

Haptic Manipulation of Serial-Chain Virtual Mechanisms

Daniela Constantinescu*

Septimiu E. Salcudean

Elizabeth A. Croft

Email: danielac@me.uvic.ca

Email: tims@ece.ubc.ca

Email: ecroft@mech.ubc.ca

Mechanical Engineering

Electrical and Computer Engineering

Mechanical Engineering

University of Victoria

University of British Columbia

University of British Columbia

Victoria, BC V8W 3P6 Canada

Vancouver, BC V6T 1Z4 Canada

Vancouver, BC V6T 1Z4 Canada

Abstract

This paper presents an approach for providing realistic force feedback to users manipulating serial-chain virtual mechanisms. In the proposed approach, a haptic device controller is designed that penalizes users' motion along the directions resisted by the virtual joints. The resisted directions span the null space the Jacobian of the virtual mechanism computed at the users' hand, and are derived via a Singular Value Decomposition-based algorithm. Haptic numerical performance is achieved by computing the resisted directions on the graphics processor, and by using them on the haptics processor to derive the control signal that restricts users' motion as required by the virtual joints. The performance of the proposed approach is validated through experimental manipulations of links with unrestricted and with restricted motion within a planar virtual world.

I. INTRODUCTION

Realistic haptic manipulation of virtual mechanisms (VMs), from any link and through singularities, is potentially beneficial in virtual prototyping [1], medical simulators, education, motion editing for animation [2], and computer games [3]. For example, operation of links with insufficient degrees of freedom (DOFs) is important in training for laparoscopy, where the scope limits the motion of the tool at the point of entry. Despite such promising applications, little research has addressed the haptic rendering of user interaction with a virtual environment through a VM [1], [4], [5], [3], [2], [6], [7]. This is partly because the dynamics of VMs are challenging to compute at guaranteed speeds of the order of hundreds of Hz, as required for haptic manipulation of virtual environments.

*Corresponding author.

To overcome the computational challenges, existing haptics research has focused primarily on the simulation of VMs. For example, efficient mechanism dynamics are the major concern in [1], [4], [2]. Accurate contact and collision models are proposed in [5], [7], [8]. Redundancy resolution and efficient dynamics of mechanisms with multiple branches are addressed in [6]. Generality and interactive performance are the key objectives of the object-oriented simulator proposed in [7]. Much of this haptics work builds on earlier developments in robot modeling [9], [10], [11], [12], and control [13], [14], in computational dynamics [15], and in graphics [16]. Its main objective is to increase the realism of the haptic manipulation via increasing the physical accuracy of interactive mechanism simulations.

In addition to depending on the physical accuracy of the virtual environment, realistic haptic manipulation of VMs hinges on the technique used to apply the simulated forces to users. Two haptic rendering methods have been developed in prior research. A virtual proxy¹ [17] has been employed to apply forces to users during point manipulations of VMs through pushing² [2], [7]. A virtual coupler³ [18], [19] has been used to apply wrenches (forces and torques) to users during rigid body manipulations [7]. Both the virtual proxy and the virtual coupler have been employed to compute penalty-like wrenches at the high speed of the force control loop while the virtual environment has been generated at lower speeds [2], [7]. Due to the lower speed at which the virtual environment has been simulated, only limited stiffness has been rendered to users [7]. The advantage of virtual coupler control is that the coupling parameters can be chosen such that the interaction is guaranteed stable. However, the virtual coupler low-pass filters the virtual interactions and does not allow users to feel physical phenomena represented through forces with frequency components above its cutoff frequency.

This paper proposes a new approach for haptic rendering of rigid body manipulations of open-loop VMs. The proposed approach is the first that enables users to operate open-loop VMs from any link and through singularities without low-pass filtering the wrenches rendering the interactions between the VM and other virtual objects. In this approach, a haptic device controller is designed as an integral part of a previously developed control architecture [20]. The new controller enforces the VM joint constraints, and the existing architecture applies to users the VM inertia and the contact interactions between the VM and the virtual environment. The proposed controller acts as generalized springs and dampers that oppose users' motion along the directions resisted by the virtual joints. These directions are selected using the orthogonal complement of the VM Jacobian, computed at the user's hand. By augmenting the existing architecture with the new controller, the present approach allows manipulations of serial-chain VMs from any user-selected link. By proposing a controller that acts orthogonal to the directions

¹The virtual proxy is a simulated small sphere connected to the user's finger through a simulated spring and a damper when constrained by other objects in the virtual environment and collocated with the user's finger otherwise.

²The virtual proxy can be used only for pushing because it behaves similar to a contact constraint by definition.

³The virtual coupler is a controller that acts as a generalized (translational and rotational) spring and damper connection between the user and the end effector of the VM.

along which users feel the VM interactions with the virtual environment, the approach does not low-pass filter the virtual contact forces. Hence, it enables users to feel physical phenomena represented through contact forces with high frequency components, such as collisions [8]. Haptic numerical performance is achieved by distributing the computations between the simulation and the haptics processors. In particular, the geometry of the joint and of the contact constraints is computed in the simulation, at lower speed (of the order of tens of Hz), and is sent to the haptics processor. On the haptics processor, the proposed controller and a local model of interaction [21] use this geometry to derive the wrenches applied to users. As illustrated experimentally in Section VIII, stiffer joint constraints can be imposed on users via this implementation of the proposed approach than via connecting users to the VM through virtual coupling.

