

The Virtual Tool Approach to Dexterous Telemanipulation *

Olac Fuentes and Randal C. Nelson
{fuentes,nelson}@cs.rochester.edu

<http://www.cs.rochester.edu/u/{fuentes,nelson}/>
Computer Science Department
University of Rochester
Rochester, New York 14627

Abstract

We propose a behavioral construct called a “virtual tool” that allows many of the advantages of high-degree-of-freedom robotic systems to be realized without excessive computational costs or modeling requirements. The basic idea of a virtual tool is drawn from the observation that the most efficient way of performing a task is generally to use a device that is specially designed for it; this suggests that a way to use redundant degrees of freedom in a system is to use them to customize it so that it becomes, in essence, a special purpose tool for the task at hand. We call the resulting instantiation a virtual tool. Formally, this tailoring takes the form of implementing constraints on the degrees of freedom of the system through lower-level control processes. Remaining degrees of freedom appear as control parameters of the virtual tool, and can be used by higher-level processes, or by a human teleoperator, to control the system.

We present experimental results showing the application of the virtual tool approach to telemanipulation using a 22 degree-of-freedom hand-arm system. Our chosen task is a fairly complex assembly task requiring the robot to pick up a small object, transport it, and insert it into a hole with matching shape. Instead of attempting to map the configuration of the teleoperator’s hand-arm system to the robot using a data-glove or similar device, we provide basic parameterized grasping and manipulation primitives (or virtual tools) that can be invoked by the teleoperator as he progresses through the stages of the complex task.

1 Introduction

Robotic arm-hand systems with capabilities that are comparable to those of their human counterparts are becoming available in research environments. One promising application of these systems is in teleoperation, where they can be guided remotely by an operator to solve tasks in environments that are too inhospitable for humans. In spite of its wide variety of potential applications, achieving reliable telemanipulation with these high degree-of-freedom systems has proven to be a very challenging task. The high dimensionality of their parameter space and the difficulty of mapping the configuration of the teleoperator’s hand-arm system to the robot make this difficult.

Fortunately, for most operations these systems can be called upon to perform, it is possible to partition the control regime in a way that requires the teleoperator to provide only a few carefully chosen control parameters, while the rest of the degrees of freedom of the system are handled by local processes at the robot site. For example, consider a teleoperation where the robot is required to pick up an object and then translate and reorient it. The traditional way of solving this task is to try to map the configuration of the teleoperator’s hand-arm system to that of the robot. A more promising way is to provide primitives for grasping and manipulation such that the teleoperator only needs to specify the amount of force to be applied to the object during grasping and a six dimensional vector indicating the desired translation and rotation of the object. This is the basic idea behind the virtual tool approach.

The key idea of virtual tools is drawn from the observation that the most efficient way of performing a task is generally to use a device that is specially designed for it. This suggests that a way to take advantage of the redundant degrees of freedom in a system is to use them to customize, or “tailor” the system so

*This work is supported by ONR grant N00014-93-I-0221 and NSF IIP Grant CDA-94-01142

that it becomes, in essence, a special purpose tool for the task at hand. We call the resulting instantiation a *virtual tool*. More precisely, a virtual tool is a set of constraints that make the system look to higher-level processes as if it were a device specialized for its current task. The remaining degrees of freedom appear as control parameters of the virtual tool, and form the interface used by higher-level control processes, such as a teleoperator. Since they are directly linked to the task and of low dimensionality, the control problem is greatly simplified, and communication between high and low level processes does not require a high bandwidth interface. Most robotic tasks, even very complicated ones, can be decomposed in a way that requires controlling only a few carefully chosen degrees of freedom at any given time. By swapping in and out different sets of constraints, the system can transition smoothly from one virtual tool to another as the task requirements change, without incurring any physical discontinuities. This is achieved by selecting transition points in such a way that the initial physical configuration of the robot for a given tool is identical to the final configuration for the previously used tool.

