A CONTROL PHILOSOPHY FOR PROSTHETIC HANDS
Raymond A. DeGennaro†, Denise Beattie†, Thea Iberalls‡, George A. Bekey‡
†Dept. of Biomedical Engineering, ‡Dept. of Computer Science,
University of Southern California; Los Angeles, CA 90089-0782

Abstract
The majority of prosthetic hands available today make no pretense of approaching the versatility and functionality of the human hand. Available prostheses generally possess only one degree of freedom, which greatly reduces the manipulative ability of the device. Additional degrees of freedom are rarely added, because of the difficulty in controlling them naturally. We have developed a control philosophy based on minimum information transfer from the amputee and maximum local autonomy of the device. To test this philosophy, we have developed a graphical simulation of a five fingered anthropomorphic device. The results presented show that this is a viable method for the control of a multi-fingered prosthesis.

Introduction
The complex nature of the human hand, under the control of the central nervous system, is highlighted by its loss, and the amputee looks to engineers for a prosthetic replacement. When people lose a hand, they are losing an intimate partner of their minds that combines fine coordinated movement, tactile sensation, proprioceptive feedback, and aesthetic appearance.

Engineers for centuries have been attempting to reproduce this amazing human structure. The first recorded instance of an artificial hand was in 200 BC when a Roman general lost his hand during a war and was fitted with an iron hand. Today, high-tech solutions using new materials and miniaturized components are emerging and new designs are attempting to satisfy active lifestyles of the differently abled. Reduced voltage requirements, battery saving features, lightweight alloys, modular designs, use of reinforced silicones for cosmetic gloves [1], and space age miniaturized motors and circuits have led to lightweight electrically powered prosthetic hands for adults and children that users can comfortably wear during an eight hour day.

Despite these advances, the majority of prosthetic devices available still make no pretense of approaching the versatility and functionality of the human hand. The available prosthetic devices, whether hands, hooks, or non-standard devices, generally have only one degree of freedom (do0 and therefore limited manipulative ability [2]. One of the major stumbling blocks to designing terminal devices with more degrees of freedom is the amputee’s inability to easily and naturally control them. In an attempt to provide a more functional prosthesis, we have developed a control philosophy for use with a dextrous anthropomorphic terminal device which minimizes the amputee’s control inputs while maximizing device functionality. In order to test this control concept, a graphical simulation of a five fingered anthropomorphic device was developed on an Apple Macintosh computer. This paper discusses the control philosophy and the simulator and presents the results of two simulated tasks.
Control Philosophy

In order to overcome limitations of currently available prosthetic devices, we have developed a control concept for a new generation of "biologically inspired" prosthetic hands. The principles of our control philosophy are:

1) minimal information transfer between the wearer and the device,
and
2) maximal local autonomy of the device.

Our intent is to put as much autonomy and intelligence as possible into the hand and its controller, utilizing the interaction between feedforward planning and feedback control. A block diagram of the controller is shown in Figure 1. The inputs to the controller include an object and the task to be performed on or with that object. The feedforward planner chooses and sets-up appropriate "opposition space" [3] and intelligently selects a control algorithm. The preshape controller uses a virtual finger model of the desired opposition space to determine preshape posture for the hand. If grasping is desired, the enclosure controller initiates the smooth closure of the hand until desired grasp forces are achieved. The task performer then performs the desired task. During performance of the task, local feedback from slip, force and position sensors is used to provide stabilization. In addition, an adaptive controller compares actual sensory information to that predicted by the feedforward planner and modifies the control scheme if necessary. Finally, given a release signal, the controller opens the hand and returns to the rest posture.

Simulation Description

This section describes the computer simulation used to test the control scheme. It is assumed that the subject is a below-elbow amputee and the prosthesis must supply the function of the hand and wrist. This requires: a user interface that will graphically display the hand and state variables and allow selection of the task and the object, a simulation of natural transport and orientation, and a method to map task information into the proper control method. In addition, the hand should be parameterized to allow the controller to operate a variety of prosthetic/robotic hands without being re-written.

User Interface

The user is initially presented with a dialog box, Figure 2, that allows him to select a task and an object from pop-up menus. The tasks supported by the simulator are taken from a database of tasks performed by the hand during daily life. The user can also select multiple state variables to be displayed in strip charts and modify the parameters in the different modules.

Transport and Pre-orientation Trajectory Generation

We assume that the prosthesis is being used by a below-elbow amputee who will provide natural transport. Therefore, we need the simulation to follow a "natural" trajectory and to coordinate preshaping with the transport. The wrist trajectory and coordination of transport and preshaping are obtained from an optimal control method [4].
User Interface
allows the user to select the object and the task to be performed

Intelligent Controller

Feedforward Planner
selects the oppositions used and the forces involved for each opposition

Preshape and Preorientation Controller
uses a VF model to control preshaping of the hand

Enclosure Controller
closes the hand around the object

Adaptive Controller
modifies control scheme based on actual sensory information

Task Performer
actually performs the task: Stabilization of the grasp is provided via local feedback from slip, force and position sensors

Release Controller
triggered by signal from user

Figure 1: Block Diagram of the Control Philosophy

Since the amputee does not have a wrist, the prosthesis must orient the hand in pitch and yaw. We assumed that pre-orientation is controlled by a method similar to transport. In the simulation, we used the same controller to transport and orient the hand. This produced a natural-looking change in orientation during transport.