The haptic rendering of joint constraints presented in this paper is related to work in [22], [23], and [24]. In that work, the tracing of a virtual surface has been implemented via the haptic manipulation of a VM⁴. The VM was designed such that it had no singularity in the workspace of interest and only allowed motion in the plane tangent to the desired surface (the configuration space of the VM coincided with the virtual surface). Furthermore, virtual springs and dampers were used to connect the haptic device to the virtual end effector. Similar to [22], [23], and [24], the approach proposed herein also penalizes users' motion along the directions resisted by the virtual joints. However, unlike those works, the present approach is designed to enable realistic and unrestricted manipulations of serial-chain VMs within dynamic multibody virtual environments rather than the tracing of virtual surfaces. Therefore, it forces users to trace the configuration manifold of the VM that users operate only when users exceed this manifold. When users move inside the workspace of the VM, the approach enables them to feel the VM inertia, as well as the VM interactions with other virtual objects.

In this paper, the control architecture implementing the proposed approach is presented in Section II. The method used to generate the VM is overviewed in Section III. The haptic rendering of the VM inertia and of the VM joint constraints is presented in Sections IV and V. Simulated haptic manipulations of a planar three-links VM from various links is discussed in Section VI. The efficient implementation of the approach is detailed in Section VII. Haptic manipulations of a planar three-links VM from various links are illustrated in Section VIII. Conclusions and directions for future work are discussed in Section IX.

II. CONTROL ARCHITECTURE

The control architecture proposed in [20] is used for haptic rendering of user interaction within rigid multibody virtual environments in this work. The architecture comprises an impedance [25] device controller and a four channel teleoperation [26] haptic controller. The device controller matches the impedance of the haptic device to the impedance of the VM held by the

⁴Haptic manipulation of a crank has been used for tracing a virtual circle in [22] and haptic manipulation of a two-link Cartesian mechanism has been used for tracing NURBS surfaces in [23] and [24].

user. The haptic controller regards the haptic device as the master and the VM as the slave [20], and applies the VM contact (environment) forces to the user’s hand. A simplified mechanical representation of one dimensional haptic interaction via the four channel haptic control architecture (excluding the communication delay and the force scaling factor) is depicted in Fig. 1. In this figure, it is considered that the dynamics of the haptic device (master) and of the VM (slave) are dominated by their respective inertias, while the dynamics of the environment are predominantly compliant. Therefore, the impedances of the haptic device Z_{HD} and of the VM Z_{VM} are represented via their mass m_{HD} and m_{VM} , respectively. The environment impedance Z_{env} is represented via the spring-damper connection that opposes the VM motion with an environment force f_{env} . The user applies a force f_h to the device, and the four channel teleoperation controller connects the haptic device to the operational point of the VM. Via the two force channels, the controller feedforwards the user-applied force to the VM and feeds back the environment force to the user’s hand. Via the two position channels, it acts as a spring-damper connection between the device and the VM operational point and applies a coordination force f_{pc} to both. High coordination gains are undesirable because they give a “sluggish” feeling in free motion [26]. Therefore, the four channel architecture uses low coordination gains and employs the coordination force only to maintain the kinematic correspondence between the device and the operational point on the VM.

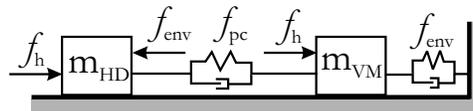


Fig. 1. Mechanical representation (excluding the communication delay and force scaling) of one dimensional haptic interaction within a virtual environment via the four channel teleoperation controller.

Note that the impedance controller is represented in Fig. 1 only through its effect on the impedance (mass) of the haptic device Z_{HD} (m_{HD}). If the impedance controller perfectly matches the impedance of the haptic device to the VM impedance at the user-selected operational point, $Z_{HD} = Z_{VM}$ ($m_{HD} = m_{VM}$), then the impedance transmitted to the users’ hand is $Z_t = Z_{VM} + Z_{env}$. Users feel as if directly manipulating the VM within the virtual environment, and the haptic controller is transparent [26]. In this existing control architecture, users feel the motion constraints due to other virtual objects through Z_{env} , represented by f_{env} in Fig. 1. They feel the VM inertia and the motion constraints due to the virtual joints through the impedance controller which matches Z_{HD} to Z_{VM} . As discussed in Section IV, the impedance controller commands the device acceleration to follow the acceleration of the VM operational point, i.e., it commands zero device acceleration along the directions of joint constraint. Numerical drift and finite device stiffness prevent zero acceleration control from effectively limiting users’ motion. Therefore, a new controller is proposed in Section V that penalizes users’ motion along the directions restricted by the virtual joints.

Before addressing the shortcoming of impedance control for haptic rendering of joint constraints, the computation of the VM dynamics in the virtual environment is presented in the following section.

III. VIRTUAL MECHANISM MODELING

The VM dynamics can be computed in task space or in configuration space. In task space, the dynamics of each link are derived independently and joint constraints are imposed via algebraic constraint equations. Hence, a computationally expensive differential algebraic system of equations must be integrated for which constraint satisfaction may be problematic [17]. In configuration space, the joint constraints (i.e., the VM topology) are embedded in the coordinate representation. Only a reduced set of coordinates must be integrated and joint (topological) constraint satisfaction is guaranteed. Therefore, VMs are simulated in configuration space in this work. Note that the VM configuration space and task space representations coincide when users choose to operate a single object, i.e., a VM with no topology.

