Proceedings of the 1998 IEEE
International Conference on Robotics & Automation
Leuven, Belgium • May 1998

Haptic Manipulation of Virtual Mechanisms from Mechanical CAD Designs

Ali Nahvi, Donald D. Nelson, John M. Hollerbach, and David E. Johnson

Depts. Computer Science and Mechanical Engineering
Univ. Utah, Salt Lake City, UT 84112

Abstract

A haptic display system is presented for manipulating virtual mechanisms derived from a mechanical CAD design. Links are designed and assembled into mechanisms using Utah's Alpha_1 CAD system, and are then manipulated with a Sarcos Dextrous Arm Master. Based on the mechanism's kinematics and the virtual grasp, the motion of the master is divided into motion of the mechanism and constraint violation. The operator experiences the dynamic forces from the mechanism plus constraint forces.

1 Introduction

Designers of complex mechanical assemblies typically create physical prototypes to evaluate part interaction, ease of assembly, and functionality. To avoid this time-consuming and costly procedure, virtual prototyping seeks instead to employ realistic simulation and immersive interfaces. Although mechanical CAD systems provide realistic visual displays, they do not permit a designer to manipulate and interact with mechanical elements in a realistic manner. A haptic interface coupled with a visual display would allow a designer to evaluate car dashboard designs [12] and to experience assembly forces [4].

Another application which is the focus of the present paper is the manipulation of virtual mechanisms, such as might be found in animatronic figures. Designers of such figures indicate that appreciation of mobility, workspace, potential collisions, and driving forces could be facilitated by virtual haptic manipulation. The experiencing of assembly forces was considered to be less important for this application, because of the difficulty in simulating press fits. A related application would be the design of mechanisms manipulated by humans, such as rehabilitation devices and exercise equipment.

We have interfaced the Sarcos Dextrous Arm Master to Utah's *Alpha_1* mechanical CAD system [5]. A major challenge has been to make the CAD system interactive, by adding real-time geometry, dynamic simulation, and haptic interface control. The kHz bandwidths required for crisp force reflection present a serious computational challenge. Our approach has been to divide the computations between a workstation, which implements the *Alpha_1* CAD system, graphical display, and global minimum distance computations, and VME-based single board computers (SBCs), which implement surface interactions, dynamic simulation, and haptic interface control (Figure 1). The real-time environment employs VxWorks (Wind River Systems, Inc.) and ControlShell (Real Time Innovations, Inc.). Instead of Ethernet communication between the workstation and SBCs, we employ the Myrinet (Myricom, Inc.) for consistent low-latency and high-bandwidth communications.

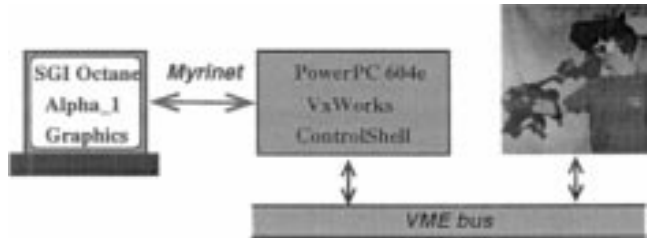

Figure 1: System configuration.

Global minimum distance calculations [6] determine where the haptic interface is about to contact a model, and is performed at a rate of a few Hz; hence it is suitable for the workstation. Surface contact and tracing is accomplished at kHz rates on the SBCs [13]. A key point in these geometric computations is the direct use of NURBS models from the design, without needing to resort to intermediate representations to make the computations sufficiently fast. We have also added a basic grasping capability [9]. It is an advantage of the Sarcos master to have a three degree-of-freedom (DOF) gripper for grasping, as well as a seven DOF arm for natural reaching.

0-7803-4300-x-5/98 $10.00 © 1998 IEEE

In the present paper, we do not yet handle the approach to a mechanism and its grasp. Instead, we assume a grasp has already been made, either fully constrained or equivalent to point contact. We also limit the mechanisms to open or single closed loops, and the grasp to fully constrain the mechanism's motion. The manipulation of multiple closed loops is under development. Based on the mechanism's kinematics and the virtual grasp, the motion of the master is divided into motion of the mechanism and constraint violation. The operator experiences the dynamic forces from the mechanism plus constraint forces.

