IEEE Expert, 10(2):9-14, Special Track on Intelligent Robotic Systems

A Basis for Distributed Control of Manipulation Tasks *

Roderic A. Grupen, Manfred Huber,
Jefferson A. Coelho Jr., Kamal Souccar

Laboratory for Perceptual Robotics, Department of Computer Science

University of Massachusetts, Amherst

Abstract

The control of complex sensorimotor systems in unstructured environments presents formidable control challenges. A distributed control approach is presented which constructs behavior on-line by activating combinations of reusable feedback control laws with formal stability and convergence properties drawn from a control basis. This control basis representation solves critical problems related to the size of the search space used to represent sensory and motor policies. Moreover, the predictability of the elemental controllers further reduces the complexity of the composition problem and simplifies the planning of composition policies. A generic control basis is constructed and applied to a 20 degree-of-freedom hand/arm system engaged in autonomous manipulation tasks.

1 Introduction

A reasonable goal in the design of robot systems is to enhance the autonomy of manipulation strategies. While introducing flexibility, dextrous robot hand/arm systems present formidable control challenges that are not amenable to traditional control methods. This is primarily because these systems are highly redundant, are applied to tasks composed of multiple goals, operate in uncertain and unstructured environments, and must capture a broad range of control contexts. Manipulation tasks also require purposeful interactions with the environment and thus crucially depend on the efficient use of the available feedback information.

The approach presented in this paper addresses these issues using a "bottom-up" approach. Behavior is composed on-line from a basis set of reusable feedback control laws with formal stability and convergence results. Control actions are derived from a sequence of concurrent activations of a subset of this control basis, bound to specific system degrees of freedom. The resulting controller inherits predictability and stability directly from the stability and robustness of the elemental controllers.

In the following sections, the control basis approach is motivated and described in detail. A control basis and a composition mechanism for manipulation tasks are constructed and applied to examples of reaching, grasping, and manipulation tasks. Examples are presented to emphasize the scalability of the approach and the reusability of the underlying controllers.

2 Control Paradigms for Complex Systems

Model-based control techniques have been applied to complex manipulation tasks [11, 5]. In these approaches, object dynamics and internal contact forces are incorporated into a single control law for the manipulation system. These systems stabilize the object by virtue of the underlying control stability and work only as well as the process model predicts system performance. For most interesting problems, the process models are highly non-linear, non-stationary, and cannot be completely specified *a priori*. Model-based planning techniques [12] are vulnerable to the same problems. Moreover, planning is fundamentally exponential in the number of degrees of freedom in the system and is sensitive to the complexity of the environment. Consequently, in addition to predictive deficiencies, model-based approaches do not scale well to the complex requirements inherent in dextrous manipulation tasks.

Hierarchical architectures [1, 14] were introduced to overcome the size of the planning problem. In this top-down process control paradigm, communication and control pathways are predetermined. Control is

*This work was supported in part by NSF CDA-8922572, IRI-9116297, IRI-9208920

passed from the higher level planner through intermediate planning levels until it reaches the lowest level servo controllers. Control is initialized at the top of the hierarchy and depends on abstractions of the process that must be constructed prior to the initiation of action. This model for control requires thus a relatively coherent and complete world representation and a corresponding control hierarchy. This requirement becomes unwieldy in broad application domains resulting in rigid, inflexible problem solving behavior.

Behavior-based architectures [2, 15, 6] were developed to address the problems of model dependence and inflexibility and to increase the robustness of the control. In this bottom-up paradigm, behavior is constructed on-line from combinations of elemental, reactive behaviors. This approach mitigates the complexity of monolithic controllers, and since models are expensive and can be misleading as predictive devices in general, traditional models are avoided. Behaviors are commonly procedural sensorimotor strategies. In order to respond to a broad range of operating contexts, the system designer must enumerate all sensor and motor alternatives, derive a behavior mapping stimulus to response, and provide some mechanism for tuning the manner in which behaviors interact. As a result, this approach can lead to extremely complex organizations of *ad hoc* behavioral elements which do not scale well to complex task domains like dextrous manipulation.