Since the coordinate representation embeds the joint constraints, only the contact and the user-applied forces must be included in the dynamic equations. Hence, the configuration space dynamics of a VM with d DOFs and c contacts are given by:

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \sum_{i=1}^c \mathbf{J}_i^T(\mathbf{q}) \mathbf{F}_i + \mathbf{J}_h^T(\mathbf{q}) \mathbf{F}_h. \quad (1)$$

In (1), notation is used as follows: $\mathbf{D}(\mathbf{q})_{d \times d}$ is the VM configuration space inertia; $\mathbf{B}(\mathbf{q}, \dot{\mathbf{q}})_{d \times 1}$ represent Coriolis and centripetal effects; $\mathbf{G}(\mathbf{q})_{d \times 1}$ are the gravitational terms; $\mathbf{J}_i(\mathbf{q})_{6 \times d}$ is the VM Jacobian computed at the i -th contact; $\mathbf{F}_{i_{6 \times 1}} = \begin{pmatrix} \mathbf{f}_i^T & \mathbf{0}^T \end{pmatrix}^T$ is the contact wrench (i.e., the force $\mathbf{f}_{i_{3 \times 1}}$ and torque $\boldsymbol{\tau}_{i_{3 \times 1}} = \mathbf{0}_{3 \times 1}$) at the i -th contact; $\mathbf{J}_h(\mathbf{q})_{6 \times d}$ is the VM Jacobian computed at the user's hand (hand Jacobian); $\mathbf{F}_{h_{6 \times 1}}$ is the user-applied wrench; and $\mathbf{q}_{d \times 1}$, $\dot{\mathbf{q}}_{d \times 1}$, and $\ddot{\mathbf{q}}_{d \times 1}$ are the VM configuration space position, velocity, and acceleration, respectively. Equation (1) describes the dynamics of a passive VM. Haptic manipulation of powered (i.e., active) VMs can be allowed by augmenting (1) with the desired actuator torques.

In (1), the contact wrenches \mathbf{F}_i model the interactions between the VM and other objects in the virtual environment. They are computed as described in [8]. Once computed, (1) can be directly solved for the VM configuration space acceleration:

$$\ddot{\mathbf{q}} = \mathbf{D}^{-1} \left(\sum_{i=1}^c \mathbf{J}_i^T \mathbf{F}_i + \mathbf{J}_h^T \mathbf{F}_h - \mathbf{B} - \mathbf{G} \right). \quad (2)$$

In (2), the instantaneous state dependence of all computed terms on the right hand side of the equation is implied. The VM state is advanced by integrating its configuration space acceleration using a fixed step integrator with time step equal to the time step of the control loop.

While the VM dynamics are computed in configuration space, users operate the VM in task space. Therefore, the VM inertia,

joint constraints, and contact forces must be applied to the user's hand in task space. The representation of the VM inertia to the user is discussed in the following section, while the approach proposed for representing the joint constraints to the haptic device is presented in Section V.

IV. HAPTIC RENDERING OF MECHANISM INERTIA AT THE USER-SELECTED POINT

Both the inertia and the joint constraints are embedded in the operational space inertia of the VM computed at the user's hand, $\Lambda_{h_{6 \times 6}}$ [13]:

$$\Lambda_h = (\mathbf{J}_h \mathbf{D}^{-1} \mathbf{J}_h^T)^{-1}. \quad (3)$$

Therefore, Λ_h is incorporated in this work in the desired VM impedance at the user-selected operational point, Z_{VM} . The impedance device controller changes the dynamics of the haptic device to match the desired VM dynamics [20]:

$$\Lambda_h \ddot{\mathbf{x}}_h + \mathbf{b}_d \dot{\mathbf{x}}_h + \mathbf{k}_d \mathbf{x}_h = \mathbf{F}_h + \mathbf{F}_{env} + \mathbf{F}_{pc}. \quad (4)$$

In (4), Λ_h , $\mathbf{b}_{d_{6 \times 6}}$, and $\mathbf{k}_{d_{6 \times 6}}$ are the desired inertia, damping, and stiffness of the VM, respectively; $\mathbf{F}_{env_{6 \times 1}}$ is the environment wrench at the user's hand (the user-perceived force and torque due to the contact, gravitational, Coriolis, and centripetal forces acting on the VM); $\mathbf{F}_{pc_{6 \times 1}}$ is the coordination wrench between the device and the VM operational point (due to the generalized spring and damper shown in Fig. 1); and $\ddot{\mathbf{x}}_{h_{6 \times 1}}$, $\dot{\mathbf{x}}_{h_{6 \times 1}}$, and $\mathbf{x}_{h_{6 \times 1}}$ are the desired body acceleration, velocity, and position of the device at the user-selected operational point, respectively.

Equation (4) is equivalent to:

$$\ddot{\mathbf{x}}_h = \Lambda_h^{-1} (\mathbf{F}_h + \mathbf{F}_{env} + \mathbf{F}_{pc} - \mathbf{b}_d \dot{\mathbf{x}}_h - \mathbf{k}_d \mathbf{x}_h). \quad (5)$$

This form is advantageous because the inverse of the VM operational space inertia at the user's hand can be computed regardless of the rank of the hand Jacobian:

$$\Lambda_h^{-1} = \mathbf{J}_h \mathbf{D}^{-1} \mathbf{J}_h^T. \quad (6)$$

In contrast, Λ_h can be computed only when \mathbf{J}_h is full row rank. When the hand Jacobian is not full row rank, Λ_h becomes infinite along certain task space directions and Λ_h^{-1} drops rank.

The rank deficiency of Λ_h^{-1} indicates that the VM joints restrict the VM instantaneous body motion at the user-selected operational point. This may happen when users hold a link with fewer than 6DOFs during spatial interaction and fewer than 3DOFs during planar interaction or when they operate the VM through a singularity. To illustrate how the VM joint constraints are represented in the rank deficiency of Λ_h^{-1} , consider the example manipulations of the planar VM depicted in Fig. 2. In

the first example, the user holds the planar VM from the centre of mass (COM) of the middle link in the position shown in Fig. 2(a). The held link has insufficient DOFs to allow arbitrary VM position and orientation at the user-selected operational point. The virtual joints instantaneously constrain the user's planar motion according to:

$$\dot{\mathbf{x}}_h = \mathbf{J}_h \dot{\mathbf{q}}, \quad (7)$$

where \mathbf{J}_h is:

$$\mathbf{J}_h = \begin{bmatrix} -l_1 s_1 - l_{c_2} s_{12} & -l_{c_2} s_{12} & 0 \\ l_1 c_1 + l_{c_2} c_{12} & l_{c_2} c_{12} & 0 \\ 1 & 1 & 0 \end{bmatrix}. \quad (8)$$