The concept of virtual tools is a generalization of some techniques that have been described previously for dealing with more specific situations. A good example is the notion of virtual fingers introduced by Arbib *et al.* [2]; in this work, a grasp involves the use of virtual fingers, which are composed of one or more real fingers working together to perform a task. Iberall [5] shows how the hand can be used as essentially three different grippers by changing the mapping of virtual fingers to real fingers. Closely related to our work is that of Michelman and Allen [7], who have used a low-degree-of-freedom interface between a teleoperator and a robotic system. In that work a simple device such as a joystick is used to command simple primitive actions, such as translation and rotations, to the Utah/MIT hand. The idea of partitioning the control of a teleoperated robotic system into a low-level layer, controlled autonomously, and a high-level layer, controlled by the teleoperator, is also used in [9].

The organization of the remainder of this paper is as follows. Section 2 describes in a little more detail the virtual tool approach and its relation to behavioral robotics. Section 3 describes the set of virtual tools used for our example dextrous telemanipulation task. Section 4 shows experimental results. Section 5 presents conclusions and offers some promising directions for future work.

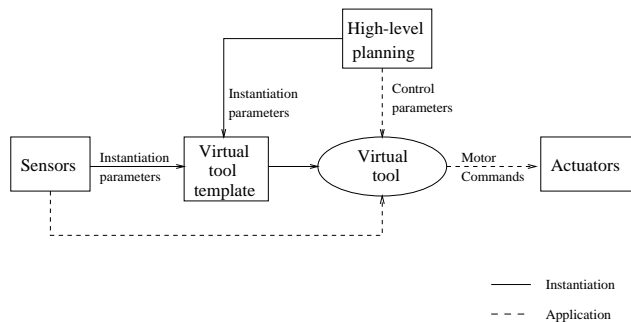


Figure 1: Instantiation and application of virtual tools

2 Virtual Tools and Behavioral Robotics

The virtual tools approach falls under the general topic of behavioral robotics [3]. In behavioral robotics, the robotic system is viewed as always being engaged in some particular task or behavior. Rather than looking for general control formulations that will permit equally simple specification and execution of all actions the robot is physically capable of, the behavioral approach considers the specific activities the robot will be called upon to perform, and looks for useful common components that can be abstracted out as primitives. Such generally useful activities are referred to as behaviors. Basically, the behavioral approach claims that the notion of a general purpose robot is an untenable concept; even the most flexible proposed systems have strong task assumptions implicitly embedded at all levels of their design. More to the point, the behavioral approach holds that there is much to be gained by making these task requirements explicit, and embedding them into the system at as low a level as possible. This can be seen as a strategy of most commitment.

In this paper, we use the term *behavioral* to imply that task knowledge is incorporated as an essential part of resolving the problem of control of dextrous manipulators. Such knowledge enters into the process at the highest level in the design of the virtual tools, at an intermediate level in determining the selection of and transition between appropriate tools, and at the lowest level in the instantiation of a tool. In the most general sense, imposing a set of dimensionality-reducing constraints on an active system can be viewed as instantiating a behavior. The approach can also be seen as a method of using domain and task knowledge to partition the control effort between high and low level processes.

A virtual tool is, in essence, a template that can be instantiated into many different parameterized be-

haviors, as shown in figure 1. Invoking a virtual tool dynamically creates a new parameterized behavior at the instantiation point on the basis of purely sensory information. For example, one could use behaviors of the form: “translate the object you’re grasping by $\langle x, y, z \rangle$ ” and “grasp the object you’re seeing.” This is somewhat related to the concept of *deictic control* [1, 11], where knowledge about the context in which actions are performed is used to simplify the representations used to solve planning and learning problems.

3 Virtual Tools for Dextrous Telemanipulation

The traditional approach to telemanipulation using dextrous hands has been to use a dataglove, exoskeleton or similar device to sense the configuration of the teleoperator’s hand and arm and then to map them to the robotic manipulator. For tasks requiring even a moderate degree of dexterity this approach usually fails for several reasons. First, existing robotic hands are only approximately anthropomorphic, thus there is no straightforward way to reliably map the joint angles of the teleoperator’s hand to those of the robotic hand. Second, with this approach we would need to obtain sixteen different parameters from the teleoperator, and each parameter is a potential source of error. Third, it is difficult and sometimes impossible to give sensory feedback from the robot to the teleoperator [7].