3 The Control Basis Approach

The approach presented in this paper attempts to circumvent the problem of behavioral proliferation and reduce the complexity of the composition problem without the loss of system responsiveness. Behavior is constructed by activating combinations of reusable, declarative feedback control laws with formal stability and convergence results. Declarative in this context implies that elements of this control basis are designed to reflect generic control strategies, largely independent of the task geometry and robot kinematics. Careful design of the control basis leads to a small set of control laws that span a broad class of tasks and are robust with respect to perturbations. As shown in Figure 1, control is derived on-line by associating sensors and system degrees of freedom with elements of the control basis. These controllers are activated concurrently according to a task-dependent composition policy. The formal stability and convergence results of the control basis lead to the predictability and stability of the composite controller. This dramatically reduces the size of the search space of possible controller bindings and combinations.

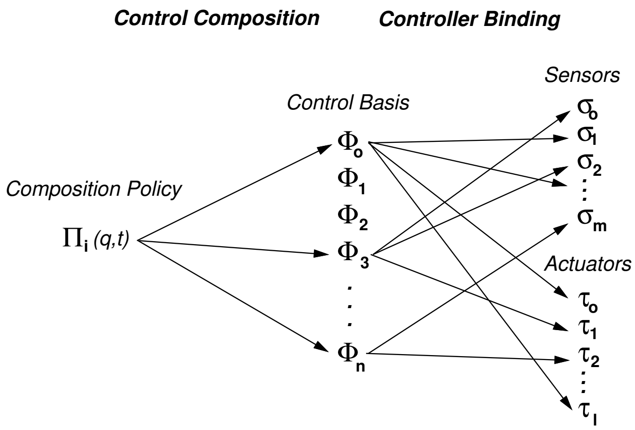

Figure 1: Control Composition

To accomplish complex tasks, the composition policy will take the form of a sequence of concurrent control activations which leads the system through a set of favorable equilibria to the goal. Different tasks are achieved by different composition functions over the same basis controllers, rather than by the design of new task-specific behaviors. In this paper, examples are given of how this composition can be accomplished and illustrate the utility of the control basis representation. In methods employing less formal behavioral specifications, the control policy has to be either given by the designer [2] or found through random exploration [13]. The formal properties of this approach allow for reasoning about end-to-end problem solving behavior and thus opens the composition problem to a large variety of control, planning, and machine learning methods. Moreover, this control architecture preserves system responsiveness and reactivity and thus scales to complex tasks such as dextrous manipulation.

4 A Control Basis for Manipulation Tasks

To demonstrate the validity of the approach described in the previous section, a control system for dextrous manipulation tasks was designed and applied to example manipulation tasks. In this section, the control basis and composition method used in the experiments is presented.

The initial control basis consists of robust solutions to three generic robot control tasks, namely configuration space motion control, contact configuration control, and kinematic conditioning. Harmonic function path controllers [8] are used to generate robust, reactive, and collision-free motion through generalized configuration spaces. A contact configuration controller [3] is employed to move contacts based on the local object geometry. Kinematic conditioning control [9] optimizes the manipulability metric of the articulated structure while it is engaged in an interaction with the world.

While a detailed description of these basis controllers is beyond the scope of this document, the following gives a brief overview. The exact set of controllers is not critical to the architecture. However, certain properties of the controllers must be established to guarantee predictable behavior in the composite controllers. In this work, it was required that candidate control functions possess unique extrema and that compositions over control functions preserve this property. Such a closure property is simple to establish for convex and harmonic control functions activated according to linear composition functions.

4.1 Configuration Space Motion Control

Harmonic functions express a control surface whose gradient leads to the goal while avoiding obstacles. Unlike other potential field approaches, harmonic functions satisfy Laplace's equation

$$\nabla^2 \phi = \sum_{i=1}^{n} \frac{\partial^2 \phi}{\partial q_i^2} = 0$$

relating the second derivatives of the potential field ϕ with respect to the configuration space parameters q_i. This results in important formal properties, the most important of which are that the harmonic function approach is complete and free of local minima [8]. For the control of a physical system this implies that gradient descent on the potential field achieves the goal if a collision-free path exists. Formally, this approach minimizes collision probabilities for the robot system [7], and can be realized in efficient VLSI architectures [17]. Given a model of obstacles in the workspace and the kinematic structure of the robot, it yields robust and reactive motion controllers.

