq 0$ does not imply that $\mathbf{F} \cdot \mathbf{n} \leq 0$. In formulating the modified third constraint it was

tacitly assumed that applied forces could only point in the opposite direction of the outward pointing surface normal. While this is true for real space forces, it will become apparent that in some situations configuration space forces may actually point away from the configuration space surface, and yet maintain static equilibrium. For the moment, however, assume that $\mathbf{F} \cdot \mathbf{n} \leq 0$.

Given these assumptions, Coulomb's law states that

$$\mu \mathbf{F} \cdot \mathbf{n}_0 \leq \mathbf{F} \cdot \mathbf{t}_s \leq -\mu \mathbf{F} \cdot \mathbf{n}_0. \tag{4.26}$$

Now, using $\mathbf{F} \cdot \mathbf{t}_r = 0$ and $\mathbf{n} \cdot \mathbf{t}_s = 0$,

$$\begin{aligned} \mathbf{F} \cdot \mathbf{n}_0 &= (\mathbf{F} \cdot \mathbf{n})\mathbf{n} \cdot \mathbf{n}_0 + (\mathbf{F} \cdot \mathbf{t}_r)\mathbf{t}_r \cdot \mathbf{n}_0 + (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{n}_0 \\ &= (\mathbf{F} \cdot \mathbf{n})\mathbf{n} \cdot \mathbf{n}_0 + (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{n}_0 \end{aligned} \tag{4.27}$$

$$\begin{aligned} \mathbf{F} \cdot \mathbf{t}_s &= (\mathbf{F} \cdot \mathbf{n})\mathbf{n} \cdot \mathbf{t}_s + (\mathbf{F} \cdot \mathbf{t}_r)\mathbf{t}_r \cdot \mathbf{t}_s + (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{t}_s \\ &= (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{t}_s \end{aligned} \tag{4.28}$$

Therefore

$$\begin{aligned} &\text{(a)} \quad (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{t}_s \leq -\mu\,[(\mathbf{F} \cdot \mathbf{n})\mathbf{n} \cdot \mathbf{n}_0 + (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{n}_0] \\ &\text{(b)} \quad (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{t}_s \geq \mu\,[(\mathbf{F} \cdot \mathbf{n})\mathbf{n} \cdot \mathbf{n}_0 + (\mathbf{F} \cdot \mathbf{t}_r^\perp)\mathbf{t}_r^\perp \cdot \mathbf{n}_0] \end{aligned} \tag{4.29}$$

Equivalently,

$$\begin{aligned} &\text{(a)} \quad (\mathbf{F} \cdot \mathbf{t}_r^\perp)\,[\mathbf{t}_r^\perp \cdot \mathbf{t}_s + \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0] \leq -\mu\,(\mathbf{F} \cdot \mathbf{n})\mathbf{n} \cdot \mathbf{n}_0 \\ &\text{(b)} \quad (\mathbf{F} \cdot \mathbf{t}_r^\perp)\,[\mathbf{t}_r^\perp \cdot \mathbf{t}_s - \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0] \geq \mu\,(\mathbf{F} \cdot \mathbf{n})\mathbf{n} \cdot \mathbf{n}_0 \end{aligned} \tag{4.30}$$

So, let

$$c_1 = \begin{cases} \frac{\mu \mathbf{n} \cdot \mathbf{n}_0}{\mathbf{t}_r^\perp \cdot \mathbf{t}_s - \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0} & \text{if } \mathbf{t}_r^\perp \cdot \mathbf{t}_s - \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0 > 0 \\ +\infty & \text{otherwise} \end{cases} \tag{4.31}$$

$$c_2 = \begin{cases} \frac{-\mu \mathbf{n} \cdot \mathbf{n}_0}{\mathbf{t}_r^\perp \cdot \mathbf{t}_s + \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0} & \text{if } \mathbf{t}_r^\perp \cdot \mathbf{t}_s + \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0 > 0 \\ -\infty & \text{otherwise} \end{cases}$$

Applied forces that satisfy the constraints are of the form

$$\mathbf{F} = c\,(\mathbf{n} + s\,c_i\,\mathbf{t}_r^{\perp}), \tag{4.32}$$

where

$$\begin{aligned} c &\leq 0 \\ c_i &\in \{\, c_1,\ c_2 \,\} \\ s &\in [\,0,\ 1\,] \end{aligned} \tag{4.33}$$

The configuration space surface may be viewed as providing reaction forces which lie within the range specified by the friction cone. The description of the possible reaction forces is identical to the definition of F given above, except that $c \geq 0$.

In general, under quasi-static assumptions, given any applied force $\mathbf{F}_A$ with $\mathbf{F}_A \cdot \mathbf{n} \leq 0$, the configuration space surface will respond with a reaction force $\mathbf{F}_R$ given by

$$\mathbf{F}_R = -(\mathbf{F}_A \cdot \mathbf{n})\,\mathbf{n} - h\,\mathbf{t}_r^{\perp}, \tag{4.34}$$

where h is chosen such that one of the following conditions (Eqs. (4.35) and (4.36)) holds:

$$c_1\,\mathbf{F}_A \cdot \mathbf{n} < h < c_2\,\mathbf{F}_A \cdot \mathbf{n}, \text{ and } \mathbf{F}_R \cdot \mathbf{t}_r^{\perp} + \mathbf{F}_A \cdot \mathbf{t}_r^{\perp} = 0 \tag{4.35}$$

or

$$h = c_i\,\mathbf{F}_A \cdot \mathbf{n},\ i = 1 \text{ or } 2, \text{ and } 0 \leq -(\mathbf{F}_R \cdot \mathbf{t}_r^{\perp})(\mathbf{F}_A \cdot \mathbf{t}_r^{\perp}) \leq (\mathbf{F}_A \cdot \mathbf{t}_r^{\perp})^2 \tag{4.36}$$

For the condition specified by Eq. (4.35), the reaction force lies inside the friction cone, while for the condition specified by Eq. (4.36), the reaction force lies on an edge of the friction cone.

In other words, given an applied force $\mathbf{F}_A$, one constructs the reaction force $\mathbf{F}_R$ by projecting $-\mathbf{F}_A$ along $\mathbf{t}_r$ into the plane containing the friction cone. If this projection lies inside the friction cone, then the projection is the reaction force. If the projection lies outside of the friction cone, then one must proceed by projecting along $\mathbf{t}_r^{\perp}$ until encountering the edge of the friction cone.

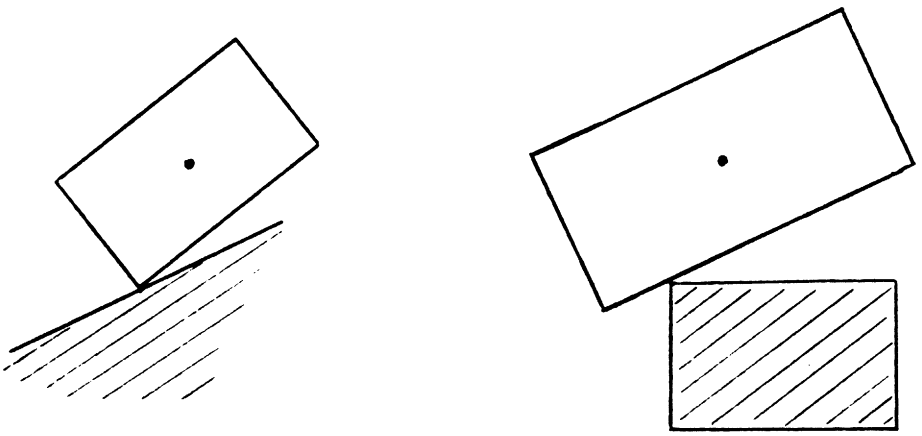


Figure 4.6. Two different types of contact between a moving object and an obstacle. Both examples are described by the same vectors at the point of contact. (The manner in which these vectors change, as the object moves, is different for the two types of contact.)

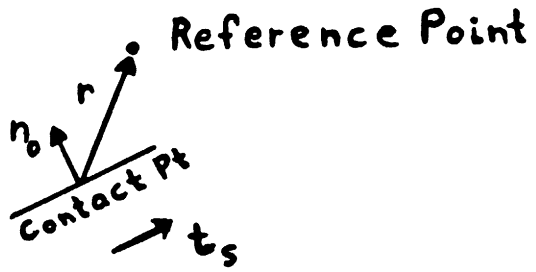


Figure 4.7. The local real space normal, sliding tangent, and radius vectors of the contacts shown in Fig. 4.6.

4.3.7. Comments

The infinite values of c_1 and c_2 correspond to friction cones with no edge constraints for applied forces that have negative normal components. This means that the configuration space friction cone actually dips below the tangent plane of the configuration space surface. In other words, it is possible to maintain static equilibrium with $\mathbf{F}_A \cdot \mathbf{n} > 0$, assuming that $\mathbf{F}_A \cdot \mathbf{t}_r^{\perp}$ is chosen properly.

The analysis in this case is similar to the one presented above, so it will not be discussed here. Suffice it to mention that the inequality constraints become lower bounds on the magnitude of $\mathbf{F} \cdot \mathbf{t}_r^{\perp}$ relative to that of $\mathbf{F} \cdot \mathbf{n}$, rather than containing upper bounds. The description of the reaction forces changes accordingly.

Finally, note that by construction only one of $\mathbf{t}_r^{\perp} \cdot \mathbf{t}_s - \mu \mathbf{t}_r^{\perp} \cdot \mathbf{n}_0$ and

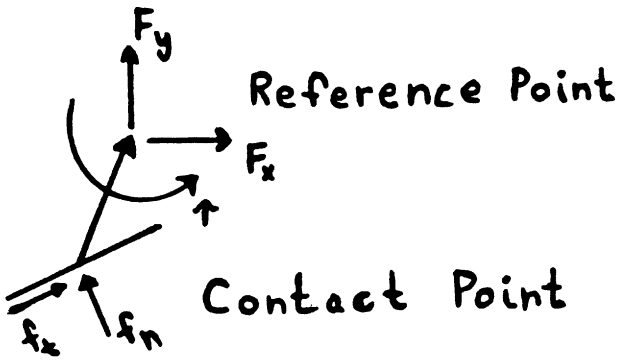


Figure 4.8. Applied forces at the reference point and reaction forces at the contact point for the contacts of Fig. 4.6.

$\mathbf{t}_r^\perp \cdot \mathbf{t}_s + \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0$ can be negative. Thus the friction cone always lies at least partially above the configuration space surface tangent plane.

At this point it should be clear that the most general formulation of constraint (iii) above, replacing Eq. (4.25), should be:

$$\begin{aligned} &\text{(iii.a)} \qquad \mathbf{F} \cdot \mathbf{n} \leq k_1 \, \mathbf{F} \cdot \mathbf{t}_r^\perp \\ &\text{(iii.b)} \qquad \mathbf{F} \cdot \mathbf{n} \leq k_2 \, \mathbf{F} \cdot \mathbf{t}_r^\perp, \end{aligned} \tag{4.37}$$

where

$$\begin{aligned} k_1 &= \frac{\mathbf{t}_r^\perp \cdot \mathbf{t}_s - \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0}{\mu \mathbf{n} \cdot \mathbf{n}_0} \\ k_2 &= \frac{\mathbf{t}_r^\perp \cdot \mathbf{t}_s + \mu \mathbf{t}_r^\perp \cdot \mathbf{n}_0}{-\mu \mathbf{n} \cdot \mathbf{n}_0} \end{aligned} \tag{4.38}$$

This formulation of the constraints follows directly from Eq. (4.30). It does not assume that $\mathbf{F} \cdot \mathbf{n} \leq 0$.

4.4. General Form of Planar One-Point Contact

The previous section on the peg-in-hole problem provided an overview of the issues involved in computing a configuration space friction cone. This section summarizes and generalizes the results of that section. It assumes that the reference point is at the center of mass, and that ρ is the radius of gyration about the reference point.

4.4.1. Vector Decomposition

Consider a planar object in one-point contact with some surface. Two contacts of different type that give rise to the same vector decomposition are shown in Fig. 4.6. Let the vector from the point of contact to the reference point of the object be given by $\mathbf{r} = (r_x, r_y)$, as in Fig. 4.7. Assume that the real space normal at the point of contact is given by $\mathbf{n}_0 = (n_x, n_y)$. Since the torque induced about the reference point by a force $\mathbf{f}$ at the point of contact is just $\mathbf{f} \times \mathbf{r}$,[1] the configuration space normal must be parallel to $\mathbf{n} = (n_x, n_y, \frac{1}{\rho} n_q)$, where n_q is the θ component of $\mathbf{n}_0 \times \mathbf{r}$. Letting, $\mathbf{t}_s$ be the sliding tangent, $\mathbf{t}_r$ the pure rotation tangent, and $\mathbf{t}_r^\perp$ the perpendicular tangent, as in the previous section, the general form for the vectors of interest is given by

$$\begin{aligned}
\mathbf{r} &= (r_x, r_y, 0) \\
\mathbf{n}_0 &= (n_x, n_y, 0) \\
\mathbf{n} &= \frac{\rho}{\sqrt{\rho^2 + n_q^2}} (n_x, n_y, \tfrac{1}{\rho} n_q) \\
\mathbf{t}_r &= \frac{1}{\sqrt{r_x^2 + r_y^2 + \rho^2}} (-r_y, r_x, \rho) \\
\mathbf{t}_r^\perp &= \pm \frac{1}{\sqrt{\delta_1^2 + \delta_2^2 + \delta_3^2}} (\delta_1, -\delta_2, \delta_3) \\
\mathbf{t}_s &= \pm (n_y, -n_x, 0),
\end{aligned} \tag{4.39}$$

where

$$\begin{aligned}
n_q &= n_x r_y - n_y r_x \\
\delta_1 &= \rho^2 n_y - n_q r_x \\
\delta_2 &= \rho^2 n_x + n_q r_y \\
\delta_3 &= \rho r_y n_y + \rho r_x n_x.
\end{aligned} \tag{4.40}$$

It is convenient to take the signs of $\mathbf{t}_r^\perp$ and $\mathbf{t}_s$ so that $\mathbf{t}_r^\perp \cdot \mathbf{n}_0 \geq 0$ and $\mathbf{t}_r^\perp \cdot \mathbf{t}_s \geq 0$.

Also note that the radius vector $\mathbf{r}$ can be recovered directly from the configuration space surface, since $\mathbf{r}$ is uniquely determined by $\mathbf{t}_r$, which can be recovered from the surface by a directional derivative in the positive q (or θ) direction.

4.4.2. Equations of Motion

It is interesting to see how the vector decomposition relates to the equations of motion. Let $\mathbf{F}_A = (F_{Ax}, F_{Ay}, F_{Aq})$ be a generalized applied force. In other words, the applied force is (F_{Ax}, F_{Ay}) and the applied torque is $\tau_A = \rho F_{Aq}$. This force is applied at the reference point. Let the reaction force at the point of contact have magnitude f_n. See Fig. 4.8. Then the equations of motion, in the absence of friction are

[1] Since the vector $\mathbf{r}$ is directed from the point of contact to the reference point, the torque about the reference point is $(-\mathbf{r}) \times \mathbf{f}$, which is $\mathbf{f} \times \mathbf{r}$.

$$\begin{aligned} f_n n_x + F_x &= m a_x \\ f_n n_y + F_y &= m a_y \\ f_n n_q + \tau &= m \rho^2 \alpha. \end{aligned} \tag{4.41}$$

In the presence of friction, there is also a tangential reaction force f_t, subject to the restriction $0 \leq |f_t| \leq \mu f_n$. Letting $v_q = n_x r_x + n_y r_y$, the equations of motion become

$$\begin{aligned} f_n n_x + f_t n_y + F_x &= m a_x \\ f_n n_y - f_t n_x + F_y &= m a_y \\ f_n n_q + f_t v_q + \tau &= m \rho^2 \alpha. \end{aligned} \tag{4.42}$$

The last set of equations makes explicit one description of the friction cone. Specifically, its edges are given by the two vectors $\Delta_n \mathbf{n} \pm \mathbf{v}_f$, where $\mathbf{v}_f$ is parallel to the vector $\mathbf{t}_s + \frac{1}{\rho}\mathbf{t}_s \times \mathbf{r}$, that is, $\mathbf{v}_f = \pm(n_y, -n_x, \frac{1}{\rho}(n_x r_x + n_y r_y))$, and $\Delta_n = \left(1 + \frac{n_q^2}{\rho^2}\right)^{\frac{1}{2}}$ is the normalizing factor used in the definition of $\mathbf{n}$ (see Eq. (4.39)).

Notice the mathematical substantiation of the intuitive description of the friction cone in Ch. 2. The configuration space normal models the direction of a real space normal reaction force and its induced reaction torque. This is clear from the cross product term in the angular component n_q of the normal (see Eq. (4.40)). Similarly, the edges of the friction cone model the direction of a real space tangential reaction force and its associated induced torque. This is apparent from the cross product in the angular component of the vector $\mathbf{v}_f$.

4.4.3. Relative Motions of the Reference Point and the Point of Contact

This subsection reviews some of the equations of motion that relate the motion of a point of contact to the motion of an object's reference point. The motion of the contact point determines the real space interaction of the moving object with constraints in its environment, while the motion of the reference point and orientation of the object represent this same interaction in configuration space. Studying the relationship between the two representations of motions builds intuition about configuration space. Additionally, the results of this subsection will be used later in computing reaction forces.

Denote by $\mathbf{v}_{xy}$ the real space velocity of the reference point, and by ω the angular velocity of the object about its reference point. Let $\mathbf{v}_0$ be the velocity of the contact point. Let $\mathbf{a}_{xy}$, α, and $\mathbf{a}_0$ have similar meanings for acceleration. Note that the configuration space representation of velocity is just $\mathbf{v} = (v_x, v_y, \rho\omega)$, where $\mathbf{v}_{xy} = (v_x, v_y)$. Similarly for acceleration. Let the vector from the contact point to the reference point be $\mathbf{r}$, as usual.

It is convenient to think of all the 2-vectors above as vectors in 3-space, with zero third components. The angular quantities ω and α can then be thought of as 3-vectors, with first two components zero. When considered as vectors, ω and α will be written as $\boldsymbol{\omega}$ and $\boldsymbol{\alpha}$, respectively. This is basically the configuration space representation. Note that the velocity at the contact point is just the velocity of the reference point with an adjustment for rotation. Thus

$$\mathbf{v}_{xy} = \mathbf{v}_0 + (\boldsymbol{\omega} \times \mathbf{r}) \tag{4.43}$$

Differentiating, while keeping in mind that the radius vector $\mathbf{r}$ need not be constant, the relationship between accelerations is simply

$$\mathbf{a}_{xy} = \mathbf{a}_0 + (\boldsymbol{\alpha} \times \mathbf{r}) + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}) \tag{4.44}$$

The third term is a centripetal term that appears because of the acceleration that occurs along $\mathbf{r}$ as a result of angular velocity. The previous analysis on the peg-in-hole problem ignored this term by assuming static or quasi-static behavior.

4.4.3.1. Sliding and Normal Contact Velocities and Accelerations

Contact sliding and normal velocities relate very simply to configuration space tangential and normal velocities. Specifically, using Eq. (4.43),

$$\mathbf{v}_0 \cdot \mathbf{t}_s = \mathbf{v} \cdot \mathbf{v}_f, \tag{4.45}$$

$$\mathbf{v}_0 \cdot \mathbf{n}_0 = \Delta_n \mathbf{v} \cdot \mathbf{n}, \tag{4.46}$$

where $\Delta_n = \left(1 + \frac{n_j^2}{\rho^2}\right)^{\frac{1}{2}}$ is the normalizing factor used in the definition of $\mathbf{n}$ (see Eq. (4.39)).

Thus the contact interaction is immediate from the configuration space velocities. Unfortunately, a similar relationship does not hold for accelerations, unless the angular velocity ω is zero, due to the centripetal term. One can, however, add a fictitious acceleration $-\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r})$ to the configuration space acceleration. Having done so, the contact accelerations may be recovered analogously to the contact velocities. Namely,

$$\mathbf{a}_0 \cdot \mathbf{t}_s = \left(\mathbf{a} + \omega^2 \mathbf{r}\right) \cdot \mathbf{v}_f, \tag{4.47}$$

$$\mathbf{a}_0 \cdot \mathbf{n}_0 = \Delta_n \left(\mathbf{a} + \omega^2 \mathbf{r}\right) \cdot \mathbf{n}, \tag{4.48}$$

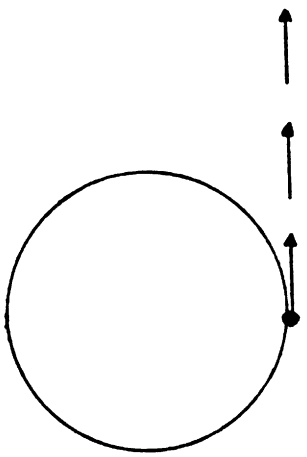


Figure 4.9. Without acceleration normal to the circle, the point will simply move in a straight line.

