Nullspace Composition of Control Laws for Grasping

Robert Platt Jr. Andrew H. Fagg Roderic A. Grupen

Laboratory for Perceptual Robotics
Department of Computer Science
University of Massachusetts, Amherst
{rplatt, fagg, grupen}@cs.umass.edu

Abstract

Much of the tradition in robot grasping is rooted in geometrical, planning-based approaches in which it is assumed that object and finger geometries are well modeled a priori. *Some recent approaches have chosen instead to deal with objects of unknown geometry. These techniques treat grasping as an active sensory-driven problem. At any given time, finger contacts are incrementally displaced along the object's local surface using a single control law. In this paper, we extend this approach by allowing multiple control laws to be active simultaneously. Three control laws are combined by projecting the actions of subordinate control laws into other control law nullspaces. The resulting composite controller finds grasps that are more robust than the component primitives in isolation. Finally, we show how this approach may be used on hand/arm manipulation systems with arbitrary kinematics.*

1 Introduction

Humans exhibit a remarkable ability to manipulate their environment by using their limbs to apply forces. Whether someone is lifting a box, typing on a keyboard, using a hammer, or simply walking, that person is intelligently applying forces to accomplish high-level goals. A component of robotics research deals with the search for algorithms that enable robots to apply forces intelligently with their effectors. We investigate this problem in the context of robotic grasping.

Significant work exists in which grasping is approached as a geometric planning problem. For example, Faverjon [9] and Nguyen [12] have developed algorithms for placing contacts on objects of known geometry. Their methods are based on a detailed exploration of geometrically sufficient conditions for developing a secure grasp on an object. For this type of approach to be viable, extensive geometrical information about the object is generally required.

An alternative approach is to make as few initial assumptions as possible about the geometry of a specific object and instead rely on tactile data. For example, Teichmann and Mishra [15] use local surface normal information to solve for the gradient of the locally minimal area triangle that encloses the object. Contacts are iteratively positioned and re-positioned near the object surface in an effort to minimize this error function.

A common thread in these approaches is that they posit a single sufficient condition for a "good grasp." This belies intuition that the way humans grasp objects often depends on task, object size, and precision or force requirements. Coelho and Grupen [6] capture some of the variety of different possible grasps by framing grasping as a controller composition problem. They posit that robust grasps result from controllers that follow net force and moment gradients. Each control law participates in an iterative improvement process. It is assumed that all contacts are touching the object before the controller begins. Contacts are repeatedly removed from the object, displaced tangentially on the surface in the direction of the negative error gradient, and placed back on the object. Eventually the hand/arm manipulator configuration converges to a point in a stable region.

Primitive behaviors have often been combined to accomplish higher-level goals. This idea has been applied in many areas including mobile robotics [1], and dynamic stability [13]. Although there are few other robust behavior-composition methods for *grasping* objects of unknown geometry, such methods do exist for manipulating objects. Michelman and Allen [10] describe how a collection of rotation and translation primitives may be sequentially combined to accomplish a manipulation task objective such as removing a childproof bottle top. Farooqi and Omata [8] describe two primitives for rotating an object of unknown geometry. One important distinction between these approaches and our work is that we combine primitive controllers concurrently while these approaches combine primitive behavior sequentially.

This paper extends the work of Coelho and Grupen [6] in two ways. First, we introduce an additional control law for kinematic conditioning which prefers contact placement such that individual fingers can apply forces normal to the object's surface (Section 2). Second, we formulate a small set of control laws that can be combined concurrently through the use of nullspace projection to accomplish a variety of grasp objectives using all available manipulator and hand degrees-of-freedom (Section 3). We demonstrate the utility of the composite grasp controller in several simulated and real robot experiments involving a Barrett Hand mounted on a Whole Arm Manipulator (Section 4).

2 Grasp Control Laws

Three primitive manipulation control laws are employed in this work to search for quality grasps on objects of unknown geometry. We utilize the control laws derived by Coelho and Grupen [6, 5] to address force and moment criteria. We outline the formulation of these control laws below and then introduce a control law for kinematic conditioning.