2 Dynamics and Haptic Interaction

A key issue in the interaction of a haptic interface with a virtual assembly is the choice of a compliance model or a stiffness model. In a compliance model, the haptic interface measures forces and produces displacements based upon a dynamic simulation [17]. In a stiffness model, the haptic interface measures displacements and returns simulated forces [16]. We have chosen a stiffness model, well suited for our haptic interface, the Sarcos Dextrous Arm Master, which is a good force source because of joint torque control.

A second key issue is the use of an inverse dynamics versus forward dynamics computation. It is common to model surface contact by springs, so that a haptic displacement is interpreted as compression of springs [16]. The resultant spring force is then applied to a forward dynamics computation. However, we consider that the contacted surface is very stiff relative to assembly motion, and so we ascribe that portion of the haptic interface motion allowed by the mechanism completely to mechanism motion rather than including surface compression. Simulations reinforce this intuition. Consequently, an inverse dynamics computation is performed, which is substantially simpler than a forward dynamics approach. For the portion of the haptic interface motion not allowed by the mechanism, we return the usual viscoelastic constraint forces.

There have been many approaches towards inverse and forward dynamics of open and closed chains, reviewed in the context of haptic interfaces by [3]. Our general approach towards closed-chain dynamics is similar to approaches of [7, 8], in that a cut joint in a loop is replaced by constraint forces. One difference is that the operator is providing the driving forces rather than an actuator. An advantage of this method over the Lagrange multiplier method is that constraint forces at the joints are explicitly calculated, which help a designer in bearing selection.

Next, we introduce a procedure to assemble virtual mechanisms by setting coordinate systems and inertias. An inverse kinematics procedure is presented by which the closest pose of the virtual mechanism to the haptic interface is found. Dynamic interaction with open chains and single closed chains are described. Finally, experimental results are presented.

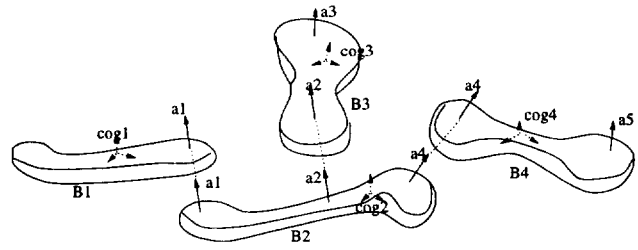

Figure 2: Assembling a mechanism by aligning anchors.

3 Synthesis of the Virtual Mechanism

In designing a link under *Alpha_1*, a user places a coordinate system, termed an anchor, at each location on the link where there is to be joint. The **z** axis defines the direction of rotation or translation, and the origin defines the joint center. The reference coordinate system is that of the drawing system. For each link, *Alpha_1* also calculates an anchor describing the location of the center of gravity, the principal axes of inertia, and the principal inertias. The mass is also an attached property of the link.

In assembling a mechanism, a user specifies the coincidence of anchors on adjacent links (Figure 2). At each joint, we extract a description of the axis as a line in space, and deduce standard Denavit-Hartenberg (DH) parameters using Sklar's method [11]. The inertial parameters of the link are then translated into the link's DH coordinate system.

Finally, a user identifies a grasping frame on the linkage by a predefined grasp. Six DH parameters are assigned to locate the grasping frame relative to the proximal joint of the grasped link.

4 Inverse Kinematics

The motion of the operator's hand has to be divided into free motion of the virtual mechanism and motions which violate geometrical constraints. For the free motion, a dynamic force due to virtual link dynamics is to be calculated and returned to the operator. For the constrained motion, a penalty force due to constraint violation is to be returned and summed with the dynamic force. We employ the terms *dynamic wrench* $\mathbf{w}_d$ and *constraint wrench* $\mathbf{w}_c$ to represent these contributions.

Consider a haptic interface with n task degrees of

freedom (DOFs) at the endpoint, interacting with a virtual mechanism with m DOFs. For the Sarcos Master, $n = 6$, although it is a redundant manipulator with 7 joints. In order to be able to move the virtual mechanism, we assume that $n \geq m$ and that the workspace of the virtual mechanism is a subset of the workspace of the haptic interface.