4.4.3.2. Contact Conditions

In order to maintain contact with a surface, it is necessary to restrict the range of velocities and accelerations. In particular, in order to remain on a surface of contact it must be the case that the object's velocity is tangential to the surface. This statement of constraint applies equally to real space and configuration space. In real space the contact velocity must be tangential to the real space surface, that is, the velocity normal to the surface must be zero. Similarly, in configuration space the reference point must be moving tangentially to the configuration space surface. Its normal velocity must be zero. The real and configuration space conditions are, of course, equivalent by Eqs. (4.45) and (4.46). Thus, in order to maintain contact with a surface, the following condition must hold.

$$\mathbf{v} \cdot \mathbf{n} = 0 \tag{4.49}$$

The same condition does not apply to accelerations. It is not the case that either the normal contact acceleration or the normal configuration space acceleration must be constrained to be zero. The peg-in-hole analysis of the previous section did in fact assume that the contact acceleration was zero. This was possible only because of the quasi-static assumption of that analysis.

The difficulty with assuming that the normal acceleration is zero does not lie with the centripetal term mentioned above. Rather, it is fundamentally inaccurate to assume that the normal acceleration is zero. To see this, suppose that a point is moving on a circle, and suppose that it has a non-zero velocity tangential to the circle, as in Fig. 4.9. If the normal acceleration were zero, then the point would not change its velocity. Thus it would move in a straight line, rather than in a circle.

The correct acceleration constraint is derived by differentiating Eq. (4.49). If the configuration space normal is constant, then the condition reduces to assuming

that the normal configuration space acceleration be zero. Otherwise, an extra term appears. Note, by Eq. (4.44), even when the normal configuration space acceleration is zero, it is not necessary that the real space normal acceleration be zero. The next subsection derives the correct constraints.

4.4.3.3. General Second Variation Constraint

In general, suppose that a surface in some configuration space of n parameters $x_1, \ldots, x_n$ is represented by the implicit equation[2]

$$F(x_1, \ldots, x_n) = 0. \tag{4.50}$$

Then the first and second variation constraints are given by

$$\frac{dF}{dt} = 0 \quad \text{and} \quad \frac{d^2F}{dt^2} = 0. \tag{4.51}$$

Intuitively, these constraints say that any curve on the surface given by F cannot leave that surface.

The first variation constraint reduces to

$$\sum_{i=1}^{n} \frac{\partial F}{\partial x_i} \frac{dx_i}{dt} = 0. \tag{4.52}$$

This is of course just the same as the velocity constraint Eq. (4.49), where the configuration space normal $\mathbf{n}$ is parallel to $(\partial F/\partial x_i, \ldots, \partial F/\partial x_n)$.

The second variation constraint is the derivative of the first variation constraint. So

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 F}{\partial x_j \partial x_i} \frac{dx_j}{dt} \frac{dx_i}{dt} + \sum_{i=1}^{n} \frac{\partial F}{\partial x_i} \frac{d^2 x_i}{dt^2} = 0. \tag{4.53}$$

The last term in the sum on the left is the dot product of a vector parallel to the configuration space normal, with the configuration space acceleration. For further details see Jellet (1872).

4.4.3.4. Second Variation Constraint For Type B Surfaces

The next two subsections derive the acceleration constraints for surfaces that appear in the three dimensional configuration space arising from the interaction of a polygonal object with polygonal obstacles. The derivation is formulated in terms of the parameters x, y, and θ. The conversion to x, y, and q parameters is straightforward (recall that $q = \rho\theta$).

There are two basic types of surfaces that arise. One arises from the interaction of an edge of the moving object and a vertex of a stationary obstacle, while the

[2]For multiple contact there would be several such surface constraints, corresponding to the intersection of surfaces in configuration space.

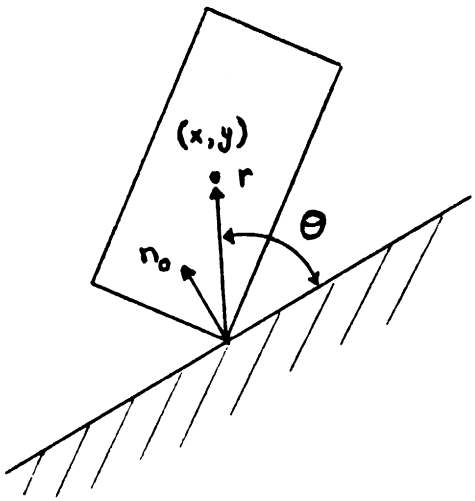


Figure 4.10. Type B contact. A vertex of the moving object is in contact with an edge of an obstacle.

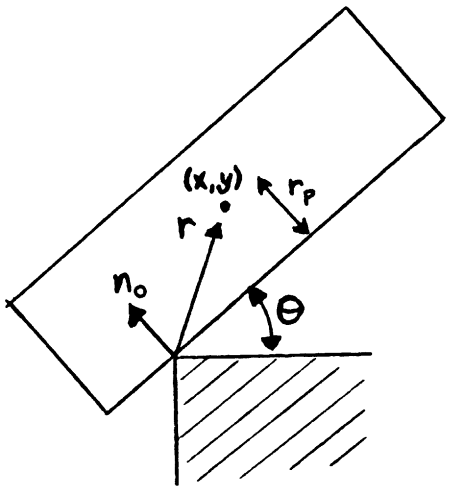


Figure 4.11. Type A contact. An edge of the moving object is in contact with a vertex of an obstacle.

---

other arises from the interaction of a vertex of the moving object and an edge of a stationary obstacle (see Fig. 4.6). In the terminology of Lozano-Pérez (1983), these are, respectively, Type A and Type B configuration space surfaces.

This subsection considers the simpler Type B surfaces. Consider Fig. 4.10. Assume, for convenience, a suitably chosen coordinate system, in which the point of contact is initially at the origin, the reference point is initially at the point (x, y), and the angle between the edge and the vector $\mathbf{r}$ is θ. The real space normal is given by (n_x, n_y), and the length of the radius vector $\mathbf{r}$ by the constant scalar r. Since it is constant, one would expect the second variation constraint to be equivalent to the condition that the normal contact acceleration be zero. This will in fact be the case.

The configuration space surface is given by

$$F(x,\, y,\, \theta) \;=\; 0, \tag{4.54}$$

where

$$F(x,\, y,\, \theta) \;=\; n_x x + n_y y - r \sin\theta. \tag{4.55}$$

The second variation partial derivatives are

$$\begin{array}{lll} \frac{\partial F}{\partial x} = n_x & \frac{\partial F}{\partial y} = n_y & \frac{\partial F}{\partial \theta} = -r\cos\theta \\ \frac{\partial^2 F}{\partial x^2} = 0 & \frac{\partial^2 F}{\partial y^2} = 0 & \frac{\partial^2 F}{\partial \theta^2} = r\sin\theta \\ \frac{\partial^2 F}{\partial x \partial y} = 0 & \frac{\partial^2 F}{\partial x \partial \theta} = 0 & \frac{\partial^2 F}{\partial y \partial \theta} = 0 \end{array} \tag{4.56}$$

A bit of fiddling shows that the second variation constraint reduces to

$$\mathbf{n} \cdot \left(\mathbf{a} + \omega^2\, \mathbf{r}\right) \;=\; 0, \tag{4.57}$$

where $\mathbf{a}$ is the configuration space acceleration.

Note that this is indeed equivalent to $\mathbf{n}_0 \cdot \mathbf{a}_0 = 0$, by Eq. (4.48).

4.4.3.5. Second Variation Constraint For Type A Surfaces

This subsection derives the second variation constraint for Type A surfaces. Consider Fig. 4.11, which depicts an edge in contact with a vertex. Again, assume a suitably chosen coordinate system, as shown. The complications with this type of surface are that the real space normal is no longer constant, and that the vector from the point of contact to the reference point can change in length. As a result, one expects that the second variation constraint is not equivalent to assuming zero normal contact velocity.

Let r_p be the perpendicular distance of the reference point from the contact point, relative to the contact edge. This scalar does not change. As a result, the configuration space surface is given by

$$F(x,\, y,\, \theta) \;=\; 0, \tag{4.58}$$

where

$$F(x,\, y,\, \theta) \;=\; -x\sin\theta + y\cos\theta - r_p. \tag{4.59}$$

Taking derivatives, one gets

$$\begin{array}{lll} \frac{\partial F}{\partial x} = -\sin\theta & \frac{\partial F}{\partial y} = \cos\theta & \frac{\partial F}{\partial \theta} = -x\cos\theta - y\sin\theta \\ \frac{\partial^2 F}{\partial x^2} = 0 & \frac{\partial^2 F}{\partial y^2} = 0 & \frac{\partial^2 F}{\partial \theta^2} = x\sin\theta - y\cos\theta \\ \frac{\partial^2 F}{\partial x \partial y} = 0 & \frac{\partial^2 F}{\partial x \partial \theta} = -\cos\theta & \frac{\partial^2 F}{\partial y \partial \theta} = -\sin\theta \end{array} \tag{4.60}$$

Again, a bit of fiddling shows that the second variation constraint reduces to

$$\mathbf{n} \cdot \left(\mathbf{a} - \omega^2\, \mathbf{r} - 2\,\omega \times \mathbf{v}_{xy}\right) \;=\; 0. \tag{4.61}$$

4.4.4. Reaction Forces

The section on peg-in-hole assembly discussed the computation of reaction forces under static or quasi-static assumptions. This subsection generalizes the results to the computation of reaction forces under dynamic conditions. For simplicity it is assumed that the configuration space friction cone lies wholly above its tangent plane. A later subsection deals with the ambiguities that arise when the friction cone dips below the tangent plane.

4.4.4.1. Fictitious Accelerations and Forces

In the quasi-static case the second variation constraint reduces to the constraint that the normal acceleration be zero. Thus, in order to compute a reaction force, one merely projects the negative applied force normally onto the friction cone. The net force is guaranteed to yield an acceleration with zero normal component. In the dynamic case the second variation constraint is of the form

$$\mathbf{n} \cdot (\mathbf{a} + \mathbf{h}) = 0, \tag{4.62}$$

where $\mathbf{h}$ is some appropriate vector function of x, y, θ, and their time derivatives.

Now define $(\mathbf{h} \cdot \mathbf{n})\,\mathbf{n}$ to be a fictitious configuration space acceleration. This of course also defines a fictitious force. One can add the fictitious force to the actual applied force, thinking of the resulting sum as the effective applied force. Relative to this effective applied force, acceleration is given by $\mathbf{a}' = \mathbf{a} + (\mathbf{h} \cdot \mathbf{n})\,\mathbf{n}$.

In other words, one has transformed the equations of motion from

$$\mathbf{F}_{applied} + \mathbf{F}_{reaction} = \mathbf{M}\,\mathbf{a} \tag{4.63}$$

into

$$\mathbf{F}'_{applied} + \mathbf{F}_{reaction} = \mathbf{M}\,\mathbf{a}', \tag{4.64}$$

where

$$\mathbf{F}'_{applied} = \mathbf{F}_{applied} + \mathbf{M}(\mathbf{h} \cdot \mathbf{n})\,\mathbf{n}, \qquad \mathbf{a}' = \mathbf{a} + (\mathbf{h} \cdot \mathbf{n})\,\mathbf{n}. \tag{4.65}$$

In the previous equations, all forces are generalized (configuration space) forces, and all accelerations are generalized accelerations. Additionally, $\mathbf{M}$ is a generalized mass matrix, combining both the mass and moment of inertia of the moving object. For the (x, y, q) configuration space of a planar object with two translational and one rotational degrees of freedom, $\mathbf{M}$ is in fact just $m\,\mathbf{I}$, where m is the usual mass of the moving object, and $\mathbf{I}$ is the 3×3 identity matrix. This is because $q = \rho\theta$ already models the object's moment of inertia. For more general configuration spaces $\mathbf{M}$ need not have such a simple structure. See, for example, Sec. 4.6.1, which describes the six dimensional configuration space of a rigid object with three translational and three rotational degrees of freedom.

The constraint on the primed acceleration is simply

$$\mathbf{n} \cdot \mathbf{a}' = 0. \tag{4.66}$$

Thus, relative to the fictitious applied force, the constraint on acceleration is equivalent to the quasi-static constraint of zero normal acceleration. In order to compute a reaction force, it is therefore sufficient to project the negative of the effective applied force, found by adding the fictitious force to the actual applied force, normally onto the friction cone.

4.4.4.2. Sliding at the Contact Point

The normal projection onto the friction cone is a valid technique when the sliding contact velocity is zero. In general, one must project the effective applied force normally into the plane of the cone, and then decide whether the reaction force lies in the interior or on the edges of the friction cone. By Coulomb's law, if the sliding contact velocity is non-zero, then the reaction force must lie on one or the other of the friction cone edges.

Additionally, one may wish to consider the sliding contact acceleration, should the sliding contact velocity be zero. Only if the sliding contact acceleration that would result from a particular reaction force is zero, does one actually choose a reaction force from the friction cone's interior. The use of the sliding contact acceleration is similar to the assumption of impending motion in quasi-static situations.

4.4.4.3. Explicit Computation of Reaction Forces

The computation of the reaction force given an applied force, based on the techniques outlined above, is presented here. Only the interesting case in which the tangential sliding velocity is zero, is considered. The other cases follow easily. It is assumed that M is a diagonal matrix with positive entries on the diagonal.

Assume that the reaction force is of the form $\mathbf{F}_{reaction} = f_n \mathbf{n} + f_t \mathbf{t}_r^{\perp}$, and that the vector $\mathbf{v}_f$ is given by $\mathbf{v}_f = v_n \mathbf{n} + v_t \mathbf{t}_r^{\perp}$, with $v_t > 0$. The reaction force is constrained by

$$\frac{1}{k_2} f_n \leq f_t \leq \frac{1}{k_1} f_n, \tag{4.67}$$

with $f_n > 0$. Also, $k_1 > 0$, and $k_2 < 0$ by the assumption that the friction cone lies above the tangent plane.[3] By the remarks of Sec. 4.4.3.1,

$$\mathbf{a}_0 \cdot \mathbf{t}_s = \left(\mathbf{a} + \omega^2 \mathbf{r}\right) \cdot \mathbf{v}_f. \tag{4.47}$$

[3]Similar results apply when the friction cone dips below the tangent plane.

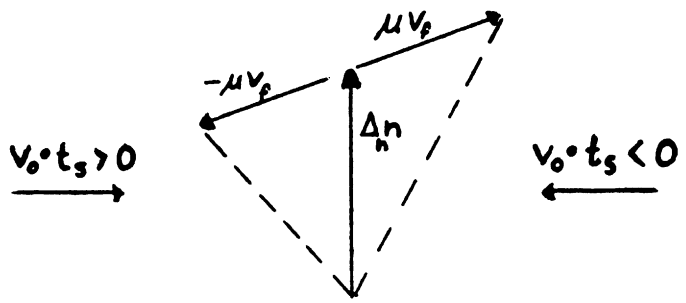


Figure 4.12. The configuration space friction cone is formed from the normal n by adding and subtracting the vector μv_f. This vector describes the frictional reaction force and torque.

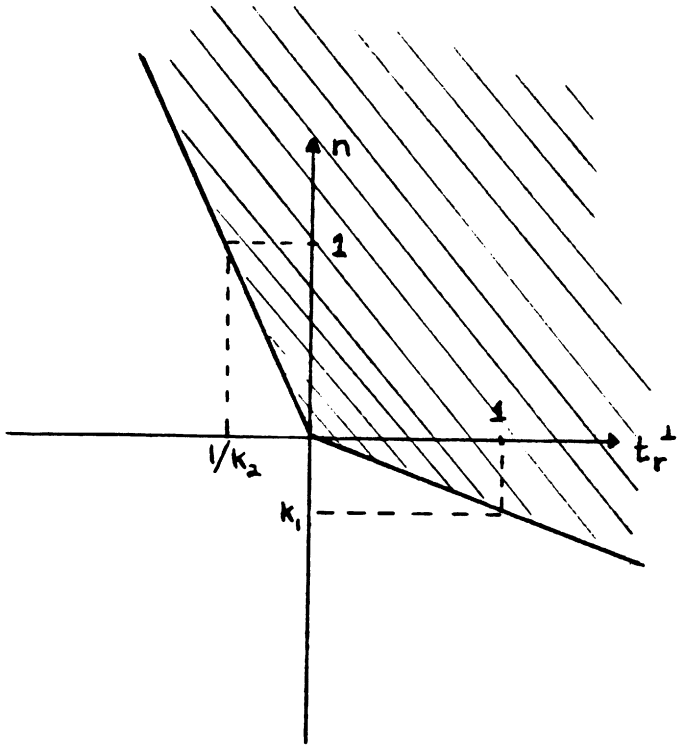


Figure 4.13. The configuration space friction cone can dip below the tangent plane. This is determined by the sign of the parameter k_1.

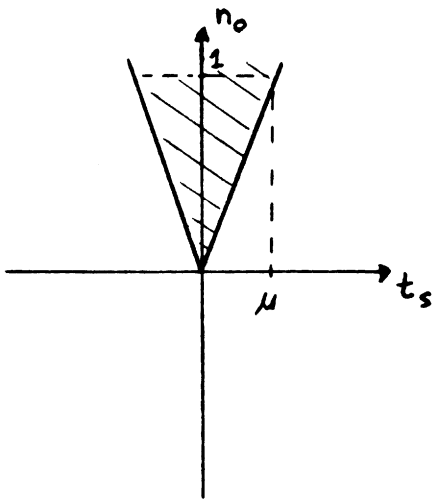


Figure 4.14. Real space friction cone.

Writing the net force as

$$\mathbf{F}_{net} = \mathbf{F}_{applied} + \mathbf{F}_{reaction}, \tag{4.68}$$

and noting that the second variation constraint (Eq. (4.62)) says that

$$\begin{aligned} \mathbf{F}_{net} \cdot \mathbf{n} &= \mathbf{M}\mathbf{a} \cdot \mathbf{n} \\ &= -\mathbf{M}\mathbf{h} \cdot \mathbf{n}, \end{aligned} \tag{4.69}$$

it follows that

$$\mathbf{M}\mathbf{a}_0 \cdot \mathbf{t}_s = f_t v_t + C, \tag{4.70}$$

where

$$C = v_t \left(\mathbf{F}_{applied} \cdot \mathbf{t}_r^{\perp}\right) - v_n \left(\mathbf{M}\mathbf{h} \cdot \mathbf{n}\right) + \omega^2 \left(\mathbf{M}\mathbf{r} \cdot \mathbf{v}_f\right). \tag{4.71}$$

Now, if $f_t > -C/v_t$, then the sliding contact acceleration is positive, which means that the reaction force should be on the "negative" friction cone edge, given by $\Delta_n \mathbf{n} - \mu \mathbf{v}_f$ (see Fig. 4.12). On the other hand, if $f_t < -C/v_t$, then the sliding contact acceleration is negative, so the reaction force should be on the "positive" friction cone edge, given by $\Delta_n \mathbf{n} + \mu \mathbf{v}_f$. Otherwise, the reaction force is in the interior of the friction cone.

In summary, the normal component of the reaction force, f_n, is determined by projecting the negative effective applied force normally onto the plane of the friction cone. The tangential component of the reaction force, f_t, is then determined from the value C above, using the following table. The table is ordered hierarchically.

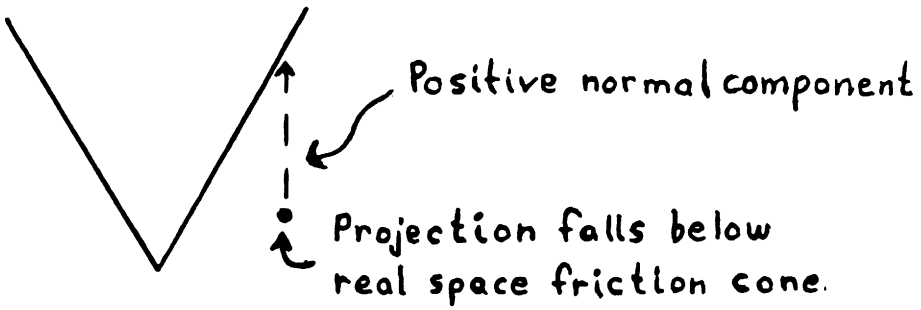


Figure 4.15. A reaction force projection that lies below the real space friction cone. A positive normal component must be added in order to reach the friction cone edge.