In (8), links are numbered from the base outward, l_i is the length of link i , l_{c_i} is the distance from the joint with the previous link to the COM of link i , s_{ij} denotes $\sin(q_i + q_j)$, and c_{ij} denotes $\cos(q_i + q_j)$. Equation (8) shows that $\text{rank}(\mathbf{J}_h) = 2$. From (6), it follows that $\text{rank}(\mathbf{\Lambda}_h^{-1}) = 2$. In the second example, the user holds the planar VM from the COM of the distal link in the position shown in Fig. 2(b). In the example position, the hand Jacobian:

$$\mathbf{J}_h = \begin{bmatrix} -(l_1 + l_2) s_1 - l_{c_3} s_{13} & -l_2 s_1 - l_{c_3} s_{13} & -l_{c_3} s_{13} \\ (l_1 + l_2) c_1 + l_{c_3} c_{13} & l_2 c_1 + l_{c_3} c_{13} & l_{c_3} c_{13} \\ 1 & 1 & 1 \end{bmatrix}, \quad (9)$$

is singular, $\text{rank}(\mathbf{J}_h) = 2$, and $\text{rank}(\mathbf{\Lambda}_h^{-1}) = 2$. These examples illustrate that $\mathbf{\Lambda}_h^{-1}$ drops rank when the virtual joints instantaneously restrict the VM body motion at the user-selected operational point.

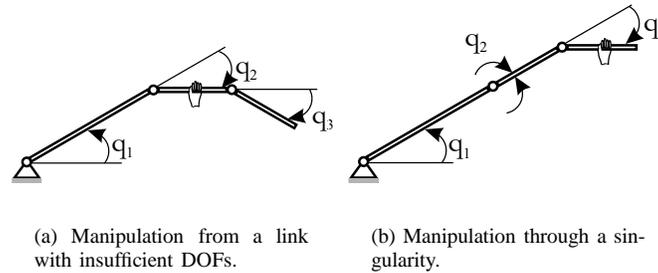


Fig. 2. Example manipulations of a planar VM where the virtual joints restrict the instantaneous body motion at the user-selected operational point.

Since $\mathbf{\Lambda}_h^{-1}$ can be computed regardless of the rank of \mathbf{J}_h , (5) is used in this work in the impedance control of the device. Equation (5) computes the desired body acceleration at the user's hand. Given the device dynamics:

$$\mathbf{M}_d \ddot{\mathbf{x}}_h + \mathbf{C}_d \dot{\mathbf{x}}_h = \mathbf{F}_h + \mathbf{u}, \quad (10)$$

where \mathbf{M}_d and \mathbf{C}_d are the inertia and the Christoffel matrices of the haptic device, and \mathbf{u} is the control signal [20], the impedance control law for the interface is obtained as:

$$\begin{aligned}
\mathbf{u} &= (\mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} - \mathbf{I}) \mathbf{F}_h + \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} (\mathbf{F}_{env} + \mathbf{F}_{pc}) \\
&+ (\mathbf{C}_d - \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{b}_d) \dot{\mathbf{x}}_h - \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{k}_d \mathbf{x}_h = \\
&= (\mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} - \mathbf{I}) \mathbf{F}_h + \mathbf{M}_d \mathbf{J}_h \mathbf{D}^{-1} \tau + \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{F}_{pc} \\
&+ (\mathbf{C}_d - \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{b}_d) \dot{\mathbf{x}}_h - \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{k}_d \mathbf{x}_h.
\end{aligned} \tag{11}$$

In (11):

$$\tau = \sum_{i=1}^c \mathbf{J}_i^T \mathbf{F}_i - \mathbf{B} - \mathbf{G} \tag{12}$$

is the configuration space torque due to the VM contact, gravitational, Coriolis, and centripetal forces, and⁵:

$$\boldsymbol{\Lambda}_h^{-1} \mathbf{F}_{env} = \mathbf{J}_h \mathbf{D}^{-1} \tau. \tag{13}$$

The control signal in (11) applies the VM contact, gravitational, Coriolis, centripetal, and joint constraint forces to the haptic device by commanding the device acceleration to follow the acceleration of the VM at the user-selected operational point. Since the VM acceleration along the directions of joint constraint is zero, (11) controls to zero the device acceleration along these directions. However, numerical drift and limited device stiffness prevent zero acceleration control from effectively restricting users' motion along the directions resisted by the virtual joints. A technique that overcomes this limitation of impedance control and enables users to realistically feel the virtual joint constraints is proposed in the following section.

V. HAPTIC RENDERING OF JOINT CONSTRAINTS AT THE USER-SELECTED POINT

An analysis of the VM dynamics suggests how the control in (11) can be augmented to enable users to perceive motion constraints imposed by the VM joints. At the user-selected point, $\boldsymbol{\Lambda}_h^{-1}$ maps all wrenches acting on the VM to the VM body acceleration at the user's hand as given in (5). In (5), $\ddot{\mathbf{x}}_h = (\dot{\mathbf{v}}_h^T \quad \dot{\boldsymbol{\omega}}^T)^T$ is an element of the space of body accelerations at the user's hand, A ; i.e., $\dot{\mathbf{v}}_{h_{3 \times 3}}$ and $\dot{\boldsymbol{\omega}}_{3 \times 3}$ are the linear and the angular acceleration of the user's hand, respectively. When $\boldsymbol{\Lambda}_h^{-1}$ is full rank, the wrenches acting on the VM, including the user-applied wrench \mathbf{F}_h , produce body acceleration of the user's hand along all directions of A . The VM behaves as an inertia along all directions of A . On the control side, (11) applies this inertia to users by controlling their acceleration to the value given in (5).

