A Control Basis for Haptically-Guided Grasping and Manipulation[1]

Kamal Souccar, Jefferson A. Coelho Jr. and Roderic A. Grupen

Laboratory for Perceptual Robotics
Departament of Computer Science
University of Massachusetts, Amherst MA 01003

Technical Report 98-46
October 21, 1998

Abstract

This report discusses the construction of adaptive control structures for teams of semi-autonomous robots. A flexible architecture for adaptive control was developed that forms control policies for allocating resources in open environments and tasks. This architecture is designed to identify the control context and to specify coordinated robot behavior. To identify the control context we form categories of distinct dynamic responses with which to augment state information — a kind of hidden state removal. To evaluate the feasibility of the approach, we have applied it to grasping and manipulation tasks with multifingered robot hands. In this class of tasks, the dynamic plant that we wish to control depends directly on the geometry of the object. We have constructed a working platform that uses the dynamic response of the controller to classify types of plants and uses this information to improve the grasp/manipulation strategy. Since resource alternatives in this platform involve various combinations of fingers, the control context is used to learn resource schedules that coordinate the allocation and de-allocation of fingers.

[1]This work is supported in part by NSF CDA-8922572, IRI-9116297, IRI-9208920, and CNPq 202107/90.6

Contents

1 Introduction

Grippers, hands, and end-tooling constitute the business-end of any robot in the world today. However, commercially available hardware has adapted little in 20 years. Manufacturing flexibility is limited by the time it takes to design and machine the gripper-jaw surfaces (supplied as rectangular soft-jaw blanks). These approaches are simple from a control standpoint, but contribute little to systems technology for reliable operation in open environments. A significant number of manufacturing tasks would be handled more efficiently if flexible tooling could be controlled to grasp a variety of part geometries. It is for this reason, that we have selected coordinated control of grasping and manipulation tasks as the experimental platform for evaluating this architecture.

Teleoperated applications seek to overcome the dearth of control technologies for flexible machines by including the human in the loop. Teleoperation applications to date rely on highly trained human operators who deal with kinematic differences and impoverished sensory feedback to execute a remote task. Operators require a great deal of skill and expend a great deal of effort to control a physical process from a distance and these approaches don't scale well to multi-robot, or complex, non-anthropomorphic robot geometries.

The answer is to introduce locally autonomous control and flexible hardware to permit the robot to function in a safe and productive manner without continuous human intervention. Dextrous robotic hands, such as those designed by Salisbury and Jacobsen in the early 1980s are important testbeds for studying physical and kinematic relationships between degrees of freedom and manipulability. But these hands were designed as sedentary testbeds, not for use on robot arms. Integrated hand-and-arm systems with these dextrous end effectors tend to be massive, complex, and fundamentally prone to failure. However, next-generation manipulator hardware is developing and is just now becoming widely available for these applications. For example, Barrett Technology now markets the Whole Arm Manipulator (WAM) with an integrated BH8 robotic hand. These technologies are inherently more flexible, but to date no control methodology is capable of exploiting the full range of sensory and motor function afforded. Barrett has identified markets in automobile manufacture waiting for practical technologies for accommodating variations in part shapes automatically. In addition to increased mechanical scale and robustness, this market hinges on the development of application software that provides an intuitive human interface for user programming, as well as a basis for automatic programming, adaptation, and optimization.

While a great deal of progress has been made concerning robot control and task level planning, robot autonomy remains beyond the reach of traditional technologies. Even when a human teleoperator directs the behavior of the system, performance is typically marginal and requires a great deal of effort on the part of the operator. New approaches are required to develop **locally competent** systems — that is, systems that exploit instantaneous sensor feedback to achieve and/or maintain critical properties during execution. To achieve this objective, the robot system must be capable of deriving its own models of the robot/world interaction rather than depending on a human operator. Such systems will be dramatically more responsive due to the tight coupling of sensing and control, they will be more robust with respect to uncertainty, and they will be more flexible and resourceful because operators will not be required to map human strategies onto unfamiliar kinematic structures with non-anthropomorphic dynamics. If such technologies existed, they would enable a wide variety of applications: hazardous waste removal, nuclear plant and underwater maintenance, and oceanographic exploration, in which, currently, humans risk life and limb.

This report describes an integrated, 14 DOF hand/arm platform that has been used to verify performance predictions and have demonstrated the feasibility of deploying this technology. It describes many of the component technologies required for locally autonomous, haptically driven manipulation systems, including:

1. a feasibility analysis of the control basis approach employing auxiliary variables derived from categories of characteristic dynamic responses,
2. the most functional, integrated experimental platform for grasping and manipulation ever de-

veloped consisting of a three-fingered Salisbury hand with Brock sensors on a GE P50 robot arm,

3. a detailed sensor characterization and calibration report,
4. a critical comparison of representations for dynamic models, and a set of experimentally-derived haptic categories for test objects,
5. results from an "open" manipulation task used to evaluate the performance of the architecture.

Innovation An important aspect of intelligent control lies in its ability to improve performance with experience. Adaptive grasping and manipulation systems, however, face the shear complexity of the domain; tasks include a broad range of contexts, object geometries vary significantly, hands can be complex and highly redundant, and the environment is often unstructured and uncertain. The central innovation in this work is the "control basis," a technology for enumerating large varieties of closed-loop controllers employing general subsets of the available sensory and motor resources, and for writing programs that automatically explore this "instruction set" in an efficient manner. The control basis solves critical problems related to the size of the space used to represent sensory and motor policies, structures exploration, provides an convenient means of incorporating external domain knowledge, and leads to safe and correct behavior during learning. We have demonstrated that grasp formation can be posed as a closed-loop response to contact feedback, with inherent error suppression, and without *a priori* models of the object's surface geometry. Such a model for reflexive grasp formation is capable of amortizing the cost of model building and grasp planning, and provides a mechanism for executing grasp plans when they exist.

Figure 1 depicts the closed-loop control scheme proposed. The plant consists of a k-contact grasp configuration on an unknown object geometry. Conceptually, the plant transforms a contact configuration into an object wrench. The grasp control basis determines how to apply inputs to the plant to achieve a reference wrench. The grasp controller is designed to be robust with respect to variations in object geometry and contact friction properties. The dynamics of grasp formation are modeled to associate the expected quality of equilibrium grasp configurations with precursor grasp states. These models form the basis for categories of control contexts and help to identify appropriate grasping and manipulation policies. Each training instance provides a trace through the state space which is labeled with an estimate of the ultimate quality of the control policy for this object. The quality metric reflects the minimum coefficient of friction required to stabilize the grasp. This representation is used to support an adaptive control compensator, π_o. The control compensator provides a means of contact resource allocation and object recognition based on the history of tactile and kinesthetic data during grasp formation.

Figure 1 depicts control laws, Φ, designed to be a basis for a family of composite controllers. The manner in which controllers interact and constraints on how they may be combined limit the size of the search space for control composition policies. A closed-loop controller is defined by associating control laws in Φ with sensors and effectors in the system. Combinations $\Phi \times 2^{\Omega_e} \times 2^{\Omega_s}$, where Ω_e and Ω_s represent the effector and sensor resources, respectively, yield a finite number of equilibria for each unique control binding. In practice, the number of bindings that must be considered can be reduced dramatically *a priori*.

In general, the composition policy, π_c, specifies combinations of controllers during the execution of the task. The composition may take the form of a concurrent control activation, where control interactions yield favorable system equilibria, as well as sequences over these concurrent control laws. In the latter case, the deterministic action of the concurrent control law supports reasoning about end-to-end problem solving behavior by generating robust intermediate equilibrium states. We have experimented with hand-crafted composition policies [2] and policies derived from DP-based learning algorithms [8]. In both cases, the composition policy was evaluated over a broad range of control contexts. Effective policies coordinate control activation robustly with respect to run-time context. We believe that such an approach to constructing control on-line yields responsive run-time systems with predictable

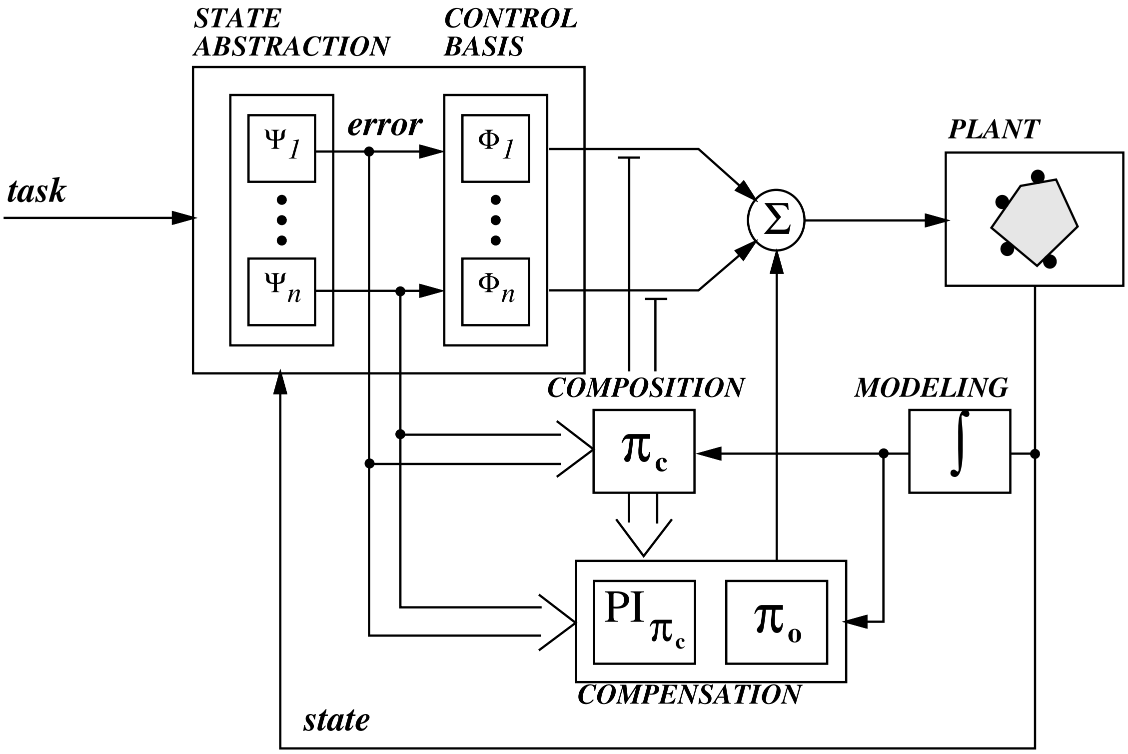

Figure 1: The Generic Element of Control and Adaptation

performance (stability, convergence, and error suppression), and as a consequence, can be robust with respect to incomplete information.