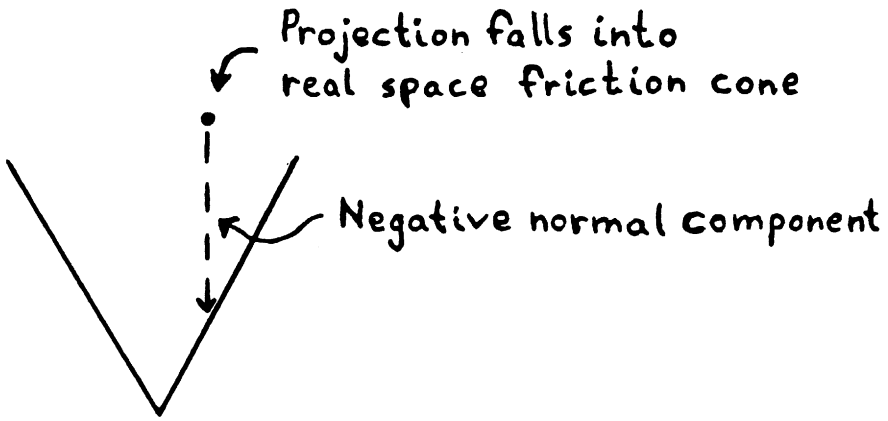


Figure 4.16. A reaction force projection that lies within the real space friction cone. A negative normal component must be added in order to reach the friction cone edge.

$$f_t = \begin{cases} \frac{1}{k_2} f_n, & \text{if } \mathbf{v}_0 \cdot \mathbf{t}_s > 0 \\ \frac{1}{k_1} f_n, & \text{if } \mathbf{v}_0 \cdot \mathbf{t}_s < 0 \\ \frac{1}{k_2} f_n, & \text{if } -\frac{C}{v_t} \leq \frac{1}{k_2} f_n \\ \frac{1}{k_1} f_n, & \text{if } -\frac{C}{v_t} \geq \frac{1}{k_1} f_n \\ -\frac{C}{v_t}, & \text{if } \frac{1}{k_2} f_n \leq -\frac{C}{v_t} \leq \frac{1}{k_1} f_n \end{cases} \tag{4.72}$$

4.4.5. Motion Ambiguities

The previous discussion has assumed that the configuration space friction cone does not dip below the configuration space tangent plane. This subsection relaxes that assumption.

Suppose that the friction cone dips below the tangent plane, as in Fig. 4.13. An applied force which points into the portion of friction cone below the tangent

plane can have two effects. Since the friction cone can provide a reaction force that balances the applied force, it is possible for static equilibrium to occur. On the other hand, it is also possible for the friction cone to offer no resistance, since the applied force has a positive normal component. The resulting motion would move away from the surface.

Both motions are valid solutions of the equations of motion. The first solution arises under the assumption that the second variation constraint must hold, while the second solution is a result of assuming zero reaction force.

4.4.5.1. Condition under which the Friction Cone Dips below the Tangent Plane

The shape of the friction cone is characterized by the two parameters k_1 and k_2 (see Eqs. (4.37) and (4.38)). By the assumption on the signs of the dot products of the vectors $\mathbf{n}_0$, $\mathbf{t}_r^\perp$, and $\mathbf{t}_s$ (see Sec. 4.4.1), the parameter k_2 is always negative. Thus only one side of the friction cone can ever dip below the tangent plane. This occurs for negative values of k_1 (see Fig. 4.13).

Consequently, whether the friction cone dips below the tangent plane is determined by the sign of the quantity

$$\mathbf{t}_r^\perp \cdot \mathbf{t}_s \; - \; \mu \, \mathbf{t}_r^\perp \cdot \mathbf{n}_0, \tag{4.73}$$

which is

$$\mathbf{t}_r^\perp \cdot (\mathbf{t}_s \; - \; \mu \, \mathbf{n}_0) \tag{4.74}$$

Each of the individual terms in the difference (4.73) is assumed to be positive. If the quantity is positive, then the configuration space friction cone lies above the tangent plane. If the quantity is negative, then the configuration space friction cone dips below the tangent plane.

Intuitively, consider what the quantity (4.74) measures. Recall that the real space component of any reaction force must lie within the real space friction cone. If the force has a positive component along the sliding axis, that is, $\mathbf{F}_{reaction} \cdot \mathbf{t}_s \geq 0$, then, in terms of Fig. 4.14, the normal component must lie above the line defined by

$$\mathbf{F}_{reaction} \cdot \mathbf{t}_s = \mu \, \mathbf{F}_{reaction} \cdot \mathbf{n}_0. \tag{4.75}$$

Now suppose that a configuration space reaction force is positively parallel to the tangent $\mathbf{t}_r^\perp$ which defines the configuration space friction cone. The sign of the quantity (4.74) specifies whether the real space component of this force lies interior or exterior to the real space friction cone. If the sign of this quantity is negative, then the real space component of the force lies above the line defined by (4.75). If the sign is positive, then the force lies below the line.

Now consider a reaction force on the configuration space friction cone's "positive" edge (see Fig. 4.12), that is, the force has a positive $\mathbf{t}_r^\perp$ component. Such

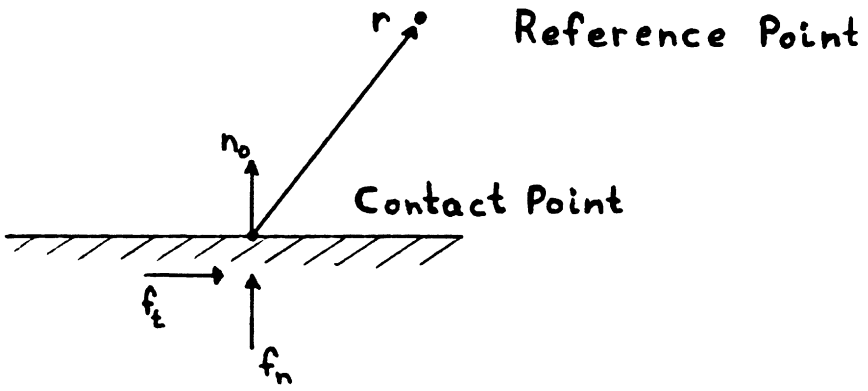


Figure 4.17. Contact normal, contact reaction forces, and radius vector, of some moving object in contact with an horizontal edge.

---

a reaction force has both a tangential component parallel to $t_r^{\perp}$, and a normal component parallel to the configuration space normal n. Since the configuration space reaction force is on an edge of the configuration space friction cone, its real space component must be on an edge of the real space friction cone, by construction. This fact allows one to deduce the sign of the normal component from the sign of quantity (4.74).

If quantity (4.74) is positive, then the tangential component of the reaction force projects below the real space friction cone. Consequently, the normal component of the configuration space reaction force must be positive, in order for the reaction force to lie on the real space friction cone's edge (see Fig. 4.15). This observation is equivalent to saying that the configuration space friction cone lies above the tangent plane. Similarly, if quantity (4.74) is negative, then the normal component must be negative, which says that the configuration space friction cone dips below the tangent plane (see Fig. 4.16).

The difference between t_s and $t_r^{\perp}$ is essentially the addition of a reaction torque. Thus the quantity (4.74) basically measures the difference between the configuration space normal reaction force and torque, on the one hand, and the induced reaction torque arising from the tangential real space frictional reaction force, on the other hand. If the component of the frictional torque along the configuration space normal is large enough, then the friction cone will dip below the tangent plane.

4.4.5.2. An Example

Consider the example of Fig. 4.17, which depicts some moving object in contact with an horizontal edge. The vector from the point of contact to the reference point is given by $\mathbf{r} = (r_x, r_y)$. Throughout this example both r_x and r_y are assumed to be positive. A simple calculation shows that the sign of quantity (4.74) is the same

as the sign of

$$\rho^2 + r_x^2 - \mu r_x r_y. \tag{4.76}$$

To see that this quantity really just measures the difference between the normal reaction force and torque, and the induced frictional reaction torque, consider the real space reaction forces. Let the normal real space reaction force be f_n, and the tangential reaction force be f_t. These are scalar magnitudes associated with the vectors shown in Fig. 4.17. The configuration space normal is parallel to the vector

$$\left(0,\ 1,\ -\frac{1}{\rho} r_x\right). \tag{4.77}$$

In fact, the configuration space normal reaction force is simply

$$f_n\left(0,\ 1,\ -\frac{1}{\rho} r_x\right). \tag{4.78}$$

The configuration space frictional reaction force is

$$f_t\left(1,\ 0,\ \frac{1}{\rho} r_y\right). \tag{4.79}$$

So the net normal component of the reaction force is found by taking a dot product of the normal with the vector

$$\left(f_t,\ f_n,\ \frac{1}{\rho}\left(f_t r_y - f_n r_x\right)\right). \tag{4.80}$$

The sign of this dot product is the same as the sign of the dot product of the vectors (4.77) and (4.80). In other words the sign is given by the sign of

$$f_n + \frac{1}{\rho^2}\left(f_n r_x^2 - f_t r_x r_y\right). \tag{4.81}$$

At the positive cone extreme $f_t = \mu f_n$, with $f_n > 0$. Therefore, letting $f_n = 1$, and $f_t = \mu$, the quantity (4.81) is immediately seen to be equivalent to the quantity (4.76).

4.4.5.3. A Numerical Example

The previous example yielded a friction cone that dipped below the tangent plane whenever the quantity (4.76) was negative. That quantity is negative when $r_x r_y$ is large in comparison to r_x^2 and ρ^2. Thus the friction cone dips below the tangent plane when the object's reference point is far away from the contact point, and is almost directly, but not quite, above the contact point.

For a numerical example, let $\mu = 1/4$, $\rho = 1$, $r_x = 1/8$, and $r_y = 35$. Then $\rho^2 + r_x^2 - \mu r_x r_y < 0$.

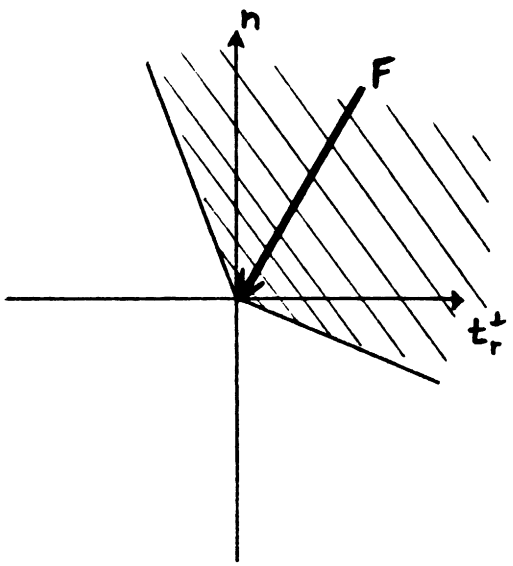


Figure 4.18. An applied force pointing into a configuration space surface and into a friction cone that dips below the surface tangent plane.

4.4.5.4. Inadequacy of Motion Equations with Friction

The following subsection demonstrates the inadequacy of classical mechanics in the presence of friction. Specifically, when the configuration space friction cone dips below the tangent plane, it is possible that the equations of motion are inconsistent under the standard assumptions about friction. See also Jellet (1872).

Suppose the configuration space friction cone dips below the tangent plane. Now apply a force which points into the friction cone and points into the configuration space surface, as in Fig. 4.18. Under static conditions, there is no difficulty in predicting a reaction force that completely cancels the applied force. However, suppose that the object is actually sliding in the negative $\mathbf{t}_r^\perp$ and $\mathbf{t}_s$ directions. According to the classical view of friction, the reaction force must lie on the friction cone edge opposing the direction of motion.

In real space, it appears that one can always tangentially project a reaction force onto the appropriate edge of the friction cone, while keeping the normal component of the reaction force fixed. It is apparent from the figure, that this is not possible in configuration space. In fact, it turns out that the equations of motion in real space may be inconsistent when the configuration space friction cone dips below the tangent plane. Thus, the assumption that the reaction force must lie on the edge of the real space friction cone is incorrect.

To see that the equations of motion may be inconsistent, consider the example

of the previous section. Assume that the object is not rotating, but is sliding towards the left. The assumption that the object is not rotating means that the following relationship holds between the acceleration (a_x, a_y) of the reference point, the acceleration (a_{0x}, a_{0y}) of the contact point, and the angular acceleration α about the reference point (see Sec. 4.4.3 and Eq. (4.44)).

$$\begin{aligned} a_x &= a_{0x} - \alpha r_y \\ a_y &= a_{0y} + \alpha r_x \end{aligned} \tag{4.82}$$

Let the applied force at the reference point be given by (F_x, F_y), and the applied torque around the reference point by τ. The normal reaction force is f_n, as in Fig. 4.17. Then the equations of motion are

$$F_x + \mu f_n = m a_{0x} - m \alpha r_y \tag{4.83}$$

$$F_y + f_n = m a_{0y} + m \alpha r_x \tag{4.84}$$

$$\tau + \mu f_n r_y - f_n r_x = m \rho^2 \alpha \tag{4.85}$$

Rearranging produces the relation

$$f_n \left(\rho^2 + r_x^2 - \mu r_x r_y \right) = -\left(F_y \rho^2 - \tau r_x \right) + m \rho^2 a_{0y}. \tag{4.86}$$

Since the applied force is assumed to be pointing into the configuration space surface, and since the surface normal is parallel to the vector (4.77), the quantity $F_y \rho^2 - \tau r_x$ is negative. The second variation constraint says that a_{0y} must be zero in order to maintain contact, and non-negative in order not to violate the surface constraint. Thus the right hand side of Eq. (4.86) is positive.

The real space reaction force f_n must also be non-negative. By assumption the quantity $\rho^2 + r_x^2 - \mu r_x r_y$ is negative. Thus the left hand side of Eq. (4.86) is negative. This is a contradiction, showing that the equations are inconsistent.

It is unclear how to resolve this situation. There are two promising possibilities. One is to assume that the sliding velocity instantaneously goes to zero, should the above scenario arise. This amounts to assuming some kind of collision-like interaction in which all energy is transferred to rotation. The second possibility is simply to dispense with the assumption that the reaction force must lie on the edge of the friction cone. It was this assumption that generated the previous contradiction.

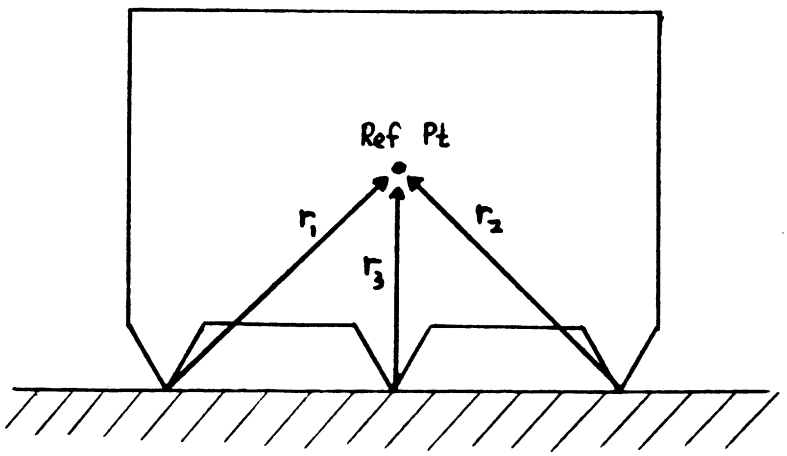


Figure 4.19. Three-point contact. Two principal motions involve sliding on the surface and rotating at the extreme contact points.

4.5. Multiple Points of Contact

During assembly operations an object is seldom in contact with only one other object. Assembly operations in the plane ultimately involve a minimum of two contact points. The classical peg-in-hole example quickly encounters two-point contact, as small angular misalignments cause it to make contact with both sides of the hole. In higher dimensions, the number of independent contacts increases directly with the dimension of the space. Furthermore, since the contacts are usually only one-sided constraints, it is possible to have more contacts than the dimension of the space, while still retaining mobility. Finally, it is possible to model the contacts of pairs of surfaces, such as the contact of a face of one object with the face of another object, as the intersection of several one-point contacts.

During multiple point contact the object is more constrained. Additionally, the range of possible frictional reaction forces increases. In the plane, for example, the configuration space friction cone of a translating and rotating object is a two dimensional subset of force space. Thus it is fairly easy to avoid sticking. As the number of contacts increases, the frictional reaction forces quickly form a three dimensional subset of force space. Thus the avoidance of sticking surfaces is more difficult. In order to predict the possible range of reaction forces, it is necessary to extend the model of the configuration space friction cone to include multiple points of contact.

4.5.1. Equations of Motion on the Intersection of Surfaces

Multiple contact in real space corresponds to the intersection of surfaces in configuration space. For each point of contact in real space, there is an associated

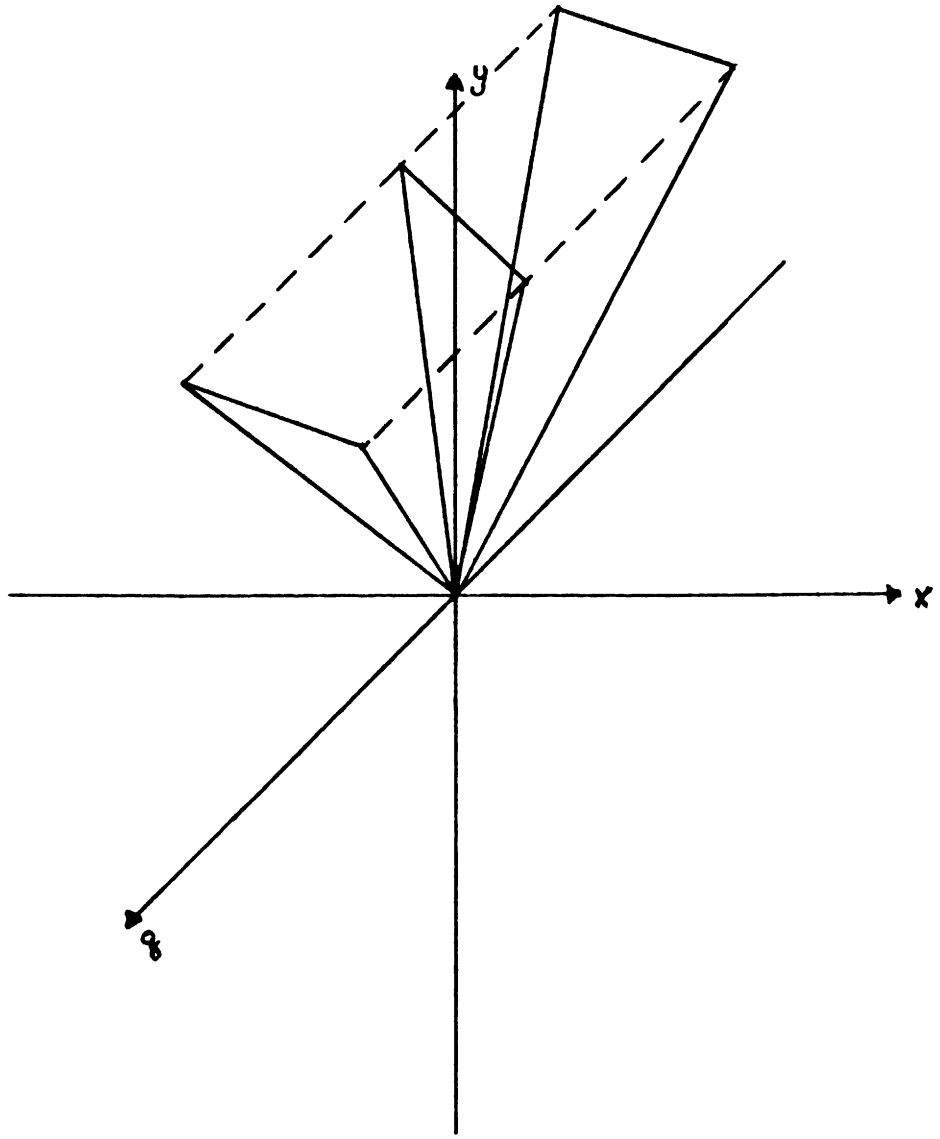


Figure 4.20. Friction cones for the example of Fig. 4.19. The three individual friction cones are described by the triangular sheets. The composite friction cone comprises the volume between the individual friction cones, as indicated by the dashed lines.

---

hyper-surface in the moving object's configuration space. The surface represents the constraint on the object's degrees of freedom imposed by the point of contact. Several points of contact impose several constraints on the degrees of freedom. The constraints are satisfied along the surface that corresponds to the intersection of all the individual hyper-surfaces.

Each surface has associated configuration space normal and tangent spaces. These permit definition of the usual one-point friction cones. Suppose that there are k points of contact. Define for each the usual vector decomposition. In particular,

let $\mathbf{n}_{0i} = (n_{xi}, n_{yi})$ be the real space normal at contact point i, and let $\mathbf{r}_i$ be the radius vector from the ith point of contact to the reference point. Let n_{qi} be formed from the cross product $\mathbf{n}_{0i} \times \mathbf{r}_i$, as in the definition of the configuration space normal $\mathbf{n}_i$, and let v_{qi} be formed from the cross product $\mathbf{t}_{si} \times \mathbf{r}_i$, as in the definition of the vector $\mathbf{v}_{fi}$ which is used to define the edges of the ith friction cone.

Finally, let the real space reaction force at point i be given by f_{ni} in the normal direction, and by f_{ti} in the tangential direction. The equations of motion are then simply

$$\begin{aligned} F_x + \sum_{i=1}^{k} f_{ni}\, n_{xi} + \sum_{i=1}^{k} f_{ti}\, n_{yi} &= m\, a_x \\ F_y + \sum_{i=1}^{k} f_{ni}\, n_{yi} - \sum_{i=1}^{k} f_{ti}\, n_{xi} &= m\, a_y \\ \tau + \sum_{i=1}^{k} f_{ni}\, n_{qi} + \sum_{i=1}^{k} f_{ti}\, v_{qi} &= m\, \rho^2\, \alpha, \end{aligned} \tag{4.87}$$

where

$$0 \leq |f_{ti}| \leq \mu\, f_{ni}, \quad \text{for every } i = 1, \ldots, k. \tag{4.88}$$

From the form of the equations it is apparent that the possible range of reaction forces is the vector sum of the range of reaction forces due to each individual point of contact. This is just the principle of superposition.