2.1 Force-Based Contact Position Control Law

The force-based contact position control law (ϕ_{force}) is a potential function that has equilibria in configurations where the contacts exert the reference net force. Without loss of generality, we hereafter assume this reference to be zero. Let $\vec{f}$ be the net force vector applied by the contacts (each contact is assumed to apply a unit force that is tangential to the sensed surface normal). The contact configuration error is defined as:

$$\epsilon_f = \vec{f}^T \vec{f}. \tag{1}$$

ϕ_f follows the negative gradient of ϵ_f with respect to the contact configuration by repositioning the contacts on the surface of the object.

Let $\vec{x} \in R^{3k}$ be a vector describing the configuration of k contacts: $\vec{x} = \left(\vec{x_1}^T \; \vec{x_2}^T \ldots \vec{x_k}^T\right)^T$ where $\vec{x_i}$ is the Cartesian location of a contact in R^3. On each probe, a step is taken in the direction of the negative gradient, $\frac{\partial \epsilon_f}{\partial \vec{x}}$. To compute the gradient, we expand $\frac{\partial \epsilon_f}{\partial \vec{x}}$ using the chain rule:

$$\frac{\partial \epsilon_f}{\partial \vec{x}} = \frac{\partial \epsilon_f}{\partial \vec{f}} \frac{\partial \vec{f}}{\partial \vec{x}}. \tag{2}$$

The first term on the right side of Equation 2 is easily derived from Equation 1:

$$\frac{\partial \epsilon_f}{\partial \vec{f}} = 2\vec{f}^T. \tag{3}$$

Calculating the second term requires that we know how $\vec{f}$ changes as the k contacts move independently over the surface of the object. In the absence of information on the geometry of the object, we assume that $\vec{f}$ changes as if each contact were moving on a finite radius sphere tangent to the object surface at the contact point. This is diagramed in Figure 1(a). The precise calculation of $\frac{\partial \vec{f}}{\partial \vec{x}}$ is detailed in [6].

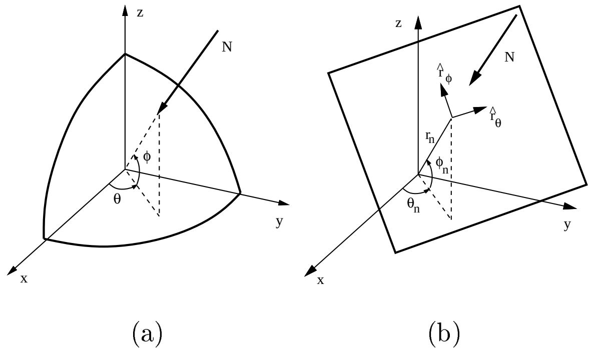


Figure 1: *(a) The force control law assumes each contact moves on the surface of a sphere. (b) The moment control law assumes each contact moves on an infinite plane. In both diagrams, $\vec{N}$ is the force applied normal to the object surface.*

2.2 Moment-Based Contact Position Control Law

The moment-based contact position control law (ϕ_{moment}) has equilibria in configurations where the contacts exert zero net moment. Let $\vec{m}$ be the net moment vector. Contact configuration error is defined as:

$$\epsilon_m = \vec{m}^T \vec{m}. \tag{4}$$

The computation of the gradient $\frac{\partial \epsilon_m}{\partial \vec{x}}$ parallels the computation of the force gradient:

$$\frac{\partial \epsilon_m}{\partial \vec{x}} = \frac{\partial \epsilon_m}{\partial \vec{m}} \frac{\partial \vec{m}}{\partial \vec{x}}. \tag{5}$$

$$\frac{\partial \epsilon_m}{\partial \vec{m}} = 2\vec{m}^T. \tag{6}$$

However, the calculation of $\frac{\partial \vec{m}}{\partial \vec{x}}$ now assumes that each contact moves on an infinite plane, as shown in Figure 1(b). See [6] for details.

2.3 Kinematic Conditioning Configuration Control Law

The kinematic conditioning configuration control law ($\phi_{kinematic}$) is a potential function with equilibria in configurations where the minor axis of the finger's force ellipsoid is parallel to the contact normal. This is advantageous for two reasons. First, in these configurations, the minor axis of the velocity ellipsoid is parallel to the local object surface. This facilitates controlled contact displacements in these directions. Second, since the force is most accurately controlled in the direction perpendicular to the object surface, the manipulator configuration is optimized for precise force control.