4.2 Contact Configuration Control

The contact configuration controller uses a context-dependent composition of two other controllers. Since our goal here is to interact with objects without building complete geometric models, these controllers represent two interpretations of contact position and normal information with respect to general wrench space tasks[1]. The FC (Force Control - not to be confused with the more common notion of force servoed controllers) controller maps errors between reference and actual contact forces to contact displacements by assuming that the local surface is a unit sphere about some center of curvature. In this *canonical* geometry, moments are ignored and forces vary smoothly and continuously with surface coordinate. The MC controller maps moment errors with respect to reference moments to contact displacements by assuming that the local surface is an infinite plane. In this canonical geometry, force do not vary while moments vary smoothly and continuously with contact coordinate. The idea is that the manner in which contact wrenches vary with surface coordinate for real surface geometries can be approximated by linear compositions of these two controllers. A fixed composition policy expressed as a linear decision on the error plane of the two controllers has been developed that requires only local contact information and has been proven to generate correct contact configurations for regular, convex objects [4].

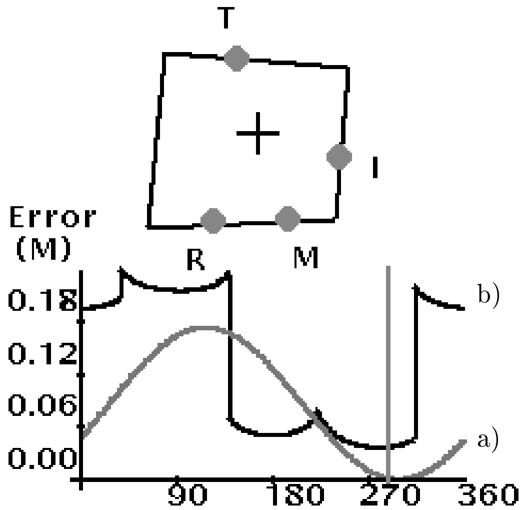

Figure 2: Object with Contact Points and *a*) FC and *b*) MC Control Surfaces for Finger M

Figure 2 shows a final grasp configuration and the two control surfaces involved in the internal composition of this controller for the middle finger (M) of a four fingered hand. Given a set of contact locations and normals, derived from visual models or directly from tactile or kinesthetic feedback [10], the

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

contact configuration controller navigates a single contact across the surface of the object. Multiple contacts run several instances of the controller asynchronously, coupled via shared contact state information.

4.3 Kinematic Conditioning Control

The manipulability metric proposed by Yoshikawa [18] is a scalar metric describing the manipulator's ability to generate forces and velocities. The kinematic conditioning controller ascends the manipulability metric of a subset of the degrees of freedom of the system, thus improving their kinematic condition. This is treated as an independent control task and optimizes conditioning explicitly in portions of an articulated mechanism by executing posture changes in other degrees of freedom.

5 Examples

To demonstrate the feasibility of the proposed approach and the efficient reusability of the elements of the control basis in different contexts, several grasping and manipulation tasks were performed. The platform used for these experiments is a 20 degree-of-freedom hand/arm system consisting of a Utah/MIT dextrous hand mounted on a GE P50 industrial arm. A camera overlooking the workspace is used to provide models for the workspace and the object surface geometry.

Composition functions were hand-crafted as finite state supervisors in a discrete event dynamic systems (DEDS) framework [16]. Due to the predictable performance of the control basis, the supervisor can be modeled on a logical level as sets of concurrent activations of basis controllers with transitions between states corresponding to convergence events. All sensory feedback is interpreted locally within the elemental controllers. The final overall controller navigates through a sequence of system equilibria. Throughout the execution, safety of the complete system can be guaranteed within the limits of the sensory and motor capabilities of the elemental controllers.

Figures 4, 5, and 7 present examples of the type of finite state supervisors employed by this work. The control policy is shown inside each state as a set of concurrent controllers. Each instance of a controller has the form Φ_τ^σ. Here, Φ is an element of the control basis, Φ_0 denotes the harmonic function path controller, Φ_1 the contact configuration controller, and Φ_2 the kinematic conditioning controller. The superscript, σ, denotes the input state set for the controller, including, conceptually, all abstractions and interpretations of the sensor data. The subscript, τ, denotes the output state set including all effectors enlisted by the controller. In these examples, the input and output state sets generally refer to a kinematic configuration and differential kinematic motion, respectively. Therefore, the σ and τ are designated by specifying subsets of the system degrees of freedom. Moreover, experience with this representation suggests that input and output state sets often consist of conjunctions over series kinematic chains. In all the examples illustrated, I/O sets were derived from groups of degrees of freedom: $finger_i, i \in [0,3]$, arm, and $object$. Resolving individual degrees of freedom has never proven to be necessary.