Given a collection of contact points, one can define a composite friction cone to be the vector sum of the individual friction cones. Any force within this cone is a potential reaction force. Thus, from a planning viewpoint, motions should be avoided which require applied forces that fall within the composite friction cone.

As an aside, note that it may not always be possible to predict the decomposition of a reaction force over the individual points of contact. This is true both in the static case and the dynamic case. For the static case, the equations of motion (4.87) have zero accelerations on the right side. Suppose that the applied configuration space force $(F_x\ F_y,\ \tau/\rho)$ is known to be pointing into the composite friction cone. One wishes to predict the reaction forces $f_{n1}, \ldots, f_{nk}$ and $f_{t1}, \ldots, f_{tk}$. Thus there are $2k$ unknowns and only three equations (for the planar case).

Consequently, as soon as there is more than one point of contact, the resolution of forces among the contact points is indeterminate. In some cases, one may know that the reaction forces lie on the individual friction cone edges. This reduces the number of unknowns. However, clearly as the number of contact points increases, the static indeterminacy will also increase.

4.5.2. Three-Point Contact Example

Consider the example of Fig. 4.19. The object shown is in three-point contact with an horizontal edge. All contacts are Type B contacts. Since the configuration space normal space is two dimensional, the object is not overconstrained. It can clearly slide horizontally, while maintaining all points of contact. Additionally, by breaking contact at the other points, the object can rotate about either of the extreme contact points. This subsection derives the composite friction cone for this example.

4.5.2.1. The Vector Decomposition

All vectors will be subscripted by the number of the contact point. The left point is number 1; the right, number 2; and the middle, number 3.

All three points have the same real space normal, namely $\mathbf{n}_0 = (0, 1, 0)$. Given the conventions of Sec. 4.4.1, the sliding tangents are

$$\begin{aligned} \mathbf{t}_{s1} &= (1, 0, 0) \\ \mathbf{t}_{s2} &= (-1, 0, 0) \\ \mathbf{t}_{s3} &= (1, 0, 0) \end{aligned} \tag{4.89}$$

The radius vectors are

$$\begin{aligned} \mathbf{r}_1 &= (1, 1, 0) \\ \mathbf{r}_2 &= (-1, 1, 0) \\ \mathbf{r}_3 &= (0, 1, 0) \end{aligned} \tag{4.90}$$

Consequently, the configuration space normal and friction cone defining vectors are

$$\begin{aligned} \Delta_1 \mathbf{n}_1 &= \left(0, 1, -\frac{1}{\rho}\right) \\ \Delta_2 \mathbf{n}_2 &= \left(0, 1, \frac{1}{\rho}\right) \\ \Delta_3 \mathbf{n}_3 &= (0, 1, 0) \end{aligned} \tag{4.91}$$

$$\begin{aligned} \mathbf{v}_{f1} &= \left(1, 0, \frac{1}{\rho}\right) \\ \mathbf{v}_{f2} &= \left(-1, 0, -\frac{1}{\rho}\right) \\ \mathbf{v}_{f3} &= \left(1, 0, \frac{1}{\rho}\right) \end{aligned} \tag{4.92}$$

4.5.2.2. The Friction Cones

Recall that the edges of the friction cone at some point of contact are given by the two vectors $\Delta_n \mathbf{n} \pm \mu \mathbf{v}_f$. Thus the three friction cones for the above points of contact are given by.

$$\begin{array}{lll} \text{Point Number 1:} & (\mu, 1, \frac{1}{\rho}(\mu-1)) & (-\mu, 1, -\frac{1}{\rho}(\mu+1)). \\ \text{Point Number 2:} & (\mu, 1, \frac{1}{\rho}(\mu+1)) & (-\mu, 1, -\frac{1}{\rho}(\mu-1)). \\ \text{Point Number 3:} & (\mu, 1, \frac{1}{\rho}\mu) & (-\mu, 1, -\frac{1}{\rho}\mu). \end{array} \tag{4.93}$$

The friction cones are drawn in Fig. 4.20, using $\rho = 1$. The composite friction cone is simply the volume between these three friction cones. In this case, the middle cone is contained in the vector sum of the two outside cones. Furthermore, the edges of the middle friction cone lie in the planes formed by the edges of the exterior friction cones. In general, this need not be the case, as demonstrated in Sec. 4.5.7.1.

4.5.3. Classes of Contacts

A multiple contact state defines more than a single set of interactions. A point moving on the intersection of a number of configuration space surfaces can certainly remain on that intersection. It can, however, also move to a higher dimensional intersection, by leaving one or more of the surfaces. In the previous example, the object could either slide, thereby maintaining all three points of contact, or it could rotate, thereby breaking two of the points of contact.

This observation suggests that in analyzing a multiple point contact, it is also necessary to analyze all smaller subclasses of contacts. In particular, the application of forces must be studied in the context of all possible contact classes. In the previous three-point example, applying a force along the horizontal sliding direction must cause sliding. This is clear from the composite friction cone. A force with no torque component will simply project onto a face of the composite friction cone. All three friction cones, or at least, the two exterior ones, are involved in this projection.

In contrast, the application of a force with a large torque component will cause rotation about one of the extreme contact points. This also is clear from the composite friction cone. The applied force projects onto one of the exterior friction cones. Neither of the other two friction cones are involved in the computation of the reaction force, indicative of the single point interaction that occurs. In summary, for the three-point example, it is necessary to consider motions resulting both from three-point contact and from one-point contact.

Consider also the two-point contact of a peg-in-hole problem, as in Fig. 4.2.c. The types of interactions possible are

1. Maintain contact at both points, while moving parallel to the common configuration space tangent. This movement entails rotating and sliding.

2. Break contact at both points, moving into free space.

3. Break contact at one of the contact points, while rotating about and/or sliding at the other contact point.

4. No motion.

The task is to predict the behavior that will occur given an arbitrary applied force.

4.5.4. Predicting Reaction Forces

Suppose that one is given a k-point contact, and an applied generalized force. In configuration space this means that the moving point is on the intersection of k hyper-surfaces. This subsection develops a method for predicting the behavior of the point in configuration space. The development continues to assume Newton's laws as the underlying motion dynamics. For generalized damper dynamics the prediction of behavior is slightly simpler. In particular, in order to decide whether sticking can occur at a k-point contact, it is sufficient to intersect the negative velocity cone with the composite friction cone (see Ch. 2). Since the composite friction cone represents the range of possible reaction forces, sticking is possible if and only if this intersection is not empty.

All possible contact classes must be considered in predicting behavior. The motion resulting from each class considered is a possible response to the applied force, if the motion does not violate any constraints. For example, in the three-point contact example of Fig. 4.19, one of the contact subclasses to consider is the one-point contact at the leftmost point. Clearly, while counterclockwise rotations about this point are legal, clockwise rotations are not.

Different contact classes can give rise to different resulting motions. The last subsection of this section presents an example in which the motion resulting from a particular force is indeterminate within a certain range. This realization makes evident once again the inherent ambiguity of classical mechanics in the presence of Coulomb friction. See also Jellet (1872) for comments on static and dynamic indeterminacy.

4.5.4.1. Consistency of Second Variation Constraints

The constraints on reaction forces are simple. They must satisfy the equations of motion subject to the second variation constraints for each of the configuration space surfaces with which there is contact. Of course, the first variation constraint must also be satisfied. In other words, the contact velocity must be tangential to the surface of contact. It is tacitly assumed that this condition holds for every point in the set of contact points.

The equations of motion were given by Eq. (4.87). The second variation constraints are of the form

$$\mathbf{a}_i \cdot \mathbf{n}_i \geq -\mathbf{h}_i \cdot \mathbf{n}_i, \quad i = 1, \ldots, k, \tag{4.94}$$

where $\mathbf{h}_i$ is some appropriate function of position and velocity, determined by Eq. (4.53) (see also Eq. (4.62)). In fact, for polygonal environments, the constraints can always be written as (using previous notation)

$$n_{xi}\, a_x + n_{yi}\, a_y + n_{qi}\, \alpha \;\geq\; -\mathbf{h}_i \cdot \mathbf{n}_{0i}, \quad i = 1, \ldots, k. \tag{4.95}$$

Notice that the constraints were written as inequality constraints. This is because not all of the points of contact need be maintained. For those contacts that are maintained, the associated constraints are of course equality constraints, but for those contacts that are broken, the constraints are inequalities. In order not to violate the surface, the inequalities must be "greater than."

All contact sets, for which a consistent set of equality constraints may be found whose solutions do not violate the inequality constraints, yield valid motion solutions. The consistency of the equality constraints insures that all contacts in the contact set can actually be maintained. The satisfaction of the remaining inequality constraints implies that the resulting motion is a legal one, as no surfaces are being violated.

4.5.4.2. Fictitious Forces and Perpendicular Projections

Once a consistent set of contact points has been found, a reaction force may be calculated in much the same fashion as was done for single point contact. The normal space of the contact set has some dimension, which is equal to the number of independent second variation constraints that are equality constraints. Suppose that the dimension of the normal space is m ($m \leq \min\{3, k\}$ for the planar case). Then the second variation equality constraints can be written as

$$\begin{pmatrix} n_{x1} & n_{y1} & n_{q1} \\ n_{x2} & n_{y2} & n_{q2} \\ \vdots & \vdots & \vdots \\ n_{xm} & n_{ym} & n_{qm} \end{pmatrix} \begin{pmatrix} a_x \\ a_y \\ \alpha \end{pmatrix} = -\begin{pmatrix} \mathbf{h}_1 \cdot \mathbf{n}_{01} \\ \mathbf{h}_2 \cdot \mathbf{n}_{02} \\ \vdots \\ \mathbf{h}_m \cdot \mathbf{n}_{0m} \end{pmatrix} \tag{4.96}$$

Now consider the system of equations

$$\begin{pmatrix} n_{x1} & n_{y1} & n_{q1} \\ n_{x2} & n_{y2} & n_{q2} \\ \vdots & \vdots & \vdots \\ n_{xm} & n_{ym} & n_{qm} \end{pmatrix} \begin{pmatrix} h_x \\ h_y \\ h_q \end{pmatrix} = \begin{pmatrix} \mathbf{h}_1 \cdot \mathbf{n}_{01} \\ \mathbf{h}_2 \cdot \mathbf{n}_{02} \\ \vdots \\ \mathbf{h}_m \cdot \mathbf{n}_{0m} \end{pmatrix} \tag{4.97}$$

Denote by $\mathbf{h}'$ a solution to this system. By consistency there is at least one such solution. Different solutions differ from each other by components that are perpendicular to all the configuration space normals $\mathbf{n}_i$, $i = 1, \ldots, m$.

The solution $\mathbf{h}'$ is a fictitious acceleration, with a corresponding fictitious force $\mathbf{M}\,\mathbf{h}'$. Relative to an effective applied force that is formed from the actual applied

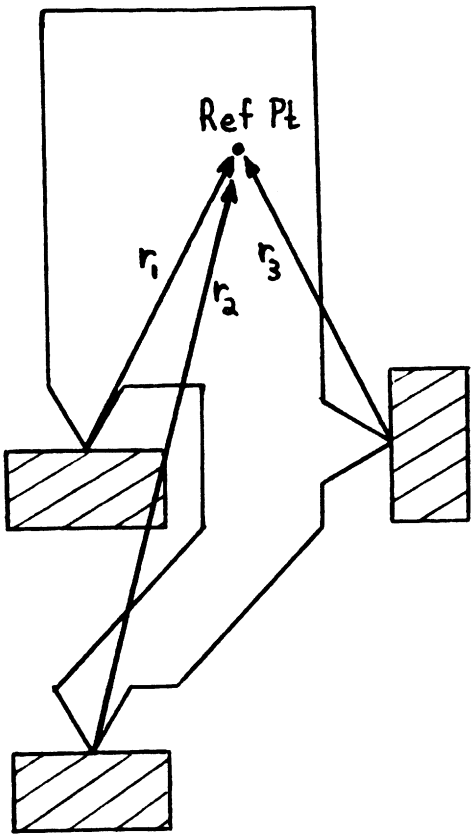


Figure 4.21. Three-point contact example. If the object is rotating, then effectively the contact state is reduced to two-point contact at points 1 and 3.

---

force by adding the fictitious force, the constraints on the acceleration are simply $\mathbf{n}_i \cdot \mathbf{a}'_i = 0$, $i = 1, \ldots, m$. This is equivalent to the old trick for computing reaction forces during one-point contact (see Eqs. (4.63)–(4.66)).

Having added the fictitious force, a reaction force may be computed by projecting the negative effective applied force perpendicular to the normal space of the contact set. This shows that the particular choice of $\mathbf{h}'$ from the set of all possible solutions is irrelevant.

If the projection of the negative effective applied force falls into the composite friction cone, then the reaction force is simply this projection. In particular, if the object is at rest, and the applied force lies inside the friction cone, then there exists a reaction force that completely cancels the applied force.

If the projection falls outside of the composite friction cone, then there does not exist a reaction force. Thus the equations of motion and the second variation constraints are satisfied with zero reaction force. Of course, there may be other contact classes, in particular, contact classes of lower normal space dimension, for which there does exist a reaction force.

Secs. 4.5.6.2 and 4.5.6.3 discuss an algorithm for computing reaction forces in more detail. Note that motion ambiguities may arise when the projection is not uniquely determined. This will become clear in Sec. 4.5.6.2. Sec. 4.5.7 provides an explicit example demonstrating motion ambiguity.

4.5.5. An Example Demonstrating Constraint Inconsistency

The following subsection considers an example in which not all points of contact can be consistently maintained. Specifically, consider the three-point example of Fig. 4.21. Assume that the surfaces are frictionless, for simplicity.

The radius vectors are given by

$$\begin{aligned} \mathbf{r}_1 &= (r_x, r_y, 0), \\ \mathbf{r}_2 &= (r_x, 2r_y, 0), \\ \mathbf{r}_3 &= (-r_x, r_y, 0), \end{aligned} \tag{4.98}$$

with both r_x and r_y positive.

The three configuration space normals are parallel to the vectors

$$\begin{aligned} \Delta_1\mathbf{n}_1 &= \left(0, 1, -\frac{1}{\rho}r_x\right), \\ \Delta_2\mathbf{n}_2 &= \left(0, 1, -\frac{1}{\rho}r_x\right), \\ \Delta_3\mathbf{n}_3 &= \left(-1, 0, -\frac{1}{\rho}r_y\right). \end{aligned} \tag{4.99}$$

Assume that the object is rotating about contact point 1 without sliding. In other words, the configuration space velocity of the reference point is parallel to $(-r_y, r_x, \rho)$. In particular, the angular velocity ω is non-zero. It is a straightforward matter to verify that the contact velocities are all tangential to the contact surfaces. This is also clear from the drawing.

The second variation constraints form the system of equations

$$\begin{pmatrix} 0 & 1 & -r_x \\ 0 & 1 & -r_x \\ -1 & 0 & -r_y \end{pmatrix} \begin{pmatrix} a_x \\ a_y \\ \alpha \end{pmatrix} \geq -\omega^2 \begin{pmatrix} r_y \\ 2r_y \\ r_x \end{pmatrix} \tag{4.100}$$

This system of equation shows that the only consistent equality constraints are formed by assuming contact either at the pair consisting of points 1 and 3, or at the pair consisting of points 2 and 3. It is clear that the equations of motion have the same form for both of these contact sets. However, the constraints, hence also the fictitious forces, are different. Note that three-point contact is effectively not possible for non-zero rotational velocity ω.

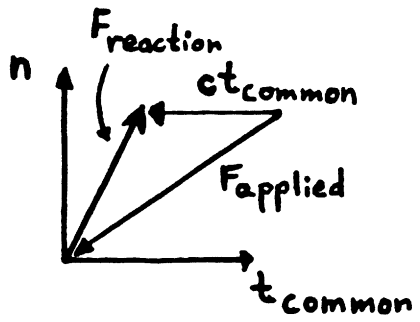


Figure 4.22. Computation of a reaction force. The negative applied force is projected perpendicular to the configuration space normal space, and parallel to the common tangent.

Finally, note that for the contact pair 1, 3, the second variation constraint for point 2 is satisfied as an inequality. In other words, $a_y - r_x \alpha = -\omega^2 r_y$ which is greater than $-2\omega^2 r_y$. Thus the contact pair 1, 3 does not violate the surface constraint at contact 2.

In contrast, if contact is satisfied for the pair 2, 3, then $a_y - r_x \alpha = -2\omega^2 r_y$ which is less than $-\omega^2 r_y$. Thus the contact pair 2, 3 violates the surface constraint at point 1. It follows that one cannot assume contact at point 2.

The legal contact classes for this three-point example are therefore simply, single point contact at point 1, single point contact at point 3, and two-point contact at points 1 and 3.

4.5.6. Reaction Force Computation for a Two Dimensional Contact Set

This subsection outlines a method for computing reaction forces based on the previous analysis. The scenario is a contact set whose normal space is two dimensional. In the three dimensional configuration space of a planar moving object, a two dimensional contact set is fairly easy to analyze. This is because the projection dimension is one. The technique outlined below generalizes to more complicated contact sets, such as those in higher dimensional configuration spaces, or those whose normal spaces are of lower dimension.

4.5.6.1. Form of the Reaction Force

Suppose there are k points of contact, which satisfy the second variation constraints with equality. Assume that the normal space dimension is two, and let t_{common} be the vector that is perpendicular to this normal space. Note that t_{common} is a common configuration space tangent for all points of contact. Let $\mathrm{F}_{fictitious} = \mathrm{M}\,\mathrm{h}'$ be the fictitious force calculated as in Sec. 4.5.4.2.

Relative to the effective applied force $\mathbf{F}'_{applied} = \mathbf{F}_{applied} + \mathbf{F}_{fictitious}$, the reaction force $\mathbf{F}_{reaction}$ is calculated by a normal projection. In this manner the fictitious force automatically insures that the second variation constraints are satisfied. Thus,

$$\mathbf{F}_{reaction} = -\mathbf{F}'_{applied} + c\,\mathbf{t}_{common}, \tag{4.101}$$

for some appropriate value of the parameter c.[4] See also Fig. 4.22.

The issue is to determine what values of c are legal values. This is most easily done by noting that any reaction force must be a combination of friction cone edges. Depending on the values of tangential contact velocities and accelerations, a given friction cone edge may not be able to participate in forming a reaction force. For example, knowing that the contact velocity in Fig. 4.12 is to the left implies that the reaction force at that contact point must be on the right friction cone edge. Thus the left friction cone edge cannot be used in forming a reaction force. This constrains the possible values of the parameter c.

4.5.6.2. Legal Friction Cone Edges

Let $\mathbf{r}_i$ be the radius vector from the contact point number i to the reference point. Recall, that the contact velocity and acceleration are related to the reference point velocity and acceleration by Eqs. (4.43) and (4.44). Recall, therefore, that the tangential contact velocity and acceleration are available as

$$\mathbf{v}_{0i} \cdot \mathbf{t}_{si} = \mathbf{v} \cdot \mathbf{v}_{fi}, \tag{4.102}$$

$$\mathbf{a}_{0i} \cdot \mathbf{t}_{si} = \left(\mathbf{a} + \omega^2\,\mathbf{r}_i\right) \cdot \mathbf{v}_{fi}, \tag{4.103}$$

where $\mathbf{t}_{si}$ is the sliding tangent, and $\mathbf{v}_{fi}$ is the vector defining the friction cone edges, at point number i. Also, $\mathbf{v}$ and $\mathbf{a}$ are the configuration space velocity and acceleration, while $\mathbf{v}_{0i}$ and $\mathbf{a}_{0i}$ are the contact velocity and acceleration at point i.

By analogy to the one-point analysis, a reaction force at a contact point must be on one or the other of the friction cone edges if the sliding contact velocity is non-zero. When the sliding contact velocity is zero, then the sliding contact acceleration is considered. Only if this also is zero, can the reaction force be in the interior of the friction cone.

Now suppose the reaction force is $\mathbf{F}_{reaction}$. In order to consider the tangential acceleration at the point i, it is necessary to consider the net force plus the term $\omega^2\,\mathbf{M}\,\mathbf{r}_i$. This follows from Eq. (4.47). Consequently, define a variant of net force

$$\begin{aligned} \mathbf{F}_i &= \mathbf{F}_{applied} + \mathbf{F}_{reaction} + \omega^2\,\mathbf{M}\,\mathbf{r}_i \\ &= -\mathbf{F}_{fictitious} + \omega^2\,\mathbf{M}\,\mathbf{r}_i + c\,\mathbf{t}_{common}, \end{aligned} \tag{4.104}$$

by Eq. (4.101). $\mathbf{F}_i$ effectively determines the acceleration of the contact point.

[4] As always, these forces are generalized forces in configuration space.