⁵after premultiplication of $\tau = \mathbf{J}_h^T \mathbf{F}_{env}$ by $\mathbf{J}_h \mathbf{D}^{-1}$ and substitution from (6).

When Λ_h^{-1} is rank-deficient, $\ddot{\mathbf{x}}_h$ lies in a subspace of A . Wrenches (in particular, user-applied wrenches) exist that lie in the null space of Λ_h^{-1} , $\mathcal{N}(\Lambda_h^{-1})$. These wrenches are opposed by the structural stiffness of the VM. They have no effect on the VM body acceleration at the user's hand. At the user-selected point, the VM behaves as an infinite stiffness along the directions in $\mathcal{N}(\Lambda_h^{-1})$ and behaves as an inertia along the directions orthogonal to them. Hence, the directions $\mathcal{N}(\Lambda_h^{-1})$ are the directions resisted by the virtual joints, while the directions in the range space of Λ_h^{-1} $\mathcal{R}(\Lambda_h^{-1})$ are the directions of motion allowed by the virtual joints. On the control side, the control in (11) enforces the VM body acceleration both along the directions of constraint and along the directions of motion. Along the constraint directions, the control in (11) enforces zero device acceleration. Due to numerical drift and finite device stiffness, this control cannot restrict users' motion as required by the virtual joints. In other words, acceleration control is unsuitable for rendering the infinite structural stiffness of the VM along the directions resisted by the virtual joints.

In the proposed implementation, the directions in $\mathcal{N}(\Lambda_h^{-1})$ are computed via a singular value decomposition (SVD) based algorithm. In particular, the SVD of \mathbf{J}_h^T (rather than the SVD of Λ_h^{-1}) is used to derive the directions resisted by the virtual joints. This is because the null spaces of Λ_h^{-1} and \mathbf{J}_h^T are equal and the two mappings have the same singular directions:

$$\mathcal{N}(\Lambda_h^{-1}) = \mathcal{N}(\mathbf{J}_h^T), \quad (14)$$

as shown in Appendix IX. Physically, the proof can be understood by considering the action of \mathbf{J}_h^T . Specifically, \mathbf{J}_h^T maps the space of wrenches F to the space of configuration torques T :

$$\tau = \mathbf{J}_h^T \mathbf{F}, \quad (15)$$

i.e., to the space of torques that result in body acceleration at the user's hand. In (15), \mathbf{F} is an element of F . In other words, in the VM configuration space dynamics, \mathbf{J}_h^T filters out the wrench components that have no effect on the body acceleration of the user's hand. Hence, the null spaces of Λ_h^{-1} and of \mathbf{J}_h^T coincide and the SVD of \mathbf{J}_h can be used to compute the directions resisted by the virtual joints.

When users hold a VM link with redundant DOFs (the hand Jacobian is full row rank):

$$\mathbf{I}_{6 \times 6} - \bar{\mathbf{J}}_h \mathbf{J}_h, \quad (16)$$

can be used to apply forces to users along the $\mathcal{N}(\Lambda_h^{-1})$ directions. In (16), $\bar{\mathbf{J}}_h = \mathbf{D}^{-1} \mathbf{J}_h^T \Lambda_h$ is the dynamically consistent inverse of the hand Jacobian [14]. When users hold a VM link with insufficient DOFs or move through a singularity, the hand Jacobian is row rank deficient and (16) cannot be used to compute forces along the directions resisted by the virtual joints. To

enable users to manipulate VMs from any link and through singularities, the restricted directions are derived using the SVD of \mathbf{J}_h^T in this work.

Realistic perception of the joint constraints is enabled via augmenting the control provided by (11) with penalties applied along the directions resisted by the virtual joints, as shown in Fig. 3. In this figure, \mathbf{x}_{VM} is the position and orientation of the user-selected operational point on the VM (obtained via forward kinematics from the VM configuration space position \mathbf{q}), and \mathbf{x}_h is the position and orientation of the user's hand on the device. For simplicity, only a one dimensional $\mathcal{N}(\mathbf{J}_h^T)$ is depicted. Note that the penalties applied to users along the directions resisted by the virtual joints serve a similar purpose to the penalties applied in stabilized integration techniques such as Baumgarte stabilization [27]. They overcome difficulties related to imposing position constraints at the acceleration level (i.e., via controlling acceleration to zero) in the presence of numerical drift.

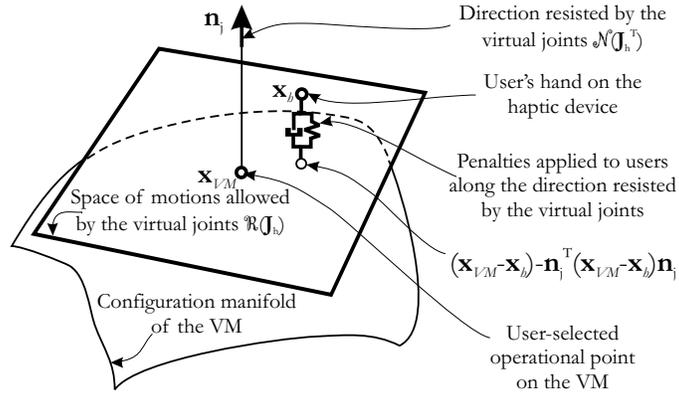


Fig. 3. Penalty wrench constraining users to the configuration manifold of the VM that they manipulate.