Alignment of the principal axes of the hand velocity ellipsoid can be viewed as optimization of manipulator posture to meet task constraints. A general expression for task optimization of manipulator kinematics was introduced by Chiu [4]. We base the error function for the kinematic control law on his derivation of the force transmission ratio.

The manipulator force ellipsoid is defined as follows, where J_i is the Jacobian of contact i with respect to the degrees of freedom in the hand:

$$\vec{f_i}^T (J_i J_i^T) \vec{f_i} = 1.$$

As in Chiu's formulation, let α be the force transmission ratio (i.e. the distance from the center to the surface of the force ellipse) in the direction of $\vec{n_i}$:

$$(\alpha \vec{n_i})^T (J_i J_i^T)(\alpha \vec{n_i}) = 1.$$

Solving for α yields:

$$\alpha = \left[\vec{n_i}^T (J_i J_i^T) \vec{n_i} \right]^{-1/2}.$$

Since we want to maximize the control of force in the direction of the contact normal, we define error as the reciprocal of α:

$$\epsilon_{c_i} = \left[\vec{n_i}^T (J_i J_i^T) \vec{n_i} \right]^{1/2}. \qquad (7)$$

In order to define a control law to descend the gradient of ϵ_{c_i}, we must specify two sets of (not necessarily disjoint) degrees of freedom (DOFs) for each contact. The first set is that used in the calculation of J_i in Equation 7. Let $\vec{\gamma_i}$ be a vector composed of the members of this set. As denoted by the subscript, $\vec{\gamma_i}$ can be different for each contact. The second set is the space in which ϵ_{c_i} is optimized. These are the DOFs which the hand/arm system controls to optimize the error function. Let $\vec{q}$ be a vector composed of members of this set. We assume that all contacts are optimized with respect to the same $\vec{q}$. For k contacts, the error function becomes:

$$\epsilon_c = \sum_{i=1}^{k} \epsilon_{c_i}$$

$$\epsilon_c = \sum_{i=1}^{k} \left[\vec{n_i}^T \left(\frac{\partial \vec{x_i}}{\partial \vec{\gamma_i}} \frac{\partial \vec{x_i}}{\partial \vec{\gamma_i}}^T \right) \vec{n_i} \right]^{1/2}. \qquad (8)$$

In order to descend this function, we take the gradient with respect to $\vec{q}$:

$$\frac{\partial \epsilon_c}{\partial \vec{q}} = \sum_{i=1}^{k} \frac{\left(\frac{\partial \vec{x_i}}{\partial \vec{\gamma_i}}^T \vec{n_i} \right)^T \left(\frac{\partial^2 \vec{x_i}}{\partial \vec{\gamma_i} \partial \vec{q}}^T \vec{n_i} \right)}{\sqrt{\left(\frac{\partial \vec{x_i}}{\partial \vec{\gamma_i}}^T \vec{n_i} \right)^T \left(\frac{\partial \vec{x_i}}{\partial \vec{\gamma_i}}^T \vec{n_i} \right)}}. \qquad (9)$$

In the special case when $\vec{\gamma_i}$ is one dimensional, the gradient reduces to a particularly simple form:

$$\frac{\partial \epsilon_c}{\partial \vec{q}} = \sum_{i=1}^{k} \vec{n_i}^T \frac{\partial}{\partial \gamma_i} \left(\frac{\partial \vec{x}}{\partial \vec{q}} \right). \qquad (10)$$

In this case, the configuration $\vec{q}$ is optimized with respect to a single joint per finger. This makes sense for us because the hand/arm manipulator used in our experiments has only one flexion DOF in each finger. The configuration of the hand/arm manipulator was optimized with respect to the task compatibility of this flexion degree of freedom.

As with the other control laws, we generate configuration displacements in the direction of the negative gradient:

$$\frac{d\vec{q_c}}{dt} \propto -\frac{\partial \epsilon_c}{\partial \vec{q}}.$$

3 Combining Manipulation Control Laws

Our objective for the three primitive control laws described is to produce interesting and useful grasping behavior. Since the three controllers can be executed independently of one another, there are at least three different behaviors which can result. However, such an approach ignores new controllers arising from combinations of control laws.