As illustrated in Figure 1, the haptic model under a fixed composition policy, π_c, is used to determine when compensation is required in the controller. Initially, this compensation can take the form of a random control action (this works very well for breaking switch instabilities, for instance). Eventually, more effective policies, π_o, can be learned through experience. The haptic model for policy π_c predicts when compensation is necessary, and manages both the training and use of the compensator, π_o.

1.1 Technical Objectives

1.1.1 System Integration

Each element of the grasp control basis maps sensor readings to control actions, establishing closed-loop controllers. Sensor performance is critical to the eventual performance of the architecture. Just as the structure and dynamics of the robot mechanism affect its solution space, so too does the fidelity of its sensors. Contact state estimates must be well-behaved to produce stable behavior in the grasp control basis. We consider only the sensors and control issues and construct a working prototype capable of grasping unknown, regular object geometries using instantaneous tactile feedback information exclusively.

In follow up studies, a fully integrated sensorimotor mechanism will be proposed that learns to exploit the intrinsic dynamics of the robot in a broad range of tasks and environments.

Task breakdown:

Sensor Characterization, Calibration, and Signal Processing Three Brock fingertip sensors are modeled with respect to noise, drift, and sensitivity to temperature and mechanical vibration. Sensor calibration is accomplished by applying known contact loads over the sensor surface. A transformation mapping sensor response to forces and moments is constructed from this calibration data.

Signal Conditioning Contact normals and positions are established by interpreting fingertip forces and moments.

Integration Recursive filtering techniques smooth raw sensor data and overcome otherwise noisy contact state estimates.

Products Delivered:

1. Detailed sensor characterization and calibration report (Section 2.2); and an
2. Integrated, haptically-guided grasping and manipulation system (Section 2).

1.1.2 Haptic Modeling/Representation

The behavior of the system can be classified by watching the evolution of the observable state over time. In some cases, this reveals distinct categories of behavior, distinguished by the observable dynamics of the plant. We have demonstrated that the identity of an object can be determined to varying degrees by watching the control error (a function of the observable state) vary over time. The resulting delay map can exhibit multiple regions of stable behavior, each of which identifies a class of repeatable system responses for which the eventual outcome can be predicted.

Task breakdown:

Model Representation The tradeoff between memory intensive (such as the error plane bitmaps illustrated in the proposal) and more compact function approximation schemes (such as neural networks) is evaluated. Error plane bitmaps are precise which may lead to higher quality compensators and object recognition, but may also take a great deal of on-line experience to construct. Neural networks should be capable of generalizing more effectively over smaller training sets, but by the same token, may loose the ability to make fine distinctions.

Model Construction Building haptic models for each object in the set of training objects requires a potentially large number of grasping trials. The potential for using combinations of on- and off-line experience together with algorithms for efficient exploration of the model's domain have been studied.

Products Delivered:

1. An analysis of tradeoffs in alternative model representations (Section 3); and
2. A set of experimentally-derived haptic models for 2 and 3 fingered grasps on a triangular, rectangular, and circular prism (Section 3). Regions in the models are labeled with an estimate of the coefficient of friction required to grasp the object if the current control policy reaches the expected outcome.

1.1.3 Compensated Grasping and Haptic Recognition

The dynamics captured by the haptic models identifies important, *auxiliary* state information that can be used to improve the evolving sensorimotor policy. For example, if it is likely that a control strategy will fail in a particular region of the model, then policies for compensating the control strategy can be identified that move to areas in the model that are likely to succeed.

Task breakdown:

Demonstration of concept: A simple, random compensation policy is examined for the integrated hand/arm platform in which random control actions are introduced whenever the haptic model predicts that excessive contact friction will be required.

Products Delivered:

1. Performance assessment report, containing a detailed characterization of benchmark performance. A statistically significant number of trials have been executed to quantify the behavior of the model-compensated control basis for grasping and object recognition. During these on-line tests, unknown objects are drawn from a set candidate objects are grasped. System performance is benchmarked by computing the average number of compensatory actions and the average total number of re-grasps per grasp/recognition task.

1.1.4 Adaptive Skill Acquisition

The experience embodied in the haptic model can be re-described as a collection of contexts, or categories of interaction between the robot hand and an object. This information can be used to fill in important state information in the grasp formation process. If the difference between a cube and a cylinder is important, then an auxiliary variable will be attached to the real state that evaluates TRUE for cubes and FALSE for cylinders, for instance. Subsequent policy can be tuned for these particular control contexts/categories.

Task breakdown:

Learning Task Adaptive skill acquisition requires (a) segmenting the haptic model into contexts, (b) on-line recognition of haptic contexts, and (c) learning which control policy to activate in each context. Task (a) can be achieved by modeling haptic models as a mixture of distributions [6], assuming a one-to-one correspondence between contexts and distributions. Task (b) is a special case of task (a). Finally, task (c) is amenable to reinforcement learning techniques.

Training Training consists of recognizing the context in which the system can be, and collecting statistics, within a reinforcement learning framework, about the performance of each control policy.

Testing The resulting system will be subjected to tests comparing grasping and object recognition performance to the benchmark system performance.

Products Delivered:

1. The component technologies have been tested on simulation. We are identifying potential applications for the control architecture developed, and working on the elaboration of follow up studies.

2 Multifingered Grasping System

The platform used is a 14 DOF hand/arm system consisting of a Salisbury dextrous hand mounted on a 5 DOF GE P50 industrial arm. The Salisbury hand is a 3-fingered hand with 3 DOF per finger. Each finger tip is equipped with a Brock sensor as described in Section 2.2.

To accomplish complex tasks, composition functions are hand-crafted as finite state supervisors in a discrete event dynamic systems (DEDS) framework [9]. By virtue of its basis in closed-loop controllers, the supervisor generates a sequence of concurrent control situations that leads the system through a set of favorable equilibria to the goal. All sensory feedback is interpreted locally within the elemental controllers. Throughout execution, safety can be guaranteed within the limits of the sensory and motor capabilities of the elemental controllers. Different tasks are achieved by different composition functions over the same basis controllers, rather than by the design of new task-specific behaviors. A more complete discussion of this approach can be found in [4].

2.1 A Control Basis for Manipulation Tasks

A detailed description of these basis controllers is beyond the scope of this document; the following provides a very brief overview. The exact set of controllers is not critical to the architecture. However, certain properties must be established to guarantee predictable behavior. In this work, it was required that candidate control functions possess unique extrema and that compositions over control functions preserve this property.

The initial control basis consists of robust solutions to three generic robot control tasks; configuration space path control (Φ_0), contact configuration control (Φ_1), and kinematic conditioning (Φ_2). Harmonic function path planning [3] is used to generate robust, reactive, and collision-free paths in configuration space. A contact configuration controller [2] is employed to move contacts based on the local object geometry. Kinematic conditioning control [5] optimizes the manipulability metric of the articulated structure while it is engaged in an interaction with the world.

Figure 2 presents the finite state supervisor employed in the reaching and grasping experiments. The control policy is a sequence of concurrent control situations (shown inside each node). Each instance of a controller has the form Φ_τ^σ. Here, Φ is an element of the control basis, σ denotes the input state set for the controller, and τ denotes the output state set. $\Phi_i \lhd \Phi_j$ indicates that controller Φ_i is running in the extended null space of controller Φ_j as presented in [7].

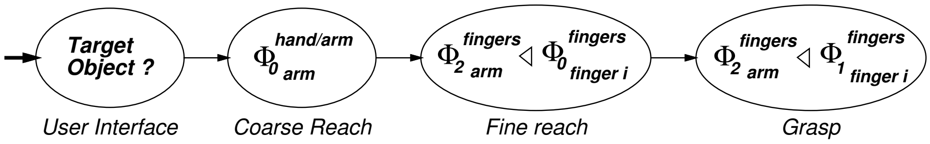

Figure 2: Finite State Machine for Reach and Grasp

Figure 2 is the most basic reach/grasp controller considered, and subsequent sections will show how to form more comprehensive controllers. The reach/grasp process begins with a user designated target object (this involves a mouse click on an overhead visual image). A path is computed (Φ_0) which navigates through the 4 dimensional configuration space of the arm in a manner consistent with constraints derived from the arm, hand, and the other objects on the table top. Next, a separate path controller (Φ_0) generates a path for each finger to the object subject to constraints derived from the other fingers. The controller that executes this path runs concurrently with a kinematic conditioning controller (Φ_2) that optimizes the conditioning of the hand by controlling the arm's configuration. This node in the reach/grasp supervisor forms fingertip contacts with the object that are subsequently manipulated. While forming a grasp, a contact configuration controller (Φ_1) for each finger runs concurrently with the same kinematic conditioning controller from the previous node.

2.2 Tactile Sensors

Each fingertip of the Salisbury hand is equipped with one six axis force/torque sensor, known as Brock sensor. Contact forces and torques are inferred by correlating the readings from eight pairs of strain gages, mounted on opposing facets of each arm of a cross, to forces and torques observed over the fingertip. Figure 3 shows sensor construction and the main physical dimensions. A more detailed technical description can be found in [10].

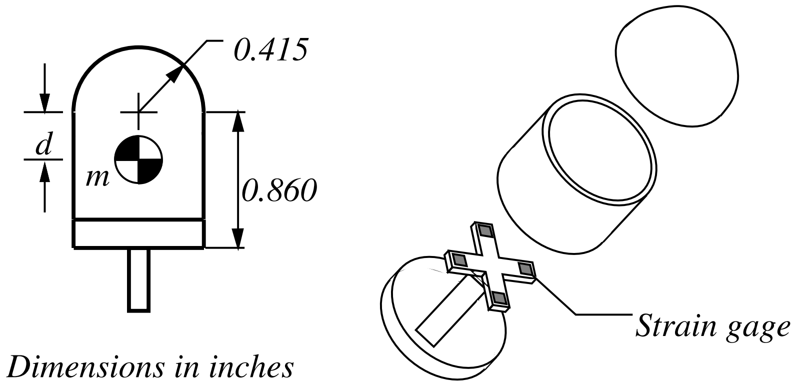

Figure 3: Geometrical configuration of the Brock sensor.