The directions resisted by the virtual joints \mathbf{n}_j are provided by $\mathcal{N}(\mathbf{J}_h^T)$. The constraint position and velocity are provided by the body position \mathbf{x}_{VM} and velocity $\dot{\mathbf{x}}_{VM}$ of the user-selected operational point on the VM. The stiffness k_j and the damping b_j of the penalties applied to users along the directions resisted by the virtual joints are used to render the infinite stiffness of the VM along these directions. Therefore, the control law in (11) is augmented with the control signal:

$$\mathbf{u}_{add} = \mathbf{M}_d \sum_{i=1}^j (k_j \mathbf{n}_{j,i}^T (\mathbf{x}_{VM} - \mathbf{x}_h) + b_j \mathbf{n}_{j,i}^T (\dot{\mathbf{x}}_{VM} - \dot{\mathbf{x}}_h)) \mathbf{n}_{j,i}, \quad (17)$$

where j is the number of directions resisted by the virtual joints. Note that $j = 6 - \text{rank}(\mathbf{J}_h)$ during spatial rigid body (6 DOFs) manipulation, and $j = 3 - \text{rank}(\mathbf{J}_h)$ during planar rigid body (3 DOFs) manipulation.

Hence, the proposed control is:

$$\begin{aligned}
\mathbf{u} = & (\mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} - \mathbf{I}) \mathbf{F}_h + \mathbf{M}_d \mathbf{J}_h \mathbf{D}^{-1} \tau + \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{F}_{pc} \\
& + (\mathbf{C}_d - \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{b}_d) \dot{\mathbf{x}}_h - \mathbf{M}_d \boldsymbol{\Lambda}_h^{-1} \mathbf{k}_d \mathbf{x}_h \\
& + \mathbf{M}_d \sum_{i=1}^j (k_j \mathbf{n}_{j,i}^T (\mathbf{x}_{VM} - \mathbf{x}_h)) \\
& + b_j \mathbf{n}_{j,i}^T (\dot{\mathbf{x}}_{VM} - \dot{\mathbf{x}}_h) \mathbf{n}_{j,i}.
\end{aligned} \tag{18}$$

The ability of this control to restrict user's motion as required by the joints of the VM is validated via simulations in Section VI and via controlled experiments in Section VIII.

VI. SIMULATIONS

In this section, two simulated manipulations of a three-link planar VM are used to compare the performance of the control in (18) to the performance of the control in (11) developed in [20]. In the simulations, users manipulate a three-link planar VM by applying a constant force $f_h = 0.4\text{N}$ along the x direction. The VM parameters are given in Table I. Initially, the VM is at rest in the position $\mathbf{q}_0 = (-\frac{\pi}{4}\text{rad} \quad \frac{\pi}{6}\text{rad} \quad -\frac{\pi}{6}\text{rad})^T$. The haptic device is also at rest, but its position differs from that of the user-selected operational point by 5mm along the x direction and by 3mm along the y direction (see Fig. 4). The VM motion is unrestricted by other virtual objects, i.e., no environment forces act on the VM during the simulated manipulations.

TABLE I
PARAMETERS OF THE THREE-LINK PLANAR VM OPERATED BY THE USER IN THE SIMULATIONS.

Link length (m)	Link mass (kg)	Link inertia ($\text{kg}\cdot\text{m}^2$)
$l_1 = 0.042\text{m}$	$m_1 = 3\text{kg}$	$I_1 = 0.015\text{kg}\cdot\text{m}^2$
$l_2 = 0.042\text{m}$	$m_2 = 3\text{kg}$	$I_2 = 0.015\text{kg}\cdot\text{m}^2$
$l_3 = 0.030\text{m}$	$m_3 = 1\text{kg}$	$I_3 = 0.005\text{kg}\cdot\text{m}^2$

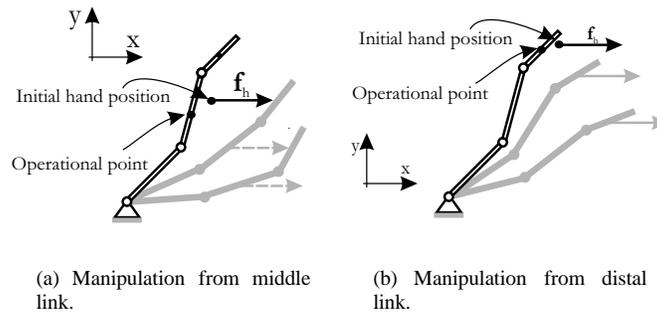


Fig. 4. Simulated manipulations of a three-links planar VM. The initial VM position is shown in black. VM positions during manipulation are shown in grey.

The SimulinkTM diagram of the manipulations is shown in Fig. 5. In this diagram, the VM simulation (corresponding to

Z_{VM} in Figure 1) is represented via the configuration space dynamics of the VM. The haptic device controlled according to (11) (corresponding to Z_{VM} in Figure 1) is represented via the VM operational space dynamics, i.e., it is assumed that the impedance device controller perfectly matches the impedance of the haptic device to the impedance of the VM ($Z_{HD} = Z_{VM}$). The additional control signal used to enforce joint constraints on users \mathbf{u}_{add} is explicitly represented. Moreover, \mathbf{M}_d is considered constant throughout the workspace and equal to unity. Note that Fig. 5 represents planar interaction, i.e., wrenches (shown in boldface upper letters) act on the haptic device and on the VM rather than forces (shown in normal lower letters), as in the 1DOF haptic interaction represented in Fig. 1. Note also that free linkage manipulation is simulated in Fig. 5, i.e., the virtual environment Z_{env} is not represented and no environment wrenches act on the device or on the VM.