Our approach to combining control laws is to project some control laws into the nullspace of others:

$$\phi_{force} \triangleright \phi_{moment} \triangleright \phi_{kinematic}$$

In this expression, ϕ_i denotes the i^{th} control law. The $\triangleright$ symbol is used to express the "subject to" relationship. This expression should read: $\phi_{kinematic}$ subject to ϕ_{moment} subject to ϕ_{force}. The "subject to" constraint is shorthand for a projection of one control law into the nullspace of another. The controller written above will reconfigure the manipulator to try to minimize net force as a first priority. If possible, it will also try to minimize net moment. Finally, it will optimize the kinematic configuration with respect to the object without disrupting the first two objectives.

This nullspace approach should be contrasted with a direct combination of control laws. Approaches which simply superimpose controllers on each other cannot characterize the behavior of the composite controller very well. If two control laws have opposite objectives in configuration space, they could cancel each other and no behavior would result. In contrast, the nullspace approach ensures that one control law is maximally effective while others participate subject to the first. In this section, we demonstrate how we accomplish this.

3.1 Combining Force and Moment Control Laws

From Section 2, we have $\frac{\partial \epsilon_f}{\partial \vec{x}}$ and $\frac{\partial \epsilon_m}{\partial \vec{x}}$ for the force and moment gradients, respectively. We want to combine these using the "subject to" constraint. Since $\frac{\partial \vec{f}}{\partial \vec{x}}$ is underconstrained, we can multiply both sides of Equation 2 by $\frac{\partial \vec{f}}{\partial \vec{x}}^{\#}$ yielding

$$\frac{\partial \epsilon_f}{\partial \vec{x}} \frac{\partial \vec{f}}{\partial \vec{x}}^{\#} = \frac{\partial \epsilon_f}{\partial \vec{f}}$$

where $\frac{\partial \vec{f}}{\partial \vec{x}}^{\#}$ denotes the pseudoinverse of $\frac{\partial \vec{f}}{\partial \vec{x}}$. For ϕ_{moment} contact displacements not to disrupt ϕ_{force}, they must not affect net force. Therefore, they must be projected into the nullspace of $\frac{\partial \vec{f}}{\partial \vec{x}}^{\#}$. The following projects the moment control law into the nullspace of the force control law:

$$\frac{\partial \epsilon}{\partial \vec{x}} = \frac{\partial \epsilon_f}{\partial \vec{f}} \frac{\partial \vec{f}}{\partial \vec{x}} + \frac{\partial \epsilon_m}{\partial \vec{m}} \frac{\partial \vec{m}}{\partial \vec{x}} \left(I - \frac{\partial \vec{f}}{\partial \vec{x}} \frac{\partial \vec{f}}{\partial \vec{x}}^{\#} \right). \tag{11}$$

The form of this equation has been shown to be robust to algorithmic singularities [3].

The utility of combining control laws in this way depends on whether the control laws are compatible or not. If the nullspace of $\frac{\partial \vec{f}}{\partial \vec{x}}^{\#}$ is orthogonal to $\frac{\partial \vec{m}}{\partial \vec{x}}$, the second term in Equation 11 will drop out. For the force control law, however, there is a equipotential surface of configurations that yield equivalent force solutions where moment can be optimized.

As before, $\frac{\partial \epsilon}{\partial \vec{x}}$ is used as the basis for gradient descent:

$$\frac{d\vec{x}}{dt} \propto -\frac{\partial \epsilon}{\partial \vec{x}}.$$

3.2 Combining the Kinematic Control Law with Force and Moment

Now that we can describe the direction of the gradient for each contact, we need to express this in terms of the manipulator joint space $\vec{q}$. Normally, this is accomplished using the pseudoinverse of the manipulator Jacobian. Here, we actually need to solve for the $\vec{q}$ which satisfies the various displacements for all k contacts. This can be accomplished using an augmented Jacobian:

$$\frac{\partial \vec{x}}{\partial \vec{q}} = \left(\frac{\partial \vec{x_1}}{\partial \vec{q}}^T \cdots \frac{\partial \vec{x_k}}{\partial \vec{q}}^T \right)^T. \tag{12}$$

In this equation, $\frac{\partial \vec{x_i}}{\partial \vec{q}}$ denotes the manipulator Jacobian for the i^{th} finger. It represents how the location of i^{th} contact point changes with DOFs in both the arm and hand. $\frac{\partial \vec{x}}{\partial \vec{q}}$ describes this relationship for every contact in the system.