The operator grasps one of the links of the virtual mechanism. In the following formulation, we assume that motion of the haptic interface completely prescribes the motion of the virtual mechanism. For example, the operator is prescribing the motion of the middle link of a 4-bar mechanism in Figure 4. We do not handle links that can flop about, such as would happen when an operator moves the first link of a two-link serial manipulator; a forward dynamics computation is required for such cases.

From a reference position $\mathbf{p}_{v.m.}$ and orientation $\mathbf{R}_{v.m.}$ of the virtual mechanism, suppose that the haptic interface is now incrementally displaced to a new position $\mathbf{p}_{h.i.}$ and orientation $\mathbf{R}_{h.i.}$. We have to find the corresponding new position of the mechanism plus the new amount of constraint violation. This is solved as a constrained optimization problem, achieved by linearization and iteration. The procedure is briefly illustrated for single closed chains. The open-chain case is a simplification of this procedure.

The linearized objective function is formulated in terms of an error vector $\mathbf{e}_1$ representing the deviation of the current pose of the grasping frame of the virtual mechanism from the pose of the haptic interface end frame:

$$\mathbf{e}_1 = \mathbf{J}\Delta\mathbf{q} = \begin{bmatrix} \Delta\mathbf{p}_{grasp} \\ \Delta\mathbf{r}_{grasp} \end{bmatrix} \tag{1}$$

where $\mathbf{J}$ represents the Jacobian of the virtual mechanism, and $\Delta\mathbf{q}$ is an incremental joint variable to be added to the current values of the joint variable $\mathbf{q}$. The position error of the grasping frame is

$$\Delta\mathbf{p}_{grasp} = \mathbf{p}_{h.i.} - \mathbf{p}_{v.m.} \tag{2}$$

while the orientation error is expressed as a differential orthogonal rotation $\Delta\mathbf{r}_{grasp}$ extracted from

$$\mathbf{S}(\Delta\mathbf{r}_{grasp}) = (\mathbf{R}_{h.i.} - \mathbf{R}_{v.m.})\mathbf{R}_{v.m.}^T \tag{3}$$

where $\mathbf{S}(\Delta\mathbf{r}_{grasp})$ is the skew-symmetric matrix formed from $\Delta\mathbf{r}_{grasp}$ [1].

The position displacements around a closed loop should sum to zero, while the rotation displacements expressed as rotation matrices should result in the identity matrix $\mathbf{I}$. Suppose the computed position and orientation of the end frame of the mechanism are $\mathbf{p}_{end}$ and $\mathbf{R}_{end}$ respectively. The linearized constraint equation is formulated as an error vector $\mathbf{e}_2$ representing the deviation from a closed loop:

$$\mathbf{e}_2 = \mathbf{C}\Delta\mathbf{q} = \begin{bmatrix} \Delta\mathbf{p}_{end} \\ \Delta\mathbf{r}_{end} \end{bmatrix} \tag{4}$$

where $\mathbf{C}$ represents the Jacobian of the virtual mechanism from the base to the last link, $\Delta\mathbf{p}_{end}$ is the position error of the end frame of the virtual mechanism:

$$\Delta\mathbf{p}_{end} = \mathbf{0} - \mathbf{p}_{end} = -\mathbf{p}_{end} \tag{5}$$

and $\Delta\mathbf{r}_{end}$ is the orientation error vector of the end frame of the virtual mechanism extracted from the following skew-symmetric matrix:

$$\mathbf{S}(\Delta\mathbf{r}_{end}) = (\mathbf{I} - \mathbf{R}_{end})\mathbf{R}_{end}^T = \mathbf{R}_{end}^T - \mathbf{I} \tag{6}$$

The linearized constrained optimization problem can be stated as:

$$\min_{\Delta\mathbf{q}} \ \mathbf{e}_1^T\mathbf{e}_1 \qquad \text{subject to } \mathbf{e}_2 = \mathbf{0} \tag{7}$$

To solve (7), we employ a standard penalty method by giving extra weight λ to the constraint violation $\mathbf{e}_2$ to force the loop to be closed:

$$\begin{bmatrix} \mathbf{J} \\ \lambda\mathbf{C} \end{bmatrix} \Delta\mathbf{q} = \begin{bmatrix} \mathbf{e}_1 \\ \lambda\mathbf{e}_2 \end{bmatrix} \tag{8}$$

This overdetermined set of equations is solved by standard least squares, and iterated as necessary. From the final solution, joint velocities and accelerations are calculated by finite differences.