Bicchi et al. [1] have presented the closed form transformation from resultant forces and torques to contact positions and normals. For the geometry of the Brock sensors, this transformation involves solving a polynomial equation of fourth degree. A comprehensive discussion on force sensors, contact types, approximate versus exact solutions can also be found in [1].

2.2.1 Sensor Interface

Figure 4 depicts the signal conditioning circuit for a typical channel, consisting of a pair of strain gages. The circuit's output V_o is proportional to the voltage differential over both strain gages, what minimizes common mode noise (such as electromagnetic noise).

2.2.2 Offset Compensation

Every channel has offset adjustment (r_{off}), and in the absence of mechanical loads the circuit above should output $V_o = 0$ volts. However, this idealized situation does not occur in practice. Over time, stray capacitances cause the circuit to output voltages different than zero in the no load situation. To compensate for this effect, software offset compensation was introduced. Before every sensor operation (in which contact is expected), the current output voltage is stored in memory, and subsequently subtracted from every reading, until a new offset is stored.

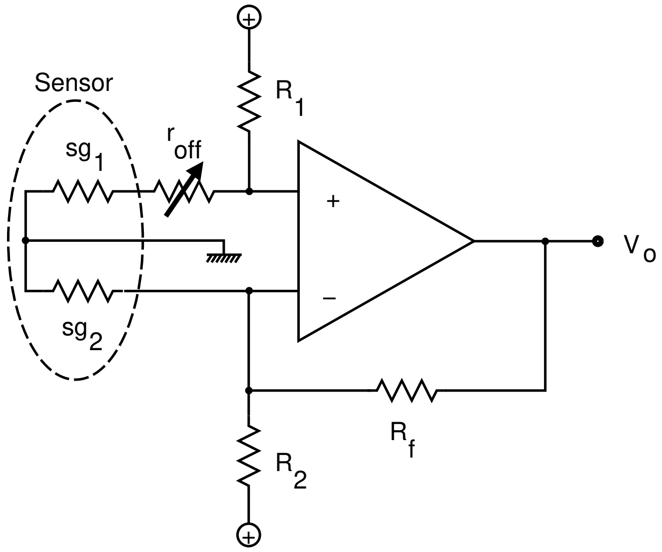

Figure 4: Sensor interface circuit.

2.2.3 Sensor Calibration

In principle, the strain gage sensor is a linear device, and there exits a linear mapping from the voltage vector to the wrench[2] vector representing the resultant forces and torques acting on the sensor surface. Equation 1 expresses this mapping in the form of a product of a matrix – the calibration matrix A — by the vector of measured voltages:

$$\vec{\omega}_{6\times1} \quad = \quad A_{6\times8} \cdot \vec{v}_{8\times1} \tag{1}$$

The purpose of sensor calibration then is to compute the matrix A that minimizes possible measurement errors. It is possible to derive A by applying eight linearly independent reference loads to the sensor and measuring the corresponding output voltages. A linear system with 48 equations and 48 variables (the elements of the calibration matrix) is then formed, and the derivation of the calibration matrix is a simple matter. This method yields good results as long as accurate estimates of the applied loads are available.

Another possibility is to apply several known loads to the sensor, and derive the calibration matrix A as the matrix that minimizes the squared error over all measurements. Compared to the first method, this method is less dependent on individual measurements, but more data are required. The sensors used in our experiments were calibrated using the second method. The decision of which method to use was influenced by the apparatus available for calibration.

The calibration procedure consists of applying known loads to the sensors and measuring the resulting voltage. Pure force loads are applied at regular intervals on the sensor surface and at five distinct orientations in order to produce reference forces and moments on the load cell. Calibration points are located on the sensor surface as shown in Figure 5.

1. **Sampling** The sensors were probed at 20 distinct positions as illustrated in Figure 5. For each position, five different pure force loads are applied to the sensor surface at four discrete load

[2] A *wrench* is a generalized, six dimensional force consisting of forces and moments on the object.

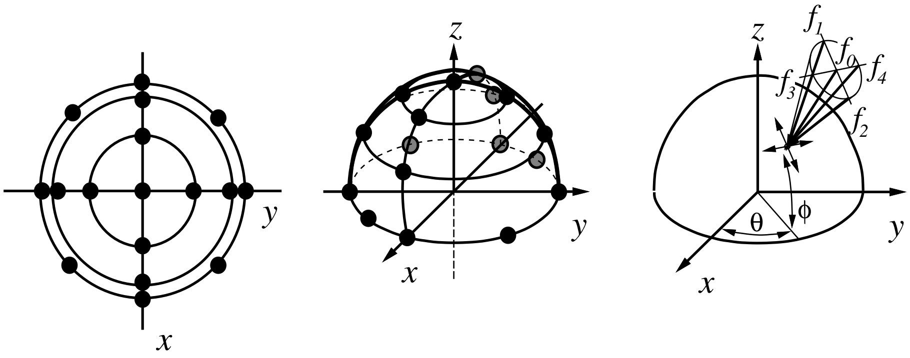

Figure 5: Load conditions used to generate reference forces and moments on the load cell.

magnitudes $\{0.4126N, 0.4787N, 0.5443N, 0.6110N\}$. That is, for each probe position, data was obtained for twenty reference wrenches.

2. **Reference Loads** During calibration, the sensor was mounted on a 2 degree of freedom fixture, capable of rotations around the x and y axes and the reference loads were applied as pure forces on the surface of the sensor. The load magnitude, and its position and orientation in sensor coordinates are recorded along with the average voltage on the eight output channels (over one hundred measurements). Measurements for which any of the eight output channels saturated are discarded. A total of 256 distinct loads were applied to each load cell and used to estimate the corresponding calibration matrix.

3. **Calibration Matrix** The calibration matrix was computed as the least squared error approximation of the experimental data (computed using the Mathematica *Fit* function). The quality of any given fit is computed as the residual error in magnitude and direction for the reference wrench. A list of points sorted by each of the error measures was produced and used to identify outliers. The estimation of the final calibration matrix involved computing a calibration matrix, eliminating significant outliers, and recalibrating. A suitable fit was obtained after a small number of iterations.

   Figure 6 contains the error histograms for the resulting fit in both magnitude and direction for the reference wrenches applied to sensor #2. The errors in magnitude for the force and moment vectors are expressed in Newtons and Newton-meters, respectively. The errors in direction are expressed in degrees. The average error for each measure is indicated in the histogram. The histograms shown are typical for the other sensors.

Table 1 shows the average errors and standard deviation for all measures and all sensors. Notice that the sensors are similar to each other, and the relative errors associated with the moments are higher than the errors associated with the force measurement, probably due to smaller signal to noise margins during measurements.

2.2.4 Gravity Compensation

The load cell responds to extraneous signals dependent on the sensor's mass as well as its position and velocity. A simple form of gravity compensator has been implemented which estimates the gravity load on the load cell as a function of the cantilevered mass of the cell, m, and the offset from the sensor frame, d, as in Figure 3. It was determined that inertial loads are insignificant in the experiments reported. The parameters, m and d were determined empirically by observing the load cell response

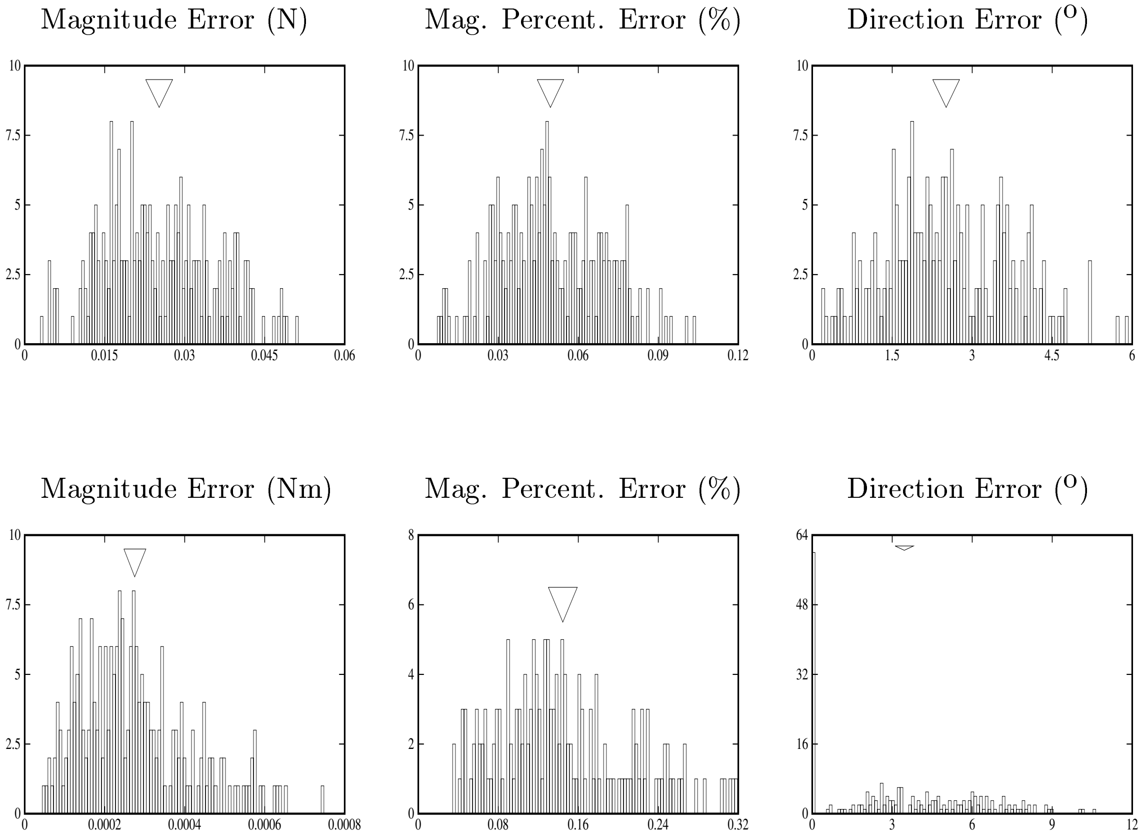

Figure 6: Error histograms in the experimentally determined calibration matrix. The top row contains the histograms for the force vector, and the bottom row contains the histograms for the moment vector.