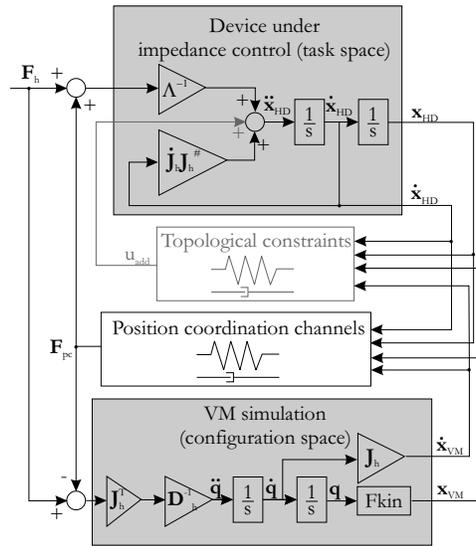


Fig. 5. Simulink diagram of the haptic manipulation of a three-links planar VM with motion unrestricted by other virtual objects.

The coordination between the device and the VM has stiffness:

$$\mathbf{K}_{pc} = \begin{pmatrix} 100\text{N/m} & 100\text{N/m} & 0.5\text{Nm/rad} \end{pmatrix}^T, \quad (19)$$

and damping:

$$\mathbf{B}_{pc} = \begin{pmatrix} 70\text{N/(m/s)} & 70\text{N/(m/s)} & 0.375\text{Nm/(rad/s)} \end{pmatrix}^T. \quad (20)$$

These values are chosen to match the values implemented in the haptic controller used in the experiments [20]. Compared to the impedance of the virtual contacts implemented in the experimental haptic interaction system ($k_{contact} = 15000\text{N/m}$ and $b_{contact} = 300\text{N/(m/s)}$), the impedance of the position coordination channels is very weak. This is because position coordination is used only to maintain kinematic correspondence between the device and the VM operational point, while the force channels are used to increase the realism of the interaction[20]. Furthermore, users would perceive a large coordination impedance as an

unphysical resistance during free motion. The stiffness and damping of the penalties used to impose joint constraints on users are $k_j = 200\text{N/m/kg}$ and $b_j = 50\text{N/(m/s)/kg}$, respectively. They also match the values implemented in the experimental haptic interaction system. These values have been chosen (via trial and error) as large as possible while preserving stability. They seem low compared to the impedance of the virtual contacts. However, note that, unlike the impedance of virtual contacts, the joint constraint impedance is applied to users at the acceleration level. In other words, it is not premultiplied by the inverse of the VM inertia matrix Λ_h^{-1} . For the virtual linkage illustrated in this work, Λ_h^{-1} is rather small and, hence, it correspondingly decreases the impedance of the virtual contacts at acceleration level.

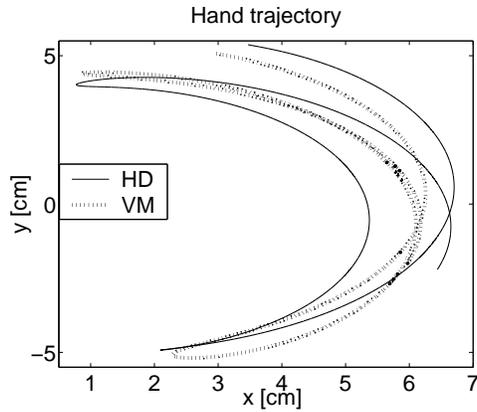
Fig. 6 depicts the results for the first simulated manipulation, in which users operate the VM from the COM of the middle link, as shown in Fig. 4(a). This link has only 2 DOFs. Hence, the VM joints constrain user's motion throughout the manipulation. The simulated trajectories of the user's hand on the haptic device ("HD") and of the operational point simulated on the VM ("VM") are shown in Fig. 6(a) for the case when the joint constraints are imposed at the acceleration level. They are shown in Fig. 6(b) for the case when user's motion along the restricted directions is penalized according to (17). Note that the user's hand drifts from the operational point when joint constraints are imposed at the acceleration level. In contrast, the drift is substantially reduced through penalizing user's motion along the directions resisted by the virtual joints.

The results for the second simulated manipulation are shown in Figure 7. This time, the user operates the VM from the COM of the distal link, as shown in Figure 4(b). Since the user holds a link with 3 DOFs, the penalties are applied only intermittently, when the user moves through a singularity. Once more, the trajectories in Figures 7(a) and 7(b) illustrate that the drift between the user's hand and the operational point on the VM is significant unless user's motion along the directions resisted by the VM joints is appropriately penalized through device control.

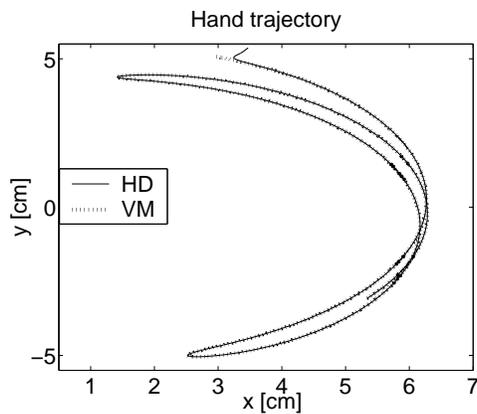
The control in (18) requires the computation of the directions resisted by the virtual joints \mathbf{n}_j . This computation is based on the SVD of \mathbf{J}_h , which may be numerically expensive for the speed requirements of the haptic control loop. The following section discusses suitable approximations of these directions that can be implemented in a local model of interaction with guaranteed real time performance [21].

VII. IMPLEMENTATION

To avoid the need to compute the directions resisted by the virtual joints at the speed of the control loop, the force control loop is decoupled from the simulation through a local model of rigid body interaction [21] (see Fig. 8). The local model is a reduced simulation that runs at the frequency of the force control loop. It approximates the interaction between the VM and the virtual environment via the interaction between the VM and nearby objects. The quality of the approximation is maintained by updating the local model at each step of the simulation.



(a) Joint constraints imposed via impedance control.