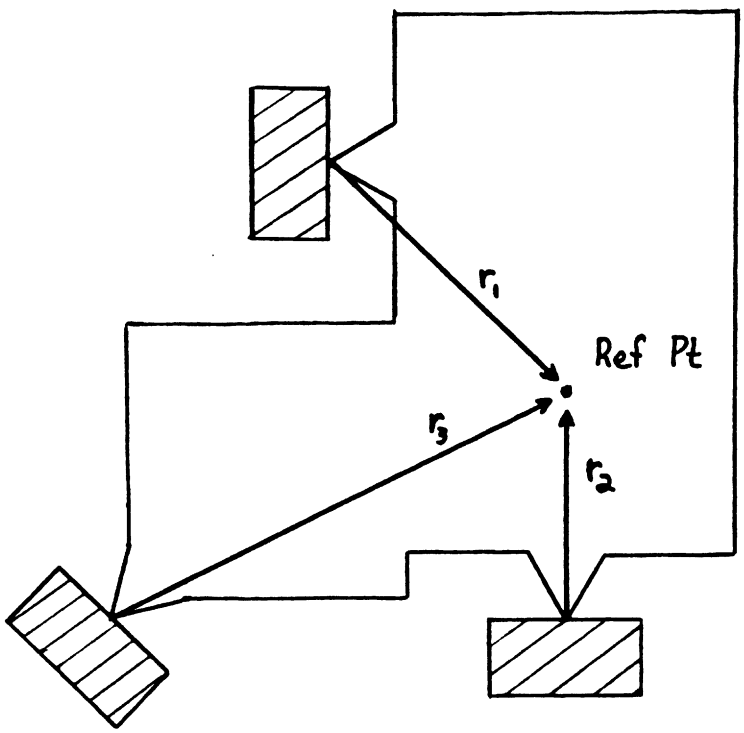


Figure 4.23. Three-point contact example demonstrating motion ambiguities. The associated three one-point friction cones do not possess three coplanar edges (see Fig. 4.24)

When the contact velocity is zero, then the sign of $\mathbf{F}_i \cdot \mathbf{v}_{fi}$ determines on which edge of the friction cone the reaction force must lie. $\mathbf{F}_i$ is a function of c. Thus to each friction cone edge one can assign a *validity interval* in c space. If c lies in this interval, then the edge can take part in forming the reaction force determined by c. Otherwise, it cannot. A reaction force that is interior to a friction cone may be viewed as the sum of two forces lying on the two friction cone edges.

Let I_i^+ be the interval corresponding to the "positive" friction cone edge, that is, to the edge $\Delta_i \mathbf{n}_i + \mathbf{v}_{fi}$. To say that $c \in I_i^+$, means that the edge $\Delta_i \mathbf{n}_i + \mathbf{v}_{fi}$ may be used in computing the reaction force $\mathbf{F}_{reaction}$, as in Eq. (4.101). Similarly, let I_i^- be the interval corresponding to the "negative" friction cone edge, that is, to the edge $\Delta_i \mathbf{n}_i - \mathbf{v}_{fi}$.

Let $\mathbf{v}$ be the configuration space velocity of the reference point. Then the intervals are given by the following table.

The table specifies a column of conditions followed by two columns of intervals. The conditions are ordered hierarchically. If a condition is not met, then the

condition below it should be considered. A condition assumes that all conditions above it are false.

Whenever a condition is met, the intervals in the adjacent columns specify the values of c for which the friction cone edges can take part in forming a reaction force. Also

$$c_i = \frac{\left(\mathbf{F}_{fictitious} - \omega^2 \mathbf{M}\mathbf{r}_i\right) \cdot \mathbf{v}_{fi}}{\mathbf{t}_{common} \cdot \mathbf{v}_{fi}}, \tag{4.105}$$

where defined.

Condition	I_i^+	I_i^-
$\mathbf{v} \cdot \mathbf{v}_{fi} > 0$	$\emptyset$	$[-\infty, +\infty]$
$\mathbf{v} \cdot \mathbf{v}_{fi} < 0$	$[-\infty, +\infty]$	$\emptyset$
$\mathbf{t}_{common} \cdot \mathbf{v}_{fi} > 0$	$[-\infty, c_i]$	$[c_i, +\infty]$
$\mathbf{t}_{common} \cdot \mathbf{v}_{fi} < 0$	$[c_i, +\infty]$	$[-\infty, c_i]$
$\mathbf{F}_i \cdot \mathbf{v}_{fi} > 0$	$\emptyset$	$[-\infty, +\infty]$
$\mathbf{F}_i \cdot \mathbf{v}_{fi} < 0$	$[-\infty, +\infty]$	$\emptyset$
$\mathbf{F}_i \cdot \mathbf{v}_{fi} = 0$	$[-\infty, +\infty]$	$[-\infty, +\infty]$

(4.106)

4.5.6.3. An Algorithm for Projecting onto the Composite Friction Cone

The previous analysis provides an algorithm for computing reaction forces. The reaction force $\mathbf{F}_{reaction}(c) = -\mathbf{F}'_{applied} + c\,\mathbf{t}_{common}$ is a function of the parameter c. Associate with every friction cone edge e in the contact set its validity interval I_e. Then $\mathbf{F}_{reaction}(c)$ is a valid reaction force if and only if it can be written as the positive sum of friction cone edges $e_1, \ldots, e_n$, such that c is in the intersection of all the validity intervals $I_{e_1}, \ldots, I_{e_n}$. For a three dimensional configuration space, as in this case, it is sufficient to consider only values of n less than or equal to three.

The algorithm generalizes to higher dimensional configuration spaces, and to other-dimensional contact sets. However, in general, the one dimensional parameter c must be replaced by a multi-dimensional parameter.

4.5.7. Indeterminate Accelerations

This subsection presents an example that demonstrates the indeterminacy of the equations of motion. This is not surprising, given that the number of variables in these equations is greater than the number of constraints. The example exhibits an applied force for which there exist several legal reaction forces.

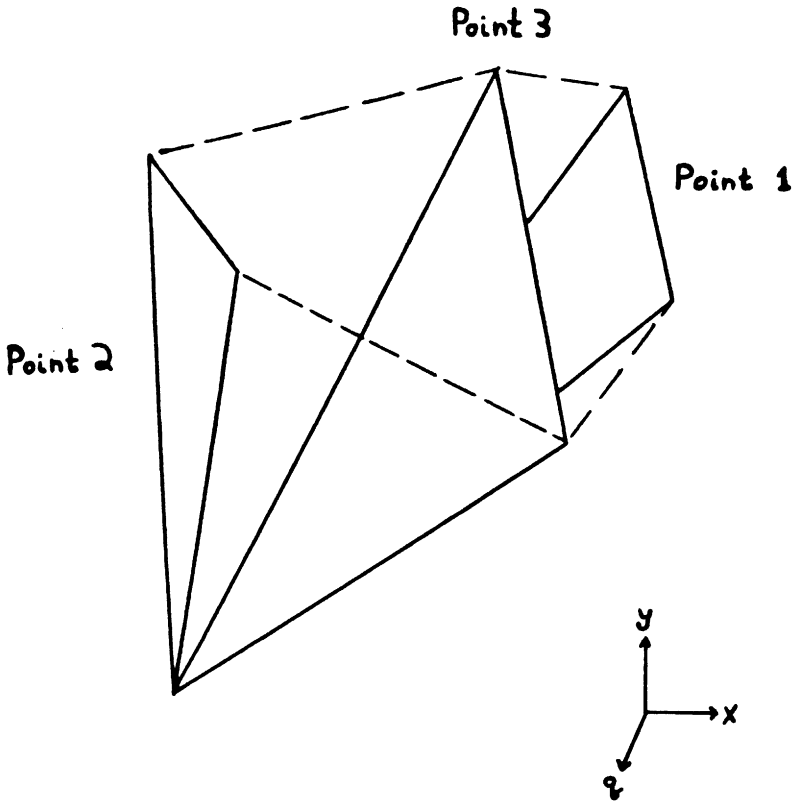


Figure 4.24. The three one-point contact friction cones for the example of Fig. 4.23. The composite friction cone is indicated by the dashed lines. Notice that the middle friction cone juts out of the planar faces formed by the two exterior friction cones. This gives rise to motion ambiguities. (The figure is not drawn to scale, but serves to describe the general relationship of the three friction cones.)

4.5.7.1. An Example

Consider Fig. 4.23 which depicts a three-point configuration. Assume that the object is stationary. For simplicity assume that the radius of gyration ρ is 1, and that the coefficient of friction μ is non-zero and less than 1/4.

The three radius vectors are

$$\begin{aligned} \mathbf{r}_1 &= (1, -1, 0) \\ \mathbf{r}_2 &= (0, 1, 0) \\ \mathbf{r}_3 &= (2, 1, 0). \end{aligned} \tag{4.107}$$

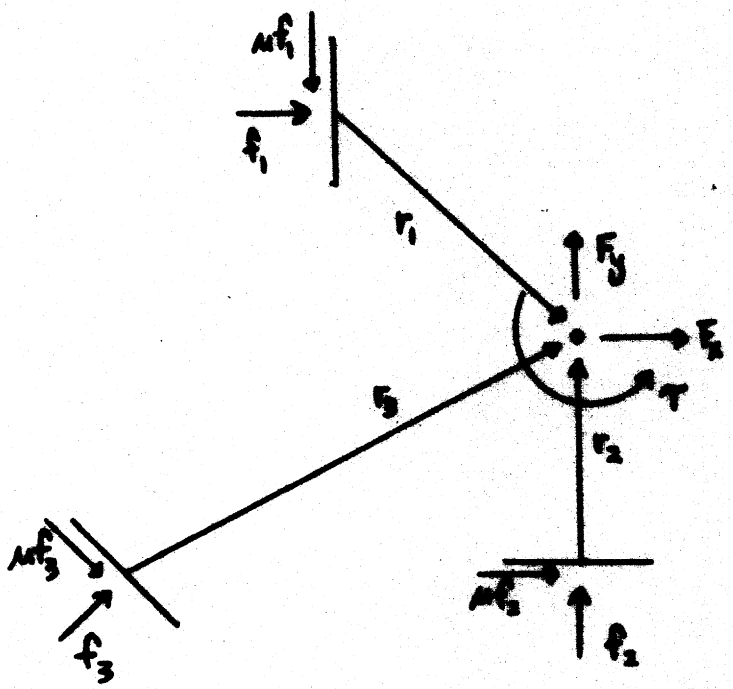


Figure 4.25. Force diagram for the example of Fig. 4.23.

The sliding tangent vectors are

$$\begin{aligned} \mathbf{t}_{s1} &= (0, -1, 0) \\ \mathbf{t}_{s2} &= (1, 0, 0) \\ \mathbf{t}_{s3} &= \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0\right) \end{aligned} \tag{4.108}$$

From this it follows that the configuration space normals and edge defining vectors are given by the following vectors.

$$\begin{aligned}
\Delta_1 \mathbf{n}_1 &= (1, 0, -1) \\
\Delta_2 \mathbf{n}_2 &= (0, 1, 0) \\
\Delta_3 \mathbf{n}_3 &= \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right) \\
\mathbf{v}_{f1} &= (0, -1, 1) \\
\mathbf{v}_{f2} &= (1, 0, 1) \\
\mathbf{v}_{f3} &= \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, \frac{3}{\sqrt{2}}\right)
\end{aligned} \tag{4.109}$$

Consequently, the friction cone edges are specified by

$$\begin{array}{lll}
\text{Point Number 1:} & (1, -\mu, -1+\mu) & (1, \mu, -1-\mu). \\
\text{Point Number 2:} & (\mu, 1, \mu) & (-\mu, 1, -\mu). \\
\text{Point Number 3:} & (1+\mu, 1-\mu, -1+3\mu) & (1-\mu, 1+\mu, -1-3\mu).
\end{array} \tag{4.110}$$

Notice that no three edges of the friction cones are coplanar (see Fig. 4.24). This is precisely the reason that there exists a continuum of solutions to the equations of motion. In the three-point example of Sec. 4.5.2 the friction cone edges were coplanar. Thus, the solution to the equations of motion was independent of the contact points assumed to be providing reaction forces.

Given a contact set with a certain normal space dimension, one might imagine that any motion that maintained all contacts was uniquely determined by any subset of those contacts, so long as that subset preserved the normal dimension. In general, as this example shows, that supposition is false.

4.5.7.2. Common Tangent

Since the object is assumed to be stationary, the second variation constraints reduce to the constraint that the net acceleration must be perpendicular to the normal space of the contact set. In other words, the net acceleration must be parallel to the common tangent $\mathbf{t}_{common}$, which is easily seen to be

$$\mathbf{t}_{common} = (1, 0, 1). \tag{4.111}$$

The fact that the friction cone edges are not coplanar means that it is possible to project an applied force along the common tangent onto two faces formed from two distinct one-point contact friction cones. Thus, the resulting motion depends on which two-point contact one wishes to consider.

4.5.7.3. Explicit Equations of Motion

The previous argument, based on configuration space information, shows that there exist motion indeterminacies for the current example. It is instructive to verify this claim using the standard motion equations (see Fig. 4.25).

Assume that one is projecting applied forces onto planar sheets formed from edges of the individual friction cones. The applied forces and torque are F_x, F_y, and τ. Then the equations of motion, in terms of the notation of Fig. 4.25, are

$$\begin{aligned} F_x + f_1 + \frac{1}{\sqrt{2}} f_3 + \mu f_2 + \frac{1}{\sqrt{2}} \mu f_3 &= m a_x \\ F_y + f_2 + \frac{1}{\sqrt{2}} f_3 - \mu f_1 - \frac{1}{\sqrt{2}} \mu f_3 &= m a_y \\ \tau - f_1 - \frac{1}{\sqrt{2}} f_3 + \mu f_1 + \mu f_2 + \frac{3}{\sqrt{2}} \mu f_3 &= m \alpha \end{aligned} \tag{4.112}$$

Now let $F_x = -1$, $F_y = -1$, and $\tau = -\mu$. For simplicity take $m = 1$. By the constraint on acceleration, $a_y = 0$ and $a_x = \alpha = a$.

Projecting along the common tangent onto the faces between the edges of friction cone 1 and friction cone 2 is equivalent to assuming that the effective points of contact are at points 1 and 2. The resulting equations of motion are.

$$\begin{aligned} -1 + f_1 + \mu f_2 &= a \\ -1 - \mu f_1 + f_2 &= 0 \\ -\mu - f_1 + \mu f_1 + \mu f_2 &= a. \end{aligned} \tag{4.113}$$

It is easily seen that the solutions to this system are given by

$$\begin{aligned} f_1 &= \frac{1-\mu}{2-\mu} \\ f_2 &= \frac{2-\mu^2}{2-\mu} \\ a &= \frac{-1+2\mu-\mu^3}{2-\mu.} \end{aligned} \tag{4.114}$$

Note that both f_1 and f_2 are positive, hence valid. Also, a is negative.

Now consider projecting along the common tangent onto a face formed by the edges from friction cone 2 and friction cone 3. This is equivalent to assuming that the effective points of contact are at points 2 and 3. The resulting equations of motion are

$$
\begin{aligned}
-1 + \frac{1}{\sqrt{2}} f_3 + \mu f_2 + \frac{1}{\sqrt{2}} \mu f_3 &= a \\
-1 + f_2 + \frac{1}{\sqrt{2}} f_3 - \frac{1}{\sqrt{2}} \mu f_3 &= 0 \\
-\mu - \frac{1}{\sqrt{2}} f_3 + \mu f_2 + \frac{3}{\sqrt{2}} \mu f_3 &= a
\end{aligned}
\tag{4.115}
$$

The solutions to this system are

$$
\begin{aligned}
f_3 &= \frac{1}{\sqrt{2}} \\
f_2 &= \frac{1}{2} + \frac{1}{2} \mu \\
a &= -\frac{1}{2} + \mu + \frac{1}{2} \mu^2
\end{aligned}
\tag{4.116}
$$

Again, both f_3 and f_2 are positive, hence valid. Also, for the given restrictions on μ, a is negative.

4.5.7.4. A Continuum of Reaction Forces

The previous analysis shows that the three-point contact of Fig. 4.23 has two different motion solutions for the applied force $(-1, -1, -\mu)$, depending on whether points 1 and 2, or points 2 and 3, are taken as the defining points of contact.

From this it is clear that any convex combination of the solutions found above is also a solution when all three points are considered. Thus there is a continuum of possible reaction forces. Along with this continuum there is a continuum of possible motions. All are directed along the common tangent t_{common}. In fact, the accelerations are all in the negative direction, but they vary in magnitude. This completes the demonstration of motion indeterminacy.

4.6. Friction in Six Dimensional Configuration Space

This section examines the representation of friction in the full six dimensional configuration space of a general moving body. The section concentrates on one-point contact. For simplicity, it is assumed that that there are exactly two translational and three rotational degrees of freedom at the point of contact. For example, for polyhedral objects, the contacts considered are vertex-face, face-vertex, and edge-edge. The ideas presented are generalizations of those discussed in previous sections for planar motions. Multiple contact may be treated in much the same fashion as outlined in the previous section for the three dimensional configuration space.

4.6.1. Configuration Space

The configuration space C of a three dimensional object is given by the six dimensional manifold $\Re^3 \times SO(3)$. $SO(3)$ is the special orthogonal group, comprising the group of rotations possible in $\Re^3$. The object's moment of inertia operator defines a field of inner products on $SO(3)$, which may be combined in a natural fashion with the standard inner product on $\Re^3$ to define a field of inner products on C. For convenience, normalize the inner product with respect to mass, that is, treat mass separately. Thus C is a Riemannian manifold (see Arnold (1978), Appendix 2).

Let $p \in C$ and let T_p be the six-dimensional tangent space to C at p. Let $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3 \in \Re^3$ be unit vectors corresponding to three orthogonal principal axes of the object in configuration p. Suppose one chooses a basis for T_p in terms of translation along and rotation about these vectors. Relative to this basis the inner product Φ_p at p is given by a diagonal matrix with entries 1, 1, 1, ρ_1^2, ρ_2^2, ρ_3^2, where ρ_i is the radius of gyration about $\mathbf{p}_i$. Specifically, if $\mathbf{v} = (v_1, v_2, v_3, v_4, v_5, v_6)$ and $\mathbf{w} = (w_1, w_2, w_3, w_4, w_5, w_6)$ are two vectors in T_p with respect to principal axes coordinates, then their inner product is given by

$$\Phi_p(v, w) = (v_1, v_2, v_3, v_4, v_5, v_6) \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \rho_1^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \rho_2^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \rho_3^2 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{pmatrix} \tag{4.117}$$

Throughout, the convention holds that the first three coordinates of a tangent vector arise from $\Re^3$, while the second three coordinates arise from $SO(3)$.

The relationship of the moment of inertia operator to kinetic energy permits a natural imbedding of forces in the tangent spaces of $\mathcal{C}$. Suppose $p : [\,0,\ 1\,] \rightarrow \mathcal{C}$ is a curve in $\mathcal{C}$, and $\mathbf{F}$ is a force field along this curve. If W is the work done by $\mathbf{F}$ along the curve p, then the representation of $\mathbf{F}$ should satisfy

$$W = \int_0^1 \Phi_{p(t)}\left(\mathbf{F}_{p(t)}, \frac{dp}{dt}\right) dt \tag{4.118}$$

Consequently, if f_1, f_2, f_3 are the forces along, and τ_1, τ_2, τ_3 are the torques about the principal axes $\mathrm{p}_1, \mathrm{p}_2, \mathrm{p}_3$ at some point $p \in \mathcal{C}$, then the representation of this force in T_p, relative to principal axes coordinates is

$$\left(f_1,\ f_2,\ f_3,\ \tfrac{1}{\rho_1^2}\tau_1,\ \tfrac{1}{\rho_2^2}\tau_2,\ \tfrac{1}{\rho_3^2}\tau_3\right) \tag{4.119}$$

4.6.2. One-Point Contact

Consider now the case of a three dimensional object in one-point contact with some real space obstacle. The geometric constraint of the one-point contact defines (within some region of $\mathcal{C}$) a five dimensional submanifold M of $\mathcal{C}$. At every point $q \in M$ there is a five dimensional tangent space T_q^M. One may of course, viewing q as a point in $\mathcal{C}$, identify T_q^M with a subspace of T_q. The inner product on $\mathcal{C}$ discussed above is inherited by M, permitting definition of a normal space N_q, for every $q \in M$. One may write $T_q = N_q \oplus T_q^M$. Here $\oplus$ denotes the direct sum of orthogonal subspaces.