Instead of attempting to map the configuration of the teleoperator’s hand-arm system to the robot, we provide parameterized grasping and manipulation virtual tools that can be invoked by the teleoperator as he progresses through the stages of the complex task. The teleoperator, using a Polhemus position and orientation sensor or similar device, supplies the values of the control parameters required as input to the virtual tools.

So far we have implemented a 1 degree-of-freedom (DOF) grasping tool, a 3 DOF gross manipulation tool and a 6 DOF fine manipulation tool. These tools are sufficient to perform a wide variety of pick-and-place and assembly tasks. The following subsections describe in more detail the design and implementation of each of the three tools.

3.1 A Grasping Tool

In order to manipulate objects with a dextrous hand, they must first be grasped with what is commonly referred to as a *precision grasp*, that is, one

that allows manipulation with the fingers. In order to avoid overconstraining the fingers, such grasps typically involve only fingertip contacts. The virtual tool we use for grasping is based on the observation that, particularly in the case where we have more than three fingers, and hence redundant stability, grasping can be viewed as a process of positioning and preshaping the hand, and then closing the fingers as a unit until a grasp stability criterion is satisfied. Local processes are associated with each finger to avoid unstable positions and to detect contact.

At a higher level of detail, the grasp tool is actually composed of two similar one-parameter *subtools* instantiated sequentially, and connected by a hidden transition. To start, we select a preshape based on rough information about the object’s shape and position the hand about the object. The fingers are then closed as a group, each towards an appropriate goal location under the guidance of a single control parameter until all fingertips touch the object. Local processes operating off tactile sensors freeze each finger as it contacts the object. A transition then takes place, and a second subtool is then instantiated that moves setpoints for the fingers inboard of the boundary, in a manner that achieves a secure and stable grasp.

The preshape information, together with the goal positions for each finger constitute the instantiation parameters of the first subtool. The positions at contact instantiate the second subtool. Because both tools are single parameter operations, and since the second can be automatically instantiated at the termination of the first, the transition is effectively hidden and the entire grasp can be viewed as a single one-parameter operation from the point of view of the user process. If the user does not need to control the speed of the grasp, even this parameter can be hidden, and the entire procedure viewed as an atomic operation. The net effect of this grouped movement is somewhat similar to Arbib’s virtual finger concept, but in a more general setting.

In general, we can select from a set of preshape and goal closures, however, for the class of objects we have dealt with so far, a single preshape has sufficed. When used in this way, the above procedure is not guaranteed to produce the optimally stable grasp, by any particular criterion. However, because we have more fingers than are needed to produce a minimally stable grasp, it produces a good grasp in most situations involving objects having a minimum diameter greater than the diameter of a finger, and a maximum diameter smaller than the length of a finger. It is possible to contrive situations (e.g. slippery wedges) where sta-

ble grasps are difficult to find, and this sort of strategy fails; such situations, however, can be reasonably considered to be pathological. Moreover, it is possible to recognize such situations and take corrective action.

3.2 A Gross Manipulation Tool

In our arm-hand system, gross manipulation is done by translating the arm’s end effector, where the hand is attached. The control parameters of this tool are the coordinates of the end effector, which are obtained from the coordinates of the Polhemus receiver and a simple re-scaling. Gross manipulation is used to take the hand to the vicinity of the object before grasping it and then to take it to the vicinity of the hole where the piece will be inserted.

3.3 A Fine Manipulation Tool

In the fine manipulation tool, the teleoperator provides the desired position and orientation of the object being manipulated and the hand autonomously performs the required manipulation on the object to obtain the desired configuration. For rigid objects, this entails re-orientation in a six-dimensional space. A general rigid manipulation tool would thus have six control parameters. Instantiation of further constraints would permit such a basic tool to be cast into more specialized tool, for instance one performing only translation or rotation about a particular axis. We refer to such specializations as *derived tools*.

The tool does not rely on *a priori* models; it is instantiated on the basis of sensory information available at the termination of the grasping process. The basic process involves a transition from the grasp tool to the manipulation tool. The instantiation parameters of the manipulation tool are derived from the commanded fingertip positions, known as *setpoints*, at the termination of the grasp. These values are directly available from the hand’s sensors.