Table 1: Average errors (standard deviation) for all sensors.

Errors for force measurements:

Sensor #	Magnitude (N)	Magnitude (%)	Direction (^{O})
0	0.0243210219 (0.0112047251)	0.0482319561 (0.0224081011)	2.3798826886 (1.3775726519)
1	0.0279158272 (0.0130327215)	0.0550721770 (0.0250028845)	2.8441973251 (1.4632614249)
2	0.0253229386 (0.0103345262)	0.0497712105 (0.0196502588)	2.5223268640 (1.1572062653)

Errors for moment measurements:

Sensor #	Magnitude (Nm)	Magnitude (%)	Direction (^{O})
0	0.0002387719 (0.0001294833)	0.1178446369 (0.0567219717)	3.2755358289 (3.0453176660)
1	0.0003962305 (0.0002456233)	0.2170348984 (0.1285218717)	5.4611102346 (5.5812415717)
2	0.0002764298 (0.0001392074)	0.1453814940 (0.0667669273)	3.4683033114 (2.7965894974)

Table 2: Inertial Parameters for Gravity Compensation

	Sensor 0	Sensor 1	Sensor 2
m (kg)	0.018437476	0.018803700	0.018722316
d (mm)	3.880194928	3.156531101	2.156939930

as the orientation is varied with respect to the inertial frame. Table 2 summarizes the values of m and d chosen to minimize the squared approximation error.

Figure 7 compares the measured force (left) and moment (right) during a finger trajectory within the $\theta_1 = \theta_2$ configuration subspace with the predicted forces and moments due to the gravity load. Gravity compensation using these parameters is satisfactory for the entire range of load cell orientations encountered during grasping experiments.

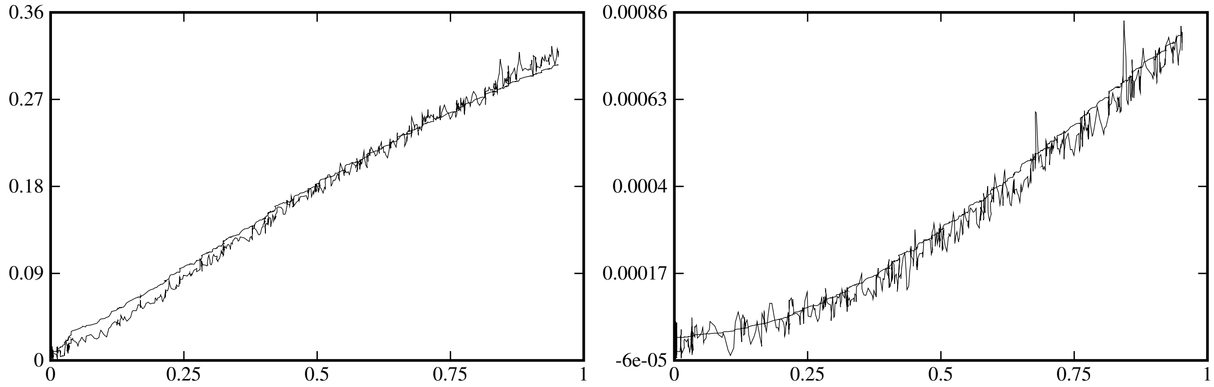

Figure 7: Magnitude of forces (left) and moments (right) observed and predicted during a test in which the finger joints executed a trajectory in the $\theta_1 = \theta_2$ configuration subspace.

2.2.5 Noise Models

This section describes how sensor data are affected by electromagnetic noise, vibration from the arm and tendon actuators, and thermal drift.

Electromagnetic Noise and Drift First, the noise level present in sensor readings at "ideal" operating conditions (at thermal equilibrium and with actuators off) is established. The noise measured in this situation provides an estimate of the background noise associated with every sensor reading. In addition, this experiment provides also a measure of absolute drift due to stray capacitances or other unmodeled effects.

50,000 consecutive force readings were read at 20 ms intervals and the mean and standard deviation were determined for each force component on each sensor. Figure 8 shows the distribution of values obtained for sensor #1 and Table 3 reports the average noise and standard deviation for all sensors. The background noise (corresponding to an average force magnitude error of $0.00133N$) is relatively low when compared to the load cell calibration error of $0.02792N$ (Table 1). No appreciable long term drift was observed.

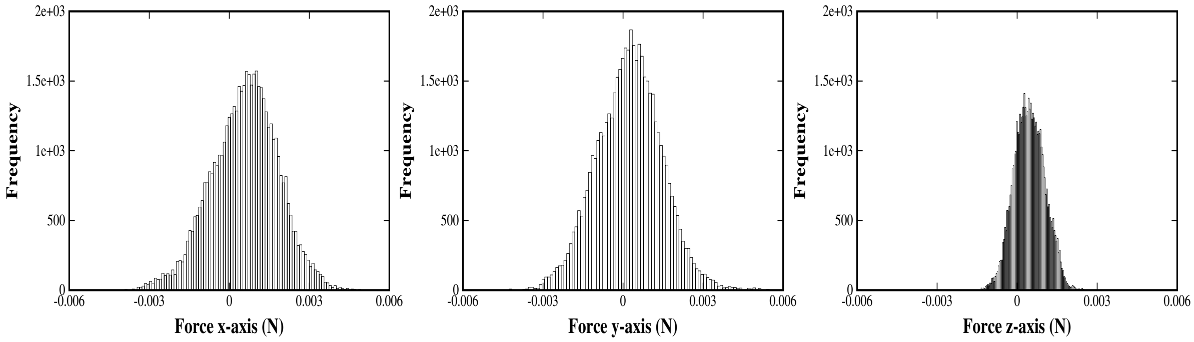

Figure 8: Distribution of values for f_x, f_y, and f_z, for sensor # 1.

Table 3: Average noise (standard deviation) for force components.

Sensor #	Force x axis	Force y axis	Force z axis
0	0.0000212852 (0.0017325197)	-0.0008681320 (0.0024146182)	0.0001473035 (0.0005090121)
1	0.0005786706 (0.0012482088)	0.0002335359 (0.0012229786)	0.0004593314 (0.0005764825)
2	-0.0013617624 (0.0014071571)	0.0000546394 (0.0014869006)	0.0001468884 (0.0003430472)

Vibration-induced noise The previous experiment was repeated with the hand motors activated to examine the effect of mechanical vibration. Results are reported in Table 4. The results show

Table 4: Average noise and standard deviation for force components, with tendon tension actuators turned on.

Sensor #	Force x axis	Force y axis	Force z axis
0	-0.0001530925 (0.0036220934)	-0.0025181098 (0.0039501413)	0.0006755186 (0.0023882254)
1	0.0005410166 (0.0034636874)	0.0002776000 (0.0036047079)	-0.0004706665 (0.0015296217)
2	-0.0018670904 (0.0036222841)	-0.0013847969 (0.0040151006)	0.0001137814 (0.0014834109)

that under these conditions, the background noise level is not increased substantially by energizing and servoing the actuators. Tables 3 and 4 shows that vibration affects the standard deviation of the measurements.

Temperature drift As the sensor circuit is powered on, the strain gages start to dissipate energy, causing temperature changes on the strain gages themselves and on their supporting surface. These temperature changes cause the readings to drift until the system reaches thermal equilibrium.

An experiment was performed to determine how readings vary during the sensor power-on transient. First, the sensor was allowed to cool down overnight; as it was turned on, we recorded the force magnitude reading every three seconds, for two hours. The resulting plot (Figure 9) shows how force magnitude changes as a function of time (in minutes). After approximately 90 minutes, the system reaches thermal equilibrium and the sensor readings stabilize.

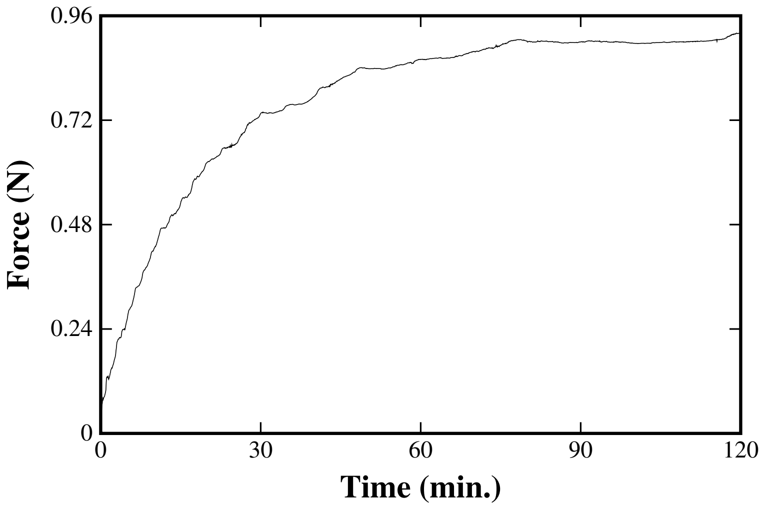

Figure 9: Transient sensor output due to thermal effects.

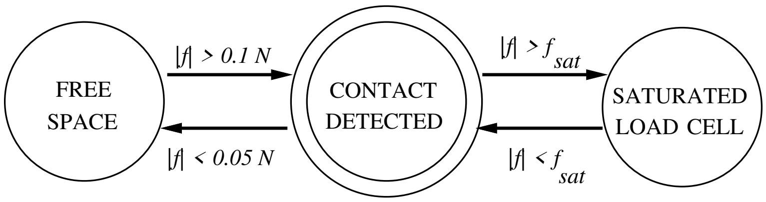

Figure 10: Sensory events during contact formation.

2.2.6 Contact Formation

All actions produced in the grasp controller depend ultimately on tactile information extracted as the fingers interact with an object. This interaction takes the form of a sequence of grasp and release operations. Contact forces must be managed over the sequence in order to minimize unwanted displacements of the object and to improve the precision of contact position and normal estimates. To avoid disturbing the object and to enhance contact precision and tactile feature detection, fingers always approach a reference contact along the local surface normal estimate. This simple approach is consistent with more complete path control treatments and allows the focus of this work to be on the use of tactile information to modify the grasp.