For every point $q \in M$, T_q^M is spanned by three vectors corresponding to pure rotation of the object about the contact point and two vectors corresponding to pure sliding of the object. The latter two vectors are simply those defining the real space tangent plane for a fixed object configuration. As with the planar configuration space discussed previously, one is primarily interested in the three tangents arising from pure rotation. Let R_q denote the subspace of T_q^M spanned by these three vectors, and let $R_q^\perp$ denote its orthogonal complement. One may then write

$$\begin{aligned} T_q^M &= R_q^\perp \oplus R_q \\ T_q &= N_q \oplus R_q^\perp \oplus R_q \end{aligned} \tag{4.120}$$

In order to examine the spanning tangents of these spaces, it is convenient to formulate the representation in terms of principal axes coordinates. Let $\mathbf{r} = (r_x, r_y, r_z)$ be the real space vector from the point of contact to the reference point, and let $\mathbf{n}_0 = (n_1, n_2, n_3)$ be the real space normal. For example, in the case of polyhedral objects, the real space normal is one of the following: the outward normal of the real space surface in the case of vertex-surface contact; the inward normal of the object under consideration in the case of surface-vertex contact; the outward normal to the plane formed by the two edges in the case of edge-edge contact. As noted previously, T_q^M is spanned by three rotational and two translational tangent vectors, namely by

$$\begin{aligned}
\mathbf{t}_x &= (\ \ 0, -r_z,\ \ r_y,\ \ 1,\ \ 0,\ \ 0) \\
\mathbf{t}_y &= (\ \ r_z,\ \ \ 0, -r_x,\ \ 0,\ \ 1,\ \ 0) \\
\mathbf{t}_z &= (-r_y,\ \ r_x,\ \ \ 0,\ \ 0,\ \ 0,\ \ 1) \\
\mathbf{t}_{s1} &= (\ n_{23},\ n_{13},\ n_{12},\ \ 0,\ \ 0,\ \ 0) \\
\mathbf{t}_{s2} &= (\ m_1,\ m_2,\ m_3,\ \ 0,\ \ 0,\ \ 0)
\end{aligned} \tag{4.121}$$

where

$$\begin{aligned}
(n_{23}, n_{13}, n_{12}) &= \mathbf{n}_0 \times \mathbf{r} \\
(m_1, m_2, m_3) &= (\mathbf{n}_0 \times \mathbf{r}) \times \mathbf{n}_0
\end{aligned} \tag{4.122}$$

Consequently N_q is spanned by

$$\mathbf{n} = \left(n_1,\ n_2,\ n_3,\ \tfrac{1}{\rho_1^2} n_{23},\ \tfrac{1}{\rho_2^2} n_{13},\ \tfrac{1}{\rho_3^2} n_{12}\right) \tag{4.123}$$

R_q is spanned by $\mathbf{t}_x, \mathbf{t}_y, \mathbf{t}_z$.

Finally, $R_q^\perp$ is spanned by

$$\begin{aligned}
\mathbf{t}_1 &= \left(\Delta_3, \quad 0, \quad -\Delta_1, \quad \tfrac{1}{\rho_1^2} r_y \Delta_1, \quad -\tfrac{1}{\rho_2^2}(r_z \Delta_3 + r_x \Delta_1), \quad \tfrac{1}{\rho_3^2} r_y \Delta_3\right) \\
\mathbf{t}_2 &= \left(0, \quad \Delta_3, \quad -\Delta_2, \quad \tfrac{1}{\rho_1^2}(r_z \Delta_3 + r_y \Delta_2), \quad -\tfrac{1}{\rho_2^2} r_x \Delta_2, \quad -\tfrac{1}{\rho_3^2} r_x \Delta_3\right)
\end{aligned} \tag{4.124}$$

where

$$\begin{aligned}
\Delta_1 &= n_1 + \tfrac{1}{\rho_3^2} r_y n_{12} - \tfrac{1}{\rho_2^2} r_z n_{13} \\
\Delta_2 &= n_2 - \tfrac{1}{\rho_3^2} r_x n_{12} + \tfrac{1}{\rho_1^2} r_z n_{23} \\
\Delta_3 &= n_3 - \tfrac{1}{\rho_1^2} r_y n_{23} + \tfrac{1}{\rho_2^2} r_x n_{13}
\end{aligned} \tag{4.125}$$

These vectors may easily be derived by solving four linear equations representing the orthogonality conditions on $\mathbf{t}_1$ and $\mathbf{t}_2$.

4.6.3. Friction

Fix $q \in M$. Henceforth, the standard dot notation is used for the inner product Φ_q on T_q.

Let $\mathbf{n}_0 \in T_q$ be a unit vector corresponding to the "outward pointing" real space normal. Let $\mathbf{t}_x, \mathbf{t}_y \in T_q$ be two orthogonal unit vectors spanning the real space tangent plane for configuration q.

Define

$$\begin{aligned}
\pi_T : T_q &\to T_q \\
\mathbf{v} &\mapsto (\mathbf{v} \cdot \mathbf{t}_x)\mathbf{t}_x + (\mathbf{v} \cdot \mathbf{t}_y)\mathbf{t}_y
\end{aligned} \tag{4.126}$$

$$\begin{aligned}
\pi_N : T_q &\to T_q \\
\mathbf{v} &\mapsto (\mathbf{v} \cdot \mathbf{n}_0)\mathbf{n}_0
\end{aligned} \tag{4.127}$$

So π_T, π_N are the orthogonal projections onto the real space tangent plane and the real space normal space, respectively, viewing these as subspaces of T_q. Note that $\|\pi_T(\mathbf{v})\| = \sqrt{(\mathbf{v} \cdot \mathbf{t}_x)^2 + (\mathbf{v} \cdot \mathbf{t}_y)^2}$, and that $\|\pi_N(\mathbf{v})\| = |\mathbf{v} \cdot \mathbf{n}_0|$.

Finally, choose $\mathbf{n} \in N_q$ such that $\mathbf{n} \cdot \mathbf{n}_0 > 0$, and $\|\mathbf{n}\| = 1$. In other words, $\mathbf{n}$ is the outward pointing normal of the configuration space surface.[5]

[5] Alternatively, as in the previous sections, one could have first defined $\mathbf{n}$, and then defined $\mathbf{n}_0$ in terms of this vector. For the sake of intuitive development this section chose the other route.

Given an arbitrary force $\mathbf{F} \in T_q$, decompose $\mathbf{F}$ into $\mathbf{F} = \mathbf{F}_n + \mathbf{F}_\varphi + \mathbf{F}_r$, where $\mathbf{F}_n \in N_q$, $\mathbf{F}_\varphi \in R_q^\perp$, $\mathbf{F}_r \in R_q$. Assuming a coefficient of friction μ, the static equilibrium conditions are

$$\mathbf{F}_r = \mathbf{0} \tag{4.128}$$

$$\|\pi_T(\mathbf{F})\| \leq \mu \|\pi_N(\mathbf{F})\|, \qquad \mathbf{F} \cdot \mathbf{n}_0 \leq 0 \tag{4.129}$$

These conditions follow, as in the planar case, from the realization that frictional forces arise in response to applied forces acting in directions of movement involving sliding, but not in response to forces acting in directions of pure rotation. Given condition (4.128), one can, for some unit vector $\mathbf{t}_\varphi \in R_q^\perp$, with $\mathbf{t}_\varphi \cdot \mathbf{n}_0 \geq 0$, write

$$\begin{aligned} \mathbf{F} &= \mathbf{F}_n + \mathbf{F}_\varphi \\ \mathbf{F}_\varphi &= (\mathbf{F} \cdot \mathbf{t}_\varphi)\mathbf{t}_\varphi \\ \mathbf{F}_n &= (\mathbf{F} \cdot \mathbf{n})\mathbf{n} \end{aligned} \tag{4.130}$$

Therefore

$$\begin{aligned} \pi_T(\mathbf{F}) &= (\mathbf{F} \cdot \mathbf{t}_\varphi)\,\pi_T(\mathbf{t}_\varphi) + (\mathbf{F} \cdot \mathbf{n})\,\pi_T(\mathbf{n}) \\ &= (\mathbf{F} \cdot \mathbf{t}_\varphi)\,\pi_T(\mathbf{t}_\varphi) \end{aligned} \tag{4.131}$$

$$\pi_N(\mathbf{F}) = (\mathbf{F} \cdot \mathbf{t}_\varphi)\,\pi_N(\mathbf{t}_\varphi) + (\mathbf{F} \cdot \mathbf{n})\,\pi_N(\mathbf{n}) \tag{4.132}$$

Since $\mathbf{F} \cdot \mathbf{n}_0 \leq 0$, $\mathbf{t}_\varphi \cdot \mathbf{n}_0 \geq 0$, and $\mathbf{n} \cdot \mathbf{n}_0 > 0$, it follows that

$$\begin{aligned} \|\pi_N(\mathbf{F})\| &= -\pi_N(\mathbf{F}) \cdot \mathbf{n}_0 \\ &= -\Big((\mathbf{F} \cdot \mathbf{t}_\varphi)\,\|\pi_N(\mathbf{t}_\varphi)\| + (\mathbf{F} \cdot \mathbf{n})\,\|\pi_N(\mathbf{n})\| \Big) \end{aligned} \tag{4.133}$$

So

$$(\mathbf{F} \cdot \mathbf{t}_\varphi)\,\|\pi_T(\mathbf{t}_\varphi)\| \geq \mu \Big((\mathbf{F} \cdot \mathbf{t}_\varphi)\,\|\pi_N(\mathbf{t}_\varphi)\| + (\mathbf{F} \cdot \mathbf{n})\|\pi_N(\mathbf{n})\| \Big) \tag{4.134}$$

$$(\mathbf{F} \cdot \mathbf{t}_\varphi)\,\|\pi_T(\mathbf{t}_\varphi)\| \leq -\mu \Big((\mathbf{F} \cdot \mathbf{t}_\varphi)\,\|\pi_N(\mathbf{t}_\varphi)\| + (\mathbf{F} \cdot \mathbf{n})\,\|\pi_N(\mathbf{n})\| \Big) \tag{4.135}$$

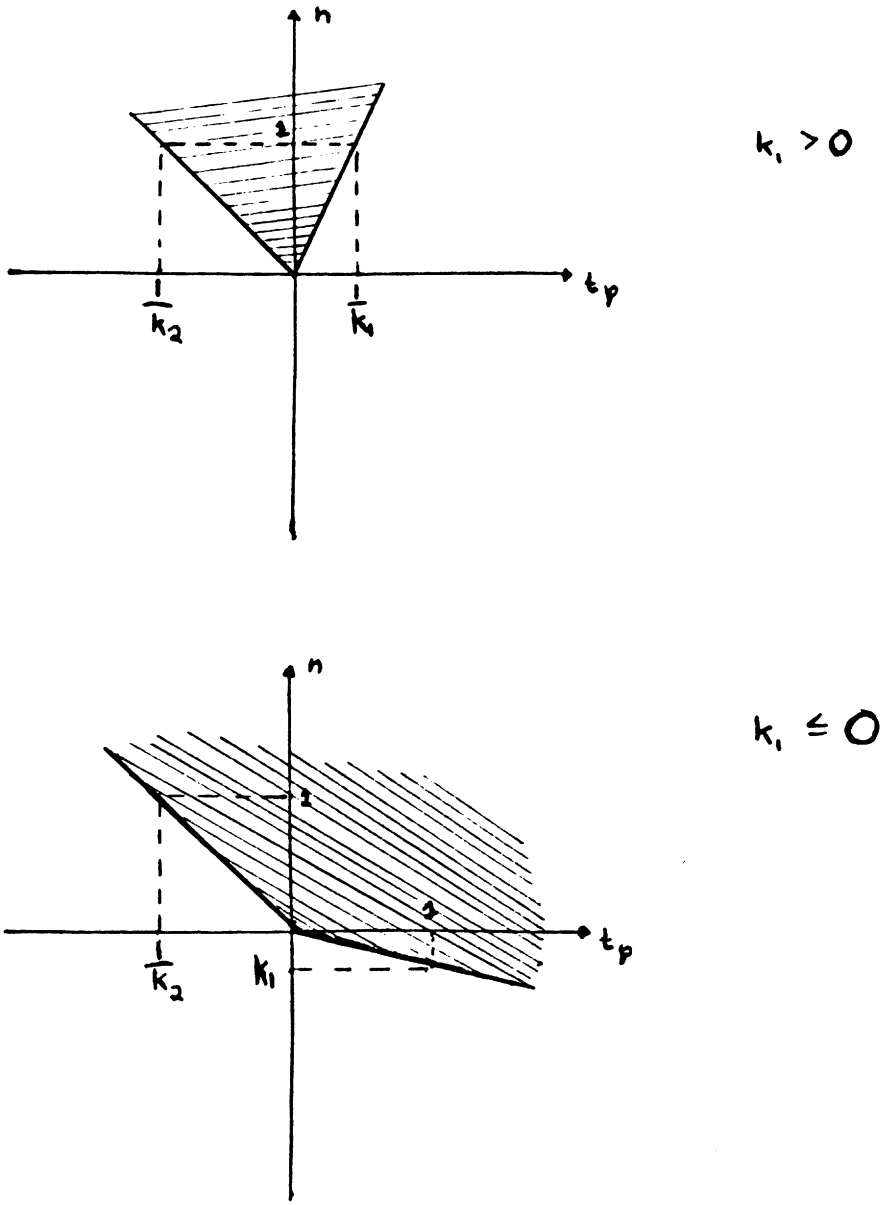


Figure 4.26. The configuration space friction cone for different values of the parameter k_1. Note that k_2 must always be non-positive.

Thus

$$(\mathbf{F}\cdot\mathbf{n}) \leq k_1(\mathbf{F}\cdot\mathbf{t}_\wp) \tag{4.136}$$

$$(\mathbf{F}\cdot\mathbf{n}) \leq k_2(\mathbf{F}\cdot\mathbf{t}_\wp), \tag{4.137}$$

where

$$\begin{aligned} k_1 &= \frac{\|\pi_T(\mathbf{t}_\wp)\| - \mu\|\pi_N(\mathbf{t}_\wp)\|}{\mu\|\pi_N(\mathbf{n})\|} \\ k_2 &= -\frac{\|\pi_T(\mathbf{t}_\wp)\| + \mu\|\pi_N(\mathbf{t}_\wp)\|}{\mu\|\pi_N(\mathbf{n})\|} \end{aligned} \tag{4.138}$$

The frictional constraints of the three dimensional configuration space analysis discussed in Sec. 4.3.6 are easily seen to be a special case of Eqs. (4.136) and (4.137). For a given $\mathbf{t}_\wp \in R_q^\perp$, Eqs. (4.136) and (4.137) define a two dimensional cone in T_q. Thus $\mathbf{t}_\wp$ acts as a parameter for a one-parameter family of such cones, the union of which comprises the complete six dimensional configuration space friction cone. The cone is therefore a three dimensional subset of T_q.

It is interesting to examine the meaning of these two constraints. In particular, note that k_2 is never positive, and in practice is always negative. Now if $k_1 > 0$, then the constraints become

$$\frac{1}{k_1}(\mathbf{F}\cdot\mathbf{n}) \leq (\mathbf{F}\cdot\mathbf{t}_\wp) \leq \frac{1}{k_2}(\mathbf{F}\cdot\mathbf{n}), \qquad \text{with} \quad \mathbf{F}\cdot\mathbf{n} \leq 0. \tag{4.139}$$

On the other hand, if $k_1 \leq 0$, then the constraints become

$$\begin{aligned} (\mathbf{F}\cdot\mathbf{t}_\wp) &\leq \frac{1}{k_2}(\mathbf{F}\cdot\mathbf{n}), \qquad \text{for} \quad \mathbf{F}\cdot\mathbf{n} \leq 0, \\ (\mathbf{F}\cdot\mathbf{n}) &\leq k_1(\mathbf{F}\cdot\mathbf{t}_\wp), \qquad \text{for} \quad \mathbf{F}\cdot\mathbf{n} > 0. \end{aligned} \tag{4.140}$$

In other words, for positive k_1, the $\mathbf{t}_\wp$ slice of the generalized friction cone lies wholly above the tangent plane, whereas for negative k_1 it lies partially below the tangent plane. See Fig. 4.26.

4.6.4. Reaction Force

Assume zero translational velocity at the point of contact. Given an arbitrary applied force $\overline{\mathbf{F}}_A = \mathbf{F}_n + \mathbf{F}_p + \mathbf{F}_r$, with $\overline{\mathbf{F}}_A \cdot \mathbf{n} \leq 0$, one would expect a reaction force of the form $\overline{\mathbf{F}}_R = -\mathbf{F}_n - t\,\mathbf{F}_p$, where $t \in [\,0\,,\,1\,]$ is the maximum value which permits $-\overline{\mathbf{F}}_R$ to satisfy the frictional constraints of Eqs. (4.136) and (4.137).[6] If these constraints cannot be satisfied for any appropriate value of t, then $\overline{\mathbf{F}}_R = \mathbf{0}$. The only issue is whether the decomposition of forces into $\mathbf{F}_n, \mathbf{F}_p$, and $\mathbf{F}_r$ components is a sound one, that is, whether the choice of inner product on C is a physically valid choice. This subsection shall investigate this issue by considering the extreme case of an object hinged at the point of contact. This problem is equivalent to the friction problem with respect to the orthogonal decomposition of forces. The difference between the two problems lies primarily in the constraints imposed on the relative maximum magnitudes of reaction force components in the frictional case. In addition, for non-zero contact velocities, the fictitious applied forces in the frictional case include terms that depend on the shape of the contact surface.

In order to analyze the problem it is convenient to translate between a configuration space formulation and a Newton-Euler formulation. The following assumptions are made:

For a given configuration, the representation of all vectors will be in terms of the principal axes coordinates of the associated tangent space.

The configuration space reference point is the object's center of mass. This assumption may be modified to permit arbitrary centers of compliance.

[6] The case $\overline{\mathbf{F}}_A \cdot \mathbf{n} > 0$ is more complicated. In particular, motion indeterminacies may arise. If the non-rotational component of the applied force, that is, $\mathbf{F}_n + \mathbf{F}_p$, lies inside the friction cone, then the object may either stick to the contact surface or accelerate away from the contact surface. The previous sections have discussed indeterminacy in the three dimensional configuration space.

The following notation is employed:

$\mathbf{r} = (r_x, r_y, r_z)$ is the real space vector
from the hinge point to the reference point.

$\mathbf{F}_A = (F_{Ax}, F_{Ay}, F_{Az})$ is the real space force applied
at the reference point.

$\tau_A = (\tau_{Ax}, \tau_{Ay}, \tau_{Az})$ is the real space torque applied
at the reference point.

$\mathbf{F}_R = (F_{Rx}, F_{Ry}, F_{Rz})$ is the real space reaction force
at the contact point.

$\tau_R = (\tau_{Rx}, \tau_{Ry}, \tau_{Rz})$ is the real space reaction torque
about the reference point.

$\mathbf{a} = (a_x, a_y, a_z)$ is the real space acceleration
of the reference point.

$\alpha = (\alpha_x, \alpha_y, \alpha_z)$ is the real space angular acceleration
about the reference point.

$\overline{\mathbf{F}}_A$ is the configuration space representation of the applied force.

$\overline{\mathbf{F}}_R$ is the configuration space representation of the reaction force.

ω is the real space angular velocity about the reference point.

m is the total mass of the object.

$\mathbf{I}$ is the moment of inertia tensor.

ρ_i is the radius of gyration about the i axis ($i = x,y,z$).

The Newton-Euler equations say (See Symon (1971))

$$\begin{aligned} m\,\mathbf{a} &= \mathbf{F}_A + \mathbf{F}_R \\ \tfrac{d}{dt}(\mathbf{I}\boldsymbol{\omega}) &= \tau_A + \tau_R \end{aligned} \tag{4.141}$$

The hinge constraint gives

$$\mathbf{a} = \boldsymbol{\alpha} \times \mathbf{r} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}) \tag{4.142}$$

Now

$$\begin{aligned} \tau_R &= (-\mathbf{r}) \times \mathbf{F}_R = \mathbf{F}_R \times \mathbf{r} \\ \frac{d}{dt}(\mathbf{I}\boldsymbol{\omega}) &= \mathbf{I}\boldsymbol{\alpha} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \end{aligned} \tag{4.143}$$

So

$$\begin{aligned} m(\boldsymbol{\alpha} \times \mathbf{r}) + m(\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r})) &= \mathbf{F}_A + \mathbf{F}_R \\ \mathbf{I}\boldsymbol{\alpha} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} &= \tau_A + \tau_R \end{aligned} \tag{4.144}$$

If one regards $-m(\omega \times (\omega \times \mathbf{r}))$ as a fictitious applied force, and $-\omega \times \mathbf{I}\omega$ as a fictitious applied torque, then one can assume without loss of generality that $\omega = 0$. Therefore Newton-Euler reduces to

$$\begin{aligned} m(\boldsymbol{\alpha} \times \mathbf{r}) &= \mathbf{F}_A + \mathbf{F}_R \\ \mathbf{I}\boldsymbol{\alpha} &= \tau_A + \tau_R, \qquad \text{with } \tau_R = \mathbf{F}_R \times \mathbf{r} \end{aligned} \tag{4.145}$$

For a hinged object the difference between N_q and $R_q^\perp$ is meaningless. One has in effect defined a new three dimensional submanifold N of C. One can thus lump the two spaces together and decompose T_q as $T_q = R_q^\perp \oplus R_q$, where R_q is as before the space of rotational tangents.