5.1 Pick-and-Place

The first example presented here is the autonomous execution of pick-and-place tasks on a cluttered tabletop. The finite state machine expressing the corresponding composition function is shown in Figure 4.

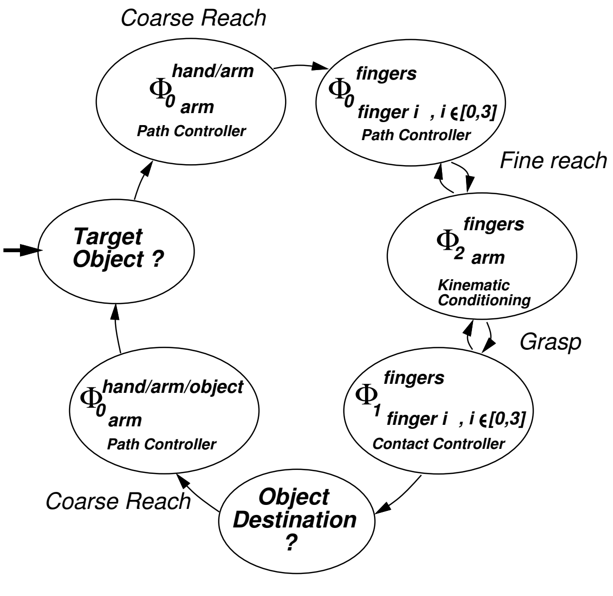

Figure 4: Finite State Machine for Pick-and-Place

This example serves to highlight several features of the proposed distributed control architecture. A user input specifying a target object (in this case, this involved a mouse click on an overhead visual image) causes a control transition to activate a harmonic function that navigates the hand/arm system through a cluttered configuration space to the neighborhood of the specified object. This coarse motion controller drives the robot arm's degrees of freedom exclusively. Following convergence, the "fine reach" portion of the

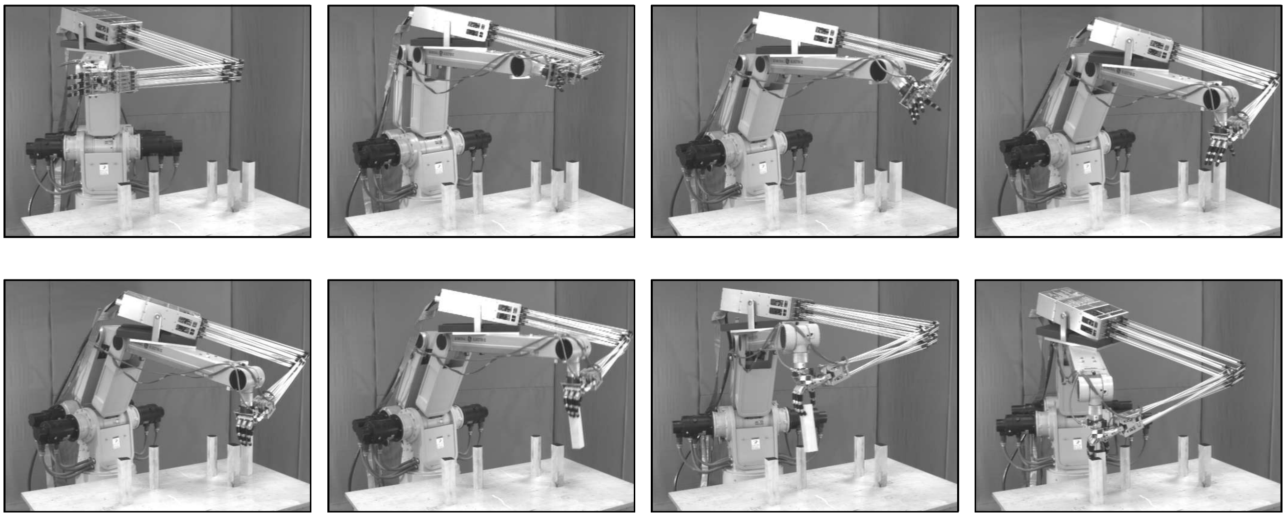

Figure 3: The Behavior of the Pick-and-Place Automaton

automaton is activated. Multiple instances of the harmonic function motion controller are activated concurrently (one for each finger) and a kinematic conditioning controller. The former navigates fingers onto the object's surface and the latter conditions the kinematics of the fingers by adjusting the posture of the arm. Upon convergence of the "fine reach" segment of the automaton, the "grasp" controller is employed to optimize the contact configuration. In this state, one instance of the contact configuration controller is activated for each finger, interleaved with an instance of the kinematic conditioning controller. This coupled control application implicitly propagates constraints imposed by the grasping task, through the hand posture, to the arm posture.