The basic idea of our technique is based on the observation that the setpoints of the fingers define a rigid object in three-dimensional space, in this case a (possibly degenerate) tetrahedron. We refer to this object as the *grasp tetrahedron*. The contact points of the fingers on the object define another object that we refer to as the *contact tetrahedron*. The locations of the vertices of the grasp and contact tetrahedra, shown in figure 2, can be obtained respectively from the measurements of the commanded and actual joint angles and the forward kinematics of the hand. Because of the compliant control system, we can model the statics of the situation by regarding each vertex

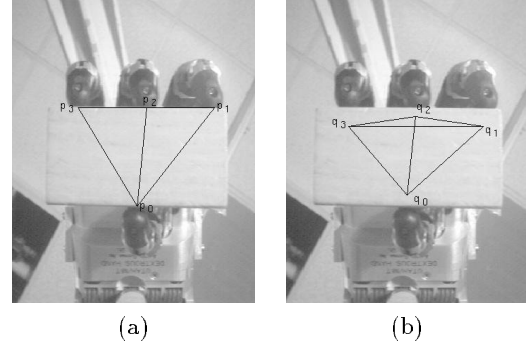


Figure 2: The contact tetrahedron (a) and the grasp tetrahedron (b).

of the contact tetrahedron as being attached to the corresponding vertex of the grasp tetrahedron by a virtual spring. In the case of an object free to move, net wrench is constrained to be zero. Moreover, for a well conditioned grasp, the wrench zero coincides with a deep local minimum of the spring energy function on a manifold defined by the rigid displacements of the grasp tetrahedron with respect to the contact tetrahedron. This means that if we treat the fingertip contacts as fixed points on the object, then the object can be rotated and translated by executing the desired rigid transformation on the grasp tetrahedron. In other words, displace the grasp tetrahedron and the object will follow. Somewhat similar approaches have been suggested in [8] and [10].

In more detail, the formulation is as follows. Let ${}^H p_0, {}^H p_1, {}^H p_2$ and ${}^H p_3$ be the fingertip to object contact points, where ${}^H p_i$ denotes that p_i is being measured with respect to a hand-centered reference frame H . Let ${}^H q_0, {}^H q_1, {}^H q_2$ and ${}^H q_3$ be the commanded fingertip positions or setpoints. Let ${}^H c = [c_x, c_y, c_z]^T$ be the coordinates of the centroid of the grasp tetrahedron with respect to frame H . We define an object-centered frame of reference C , with its axes parallel to H ’s and with origin at c , as shown in figure 3.

Given the manipulation command $\langle x, y, z, \alpha, \beta, \gamma \rangle$, where x, y and z represent displacements of the object with respect to its initial position and α, β, γ are interpreted as sequential rotations around the x, y and z axes of C , the resulting setpoints q'_0, \dots, q'_3 are given by:

$$\begin{bmatrix} q'_{ix} \\ q'_{iy} \\ q'_{iz} \\ 1 \end{bmatrix} = T(\langle x, y, z, \alpha, \beta, \gamma \rangle) \begin{bmatrix} q_{ix} - c_x \\ q_{iy} - c_y \\ q_{iz} - c_z \\ 1 \end{bmatrix}$$

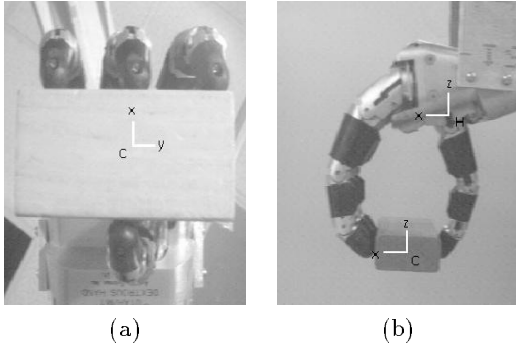


Figure 3: The hand-centered reference frame H and the object-centered reference frame C

Where $T((x, y, z, \alpha, \beta, \gamma))$ is given by:¹

$$T = \begin{bmatrix} cac\beta & cas\beta s\gamma - sac\gamma & cas\beta c\gamma + sas\gamma & x + c_x \\ sac\beta & sas\beta s\gamma + cac\gamma & sas\beta c\gamma - cas\gamma & y + c_y \\ -s\beta & c\beta s\gamma & c\beta c\gamma & z + c_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Although, due to non-zero finger size and other effects, the assumption of fixed contact points is not strictly correct, our experiments have shown that the compliance of the Utah/MIT hand compensates for the errors introduced by the use of this assumption [4, 6].