Using the pseudoinverse of the augmented Jacobian is a way of satisfying multiple objectives on an equal footing. If there is no solution that satisfies all objectives, it selects the minimum norm solution. This method has been used by many including [14] to satisfy multiple objectives. An augmented Jacobian is likely to possess singularities not present in the original manipulator Jacobian. For this reason, we used the SR-inverse instead of the pseudoinverse in our computations. We denote the SR-Inverse by $(\cdot)^*$.

The contact configuration displacement $\frac{\partial \vec{x}}{\partial t}$ is projected into manipulator configuration space and optimizes with respect to the kinematic control law in the standard way:

$$\frac{\partial \vec{q}}{\partial t} = \frac{\partial \vec{x}}{\partial \vec{q}}^* \frac{\partial \vec{x}}{\partial t} + \left(I - \frac{\partial \vec{x}}{\partial \vec{q}}^* \frac{\partial \vec{x}}{\partial \vec{q}} \right) \frac{\partial \vec{q_c}}{\partial t}. \tag{13}$$

4 Experiments

We conducted experiments to empirically demonstrate that the controllers converged, and to show that the regions of convergence correspond to reasonable grip configurations. To accomplish this, a series of trials were run on different objects. On each trial, the controller was initialized in a random configuration and run until convergence. We show that on average, control law error converges regardless of the starting location and orientation of the manipulator. Figures 4, 5, and 6 characterize the points in configuration space the manipulator converges to.

The UMass Torso was used to test our approach. The UMass Torso is a humanoid platform consisting of two Barrett WAMs (Barrett Technologies, Cambridge MA) mounted on a frame. Each Barrett WAM is equipped with a 3-finger, 4 DOF Barrett

Figure 2: The Barrett Hand has four DOFs. These include one flexion DOF for each of the three fingers and one adduction DOF. The adduction DOF spreads two of the fingers about axes in the palm.

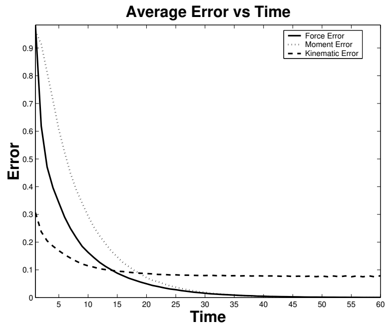


Figure 3: $\epsilon_f, \epsilon_m, \epsilon_k$ as a function of time for the cylinder. Data is averaged across 40 trials. ϵ_f is in Newtons, ϵ_m is in 1/2 Newton-millimeters, and ϵ_k is in radians.

Hand as shown in Figure 2. Mounted on the tip of each Barrett hand finger is a 6-axis force-torque sensor. The force-torque sensor is used to compute fingertip contact location and the direction and magnitude of the contact normal vector [2]. In order to test our controllers in the absence of sensor noise and actuator error, some experiments were conducted in simulation.

Simulated experiments were run for 40 trials. Experiments on the physical system were run for approximately 20 trials. We tested the controller in simulation on a cylinder, a prismatic hexagon, and an irregular six-sided polygon. We ran experiments on the physical system for a cylinder. For simulated trials, the controller was run for approximately 60 tactile probes. On the physical system, the controller was run for approximately 25 tactile probes per trial. It took the physical system an average of two minutes to complete each 25-probe trial. For both simulated and physical experiments, initial degree of adduction randomly varied between 10 and 90 degrees. Initial orientation of the hand varied between 30 and 50 degrees from the x/y plane. Initial x/y location varied between -18 and +18 cm from the object center.

4.1 Convergence of control law error functions

Our first goal is to demonstrate that the control laws function as intended. The composite controller $\Phi_{fmk} = \phi_{force} \triangleleft \phi_{moment} \triangleleft \phi_{kinematic}$ was executed in simulation for 40 trials on the cylinder. Error was calculated at each probe and averaged over the 40 trials for ϵ_f, ϵ_m, and ϵ_c. Figure 3 shows that force and moment error converge to zero while kinematic error hovers near 0.1 radians. Kinematic error never reaches zero because in this composite controller operates subject to both the force and moment control laws.