The quality of the sensor signal during contact formation and during manipulation is affected by several factors, including control parameters, actuator performance, and tendon dynamics. Moreover, the fingertip sensors may go through several sensory events before reaching a configuration in the grasp state may be measured. The sequence of sensory events is described by the finite state machine depicted in Figure 10. To guarantee good tracking characteristics while employing the existing PID tendon tension controller, each path is resolved into relatively small incremental displacements and every incremental finger displacement is monitored. The next action is applied if the displacement is completed or if a tactile event signaling contact occurs as depicted in Figure 10.

Detecting a contact load greater than $0.1N$ causes the transition from “freespace” to the “contact detected" state. Transitions to the "saturated load cell" state are triggered whenever at least one of the eight strain gage channels of the sensor becomes saturated, what can be caused by excess contact forces. From the "contact detected" state, the system may also transition back to "freespace" if the magnitude of the contact force decreases below $0.05N$. A valid grasp state can be observed only when all contacts are in the "contact detected" state simultaneously. In all cases, recursive filters are used over several measurements on the load cell to minimize noise when estimating force loads.

The computational overhead of this monitoring and path control scheme during contact formation slows the grasp formation process significantly. More complete approaches to path control must be incorporated into the platform. Moreover, skillful contact formation control can take tremendous advantage of passive compliance properties in the hand. Several designers are examining this issue in the form of new actuators in series with elastic members. Such technology promises to increase the dynamic range of sensing technology and to improve contact formation performance.

3 Haptic Models for Control

The control basis approach to manipulator control provides a means of representing a continuous state space in the form of a discrete set of control equilibria. We have shown that these equilibria constitute the best $2-$ and $3-$contact configuration for disembodied contacts on regular, convex polyhedra. However, for other object geometries and embodied manipulators, the native grasp controller is not guaranteed to work. This leads to situations in which the control strategy works in some situations and does not work in others. Moreover, the control basis methodology alone does not provide enough information to decide how to predict the outcome of all system behaviors.

Fortunately, results in nonlinear dynamics suggest a mechanism for overcoming some forms of this nondeterminism. If the state of the system is not completely observable, Taken's theorem provides a powerful technique for reconstructing the phase space by using a time series or stream of state information. From this perspective, structure in the run-time data can be used to augment the state and can be used to overcome some forms of non-Markovian behavior in the plant. This is the insight underlying haptic models. When faced with an unknown plant that may be irregular, concave, and/or smoothly contoured, the evolution of the grasp state can be used to predict the eventual quality of the grasp. Figure 11 illustrates several synthetic models of the wrench residual controller for a variety of shapes. For the convex polygonal objects, these models show accessible states in the lower half plane ($\dot{\epsilon}_w < 0$) exclusively as the theory predicts. For smoothly contoured and concave objects, the upper half plane ($\dot{\epsilon}_w > 0$) shows trajectories where the control model is inconsistent with actual object geometry. These trajectories cause a local increase in the contact wrench residual and come to equilibrium eventually at some attractor.

Haptic models capture the interaction between the controllers, the robot and the object geometry. Given enough experience with a particular object, the dynamics of the interaction expressed in the model may be used to distinguish functionally distinct types of robot/object behavior. We have collected models of grasp performance in which prior experience with a range of objects is compiled. The control error and rate of change in the error are mapped and form the functional signature of each plant. They provide a prediction about the expected quality of the current grasp trial. In the work reported here, the quality measure is the projected minimum coefficient of friction required for grasping.

In the following section, we provide examples of how models can be used to improve the expected performance of the grasp controller and to manage the engagement of resources (fingers) for grasping tasks. We will present examples of experimental models derived from our grasping platform, and discuss issues regarding efficient representations for haptic models.

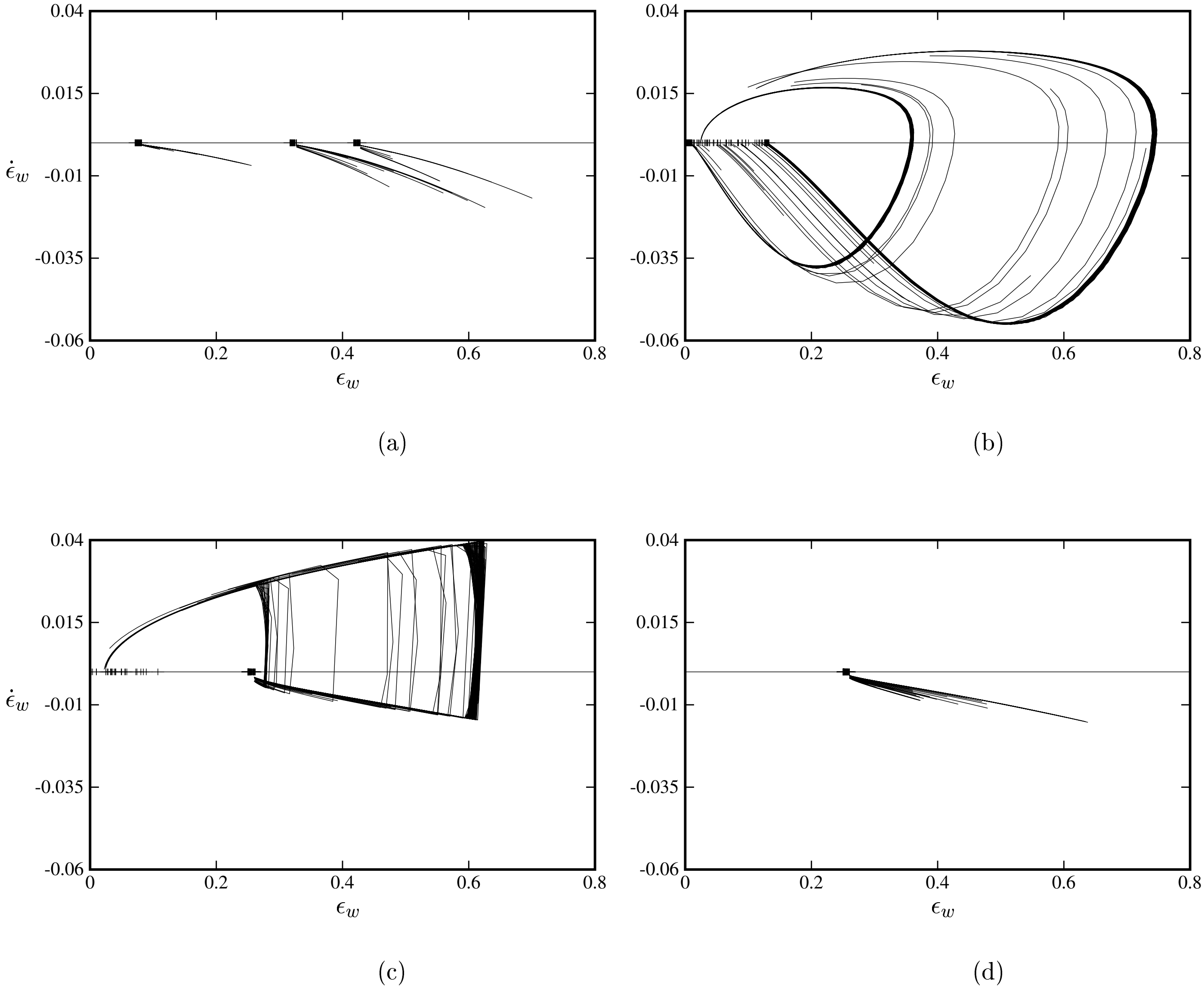

Figure 11: Theoretical phase plane diagrams for (a) an irregular triangle, (b) a smooth concave object, (c) a regular triangle with rounded corners, and (d) a regular triangle.

3.1 Experimental Haptic Models

3.1.1 Data Collection Protocol

The construction of the haptic models requires physical interaction over several trials. The data collection protocol implemented manages the construction of complete models with a minimum number of grasp trials. If three contacts are effecting the grasp, and these contacts can move freely over the object surface, then the grasp state space is a subspace of $\Re^6$ bounded by constraints derived from the hand and arm, and the control dynamics.

Effective strategies for sampling grasp states are required for modeling the dynamics of grasp formation. Moreover, it is difficult to know when a given model is complete, i.e., that every accessible state has been visited. On the other hand, we have demonstrated that a sufficiently complete model is typically available for critical regions (regions in the neighborhood of singular points, such as attractors and saddle points) early in the construction process. This is a consequence of the behavior of the controller — most of the data will be acquired in the neighborhood of critical points, where the control gradient is small. Figure 12 illustrates this phenomenon. Arguably, these points are exactly in regions critical to the final outcome of the grasp operation. Nevertheless, the (possible) incompleteness of the model is an important specification of the final grasp controller.

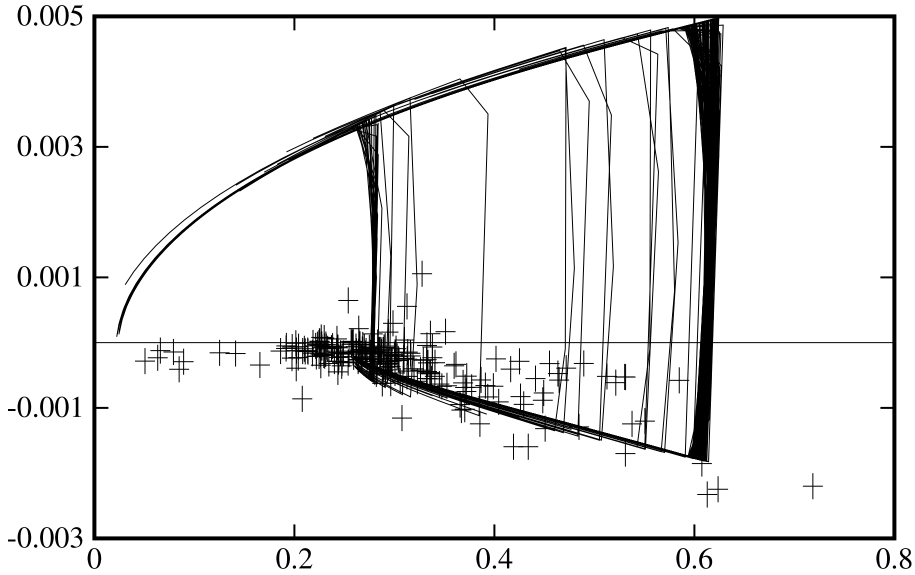

Figure 12: Sampling the phase plane; experimental data (crosses) overlayed on the theoretical model of a triangle with rounded corners.