4 Experimental Results

We implemented a dextrous telemanipulation system based on the virtual tool idea using a 6 DOF PUMA arm and a 16 DOF Utah/MIT dextrous hand. We will show how the virtual tools approach enables us to perform dextrous telemanipulation reliably with this complex robotic system.

The chosen task is assembling a puzzle consisting of several pieces with different shapes that have to be inserted into matching holes. An individual insertion requires translating the hand to the vicinity of the object, grasping it, transporting the hand (and the object) to the vicinity of the matching hole, inserting the object into the hole and dropping the object. This task was performed using the three different virtual tools described in the previous section: a 1-DOF grasping tool, a 3-DOF gross manipulation tool and a 6-DOF fine manipulation tool.

The teleoperator uses a Polhemus magnetic sensor to provide the control parameters to the virtual

¹we use $c\theta$ and $s\theta$ as shorthand for $\cos(\theta)$ and $\sin(\theta)$, respectively

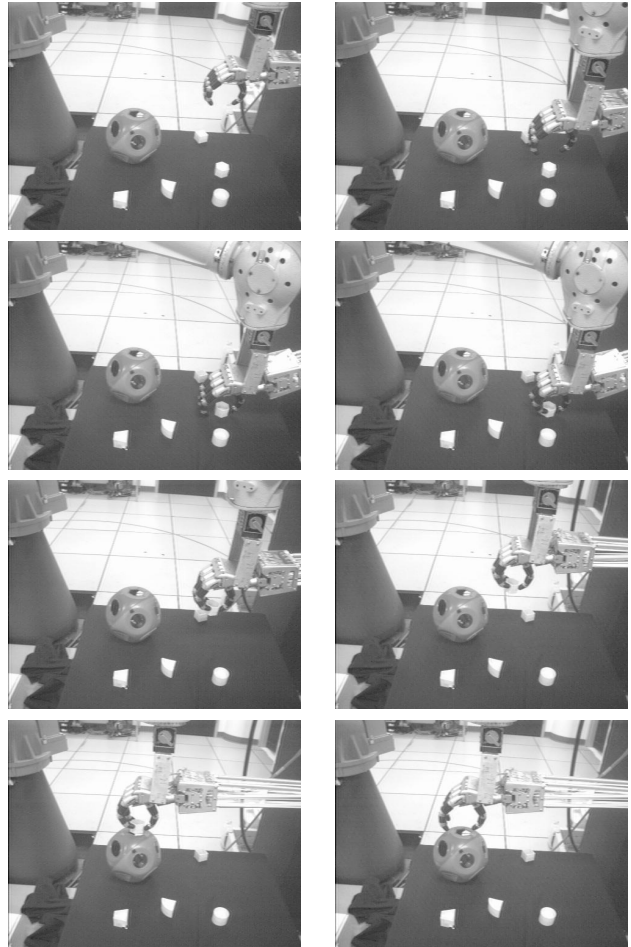


Figure 4: An assembly task using the virtual tool approach to telemanipulation

tools. For the 6 DOF fine manipulation tool, changes in the position and orientation of the Polhemus receiver are mapped to changes in position and orientation of the object. Similarly, for the gross manipulation tool, changes in the position of the sensor are mapped to displacements of the end-effector. For the 1-DOF freedom grasping tool, up and down displacements of the sensor are mapped to the degree of closure of the grasp. The teleoperator can cycle among the tools by pressing a pressure-sensitive pad taped to the back of the Polhemus receiver.