Figure 4 characterizes the performance of these controllers in terms of degree of adduction and orientation of the Barrett hand. It shows the performance of the Φ_{fmk} controller on a cylinder, a hexagon, and an irregular 6-sided prismatic polygon in simulation. The graphs show that the region where the controller converges depends on object geometry. For the cylinder, all trials converge to a very small region. For the hexagon (Figure 4b), the composite controller converges to a line segment that corresponds to equipotential solutions for contacts moving along a side of the hexagon. The convergence region is even larger for the irregular object due to the existence of multiple possible robust grasps. Figure 5 shows the performance of the controller grasping a cylinder on the physical UMass Torso system. Although the convergence points are significantly more distributed on the physical system, the results are qualitatively similar to those obtained in simulation.

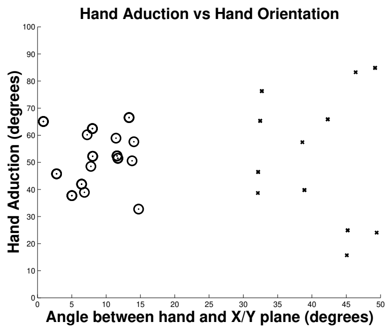
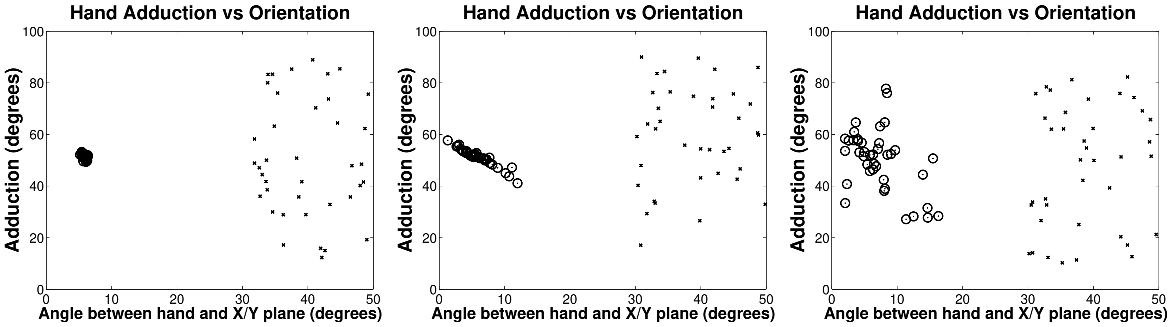


Figure 5: Hand orientation and degree of adduction for grasp trials on a 6.5cm radius cylinder using Φ_{fmk} on the physical system.

4.2 Alternative Control Law Combinations

The force and moment criteria are necessary conditions for establishing a stable grasp. However, the kinematic conditioning constraint does not directly



(a) (b) (c)

Figure 4: Hand orientation and degree of adduction at starting points and convergence points for simulated grasp trials on: (a) a 5cm radius cylinder, (b) a prismatic hexagon with 6cm sides, and (c) a prismatic, irregular six-sided polygon. The dots mark where trials begin and circles indicate where the hand/arm manipulator converged.

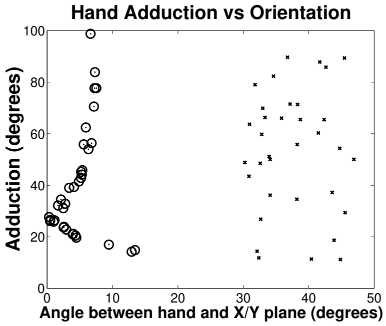


Figure 6: Hand orientation and degree of adduction for grasp trials on a 5cm radius cylinder using Φ_{fm} in simulation.

affect grasp stability. Here, we explore the class of solutions that are discovered when kinematic constraint is not included in the controller formulation.