3.1.2 Interpolation of Haptic Data

The haptic models are built from a collection of discrete data points, acquired one by one in each grasp probe. A relatively small number of states are visited in each grasp trial (averaging less than 40 contact states per grasp) suggests that many trials may be required before a sufficiently complete model becomes available. We explored the application of various forms of interpolation to reduce the total number of grasp trials necessary.

We tried both first and third order polynomial interpolants, adopting the first order interpolation by virtue of its simplicity and efficiency. The contact position is assumed to vary linearly between two adjacent contacts, while the contact normal is kept constant (equal to the average between the consecutive measurements). The associated control error trajectory is mapped into the model. We have found this to be an effective way of improving early performance.

3.1.3 Example Models

We have constructed the models corresponding to two and three fingered grasps over three objects: a cylinder with a diameter of 70 mm and 137 mm in height, plus two prismatic objects, with triangular and square base, respectively. The bases for the prismatic objects are (approximately) regular polygons; the triangular prism is 100 mm in height, and has a 60 mm base. The square prism is 75.5 mm in height and has a 62 mm base.

Haptic models convey both the predicted grasp quality, μ_0, and the local grasp control gradient. We model a particular hand-object interaction by associating a value of μ_0 and the normalized control gradient with each cell in a discrete phase plane. The model is plotted on a logarithmic scale in phase variables $(\epsilon_w, \dot{\epsilon}_w)$, the squared wrench residual and its rate of change, respectively. These state variables are derived directly from a grasp controller which optimizes the squared wrench residual locally — so the model is just a trace of the performance of this controller in the context of a particular object.

Figure 13 depicts the model of the triangular prism for two and three contacts. The horizontal axis is (ϵ_w) and the vertical axis is the rate of change $(\dot{\epsilon}_w)$. The dark circles in Figure 13 indicate the states achieved after convergence of the grasp controller. The distribution of these equilibria is due to the nonzero convergence criteria of the controller and is exaggerated somewhat by the logarithmic scale. Clusters of these convergence states identify regions in which attractors are located. In Figure 13 (b), two such clusters are identified.

Figures 14 and 15 show the corresponding models for the square and circular prisms, respectively. All these models were derived from many grasping trials on the hand/arm platform.

The models (Figures 13, 14, and 15) reveal the characteristic response of the grasp controller with the Salisbury hand, and the corresponding object. With the exception of the circular prism (Figure 15), all other model have two clusters of equilibrium states and usually one attractor dominates the behavior of the system (see for example Figure 13(a)). Less frequently visited attractors can be often be associated with edge- or vertex-contacts which are statistically less likely in this approach, but are nonetheless, reasonable candidate grasps given our quality criterion.

The total number of grasp trials, and the number of grasp probes for each case is shown in Table 5. The average number of probes per trial is also shown. Data interpolation increases the contribution of each probe to the model. Table 5 reports the number of states visited and labeled in the models as of March 11, 1998. For the objects considered, interpolation increased the number of states visited significantly, from 2.5 states per probe for the triangular prism with three contacts, to 8.4 states per probe for the square prism with two contacts.

The number of states visited should plateau over time as experience with an object grows more complete. No experimental model has yet reached a plateau.

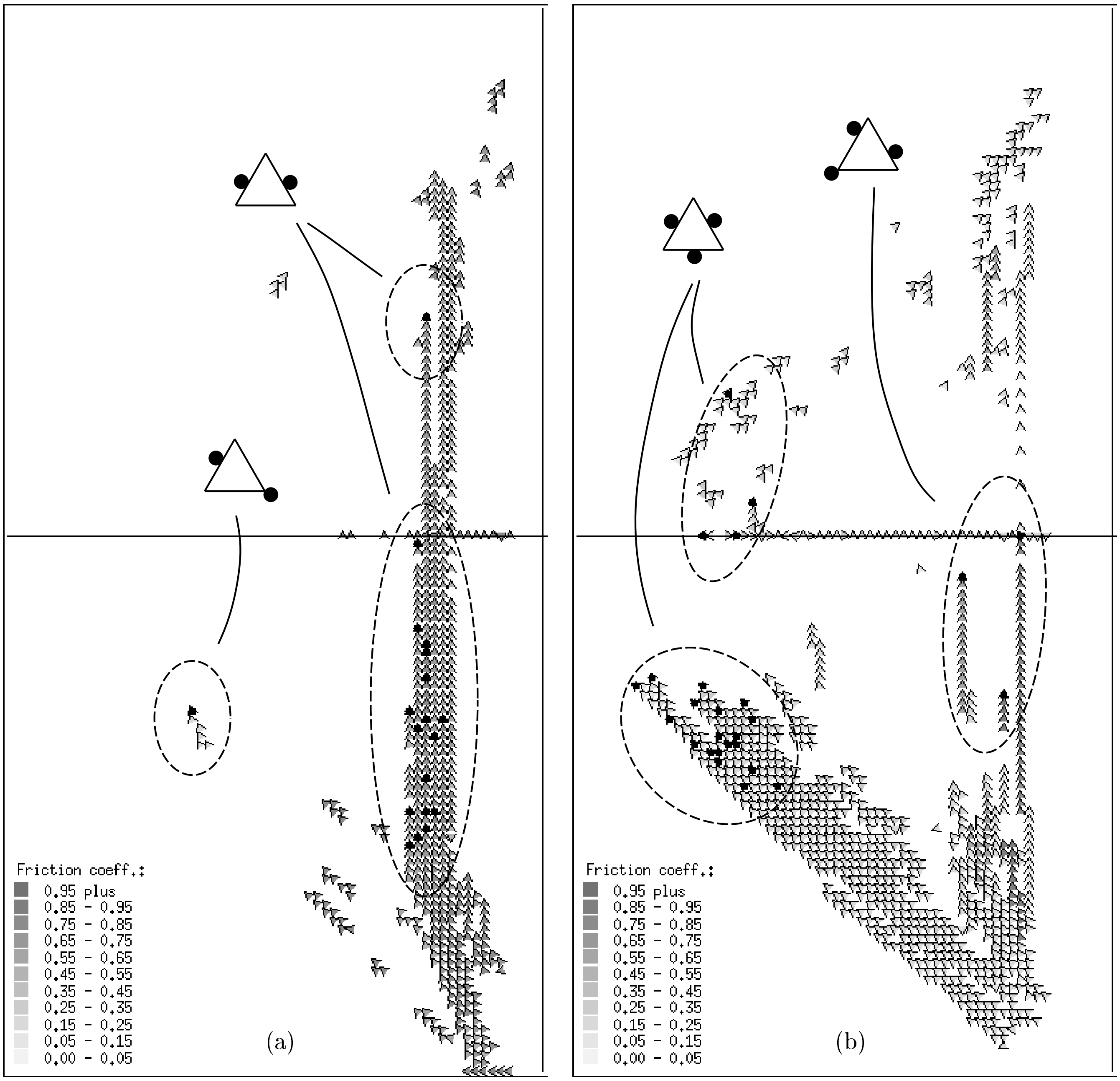

Figure 13: Haptic model for (a) 2 fingered grasps and (b) 3 fingered grasps on the triangular prism.

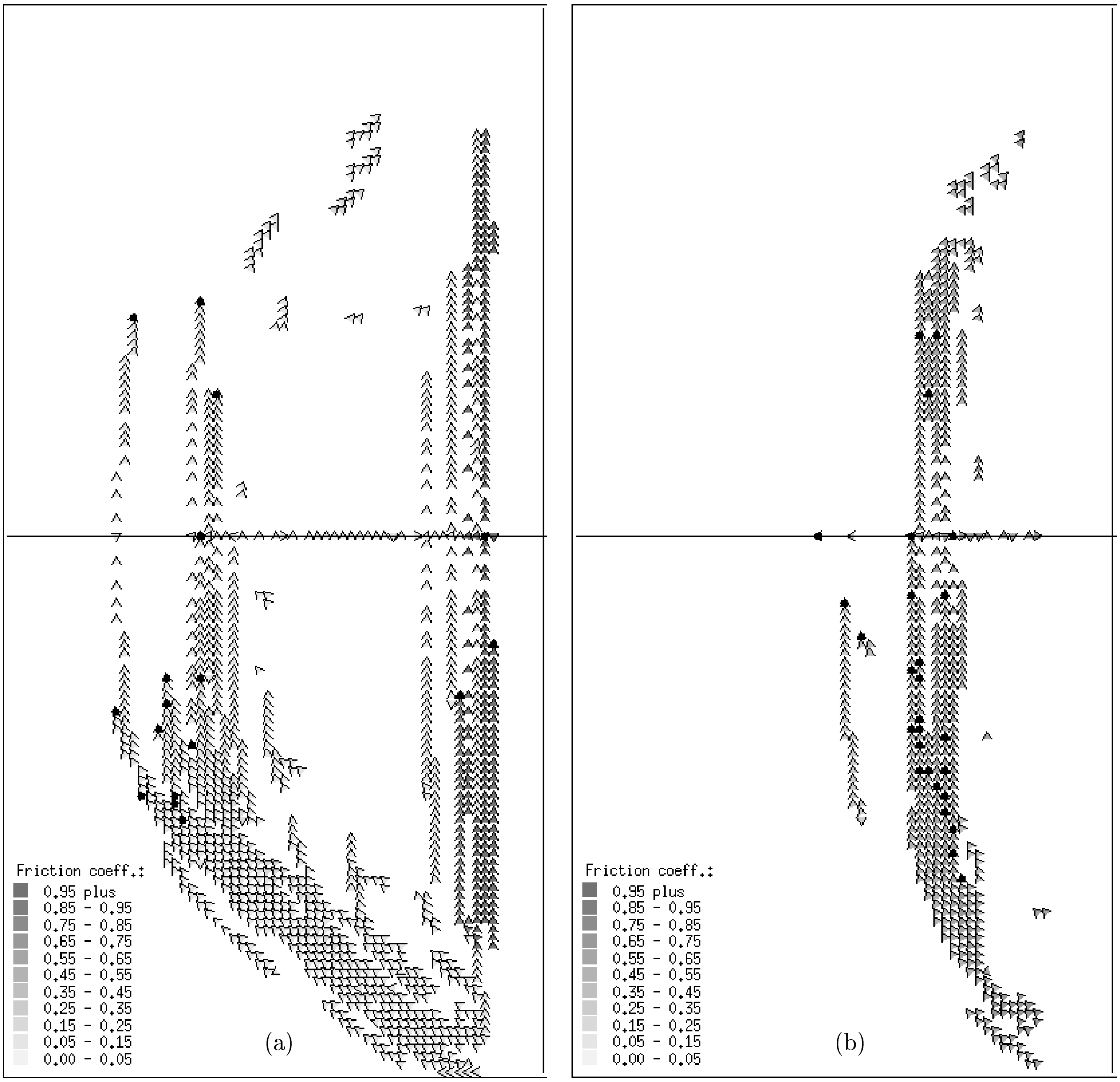

Figure 14: Haptic model for (a) 2 fingered grasps and (b) 3 fingered grasps on the square prism.