A user may not wish to experience orientation errors due to constraint violations when rigidly grasping a four-bar linkage, for example. We allow the user to specify a point grasp in order to perceive just the positional displacements. Only the positional components are then included in $\mathbf{e}_1$, and no constraint forces are returned for orientation.

5 Dynamic Interaction with Open Chains

Consider a virtual open chain grasped by an operator who holds a haptic interface. A simple example is a two-link planar mechanism in Figure 3. There is no direct connection between the virtual open chain and the haptic interface; the operator represents this connection. The haptic interface reflects interaction wrench of the virtual open chain to the operator. Three steps are followed to solve for the operator wrench:

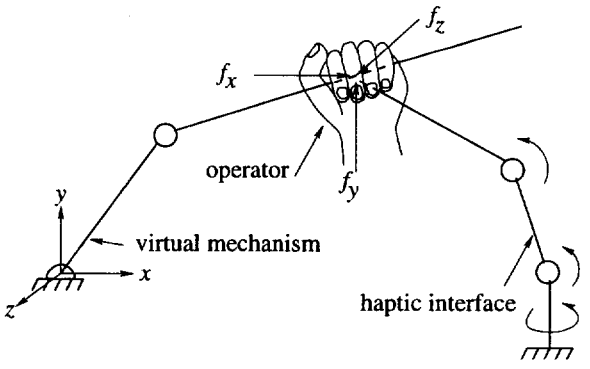

Figure 3: A typical system of virtual open chain, operator, and haptic interface.

- Step 1. Solve for the dynamic wrench $\mathbf{w}_d$.

  The equation of motion of the virtual chain is:

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{J}^T \mathbf{w}_d \qquad (9)$$

  where $\boldsymbol{\tau}$ is an $m \times 1$ vector of the joint torques, $\mathbf{M}$ is an $m \times m$ inertia matrix, $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ are $m \times 1$ vectors of joint angle, velocity, and acceleration, $\mathbf{h}$ is an $m \times 1$ vector representing centrifugal, Coriolis, and gravity forces, and $\mathbf{J}$ is a $6 \times m$ Jacobian of the open chain assuming the end point is the grasping point. Assuming that friction is zero, $\boldsymbol{\tau}$ is zero for all joints since they are not actuated. Thus, $\mathbf{w}_d$ is obtained using the pseudo inverse of $\mathbf{J}^T$:

$$\mathbf{w}_d = \mathbf{J}(\mathbf{J}^T\mathbf{J})^{-1}[\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})] \qquad (10)$$

  Because of its minimum norm properties, the pseudo inverse automatically isolates the dynamic wrench. The open chain dynamics can be found efficiently from the recursive Newton-Euler method.

- Step 2. Solve for the constraint wrench $\mathbf{w}_c$.

  This wrench is a function of the error vector $\mathbf{e}_1$ found in (1) and can be computed by any contact model. The simplest model is to make it proportional to $\mathbf{e}_1$:

$$\mathbf{w}_c = \mathbf{G}\,\mathbf{e}_1 \qquad (11)$$

  where $\mathbf{G}$ is a diagonal stiffness matrix selected based on the strength of the haptic interface. Other contact models which consider both penetration and the rate of penetration can also be used [10].

- Step 3. Reflect wrenches to the haptic interface.