Following convergence of the contact configuration controllers, the user may designate a destination for the acquired object. Doing so activates a harmonic function path controller that navigates the hand/arm/object volume through the configuration space of the arm and releases the object. Figure 3 illustrates a cycle in this control "gait." All sensing and motion control is executed on-line in a cycle time of approximately 20 seconds.

5.2 Object Reorientation

A direct extension of the pick-and-place task is the reorientation of an object by picking it up and placing it down in the same spot with a different orientation. The control composition shown in Figure 5 displays many of the same components as the pick-and-place example.

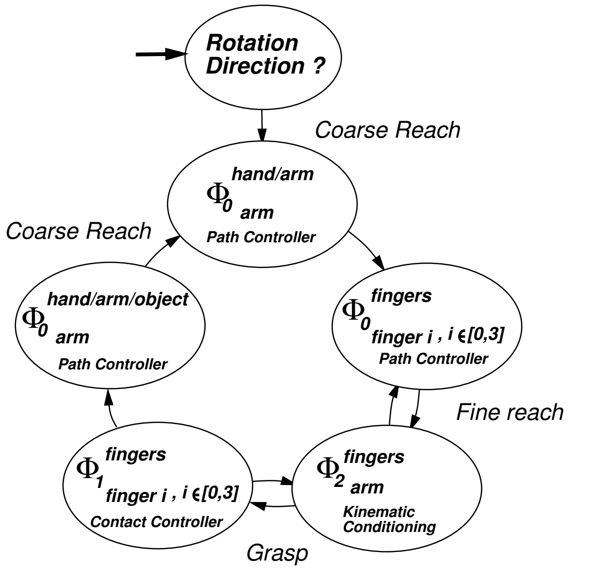

Figure 5: Finite State Machine for Reorientation Task

Here the initial user input specifies the direction of the reorientation by means of fixing the goal orientation for the configuration space motion controller. Changing the goal orientation in the second coarse reach state leads then to the reorientation shown in Figure 6.

This example shows how composition policies can be easily adjusted to similar task objectives without the need for a complete redesign. This suggests that within certain bounds additional task constraints can

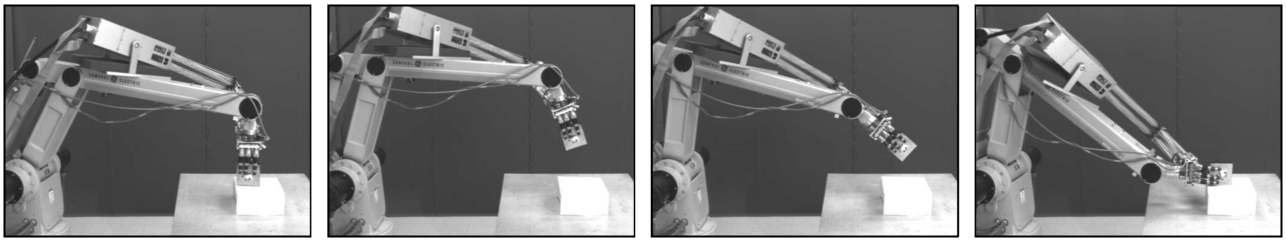

Figure 6: Robot System Performing Reorientation Task

be added incrementally and in a modular fashion into these composition functions.

5.3 Object Rotation

The last example is used to demonstrate the span of the control basis and the power of controller bindings. The task objective is to rotate an object within the hand while maintaining object stability at all times. Again the predictable properties of equilibria achieved by concurrent control activations allow for easy detection of task constraint violations, greatly simplifying the design of a valid composition policy. The policy used is shown in Figure 7.

Initially, the "coarse reach", "fine reach", and "grasp" automata illustrated in Figure 3 establish a stable 4 fingered grasp. Thereafter, the system cycles through a set of stable, three-fingered grasps on the object. The resulting grasps leave the fingers in sub-optimal kinematic configurations due to the geometry of the task. This is compensated by the posture controller through changes in the arm and the rotational axis of the object, leading to the object rotation shown in Figures 8 and 9.