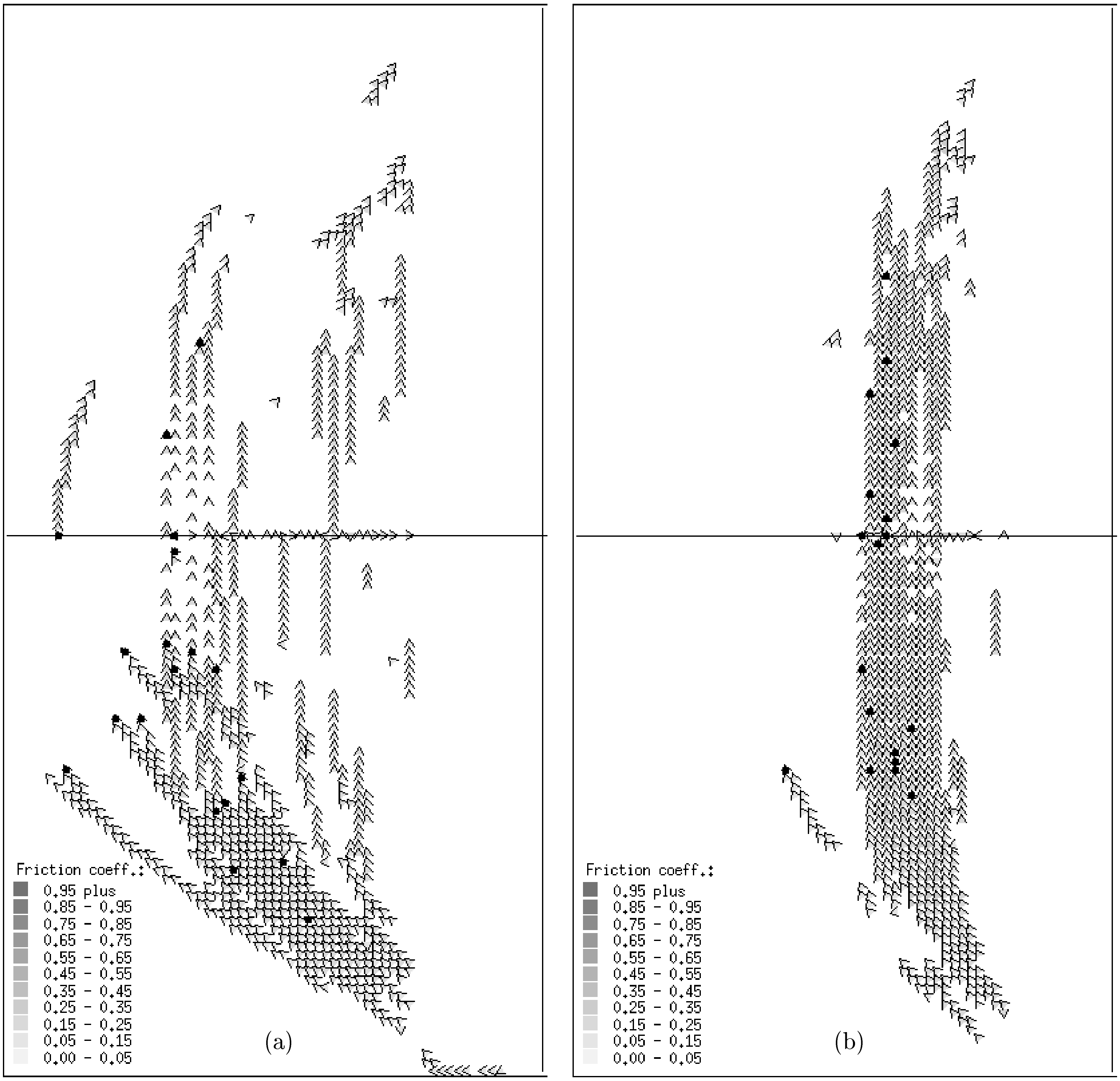

Figure 15: Haptic model for (a) 2 fingered grasps and (b) 3 fingered grasps on the circular prism.

Table 5: Summary of empirical study as of 7/1/97.

Object	# contacts	# trials	# probes	probes/trial	# visited states
Triangular	2	20	207	10.4	587
Prism	3	27	352	13.0	884
Square	2	20	148	7.4	1241
Prism	3	30	167	5.6	539
Circular	2	12	82	6.8	681
Prism	3	20	213	10.7	908

3.2 Representational Issues

Haptic models express the error dynamics of the grasp formation process for a given combination of hand, arm, and object. Attractors in this model identify discrete equilibria in the grasp configuration and so reveal properties of the object; they also provide a basis for predicting the expected grasp quality — in this case, the minimum value of the coefficient of friction required to produce a null space in the grip Jacobian.

The first form of query supported by the haptic models is a comparison of the observed dynamics with the characteristic dynamics of a given plant. Answers to these kind of queries can be used to distinguish candidate plants. The second kind of query is a prediction of the expected quality of the equilibrium contact configuration given a set of candidate plants and the current contact state. Answers to this kind of query contribute to policies for compensating grasping strategies on unknown objects.

Experiments in which models are constructed using state-based, path-based, and region-based representations have been conducted. We have verified the proposition that objects can be recognized under some circumstances by matching observed contact dynamics to a family of haptic models. More precisely, we have shown that variations in object geometry are reflected in the dynamics of grasp formation and have verified this proposition experimentally. State occupancy and/or path-based features of the haptic model can be used to match run-time haptic information to candidate models. However, purely state-based (bitmap) representations with no temporal information are less powerful. A path-based representation has been implemented that constructs a linked-list of experimental observations $(g, \mu_0)_i$. Path-based representations are economical in terms of memory, but are relatively inefficient in answering queries about set membership (i.e. whether g is an element of the set of accessible states for a given model).

Other region-based function approximation schemes (radial basis functions, for example) have been used successfully to model $\mu_0 = f(g)$, where g is a grasp state. As expected, these techniques can be used successfully to approximate this function given a collection of n samples $(g, \mu_0)_i$. However, these techniques can over generalize with insufficient training g. This generalization interferes with the ability to sharply distinguish accessible and inaccessible states.

The path-based representation in conjunction with an interpolation scheme (Section 3.1.2) was chosen for the experimental study since this approach facilitates model construction and alleviates some of the difficulty associated with membership queries, i.e. is this data stream consistent with a particular model.

4 Compensated Grasp Policy

We are proposing that haptic models can be used to predict the future quality of a controller for a given plant. The "plant," in this context, is the dynamical system composed of the robot (with its current control policy) and the object. So our desire to design control policies for unique types of plants ultimately depends on the similarity (or lack of similarity) between the models for the candidate objects.

Tables 6 and 7 report on the similarity of each pair of objects for two and three contact grasps, respectively. Similarity is measured as the number of model cells which are occupied in both models and for which the inner product of their normalized control gradients exceeds 0.99. The absolute number of cell-wise matches is reported along with the percentage of matching cells (in parenthesis).

Table 6: Similarity between models, for two-fingered grasps.

2 contacts	Triangular prism	Square prism	Circular prism
Triangular prism	587 (100%)	111 (19%)	6 (1%)
Square prism	111 (9%)	1241 (100%)	171 (13 %)
Circular prism	6 (0.9%)	171 (25%)	681 (100%)

Table 7: Similarity between models, for three-fingered grasps.

3 contacts	Triangular prism	Square prism	Circular prism
Triangular prism	884 (100%)	24 (3%)	6 (0.7%)
Square prism	24 (4%)	539 (100%)	246 (47 %)
Circular prism	6 (0.7%)	246 (27%)	908 (100%)

Tables 6 and 7 indicate that the predicted future grasp quality does not depend exclusively on the object identity. For example, in Table 7, there are 246 states in common between the square and circular prisms (and this amounts to 47% of the accessible states in the model for the square prism). This implies that there are many cases in which the square and circular prisms are functionally indistinguishable during grasp formation.

This ambiguity regarding plant identity leads to nondeterminism as shown in the state transition diagram depicted in Figure 16. In this situation, a given control activation, Φ, leads nondeterministically to a set of acceptable solutions and to a set of unacceptable solutions. The grasp state p is not sufficient to determine which of the two outcomes is likely. However, if it is possible to match the unfolding grasp state to that predicted in the model as depicted in Figure 16, then a locally enriched grasp state can be constructed by concatenating p with auxiliary state variables ρ that identify the attractor that dominates system behavior. The locally enriched state is illustrated in Figure 17 in which controller Φ leads deterministically from $(p|\rho = 1)$ to q and from $(p|\rho = 0)$ to q'.

Now, the grasp policy can be augmented to include an appropriate action to execute from state $(p|\rho = 0)$ to the set of solutions. We have explored two alternatives for selecting new actions.