In terms of principal axes coordinates, R_q is spanned by

$$\begin{aligned} \mathbf{t}_x &= (\ 0, -r_z,\ \ r_y,\ \ 1,\ \ 0,\ \ 0) \\ \mathbf{t}_y &= (\ r_z,\ \ 0, -r_x,\ \ 0,\ \ 1,\ \ 0) \\ \mathbf{t}_z &= (-r_y,\ \ r_x,\ \ 0,\ \ 0,\ \ 0,\ \ 1) \end{aligned} \tag{4.146}$$

The orthogonal complement $R_q^\perp$ is spanned by

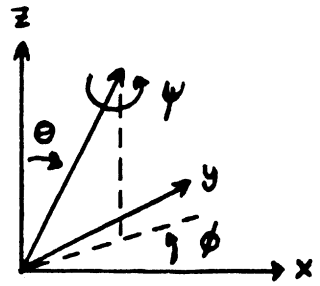


Figure 4.27. Spherical coordinates.

$$
\begin{aligned}
\mathbf{t}_1 &= \left(1,\ 0,\ \frac{r_x r_z}{1+r_x{}^2+r_y{}^2},\ -\frac{1}{\rho_x^2}\frac{r_x r_y r_z}{1+r_x{}^2+r_y{}^2},\ -\frac{1}{\rho_y^2} r_z \frac{1+r_y{}^2}{1+r_x{}^2+r_y{}^2},\ \frac{1}{\rho_z^2} r_y\right) \\
\mathbf{t}_2 &= \left(0,\ 1,\ \frac{r_y r_z}{1+r_x{}^2+r_y{}^2},\ \frac{1}{\rho_x^2} r_z \frac{1+r_x{}^2}{1+r_x{}^2+r_y{}^2},\ \frac{1}{\rho_y^2}\frac{r_x r_y r_z}{1+r_x{}^2+r_y{}^2},\ -\frac{1}{\rho_z^2} r_x\right) \\
\mathbf{t}_3 &= \left(0,\ 0,\ 1,\ -\frac{1}{\rho_x^2} r_y,\ \frac{1}{\rho_y^2} r_x,\ 0\right)
\end{aligned}
\tag{4.147}
$$

The choice of inner product in terms of the moment of inertia operator is justified by the following

Proposition Let $\overline{\mathbf{F}}_A$ be an applied force at some point $q \in N \subset C$. Write $\overline{\mathbf{F}}_A = \overline{\mathbf{F}}_{A_p} + \overline{\mathbf{F}}_{A_r}$, where $\overline{\mathbf{F}}_{A_p} \in R_q^{\perp}$, $\overline{\mathbf{F}}_{A_r} \in R_q$. Then $\overline{\mathbf{F}}_R = -\overline{\mathbf{F}}_{A_p}$.

Proof

It is sufficient to show

$$
\begin{aligned}
\overline{\mathbf{F}}_R \cdot \mathbf{t} = 0 \quad \forall\, \mathbf{t} \in R_q \\
(\overline{\mathbf{F}}_A + \overline{\mathbf{F}}_R) \cdot \mathbf{t} = 0 \quad \forall\, \mathbf{t} \in R_q^{\perp}
\end{aligned}
\tag{4.148}
$$

These statements are fairly evident from the definitions of R_q, $R_q^{\perp}$, and the Newton-Euler equations. However, the proof is presented here for the sake of completeness.

I. Relative to principal axes coordinates, since $\tau_R = \mathbf{F}_R \times \mathbf{r}$,

$$\overline{\mathbf{F}}_R = \left(F_{Rx},\ F_{Ry},\ F_{Rz},\ \tfrac{1}{\rho_x^2}(F_{Ry}r_z - F_{Rz}r_y),\ \tfrac{1}{\rho_y^2}(F_{Rz}r_x - F_{Rx}r_z),\ \tfrac{1}{\rho_z^2}(F_{Rx}r_y - F_{Ry}r_x)\right)$$

So

$$\begin{aligned}\overline{\mathbf{F}}_R \cdot \mathbf{t}_x &= -F_{Ry}r_z + F_{Rz}r_y + F_{Ry}r_z - F_{Rz}r_y \\ &= 0\end{aligned} \tag{4.149}$$

$$\begin{aligned}\overline{\mathbf{F}}_R \cdot \mathbf{t}_y &= F_{Rx}r_z - F_{Rz}r_x + F_{Rz}r_x - F_{Rx}r_z \\ &= 0\end{aligned} \tag{4.150}$$

$$\begin{aligned}\overline{\mathbf{F}}_R \cdot \mathbf{t}_z &= -F_{Rx}r_y + F_{Ry}r_x + F_{Rx}r_y - F_{Ry}r_x \\ &= 0\end{aligned} \tag{4.151}$$

So $\overline{\mathbf{F}}_R \cdot \mathbf{t} = 0$, for every $\mathbf{t} \in R_q$.

II. The Newton-Euler equations

$$\begin{aligned} m(\boldsymbol{\alpha} \times \mathbf{r}) &= \mathbf{F}_A + \mathbf{F}_R \\ \mathbf{I}\,\boldsymbol{\alpha} &= \tau_A + \tau_R \end{aligned} \tag{4.152}$$

yield the following linear equations

$$\begin{aligned} m(\alpha_y r_z - \alpha_z r_y) &= F_{Ax} + F_{Rx} \\ m(\alpha_z r_x - \alpha_x r_z) &= F_{Ay} + F_{Ry} \\ m(\alpha_x r_y - \alpha_y r_x) &= F_{Az} + F_{Rz} \\ m\rho_x^2 \alpha_x &= \tau_{Ax} + \tau_{Rx} \\ m\rho_y^2 \alpha_y &= \tau_{Ay} + \tau_{Ry} \\ m\rho_z^2 \alpha_z &= \tau_{Az} + \tau_{Rz} \end{aligned}$$

So

$$\overline{\mathbf{F}}_A + \overline{\mathbf{F}}_R = \left(m(\alpha_y r_z - \alpha_z r_y),\ m(\alpha_z r_x - \alpha_x r_z),\ m(\alpha_x r_y - \alpha_y r_x),\ m\alpha_x,\ m\alpha_y,\ m\alpha_z\right)$$

Therefore

$$\begin{aligned}(\mathbf{F}_A + \mathbf{F}_R)\cdot \mathbf{t}_1 &= m(a_y r_z - a_z r_y) + \frac{r_x r_z}{1+r_x^2+r_y^2} m(a_x r_y - a_y r_x) \\ &\quad - \frac{r_x r_y r_z}{1+r_x^2+r_y^2} m a_x - r_x \frac{1+r_y^2}{1+r_x^2+r_y^2} m a_y + m r_y a_z \\ &= 0 \end{aligned} \tag{4.153}$$

$$\begin{aligned}(\mathbf{F}_A + \mathbf{F}_R)\cdot \mathbf{t}_2 &= m(a_z r_x - a_x r_z) + \frac{r_y r_z}{1+r_x^2+r_y^2} m(a_x r_y - a_y r_x) \\ &\quad + r_z \frac{1+r_x^2}{1+r_x^2+r_y^2} m a_x + \frac{r_x r_y r_z}{1+r_x^2+r_y^2} m a_y - m r_x a_z \\ &= 0 \end{aligned} \tag{4.154}$$

$$\begin{aligned}(\mathbf{F}_A + \mathbf{F}_R)\cdot \mathbf{t}_3 &= m(a_x r_y - a_y r_x) - m a_x r_y + m a_y r_x \\ &= 0 \end{aligned} \tag{4.155}$$

So $(\mathbf{F}_A + \mathbf{F}_R)\cdot \mathbf{t} = 0$, for every $\mathbf{t} \in R_q^\perp$.
This completes the proof. ∎

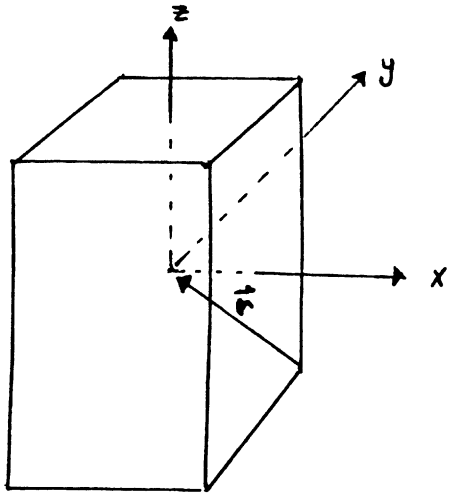


Figure 4.28. An object whose principal axes are aligned with the x, y, z axes.

4.6.5. Application

This subsection briefly address the issue of representing C, M, T_q, N_q, R_q, and R_q^+ in a global fashion. It is naturally desirable to represent C as a six dimensional vector space, so that the tangent spaces T_q may be directly identified with C itself. Since C is a six dimensional manifold this is always possible locally. Unfortunately it is not strictly possible globally. If, however, one is willing to tolerate singularities in the representation, then one can approximate C by a vector space fairly well.

The representation employs six parameters $(x, y, z, \theta, \phi, \psi)$, where x, y, z are the usual translational coordinates of $\Re^3$, and θ, ϕ, ψ are three angles which represent orientation. See Fig. 4.27. The angular coordinates are exactly the "joint angles" employed in the representation of a spherical wrist. They are also referred to as Cardan suspension coordinates (see Kane and Levinson (1978)). Singularities occur for θ values that are multiples of π. Furthermore, note that without restriction on the range of the joint angles, there are multiple representations of any given orientation.[7] The $(x, y, z, \theta, \phi, \psi)$ parameterization approximates C as a vector space. In particular, rotations about the three different axes of rotation commute. The parameterization is desirable on this basis.

Now consider an object, such as the block in Fig. 4.28, whose principal axes are aligned with the x, y, z axes when θ, ϕ, ψ are all zero. Suppose the object is in contact with a real space surface given by the x-y plane (see Fig. 4.29). Let

[7]For example, the two representations $(\theta_0, \phi_0, \psi_0)$, and $(-\theta_0, \pi + \phi_0, \pi + \psi_0)$ correspond to the same physical orientation. This is to be expected, given the singularities. Also note that there are multiple representations of the same physical rotation.

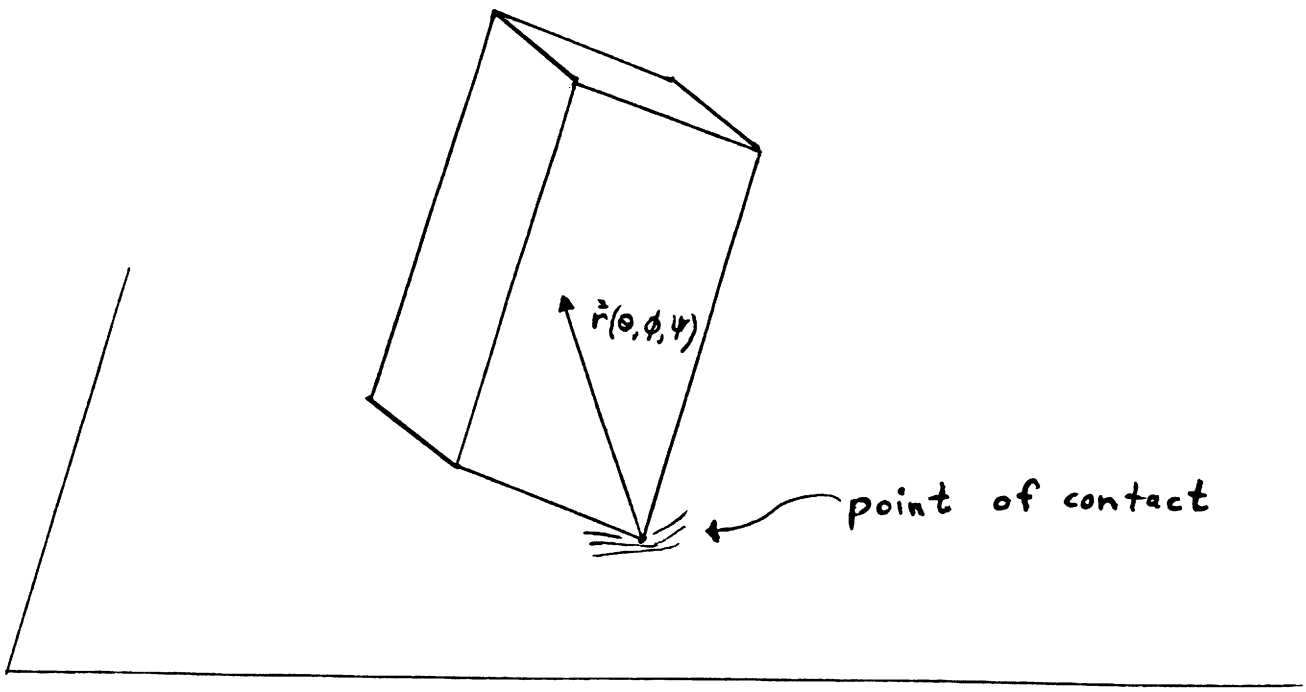


Figure 4.29. An object in one-point contact with the x-y plane

---

$\mathbf{r} = (r_1, r_2, r_3)$ be the vector in the zero position from the vertex of contact to the reference point.[8]

Let $\mathbf{R}_{\theta\phi\psi}$ be the standard rotation matrix[9]

$$\mathbf{R}_{\theta\phi\psi} = \begin{pmatrix} c_\phi c_\theta c_\psi - s_\phi s_\psi & -c_\phi c_\theta s_\psi - s_\phi c_\psi & c_\phi s_\theta \\ s_\phi c_\theta c_\psi + c_\phi s_\psi & -s_\phi c_\theta s_\psi + c_\phi c_\psi & s_\phi s_\theta \\ -s_\theta c_\psi & s_\theta s_\psi & c_\theta \end{pmatrix} \tag{4.156}$$

and let

$$\mathbf{r}(\theta, \phi, \psi) = (r_x, r_y, r_z) = \mathbf{R}_{\theta\phi\psi}\, \mathbf{r}^t \tag{4.157}$$

Then the configuration space surface M associated with the reference point $\mathbf{r}(\theta, \phi, \psi)$ is given by

---

[8]Assumed to be the center of mass or compliance.

[9]Let c_i and s_i denote $\cos(i)$ and $\sin(i)$, respectively $(i = \theta, \phi, \psi)$.

$$
\begin{aligned}
x(\overline{x},\overline{y},\theta,\phi,\psi) &= r_x + \overline{x} \\
y(\overline{x},\overline{y},\theta,\phi,\psi) &= r_y + \overline{y} \\
z(\overline{x},\overline{y},\theta,\phi,\psi) &= r_z \\
\theta(\overline{x},\overline{y},\theta,\phi,\psi) &= \theta \\
\phi(\overline{x},\overline{y},\theta,\phi,\psi) &= \phi \\
\psi(\overline{x},\overline{y},\theta,\phi,\psi) &= \psi
\end{aligned}
\tag{4.158}
$$

where $\overline{x}$, $\overline{y}$ are the x-y coordinates of the contact point.

Now suppose that $q = q(\overline{x},\overline{y},\theta,\phi,\psi)$ is a point in M. One wishes to examine N_q, R_q, and $R_q^{\perp}$. Clearly one can determine R_q without resorting to principal axes coordinates. One need merely differentiate the expression for q with respect to θ, ϕ, and ψ. Therefore, R_q is spanned by

$$
\begin{aligned}
&\left(\frac{\partial r_x}{\partial\theta}, \frac{\partial r_y}{\partial\theta}, \frac{\partial r_z}{\partial\theta}, 1, 0, 0\right) \\
&\left(\frac{\partial r_x}{\partial\phi}, \frac{\partial r_y}{\partial\phi}, \frac{\partial r_z}{\partial\phi}, 0, 1, 0\right) \\
&\left(\frac{\partial r_x}{\partial\psi}, \frac{\partial r_y}{\partial\psi}, \frac{\partial r_z}{\partial\psi}, 0, 0, 1\right)
\end{aligned}
\tag{4.159}
$$

These derivatives are easily calculable.

Now let

$$
\mathbf{B}_{\theta\phi\psi} = \begin{pmatrix} s_\psi & c_\psi & 0 \\ -\frac{c_\psi}{s_\theta} & \frac{s_\psi}{s_\theta} & 0 \\ \frac{c_\psi c_\theta}{s_\theta} & -\frac{s_\psi c_\theta}{s_\theta} & 1 \end{pmatrix}
\tag{4.160}
$$

The columns of $\mathbf{B}_{\theta\phi\psi}$ are the principal axes of the object for some configuration q expressed in (θ,ϕ,ψ) coordinates.

This matrix may be derived by noting that rotations in the θ, ϕ, and ψ directions correspond to rotations about the three vectors $(-\sin\phi, \cos\phi, 0)$, $(0, 0, 1)$, and $(\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta)$, respectively, where these vectors are expressed in terms of standard x, y, and z coordinates.

$\mathbf{R}_{\theta\phi\psi}$ maps principal axes coordinates to x,y,z coordinates, while $\mathbf{B}_{\theta\phi\psi}$ maps principal axes coordinates to θ, ϕ, ψ coordinates.

So, if

$$
\mathbf{v} \in T_q,
\tag{4.161}
$$

and

$$\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2) \tag{4.162}$$

when expressed in principal axes coordinates, then

$$\mathbf{v} = (\mathbf{R}_{\theta\phi\psi}\,\mathbf{v}_1,\ \mathbf{B}_{\theta\phi\psi}\,\mathbf{v}_2) \tag{4.163}$$

when expressed in $x, y, z, \theta, \phi, \psi$ coordinates.

This permits calculation of N_q and $R_q^{\perp}$ in terms of $x, y, z, \theta, \phi, \psi$ coordinates, given the earlier derivations relative to principal axes coordinates. Of course, for any given object, such as one with a uniform mass distribution, simpler methods may exist.

Finally, note that the sliding tangent space is spanned by

$$\begin{aligned} \mathbf{t}_x &= (1,\ 0,\ 0,\ 0,\ 0,\ 0) \\ \mathbf{t}_y &= (0,\ 1,\ 0,\ 0,\ 0,\ 0) \end{aligned} \tag{4.164}$$

while the real space normal is given by

$$\mathbf{n}_0 = (0,\ 0,\ 1,\ 0,\ 0,\ 0) \tag{4.165}$$

This shows that the projections π_T and π_N are easy to calculate.

4.7. Summary

This chapter has developed a generalization of the real space friction cone for representing friction in configuration space. The configuration space friction cone models both the reaction forces and the reaction torques generated by a point of contact. The friction cone thus represents the range of generalized reaction forces that can be generated by a point of contact. For multiple points of contact, the composite configuration space friction cone is simply the vector sum of the individual one-point contact friction cones.

The friction cone is a useful geometric tool. In the case of generalized damper dynamics, the friction cone offers a simple geometric test for sticking. A commanded velocity will cause sticking precisely when it points into the friction cone. More generally, a range of velocities, such as those in the velocity uncertainty cone, can cause sticking exactly when the intersection of the negative velocity range with the friction cone is not empty.

A planning scheme can use this geometric sticking test to determine surfaces in configuration space on which a motion might stick. In particular, the backprojection algorithm can use the sticking test to mark regions which should be avoided. The remaining regions are sliding regions.

The friction cone also makes geometrically explicit points at which motion ambiguities might occur. For example, if a one-point friction cone dips below the tangent plane, then a motion may or may not stick at that point. Additionally, when the edges of the individual friction cones which comprise a composite friction cone are not coplanar, then a variety of reaction forces can arise in response to an applied force. Effectively, the distribution of reaction forces among the points of contact is indeterminate. Consequently, the resulting motion is ambiguous.

The chapter also considered a method for computing reaction forces within Newton's world, under arbitrary initial conditions. This method takes account of second order terms arising from non-zero rotational velocity. Computing reaction forces is useful to a planner that operates in Newton's world. Additionally, this capability is required of any simulator that wishes to model or verify the results of a planning scheme.

Finally, while most of the chapter considered the three dimensional configuration space corresponding to a rigid object translating and rotating in the plane, the end of the chapter presented an analysis of friction in the full six dimensional configuration space of a three dimensional rigid object.

This empty page was substituted for a blank page in the original document.

Chapter Five

Conclusion

5.1. Summary

This thesis has developed geometrical tools for planning motions in the presence of uncertainty. An algorithm was presented for backprojecting from goal states. The backprojection algorithm computed regions from which certain motions were guaranteed to successfully accomplish a desired task. The connection between these backprojection regions and the structure of goal sets and termination conditions was investigated. Finally, a representation of friction in configuration space was developed.

The thesis employed a configuration space representation in order to reduce motions of objects to motions of points. The configuration space of a moving object is the parameter space corresponding to the object's degrees of freedom. Tasks are specified in configuration space by desired goal regions. These regions correspond to sets of configurations that the moving object should attain in order that the task be successfully accomplished.

Uncertainty appeared both in the sensors that interrogated the state of a system, as well as in the control commands that changed the state of the system. The sensors assumed throughout this thesis were position and force sensors. Uncertainty was represented by a ball of possible interpretations of measured sensor values. The dynamics of the underlying system were modelled by the generalized damper. This is a first order system which permits identification of velocities and forces. Thus, the force sensors were equivalent to velocity sensors. Additionally, with generalized damper dynamics, the control commands were desired velocities. The effective velocity commands were assumed to lie in some uncertainty cone about the desired velocity command.