Each stable and kinematically conditioned grasp configuration represents a system equilibrium established by the concurrent activation of two controllers. The first is a contact configuration controller with input state set consisting of three fingers and an output state set consisting of one of these three. The other controller is a kinematic conditioning controllers that optimizes the configuration of all four fingers by performing posture changes in the arm and along the rotational axis of the object.

This example further demonstrates the reusability of the controllers. The same controllers as in the pick-and-place task are used here to accomplish a conceptually very different task. Moreover, dynamically rebinding this control basis to degrees of freedom results in solutions that are robust with respect to task geometry and robot kinematics.

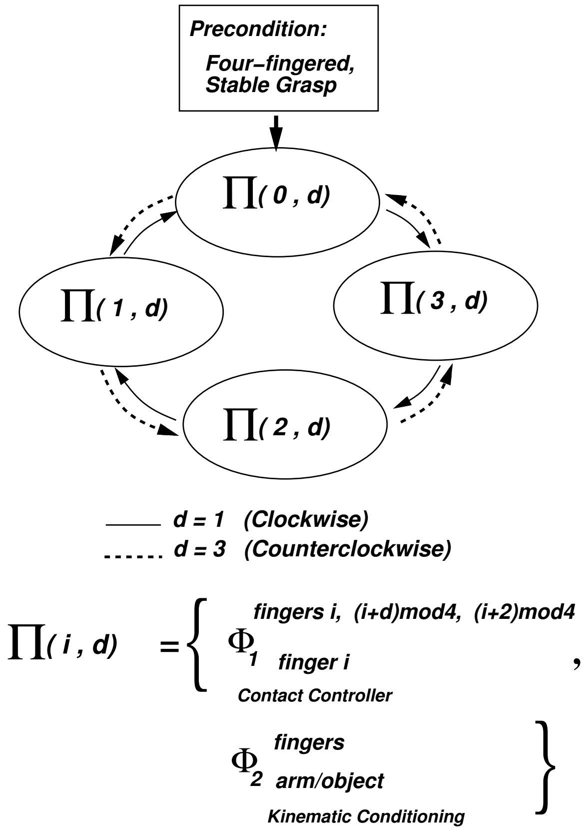

Figure 7: Finite State Machine for Rotation Task

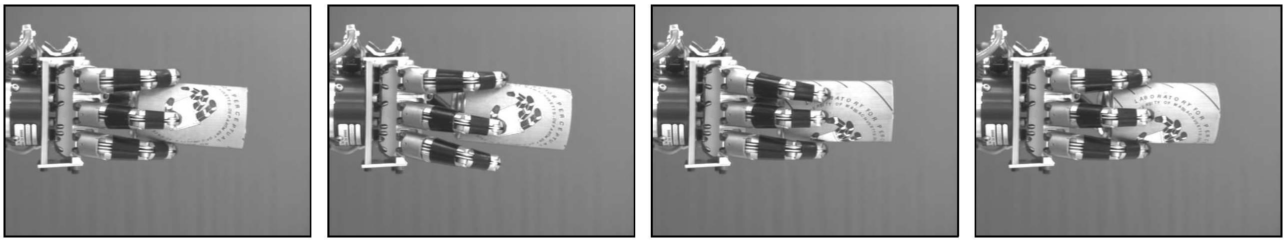

Figure 8: Utah/MIT hand Rotating Soda Can in Free-space

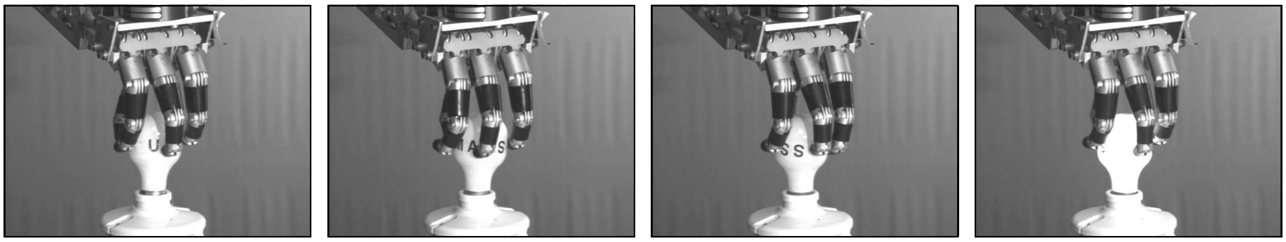

Figure 9: Utah/MIT hand Screwing in a Light Bulb

6 Conclusions

Neither traditional model-based nor behavior-based approaches scale well to complex task domains such as manipulation. To overcome this, the distributed control approach presented in this paper uses a carefully constructed basis of reusable feedback control laws with formal stability and convergence properties which are composed on-line to span a wide variety of tasks. To illustrate this approach, a control basis for manipulation tasks and composition policies for a set of tasks was designed. The resulting composite controllers were applied to a 20 degree-of-freedom hand/arm system and were able to perform tasks robustly with respect to task geometry. Moreover, the construction of correct composition functions was greatly enhanced by the compact control basis representation, and the predictable behavior of the constituent controllers, dramatically reducing the dimension of the space of possible motor alternatives.

Overall, the approach seems to scale well to the complexity of manipulation tasks and the generic character of the control basis allows it not only to generalize across task domains but also appears to improve generalization across a wide variety of hardware platforms.

References

[1] J.S. Albus, R. Lumia, and H. McCain. Hierarchical control of intelligent machines applied to space station telerobots. *IEEE Trans. Aerospace and Electronic Systems*, 24(5):535–541, 1988.