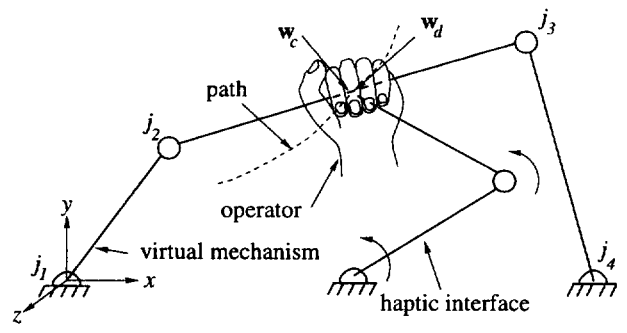

Figure 4: A typical system of virtual closed chain, operator, and haptic interface.

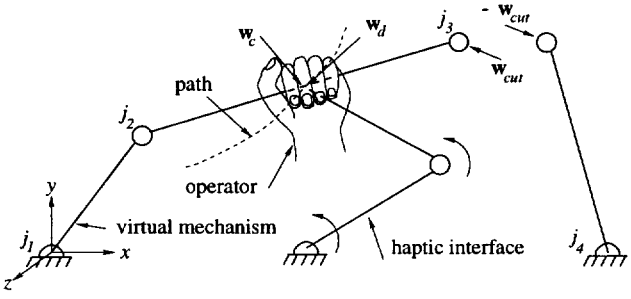

Figure 5: Breaking a joint of a virtual closed chain.

The dynamic and constraint wrenches are reflected to the operator via the haptic interface torque vector:

$$\boldsymbol{\tau}_{h.i.} = \mathbf{J}_{h.i.}^T(\mathbf{w}_d + \mathbf{w}_c) \qquad (12)$$

where $\mathbf{J}_{h.i.}$ is the Jacobian of the haptic interface.

6 Dynamic Interaction With Single Closed Chains

The procedure for single closed chains is similar to that for open chains, but we cut a joint and solve two open branches simultaneously. Figure 4 shows an example where the virtual mechanism is a planar 4-bar linkage and the haptic interface is a planar 2-link manipulator. Step 1 of the previous procedure is replaced by the following, while steps 2 and 3 are the same.

- Step 1. Cut a joint and solve for $\mathbf{w}_d$.

  A joint is cut and two open chains are formed instead; the place to cut follows the procedure of [15]. The cut joint is then replaced by a wrench $\mathbf{w}_{cut}$ to preserve the motion. For example, in Figure 4, joint j_3 is cut and replaced $\mathbf{w}_{cut}$. This allows the left and right open chains in Figure 5 to

have the same motion as the uncut closed chain in Figure 4. Suppose the operator is grasping the left chain. The equations of motion are written for both left and right open chains:

$$\begin{aligned}\boldsymbol{\tau}_l &= \mathbf{M}_l(\mathbf{q}_l)\ddot{\mathbf{q}}_l + \mathbf{h}_l(\mathbf{q}_l, \dot{\mathbf{q}}_l) - \mathbf{J}_{l1}^T\mathbf{w}_d - \mathbf{J}_{l2}^T\mathbf{w}_{cut} \\ \boldsymbol{\tau}_r &= \mathbf{M}_r(\mathbf{q}_r)\ddot{\mathbf{q}}_r + \mathbf{h}_r(\mathbf{q}_r, \dot{\mathbf{q}}_r) + \mathbf{J}_r^T\mathbf{w}_{cut}\end{aligned} \tag{13}$$

where the symbols are almost similar to those we used for open chains. Subscripts l and r denote left and right chains, $\mathbf{J}_{l1}$ is Jacobian of the left chain assuming the end point is at the grasping point, $\mathbf{J}_{l2}$ is Jacobian of the left chain assuming the end point is at the cut joint, and $\mathbf{J}_r$ is Jacobian of the right chain assuming the end point is at the cut joint. In (13), $\boldsymbol{\tau}_l$ and $\boldsymbol{\tau}_r$ are zero since they are not actuated and friction is assumed to be zero. Rewriting (13):

$$\begin{bmatrix} \mathbf{J}_{l1}^T & \mathbf{J}_{l2}^T \\ \mathbf{0} & -\mathbf{J}_r^T \end{bmatrix} \begin{bmatrix} \mathbf{w}_d \\ \mathbf{w}_{cut} \end{bmatrix} = \begin{bmatrix} \mathbf{M}_l\ddot{\mathbf{q}}_l + \mathbf{h}_l \\ \mathbf{M}_r\ddot{\mathbf{q}}_r + \mathbf{h}_r \end{bmatrix} \tag{14}$$

which is solved for $[\mathbf{w}_d\ \mathbf{w}_{cut}]^T$ by the pseudo inverse method similar to (10).