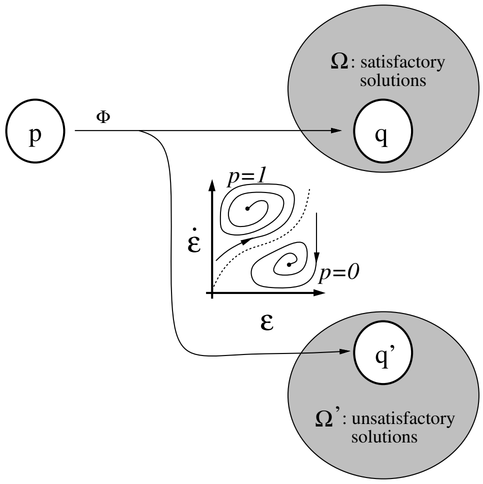

Figure 16: Nondeterminism in the Behavioral State Transition for Two Classes of Plants

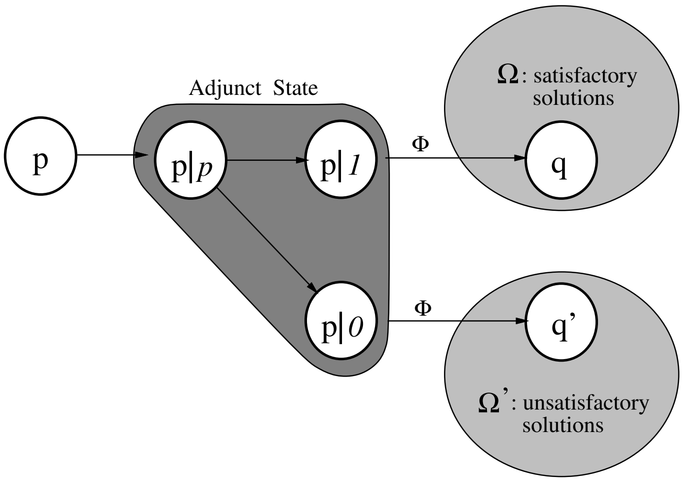

Figure 17: State resolution via auxiliary state.

4.1 Randomization

The first model-based compensation consists of superimposing a random control action ϵ onto controller Φ and to execute a random walk on the model until the systems enters the $(p|\rho = 1)$ state. From here there exists a control policy for achieving q. Random compensatory actions have been implemented

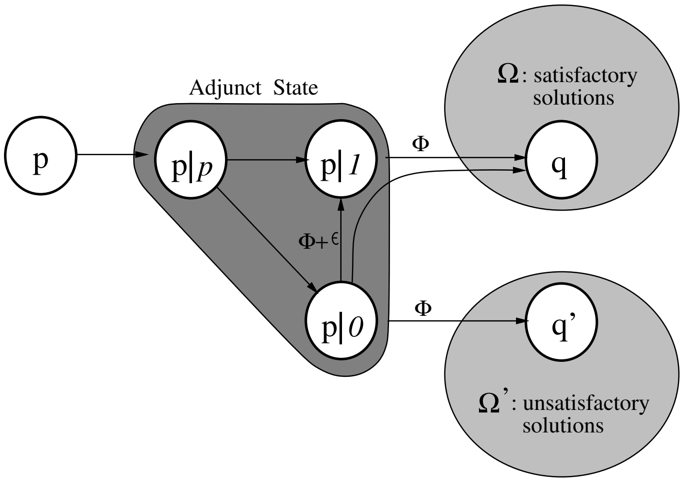

Figure 18: Randomized compensation based on auxiliary state.

and appear to behave as predicted. When the predicted outcome of a grasping trial is negative, a random walk has eventually reached a state from which positive outcomes are expected. From this point, the policy produces grasps that are stable.

4.2 Resource (de)Allocation

Randomization works in practice, but in the simplest implementation, does not attempt to construct a compensation *policy* that can be reused whenever this control context arises. The next time the system passes through these states, it will again employ a random walk if it achieves state $(p|\rho = 0)$. A better alternative is to treat the policy from state $(p|\rho)$ in the same fashion as the policy from p to q, that is, by exploring the space of possible action sequences (resource engagements) that transition from state $(p|\rho)$ to the Ω. Figure 19 illustrates the idea. In this case, the model is used to supply auxiliary state information and the control policy is enhanced permanently by grasp policies for the types of plant corresponding to states $(p|\rho)$.

We have programmed compensators by (de)allocating resources in the grasping policy. Contacts can be allocated (added) whenever all objects in the set of candidate hypotheses requires a larger friction coefficient than allowed in the task specification. Contact placement can be selected so as maximize overall manipulability. Contact deallocation is possible if the worst case friction requirement after contact removal still satisfies the task specification. Frequently, contact removal leads to the elimination of competing candidate object hypotheses. Thus, the decision of which contact to remove can be made so as to maximize the number of eliminated candidate object hypotheses. These rules for contact (de)allocation can only be implemented if a haptic model for each object is available.

Figure 20 illustrates the application of contact (de)allocation rules on a typical grasp for a trapezoidal object, when $\mu_{task} = 0.1$. The set of candidate object hypothesis consists of the nine objects depicted in Figure 20. The initial grasp configuration has two contacts ($\theta = (0, \pi)$). At $t = 0$, the regular and

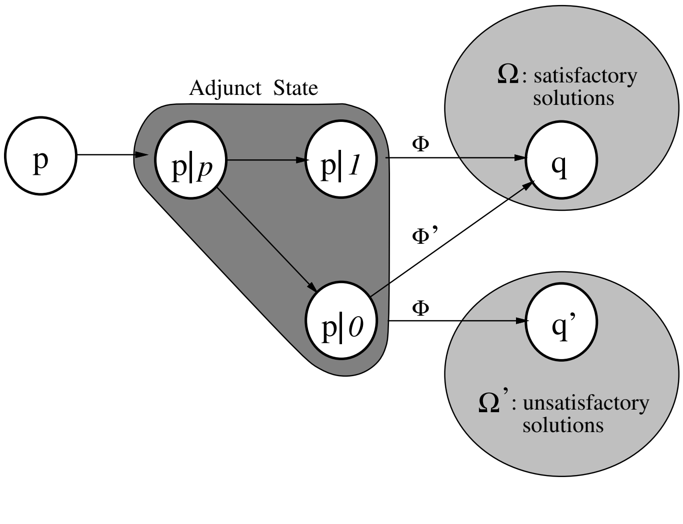

Figure 19: Comprehensive policies based on auxiliary state

the irregular triangle (depicted in grey in Figure 20) are eliminated. All other objects have plausible grasp interpretations for the $(\epsilon, \dot{\epsilon})$ state derived from the trapezoid. At $t = 1$, the controller inserts a third contact in the grasp, at $\theta_2 \approx \pi$. At $t = 21$, the controller removes contact 0 ($\theta_0 \approx 0$), and subsequently rejects the pentagon, the hexagon, the leaf, and the bulb objects. At this point, the set of candidate objects contains only the square and the trapezoid; the controller converges at $t = 87$ to a suitable grasp configuration, requiring a friction coefficient $\mu \leq \mu_{task}$.

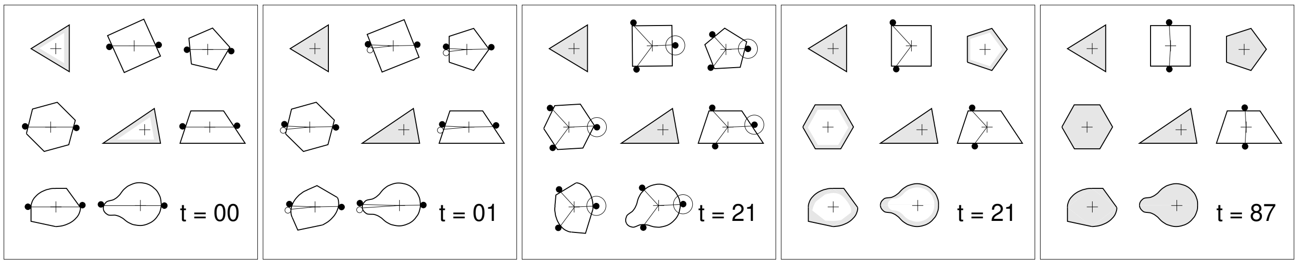

Figure 20: Grasping a trapezoid with two contacts, with $\mu_{task} = 0.1$.

The resulting grasping/manipulation controller has been used in a number of preliminary demonstrations. A notable focus of this experimental work resulted in a grasp controller that typically identifies the object during grasp formation and subsequently tunes the grasp policy for that target object. While detailed experimental data is not yet complete, it seems clear from our experience so far, that this approach to robust control in open environments is practical and more functional than existing technologies.

A major theoretical issue remains: how should a system determine automatically when to construct a model of dynamics? We have our ideas on the matter but have not yet developed a general answer to this question. Fortunately, this issue does not preclude the application of these techniques to a wide variety of interesting and important systems and tasks.

References

[1] Bicchi, A., Salisbury, J., and Brock, D. Contact sensing from force measurements. *International Journal of Robotics Research 12*, 3 (1993), 249–262.

[2] Coelho Jr., J., and Grupen, R. Control pre-imaging for multifingered grasp synthesis. In *Proc. 1994 IEEE Int. Conf. Robotics Automat.* (San Diego, CA, May 1994), vol. 4, pp. 3117–3122.

[3] Connolly, C. I., and Grupen, R. The applications of harmonic functions to robotics. *Journal of Robotic Systems 10*, 7 (Oct. 1993), 931–946.

[4] Grupen, R., Huber, M., Coelho Jr., J. A., and Souccar, K. Distributed control representation for manipulation tasks. *IEEE Expert 10* (Apr. 1995).

[5] Grupen, R., and Souccar, K. Manipulability-based spatial isotropy: A kinematic reflex. In *Proc. of Int. Workshop Mechatronical Comp. Systems for Perception and Action* (Halmstad, SWEDEN, June 1993), pp. 157–163.

[6] Hanson, R., Stutz, J., and Cheeseman, P. Bayesian classification theory. Tech. Rep. Technical Report FIA-90-12-7-01, Artificial Intelligence Research Branch, NASA Ames Research Center, 1990.

[7] Huber, M., and Grupen, R. A. A feedback control structure for on-line learning tasks. *Robotics and Autonomous Systems 22*, 3-4 (December 1997), 303–315.

[8] Huber, M., and Grupen, R. A. Learning to coordinate controllers - reinforcement learning on a control basis. In *Fifteenth International Joint Conference on Artificial Intelligence* (Nagoya, Japan, August 1997), IJCAII, pp. 1366–1371.

[9] Ramadge, P. J., and Wonham, W. M. The control of discrete event systems. *Proceedings of the IEEE 77*, 1 (January 1989), 81–97.

[10] Salisbury, J. K., and Brock, D. L. Force sensing apparatus. U.S. Patent 4,635,479.