When $\Phi_{fm} = \phi_{force} \triangleright \phi_{moment}$ is used instead of Φ_{fmk}, the regions of convergence are categorically different. Figure 6 shows that on the cylinder, Φ_{fm} converges to a range of different hand adduction angles (compare to Figure 4). As velocity control tangent to the object surface decreases, the composite controller may reach configurations where ϵ_f and ϵ_m gradients are limited by poor kinematic configuration. In these cases, the composite controller converges with different angles of adduction.

5 Conclusions and Future Work

In this paper, we have shown how to treat grasping as a multi-objective control problem. This is valuable when information concerning the geometry and location of the object is imprecise or not available. We describe three primitive control laws which can be combined to produce useful behavior. We show how to combine these control laws into composite controllers using nullspace projections.

There are numerous areas for future work. Recent work [7] suggests that visual features can be learned on the basis of how well they predict grasping success. The presence of visual features in subsequent grasp targets can provide additional information which can reduce the number of probes required until a suitable grasp is found.

In addition, recent work has incorporated knowledge of manipulator kinematics into a geometric grasp planning framework [11]. We would like to explore how to combine geometric methods with the work presented here.

Finally, although we assume here that the manipulator makes contact with the object at the fingertips, this is not a necessary assumption. We plan to explore the possibility of contacting the object at different points on the manipulator. This could result in whole-hand grasps, two-handed grasps, and two-armed grasps.

Acknowledgments

The authors would like to thank David Wheeler for building a significant part of the UMass Torso infrastructure, and Michael T. Rosenstein and Antonio Morales for their intellectual contributions to this work.

This work was supported by the National Science Foundation under grants CISE/CDA-9703217, and IRI-9704530, DARPA MARS DABT63-99-1-0004, and NASA/RICIS (University of Houston #215).

References

[1] T. Balch and R. Arkin. Behavior-based formation control for multi-robot teams. *IEEE Transactions on Robotics and Automation*, 1999.

[2] A. Bicchi, J. Salisbury, and D. Brock. Contact sensing from force measurements. *Int'l Journal of Robotics Research*, 12(3), 1993.

[3] S. Chiaverini. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Transactions of Robotics and Automation*, 1997.

[4] S. Chiu. Task compatibility of manipulator postures. *Int'l Journal of Robotics Research*, 1988.

[5] J. Coelho. *Multifingered Grasping: Grasp Reflexes and Control Context*. PhD thesis, University of Massachusetts, Amherst, 2001.

[6] J. Coelho and R. Grupen. A control basis for learning multifingered grasps. *Journal of Robotic Systems*, 1997.

[7] J. Coelho, J. Piater, and R. Grupen. Developing haptic and visual perceptual categories for reaching and grasping with a humanoid robot. *Robotics and Autonomous Systems Journal, special issue on Humanoid Robots*, 2001.

[8] M. Farooqi, T. Tanaka, Y. Ikezawa, and T. Omata. Sensor based control for the execution of regrasping primitives on a multifingered robot hand. In *1999 IEEE Int'l Conference on Robotics and Automation*, volume 4, pages 3217–3223, 1999.

[9] B. Faverjon and J. Ponce. On computing two-finger force-closure grasps of curved 2d objects. In *IEEE Int'l Conf. Robotics Automation*, 1991.

[10] P. Michelman and P. Allen. Forming complex dextrous manipulations from task primitives. In *IEEE Int'l Conference on Robotics and Automation*, 1994.

[11] A. Morales, P. Sanz, A. Pobil, and A. Fagg. An experiment in constrainig vision-based finger contact selection with gripper geometry. In *IEEE Int'l Conference on Intelligent Robots and Systems*, 2002.

[12] V. Nguyen. Constructing stable grasps. In *Int'l Journal of Robotics Research*, 1989.

[13] M. Raibert, H. Brown, and M. Chepponis. Experiments in balance with a 3d one-legged hopping machine. *Int'l Journal of Robotics Research*, 1984.

[14] L. Sciavicco and B. Siciliano. A solution algorithm to the inverse kinematic problem for redundant manipulators. *IEEE Journal of Robotics and Automation*, 4(4), 1988.

[15] M. Teichmann and B. Mishra. Reactive algorithms for 2 and 3 finger grasping. In *Proceedings of the 1994 Int'l Workshop on Intelligent Robots and Systems*.