7 Experimental Results

We have successfully implemented the open and closed chain computations into our real-time framework. Details of the master controller, including gravity compsensation, and of the CAD environment implementation are presented in [13, 14].

For open chains, the process of inverse kinematics, Jacobian pseudo-inverse, and recursive inverse dynamics runs in excess of 10kHz for the 1 DOF case of the open loop crank, shown in Figure 6. The assumption is made that the operator grasps the end of the virtual linkage, indicated by a disk. A second disk shows the location of the haptic interface for this example.

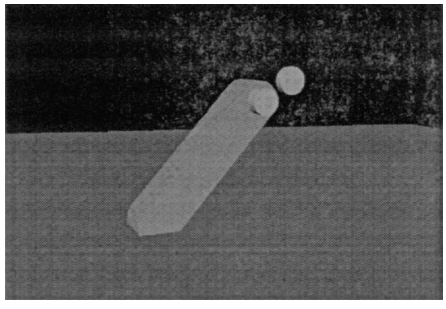

Figure 6: Example crank mechanism.

The crank example touches all stages of our computation: inverse kinematics, Jacobian formation, virtual mechanism torque to operator grasp force, and projection of the operator force to the Master Arm. This procedure has been generalized to arbitrary mechanisms that are fully constrained, following the outline of the previous section. The open chain examples which are in the fully constrained class for prescribed position include all planar or spatial two link, three joint (or less complex) mechanisms with prismatic or revolute joints.

For closed chains, the real-time constraints are significantly more costly for the slider-crank shown in Figure 7 than for the open loop crank, but still runs in excess of kHz rates. The class of mechanisms in the prescribed position case for closed loops is more vague. Most planar or spatial three-bar, four-joint mechanisms and some spatial four-bar, five-joint mechanisms can be prescribed. We have successfully kept the closed algorithm general to this point.

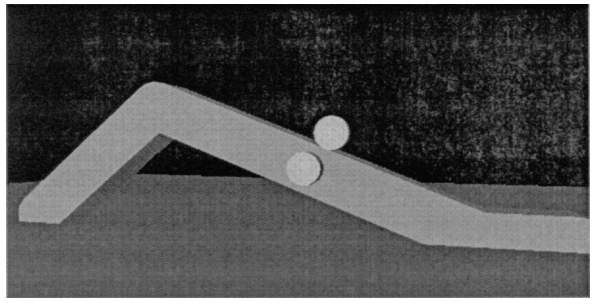

Figure 7: Example slider-crank mechanism.

8 Discussion

This paper has presented an approach towards haptic manipulation of virtual mechanisms derived from CAD designs, such as arise in the design of animatronic figures. The purpose is to aid the design process by providing haptic feedback for appreciating mobility, workspace, and driving forces of a hypothetical mechanism. A stiffness model of haptic interaction and an inverse dynamics approach have been developed for this purpose.

A numerical optimization procedure first partitions a haptic motion into mechanism motion and constraint motion. In the constrained directions we apply typical viscoelastic surface models to return constraint forces. In the motion directions allowed by a mechanism, we equate motion of the haptic interface with motion of the mechanism. The rationale is that real surface stiffnesses are much higher than the stiffness achievable with a haptic interface [2], and it is not necessary to partition the haptic interface motion partly into surface compression in a direction where a mech-

anism can move. We perform an inverse dynamics computation to return dynamic forces for single open-loop and single closed-loop chains.

This work has had the specific aim of adding value to the design process of an important class of practical mechanisms. It was not the intent at this point to provide a general simulation environment for virtual reality, or to model assembly forces. We took advantage of the features of this domain that the mechanism kinematics are fully prescribed by haptic interface motion, which permits an inverse dynamics approach towards simulation. Otherwise, a forward dynamics computation is required in general. We are able to handle single open-loop and many single closed-loop chains; the generalization to multiple-loop chains is under development. Future studies will also evaluate the utility of haptic feedback to a designer.