**Determination of Control Method**

After the task to be simulated and an appropriate object are selected, the feedforward controller sets-up an "opposition space" and intelligently selects a control algorithm. This requires information about the task (extracted from a database) and the object (collected by a "vision system"). This information is mapped into an appropriate control scheme.
In previous work, Iberall, et. al. collected and categorized an extensive list of tasks performed by the hands [5]. The database was expanded to include additional information needed to plan the grasp and to select a control method. Fields were added to specify movement primitive, preference for more or fewer fingers, position of unused fingers and, if it is used, the initial position of virtual finger 3 (VF3) [3]. Additional information about the object (e.g. diameter, width, shape, position and orientation) is supplied by the “vision system.” This is a black box that supplies information about the object to the controller. Future work will include the development of a new user interface and sensory input modules which allow the prosthetic hand to acquire the needed information.

The Feedforward Planner takes the general information supplied by the database and determines the actual preshaping of the hand. Based upon the object’s width and the preference for a greater or lesser number of fingers, the planner maps a number of real fingers into each of the virtual fingers (VF’s) used in the opposition. The object’s diameter is used to determine the desired preshape aperture. The object’s shape, number of real fingers in each VF and position of unused fingers are mapped into a preshape posture [6]. Finally, the final position and orientation are determined based upon the object’s location, the opposition space selected and the movement primitive. This is the extent of the feed-forward planning.

Next, the transport, pre-orientation, and preshape sub-controllers are used to position the hand. If the task requires the object to be grasped, then the enclosure controller is activated. At this point, based on the movement primitive, the appropriate execution sub-controller is run. Upon completion of the task, the object is released. These actions make use of local feedback to maintain grasp stability and control the task.
Parameterization of Hands

Since a practical dextrous prosthetic hand has not been designed yet, it is important for the controller to be able to drive a variety of robotic hands. This ability is achieved by parameterizing the hands. The hand specific parameters are maximum aperture, maximum number of VF’s, preshape and enclose times. In addition, the real to virtual finger mapping, which is also hand specific, is performed by the hand driver. The communication between the hand and the controller is divided into routines that every potential dextrous prosthetic hand needs to implement. The controller drives the hand by calling routines that manipulate the force(s) and/or aperture between VF’s. This allows one controller to drive numerous hands without modification.

We have developed a C++ class hierarchy with abstract classes for each module (the controller, graphical displays, database interfaces, and robotic hand and arm drivers). Actual instances of these modules are descendants of the abstract classes. This allows the modules to communicate through a consistent set of routines and specific implementations can be changed without the need to rewrite the other modules.

Simulation Results

The two tasks chosen from the database for simulation represent two classes of hand motion. Flicking on a light switch incorporates a preshape motion and a task motion but no grasping is required while turning a doorknob requires preshaping, grasping and task motion. The results presented show the hand configurations at various points during the simulation. Figures 3-6 show the side and top views of the hand posture for flicking a light switch. Figure 3 displays the hand in the initial position and Figure 4 shows the hand preshaping. Figure 5 displays the hand just prior to execution of the task and Figure 6 shows the task being executed. Figures 7-9 show the side and top views of the hand configurations for turning a doorknob. Figure 7 show the hand preshaping, figure 8 shows the hand just prior to execution and figure 9 shows the task being executed.

Conclusions and Future Work

Advances in materials and components have given us the opportunity to develop prostheses that are closer to the versatility and functionality of a natural hand than currently available devices. The limitations in information transfer between the wearer and the device, however, impede the development of more functional prostheses. It is our belief that the next generation of prosthetic devices should be as autonomous as possible. Our simulation demonstrates that self-governing prostheses are realizable.

In the future, we plan to increase the number of tasks from the database that can be simulated and replace the stick-figure representation with rendered 3D animation. With the addition of other related aspects of grasping, we plan to develop a command language to describe and control automated grasping. This command language will be applied to the development of the next generation of prosthetic hands.
Figure 3: Initial hand posture and position.

Figure 4: The hand in the process of preshaping for the task, Flicking a Light switch
Figure 5: The hand just prior to executing the task, Flicking a Light switch

Figure 6: The hand executing the task, Flicking a Light switch
Figure 7: The hand in the process of preshaping and orienting for the task, Opening a Doorknob

Figure 8: The hand just prior to executing the task, Opening a Doorknob
Figure 9: The hand executing the task, Opening a Doorknob

References