[2] Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, March 1986.

[3] J.A. Coelho Jr. and R.A. Grupen. Control preimaging for multifingered grasp synthesis. In *Proceedings of the IEEE International Conference on Robotics and Automation*, San Diego, CA, May 1994.

[4] Jefferson A. Coelho Jr. and Roderic A. Grupen. Effective multifingered grasp synthesis. Technical Report 94-12, CMPSCI Dept., Univ. of Mass., Amherst, February 1994.

[5] A. Cole, P. Hsu, and S. Sastry. Dynamic regrasping by coordinated control of sliding for a multifingered hand. In *Proceedings of the 1989 Conference on Robotics and Automation*, pages 781–786, Scottsdale, AZ, May 1989. IEEE.

[6] J.H. Connell. SSS: A hybrid architecture applied to robot navigation. In *Proc. IEEE Int. Conf. Robot. Automat.*, pages 2719–2724, Nice, France, May 1992. IEEE.

[7] Christopher I. Connolly. Harmonic functions and collision probabilities. In *Proc. IEEE Int. Conf. Robot. Automat.*, San Diego, CA, May 1994. IEEE.

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

[9] R. Grupen and K. Souccar. Manipulability-based spatial isotropy: A kinematic reflex. In *Proceedings of the International Workshop on Mechatronical Computer Systems for Perception and Action*, pages 157–163, Halmstad, Sweden, June 1993. Halmstad University, Sweden.

[10] Manfred Huber and Roderic A. Grupen. 2d contact detection and localization using proprioceptive information. *IEEE Trans. Robotics Automat.*, 10(1):23–33, 1994.

[11] Z. Li, P. Hsu, and S. Sastry. Grasping and coordinated manipulation by a multifingered robot hand. *Int. J. Robotics Res.*, 8(4):33–50, August 1989.

[12] T. Lozano-Pérez, J.L. Jones, E. Mazer, and P.A. O'Donnell. *Handey*. MIT Press, Cambridge, MA, 1992.

[13] P. Maes. Learning to coordinate behaviors. In *Proceedings of the* 8^{th} *National Conference of Artificial Intelligence*, pages 796–802, Boston, MA, 1990.

[14] A. Meystel. Planning in a hierarchical neested controller for autonomous robots. In *Proceedings of the 25th Conference on Decision and Control*, Athens, Greece, 1986. IEEE.

[15] M.H. Raibert. *Legged Robots that Balance*. MIT Press, Cambridge, MA, 1986.

[16] Peter J.G. Ramadge and W. Murray Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–97, January 1989.

[17] M. Stan, W. Burleson, C. Connolly, and R. Grupen. Analog vlsi for robot path planning. *Journal of VLSI Signal Processing*, 8(1):61–73, Aug 1994.

[18] T. Yoshikawa. Manipulability of robotic mechanisms. *International Journal of Robotics Research*, 4(2):3–9, Summer 1985.