Acknowledgments

Support for this research was provided by NSF Grant MIP-9420352, by DARPA grant F33615-96-C-5621, and by NSERC NCE IRIS Project HMI-8.

References

[1] An, C.H., Atkeson, C.G., and Hollerbach, J.M., *Model-Based Control of a Robot Manipulator.* Cambridge, MA: MIT Press, 1988.

[2] Colgate, J.E., and Brown, J.M., "Factors affecting the z-width of a haptic display," *Proc. IEEE Intl. Conf. Robotics and Automation*, pp. 3205-3210, 1994.

[3] Gillespie, R.B., and Colgate, J.E., "A survey of multibody dynamics for virtual environments," in *Proc. ASME Dynamic Systems and Control Division*, Dallas, pp. 45-54, Nov. 15-21, 1997.

[4] Gupta, R., Sheridan, T., and Whitney, D., "Experiments using multimodal virtual environments in design for assembly analysis," *Presence*, vol. 6, pp. 318-338, 1997.

[5] Hollerbach, J.M., Cohen, E.C., Thompson, W.B., Freier, R., Johnson, D.E., Nahvi, A., Nelson, D.D., and Thompson II, T.V., "Haptic interfacing for virtual prototyping of mechanical CAD designs," in *ASME Design for Manufacturing Symposium*, Sacramento, Sept. 14-17, 1997.

[6] Johnson, D.E., and Cohen, E., "A framework for efficient minimum distance computations," in *IEEE Intl. Conf. Robotics & Automation*, Leuven, May 16-21, 1998.

[7] Lilly, K.W., and Orin, D.E., "Efficient dynamic simulation of multiple chain robotic mechanisms," *J. Dynamic Systems, Meas., Control*, vol. 116, pp. 223-231, 1994.

[8] Lin, S.-K., "Dynamics of the manipulator with closed chains," *IEEE Trans. Automation and Robotics*, vol. 6, pp. 496-501, 1990.

[9] Maekawa, H., and Hollerbach, J.M., "Haptic display for object grasping and manipulation in a virtual environment," in *IEEE Intl. Conf. Robotics & Automation*, Leuven, May 16-21, 1998.

[10] Marhefka, D.W., and Orin, D.E., "Simulation of contact using a nonlinear damping model," in *IEEE Intl. Conf. Robotics and Automation*, Minneapolis, pp. 1662-1668, April 21-28, 1996.

[11] Mooring, B.W., Roth, Z.S., and Driels, M.R., *Fundamentals of Manipulator Calibration.* NY: Wiley Interscience, 1991.

[12] Stewart, P., Buttolo, P., and Chen, Y., "CAD data representations for haptic virtual prototyping," in *ASME Design Engineering Technical Conferences*, Sacramento, Sept. 14-17, 1997.

[13] Thompson II, T.V., Johnson, D.E., Cohen, E., "Direct haptic rendering of sculptured models," in *Proc. Symposium on Interactive 3D Graphics*, Providence, pp. 167-176, April 27-30, 1997.

[14] Thompson, T.V. II, Nelson, D.D., Cohen, E., and Hollerbach, J., "Maneuverable NURBS models within a haptic virtual environment," in *Proc. ASME Dynamic Systems and Control Division*, Dallas, pp. 37-44, Nov. 15-21, 1997.

[15] Tsai, F.-F., and Haug, E.J., "Real-time multibody system dynamic simulation: Part I. A modified recursive formulation and topological analysis," *Mech. Struct. & Mach.*, vol. 19, pp. 99-127, 1991.

[16] Yoshikawa, T., and Ueda, H., "Haptic virtual reality: display of operating feel of dynamic virtual objects," *Robotics Research: The Seventh International Symposium*, edited by G. Giralt and G Hirzinger. London: Springer, pp. 214-221, 1996.

[17] Yoshikawa, T., Yokokohji, Y., Matsumoto, T., and Zheng, X.-Z., "Display of feel for the manipulation of dynamic virtual objects," *J. Dynamic Systems, Meas., Control*, vol. 117, pp. 554-558, 1995.