The backprojection algorithm operated by erecting the negative commanded velocity cone at undesirable regions in configuration space. In essence, the algorithm geometrically solved the generalized damper differential equation, in order to determine all regions from which a point could possibly move to a region in which the point could stick on a non-goal surface. From the remaining regions all points were guaranteed to reach the goal. They would do so either by moving through free space, or by sliding on configuration space surfaces.

The thesis considered the power of termination predicates in relation to the structure of backprojection regions. In order to accomplish a task, it is not sufficient

for a point to merely reach its goal. Additionally, it is necessary for the plan executor to recognize entry of the point into the goal. Termination predicates that use sensors to interrogate the state of the moving point must be employed. These termination predicates halt a motion once they are certain that all interpretations of the sensor values are included in a goal (see Lozano-Pérez, Mason, and Taylor (1983)).

A structure equation was presented which showed that all regions from which motions are guaranteed to recognizably enter a goal are backprojections of particular subsets of the goal. These subsets are the first entry sets of trajectories, that is, the first possible positions at which a trajectory is about to enter a goal region. The termination predicate assumed in formulating this structure equation was allowed only to consider current sensor values and remember the initial starting region. It was not permitted to remember previous sensor values, or to possess a clock with which to measure time.

The basis of the backprojection algorithm lay in exploiting surfaces that could guide motions to the goal, while avoiding surfaces on which motions could stick. In order to devise a systematic method for deciding whether a motion may stick on a given surface, it was necessary to develop a representation of friction in configuration space. Friction was represented by a friction cone analogous to the real space friction cone. The configuration space friction cone differs from its real space counterpart by the addition of torques. These torques are the induced torques generated by the normal and tangential components of force contained in the real space friction cone.

One point contact in real space corresponds to movement along a hyper-surface in configuration space, while multiple point contact in real space corresponds to movement along the intersection of multiple hyper-surfaces in configuration space. The friction cone for multiple point contact was found to be the vector sum of the individual one-point contact friction cones.

Within the generalized damper framework, the configuration space friction cone permitted a simple geometrical test for sticking. Sticking could occur exactly when a commanded velocity pointed into the friction cone. Thus sticking was possible whenever the negative commanded velocity cone intersected the configuration space friction cone.

The thesis also considered the friction cone representation within the framework of Newton's world. In particular, the thesis investigated the computation of reaction forces given arbitrary applied forces and arbitrary initial velocities. The computation amounted to a normal projection of an effective applied force. The effective applied force took into account the effect of second order terms, such as centripetal and coriolis forces.

Finally, the configuration space representation made explicit in geometrical terms conditions under which motion ambiguities could arise. These could occur for multiple contact whenever the edges of the individual friction cones were non-planar. Additionally, for single point contact, ambiguities were found to be possible

whenever the configuration space friction cone dipped below the configuration space tangent plane.

5.2. Suggestions for Future Work

5.2.1. Model Uncertainty

The thesis assumed that uncertainty was concentrated in sensors and control. The basic shape of the environment was assumed to be modelled perfectly. Future work should consider relaxing this assumption. Undoubtedly some model error can be incorporated into the position and velocity sensors' uncertainties. For example, an error in the relative positions of two objects may be represented by uncertainty in the position sensor, while an error in the incline of a surface may be represented by uncertainty in the velocity sensor. Furthermore, all errors in modelling are easily accounted for by the formal framework of pre-images.

At issue is the problem of computing pre-images or backprojections. Backprojections reflect the possible trajectories that a point may follow, subject to the uncertainty in commanded velocities. Uncertainty in the basic shape of environment models implies an extra degree of uncertainty in the trajectories that emanate from a given point. Clearly, if arbitrary model uncertainty is allowed, then very little can be said about these trajectories. Thus the question is twofold. First, one should decide on the type of model uncertainty that one is willing to tolerate. Second, one should represent this uncertainty in a fashion that permits pre-image or backprojection computation. In particular, one must worry about the meaning of backprojections in the presence of model uncertainty.

5.2.2. Implementation of Backprojections for Full Six Dimensions

The thesis outlined an algorithm for computing backprojections in two and three dimensional configuration spaces. Future work should consider implementations for the full six dimensional configuration space arising from a rigid body with three translational and three rotational degrees of freedom. A slice algorithm employing three dimensional slices, much as the algorithm of this thesis employed two dimensional slices, seems attractive.

Additional work might focus on a replacement for the slice algorithm that avoids linearization approximations. Specifically, a method for computing backprojections, using the exact generalized damper differential equation while in frictional contact with surfaces, is desirable. Such a method might consider alternate means to computing backprojections in six dimensions, while retaining the notions of goal reachability and recognizability.

5.2.3. Termination Predicates with Time

The thesis showed that backprojections are related via the structure equation to pre-images that use termination predicates without history or time. More powerful termination predicates remember previous sensor values and have a sense of time. Future work should consider re-introducing time and state. The task that arises, within the framework of this thesis, is the definition and computation of

backprojections and forward projections. Definitions of these projections should incorporate some dependence on time. The computation of backprojections indexed by time, and a description of their relationship to pre-images that employ termination predicates with a sense of time, are open problems whose solutions would considerably increase the power of motion planners.

5.2.4. Representation of Infinite Sets

The thesis briefly outlined a method for reducing infinite collections of sets to finite collections using the notion of distinguishability. Future work should consider more general reduction operations. For example, although it may not be possible to reduce an infinite collection of sets to a finite number of sets, it may be possible to characterize the types of sets in the collection in a finite fashion. Examples of this reduction were presented earlier in Chapter 3. More general work in this area is needed.

5.2.5. Constraints Between Forward Projections and Goal Sets

The end of Chapter 3 developed constraints that must exist between goal sets and forward projections, in order that the goals be good bases for backprojections. It was noted that in practical applications these infinite constraints must be approximated by finite representations. Further investigation of the dependencies between forward projections and good goal sets is necessary in order to improve this approximation. Additionally, there is considerable room for work on intra-goal dependencies. The computation of all goal sets that satisfy the intra-goal dependencies is still an open problem. A better understanding of the shape of good goal sets would increase the range of pre-images that are calculable using backprojections.

5.2.6. Frictional Ambiguities

It would be interesting to better understand the situations in which frictional ambiguities can arise. This entails abandoning the models of Coulomb friction and classical mechanics. A better prediction of possible motions in ambiguous conditions would permit more accurate planning.

5.2.7. Second Order Systems

This thesis assumed a generalized damper as the underlying model of dynamics. Much work remains to be done in extending the results of this thesis to general second order systems. Planning in a second order system might contract the position and velocity uncertainty balls. It would certainly add another type of interpretation restriction on the set of sensor values, thereby possibly increasing the power of termination predicates. Furthermore, using second order systems, it would become possible to specify both position and velocity goals, thereby enlarging the class of tasks solvable by a planner. At issue is the computation of pre-images and backprojections in phase space. Both a theoretical investigation and a practical implementation of a planner for a second order system deserve focused attention.

5.2.8. Probabilistic Uncertainty

This thesis assumed a very simple model of uncertainty. In particular, errors in sensing and control were conservatively bounded by a ball or cone about the nominal sensor value or control command. Pre-images and backprojections define regions from which particular motions are guaranteed to succeed for all possible sensor and command errors. In general, one would expect sensor and control uncertainties to underlie some set of probability distributions. Consequently, one should consider constructing regions from which motions are guaranteed to reach goal regions with particular probabilities.

Within the pre-image framework, motions are guaranteed to attain goals, although the particular goals attained may not be predictable at the beginning of a motion. Placing probability distributions on the shape of uncertainty extends this model. A continuum of goals is attainable, subject to some probability distribution. Failure to achieve a particular goal is recognized with the aid of sensors, much as a termination predicate decides which goal has been achieved in the pre-image model of planning.

The precise consequences of permitting more general types of uncertainty warrant further study.

5.2.9. Computability

Further work is needed in the area of computability. A characterization of the types of constraints in which the pre-image problem is computable is of considerable interest. For example, this thesis conjectured that backprojections may be computed in environments with a finite number of algebraic constraints. Future research should either substantiate this conjecture, or suggest further restrictions on the environment required to solve the backprojection problem.

References

Arnold, V. I. *Mathematical Methods of Classical Mechanics*, Springer-Verlag, New York, 1978.

Boothby, W. M. *An Introduction to Differentiable Manifolds and Riemannian Geometry*, Academic Press, New York, 1975.

Brady, M. et. al. (eds) *Robot Motion: Planning and Control*, MIT Press, Cambridge, Massachusetts, 1982.

Brooks, R. A. "Symbolic Error Analysis and Robot Planning," *International Journal of Robotics Research* **1**, 4 (1982).

Brooks, R. A. "Solving the Find-Path Problem by Good Representation of Free Space," *IEEE Transactions on Systems, Man, and Cybernetics* **SMC–13** (1983).

Brooks, R. A., and T. Lozano-Pérez "A Subdivision Algorithm in Configuration Space for Findpath with Rotation," *Proceedings, 8th International Joint Conference on Artificial Intelligence*, Karlsruhe, Germany, August 1983.

Burridge, R., V. T. Rajan, and J. T. Schwartz "The Peg-In-Hole Problem: Statics and Dynamics of Nearly Rigid Bodies in Frictional Contact," 1983, to appear.

Donald, B. R. "Motion Planning with Six Degrees of Freedom," S. M. Thesis, AI-TR-791, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1984.

Donald, B. R. personal communication, August 1984.

Dufay, B., and J. Latombe "An Approach to Automatic Robot Programming Based on Inductive Learning," in Brady, M., and R. Paul (eds), *Robotics Research: The First International Symposium*, MIT Press, 1984.

Erdmann, M. A. On Motion Planning with Uncertainty, S. M. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1984.

Hopcroft, J. E., D. A. Joseph, and S. H. Whitesides "On the Movement of Robot Arms in 2-Dimensional Regions," Department of Computer Science, Cornell University, TR 82–486, March 1982.

Hopcroft, J. E., and G. Wilfong "On the Motion of Objects in Contact," Department of Computer Science, Cornell University, TR 84–602, May 1984.

Inoue, H. "Force Feedback in Precise Assembly Tasks," Artificial Intelligence Laboratory, Massachusetts Institute of Technology, AIM–308, August 1974

(Reprinted in Winston, P. H. and R. H. Brown (eds), *Artificial Intelligence: An MIT Perspective*, MIT Press, 1979).

Jellet, J. H. *A Treatise on the Theory of Friction*, MacMillan, London, 1872.

Kane, T. R., and D. A. Levinson "Successive Finite Rotations," *Journal of Applied Mechanics* **Vol 5** (December 1978), 944–5.

Lewis, H. R., and C. H. Papadimitriou *Elements of the Theory of Computation*, Prentice-Hall, Englewood Cliffs, New Jersey, 1981.

Lozano-Pérez, T. "The Design of a Mechanical Assembly System," Artificial Intelligence Laboratory, Massachusetts Institute of Technology, AI-TR-397, 1976 (Reprinted in part in Winston, P. H. and R. H. Brown (eds), *Artificial Intelligence: An MIT Perspective*, MIT Press, 1979).

Lozano-Pérez, T. "Automatic Planning of Manipulator Transfer Movements," *IEEE Transactions Systems, Man, Cybernetics* **SMC–11**, 10 (1981), 681–689 (Reprinted in Brady, M. et. al. (eds), *Robot Motion*, MIT Press, 1982.).

Lozano-Pérez, T. "Spatial Planning: A Configuration Space Approach," *IEEE Transactions Computers* **C-32**, 2 (February 1983).

Lozano-Pérez, T. "Task Planning," in Brady, M. et al. (eds), *Robot Motion*, MIT Press, 1982.

Lozano-Pérez, T., M. T. Mason, and R. H. Taylor "Automatic Synthesis of Fine-Motion Strategies for Robots," *Proceedings, International Symposium of Robotics Research*, Bretton Woods, New Hampshire, August 1983 (Also published in *International Journal of Robotics Research* **3**, 1 (1984)).

Lozano-Pérez, T., and M. A. Wesley "An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles," *Communications of the ACM* **22**, 10 (1979).

Mason, M. T. "Compliance and Force Control for Computer Controlled Manipulators," *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-11**, 6 (June 1981), 418–432 (Reprinted in Brady, M. et. al. (eds), *Robot Motion*, MIT Press, 1982.).

Mason, M. T. "Manipulator Grasping and Pushing Operations," Ph.D. Thesis, AI-TR-690, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, June 1982.

Mason, M. T. "Compliant Motion," in Brady, M. et al. (eds), *Robot Motion*, MIT Press, 1982.

Mason, M. T. "Automatic Planning of Fine-Motions: Correctness and Completeness," Computer Science Department and Robotics Institute, Carnegie-Mellon University, October 1983.

Nievergelt, J., and F. P. Preparata "Plane-Sweep Algorithms for Intersecting Geometric Figures," *Communications of the ACM* **25**, 10 (1982).

Nilsson, N. J. *Principles of Artificial Intelligence*, Tioga Publishing Co., Palo Alto, California, 1980.

Ó'Dúnlaing, C., and C. Yap "The Voronoi Diagram Method of Motion-Planning: I. The Case of a Disc.," Computer Science Department, Courant Institute of Mathematical Sciences, September 1982.

Ó'Dúnlaing, C., M. Sharir, and C. K. Yap "Retraction: A New Approach to Motion-Planning," Computer Science Department, Courant Institute of Mathematical Sciences, and Tel-Aviv University, November 1982.

Ohwovoriole, M. S., J. W. Hill, and B. Roth "On the Theory of Single and Multiple Insertions in Industrial Assemblies," *10th International Symposium on Industrial Robots, 5th International Conference on Industrial Robot Technology*, Milan, Italy, March 5–7, 1980.

Ohwovoriole, M. S., and B. Roth "A Theory of Parts Mating for Assembly Automation," *Proceedings of the Robot and Man Symposium 81*, Warsaw, Poland, September 1981.

Paul, L. *Robot Manipulators: Mathematics, Programming, and Control*, MIT Press, Cambridge, Massachusetts, 1981.

Raibert, M. H., and J. J. Craig "Hybrid Position/Force Control of Manipulators," *Journal of Dynamic Systems, Measurement, and Control* **102** (June 1981), 126–133 (Reprinted in Brady, M. et. al. (eds), *Robot Motion*, MIT Press, 1982.).

Reif, J. H. "The Complexity of the Mover's Problem and Generalizations," *Proceedings, 20th Symposium on the Foundations of Computer Science*, 1979.

Salisbury, J. K. "Active Stiffness Control of a Manipulator in Cartesian Coordinates," *IEEE Conference on Decision and Control*, Albuquerque, New Mexico, November 1980.

Schwartz, J. T., and M. Sharir "On the Piano Movers' Problem: I. The Case of a Two-Dimensional Rigid Polygonal Body Moving Amidst Polygonal Barriers," Computer Science Department, Courant Institute of Mathematical Sciences, New York University, Technical Report No. 39, 1981.

Schwartz, J. T., and M. Sharir "On the Piano Movers' Problem: II. General Techniques for Computing Topological Properties of Real Algebraic Manifolds," Computer Science Department, Courant Institute of Mathematical Sciences, New York University, Technical Report No. 41, 1982.

Schwartz, J. T., and M. Sharir "On the Piano Movers' Problem: III. Coordinating the Motion of Several Independent Bodies: The Special Case of Circular Bodies Moving Amidst Polygonal Barriers," *International Journal of Robotics Research* **2**, 3 (1983).

Simunovic, S. N. "Force Information in Assembly Processes," *Proceedings, 5th International Symposium on Industrial Robots*, Chicago, Illinois, September 22–24, 1975.

Simunovic, S. N. An Information Approach to Parts Mating, Ph.D. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1979.

Symon, K. R. *Mechanics*, Addison-Wesley, Reading, Mass., 1971.

Taylor, R. H. "The Synthesis of Manipulator Control Programs from Task-Level Specifications," Artificial Intelligence Laboratory, Stanford University, AIM-282, July 1976.

Udupa, S. M. "Collision Detection and Avoidance in Computer Controlled Manipulators," *5th International Joint Conference on Artificial Intelligence*, Massachusetts Institute of Technology, 1977.

Udupa, S. M. Collision Detection and Avoidance in Computer Controlled Manipulators, Ph.D. Thesis, Department of Electrical Engineering, California Institute of Technology, 1977.

Whitney, D. E. "Force Feedback Control of Manipulator Fine Motions," *Journal of Dynamic Systems, Measurement, and Control* (June 1977), 91–97.

Whitney, D. E. "Quasi-Static Assembly of Compliantly Supported Rigid Parts," *Journal of Dynamic Systems, Measurement, and Control* **104** (March 1982), 65–77 (Reprinted in Brady, M. et. al. (eds), *Robot Motion*, MIT Press, 1982.).

Will, P. M., and D. D. Grossman "An Experimental System for Computer Controlled Mechanical Assembly," *IEEE Transactions on Computers* **C-24**, 9 (1975), 879–888.

CS-TR Scanning Project
Document Control Form

Date : 1/25/96

Report # AI-TR-810

Each of the following should be identified by a checkmark:
Originating Department:

☒ Artificial Intellegence Laboratory (AI)
☐ Laboratory for Computer Science (LCS)

Document Type:

☒ Technical Report (TR) ☐ Technical Memo (TM)
☐ Other:_____

Document Information

Number of pages: 264 (272-IMAGES)
Not to include DOD forms, printer intstructions, etc... original pages only.

Originals are:
☒ Single-sided or
☐ Double-sided

Intended to be printed as :
☐ Single-sided or
☒ Double-sided

Print type:
☐ Typewriter ☐ Offset Press ☒ Laser Print
☐ InkJet Printer ☐ Unknown ☐ Other:_____

Check each if included with document:

☒ DOD Form (2) ☐ Funding Agent Form ☒ Cover Page
☒ Spine ☐ Printers Notes ☐ Photo negatives
☐ Other: _____

Page Data:

Blank Pages (by page number): _____

Photographs/Tonal Material (by page number): _____

Other (note description/page number):

Description :	Page Number:
IMAGE MAP: (1-264) UN#'ED TITLE PAGE, BLANK,	
FUNDING AGENT, BLANK, 2-9, UN#'ED BLANK,	
~~[illegible]~~ 11-27, UN# BLANK, 29-251,	
UN#'ED BLANK, 253-261	
(265-272) SCAN CONTROL, COVER, SPINE, DOD (2), TARGET'S (3)	

Scanning Agent Signoff:

Date Received: 1/25/96 Date Scanned: 2/9/96 Date Returned: 2/15/96

Scanning Agent Signature: Michael W. Cook

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AI-TR-810	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* On Motion Planning with Uncertainty		5. TYPE OF REPORT & PERIOD COVERED memorandum 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR*(s)* Michael Andreas Erdmann		8. CONTRACT OR GRANT NUMBER*(s)* N00014-81-K0494 N00014-80-C-0505 N00014-82-K-0344
9. PERFORMING ORGANIZATION NAME AND ADDRESS Artificial Intelligence Laboratory 545 Technology Square Cambridge, Massachusetts 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Projects Agency 1400 Wilson Blvd Arlington, Virginia 22209		12. REPORT DATE memorandum 13. NUMBER OF PAGES 261
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)* Office of Naval Research Information Systems Arlington, Virginia 22217		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Distribution of this document is unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

Distribution is unlimited

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Motion planning
Mechanical assembly
Parts mating
Robotics
Configuration space
Friction
Compliance
Uncertainty

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

Robots must successfully poan and execut tasks in the presence of uncertainty. Uncertainty arises from errors in modelling, sensing, and control. Plannning in the presence of uncertainty constitutes one facet of the general motion planning problem in robotics. This problem is concerned with the automatic sythesis of motion strategies from high level task specifications and geometric models of environments.

(OVER)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

Block 20 cont.

In order to develop successful motion strategies, it is necessary to understand the effect of uncertainty on the geometry of object interactions. Object interactions, both static and dynamic, may be represented in geometrical terms. This thesis investigates geometrical tools for modelling and overcoming uncertainty.

The thesis describes and algorithm for computing backprojections of desired task configurations. Task goals and motion states are specified in terms of a moving object's configuration space. Backprojections specify regions in configuration space from which particular motions are guaranteed to accomplish a desired task. The backprojection algorithm considers surfaces in configuration space that facilitate sliding toward the goal, while avoiding surfaces on which motions may prematurely halt.

In executing a motion from a backprojection region, a plan executor must be able to recognize that a desired task has been accomplished. Since sensors are subject to uncertainty, recognition of task success is not always possible. The thesis considers the structure of backprojection regions and of task goals that ensures goal recognizability.

The thesis also develops a representation of friction in configuration space, in terms of a friction cone analogous to the real space friction cone. The friction cone provides the backprojection algorithm with a geometrical tool for determining points at which motions may halt.

Scanning Agent Identification Target

Scanning of this document was supported in part by the **Corporation for National Research Initiatives,** using funds from the **Advanced Research Projects Agency** of the **United states Government** under Grant: **MDA972-92-J1029.**

The scanning agent for this project was the **Document Services** department of the **M.I.T Libraries.** Technical support for this project was also provided by the **M.I.T. Laboratory for Computer Sciences.**

Scanned

Date: 2/9/1996

M.I.T. Libraries
Document Services