Figure 4 shows a telemanipulated insertion operation. Initially the teleoperator uses the gross manipulation tool to position the hand near the desired object, then, using the pressure-sensitive pad, he signals a transition to the grasping tool. When the object has been grasped, the teleoperator switches back to the gross manipulation tool to take the object to the

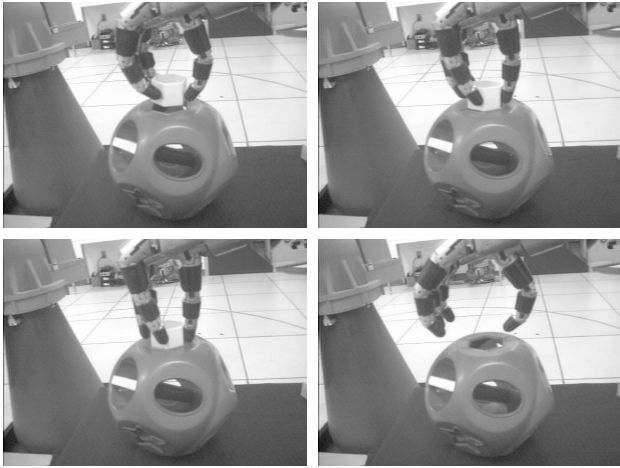


Figure 5: Closeup of the insertion operation

vicinity of the hole using arm motion only. After that, the fine manipulation tool, is used to insert the piece into the hole. Finally, the grasping tool is used again to open the hand and drop the object into the hole, completing the operation. A closeup of the final phase is shown in figure 5.

5 Conclusions and Future Work

We have presented the virtual tools approach as a simple and intuitive way of controlling high-degree of freedom robotic systems. We showed the application of the virtual tools idea to dextrous manipulation, showing how it can be used to perform complex tasks that need a significant degree of dexterity in a robust and reliable way. Virtual tools do not require extensive modeling, a crucial feature for operation in unknown and dynamic environments. They also allow the teleoperator to issue complex commands using a low-bandwidth communication channel with the robot.

Future extensions of this work include integrating a large number of virtual tools, including power grasping tools and extension of the 6DOF manipulation tool to manipulate flexible objects, allowing non-rigid transformations of the grasp tetrahedron, such as bending and twisting.

References

[1] P. E. Agre and D. Chapman. Pengi: an implementation of a theory of activity. In *AAAI*, pages

268–272, 1987.

- [2] M. Arbib, T. Iberall, and D. Lyons. Coordinated Control Programs for Movements of the Hand. Technical report, Department of Computer and Information Science, University of Massachusetts at Amherst, Amherst, Massachusetts, 1985.
- [3] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–22, April 1986.
- [4] O. Fuentes and R. C. Nelson. Experiments on dextrous manipulation without prior object models. Technical Report 606, Computer Science Department, University of Rochester, Rochester, New York, January 1996.
- [5] T. Iberall. The nature of human prehension: Three dextrous hands in one. In *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*, pages 396–401, Raleigh, North Carolina, 1987.
- [6] M. Jägersand, O. Fuentes, and R. C. Nelson. Acquiring visual-motor models for precision manipulation with robot hands. In *Proceedings of the Fourth European Conference on Computer Vision*, Cambridge, U. K., 1996.
- [7] P. Michelman and P. Allen. Shared autonomy in a robot hand teleoperation system. In *Proceedings of the 1994 Conference on Intelligent Robotics Systems*, pages 253–259, 1994.
- [8] T. Okada. Computer control of multijointed finger system for precise object-handling. *IEEE Transactions on Systems, Man, and Cybernetics*, 12(3), 1982.
- [9] P. K. Pook and D. H. Ballard. Teleassistance: Contextual guidance for autonomous manipulation. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 1291–1296, Seattle, WA, 1994.
- [10] K. Salisbury, D. Brock, and P. O’Donnell. Using an articulated hand to manipulate objects. In M. Brady, editor, *Robotics Science*. MIT Press, 1989.
- [11] S. D. Whitehead. *Reinforcement Learning for the Adaptive Control of Perception and Action*. PhD thesis, Department of Computer Science, University of Rochester, Rochester, NY, November 1991.