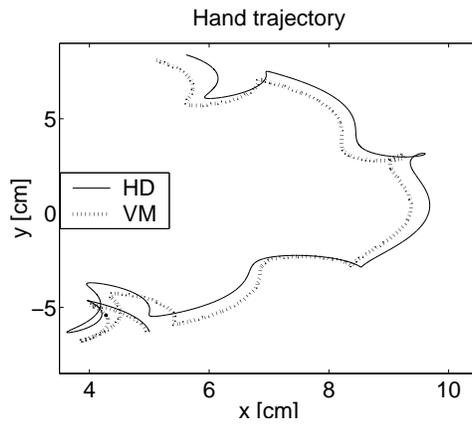


(b) Joint constraints imposed via penalties.

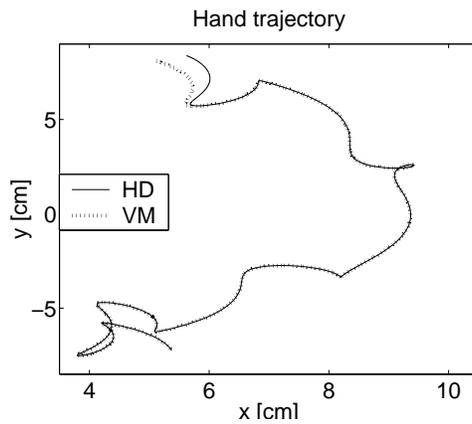
Fig. 6. Simulated planar manipulation of a three-links planar VM held from the COM of the middle link.

The local model of interaction allows the computations to be distributed between the simulation and the haptics processors. In particular, the contact geometry and the directions resisted by the virtual joints are computed on the simulation processor. The VM dynamics and the interaction and device control (including the penalties along the restricted directions) are computed on the haptics processor. This significantly reduces the load of the haptics processor, because the computationally intensive collision detection and SVD of the hand Jacobian are eliminated from the local model of the interaction. As a result of the increased numerical efficiency of the local model compared to the VM simulation, the wrenches enforcing the VM joint and contact constraints can be computed at the speed of the haptic loop⁶, and users can manipulate VMs with larger number of links. Furthermore, the local joint constraint geometry alleviates the computational delay of the virtual environment. This allows stiffer joint constraints to be imposed on users via the proposed approach than via connecting users to the virtual environment

⁶Attempting to compute the SVD of the hand Jacobian on the haptics processor renders the local model too slow for the force feedback loop when implemented on the haptic planar interaction system described in Section VIII.



(a) Joint constraints imposed via impedance control.



(b) Joint constraints imposed via penalties.

Fig. 7. Simulated planar manipulation of a three-links planar VM held from the COM of the distal link.

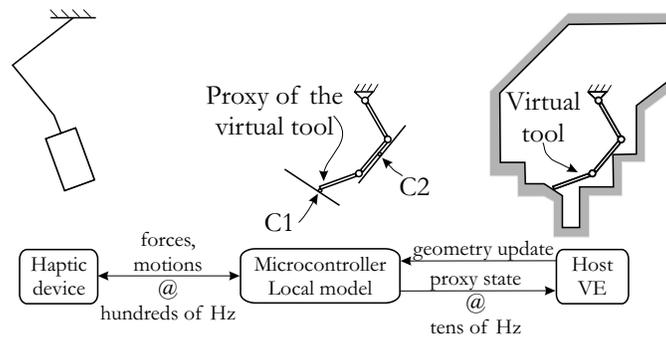


Fig. 8. Decoupling of the force control loop from the virtual environment (VE) simulation through a local model of rigid body interaction.

through virtual coupling (as validated experimentally in the following section).

During haptic manipulation of VMs, the numerical efficiency of the local model is further increased by approximating the VM configuration space inertia and the gravitational terms via their values in the virtual environment at the moment of the update, and by ignoring Coriolis and centripetal effects (similar to work in [4] and [2]), i.e., by simulating:

$$\widehat{\mathbf{D}}\ddot{\mathbf{q}} + \widehat{\mathbf{G}} = \sum_{i=1}^c \mathbf{J}_i^T \mathbf{F}_i + \mathbf{J}_h^T \mathbf{F}_h. \quad (21)$$

In (21), $\widehat{\mathbf{D}}$ and $\widehat{\mathbf{G}}$ denote the values of \mathbf{D} and \mathbf{G} , respectively, computed by the simulation and sent to the local model at the update. Furthermore, the directions resisted by the VM joints are approximated through their values in the virtual environment at the update. In other words, the SVD of \mathbf{J}_h is computed by the simulation:

$$\widehat{\mathbf{J}}_h = \widehat{\mathbf{U}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{V}}^T \quad (22)$$

and the columns of $\widehat{\mathbf{U}}$ are used in the local model to estimate the directions restricted by the virtual joints. Lastly, $\mathbf{\Lambda}_h^{-1}$ is approximated locally by:

$$\widehat{\mathbf{\Lambda}}_h^{-1} = \widehat{\mathbf{J}}_h \widehat{\mathbf{D}}_{vt}^{-1} \widehat{\mathbf{J}}_h^T. \quad (23)$$

Operation of VMs using the approximations provided by equations (21), (22), and (23) is demonstrated experimentally in the following section.

VIII. EXPERIMENTS

In this section, VM manipulations from various user-selected links are illustrated via two sets of experiments. The first set of experiments investigates controlled manipulations within the virtual environment depicted in Fig. 9. The second set of experiments illustrates unrestricted manipulations by human users within the dynamic virtual environment shown in Fig. 10. In all experiments, users operate the three-link planar VM (dimensions and inertial properties are given in Table I) via the planar haptic interface available in the Robotics and Controls Laboratory at the University of British Columbia [21]. The VM is initially at rest, in the configuration space position $\mathbf{q}_0 = (0\text{rad} \quad \frac{\pi}{6}\text{rad} \quad -\frac{\pi}{3}\text{rad})^T$. The stiffness and damping of the penalties enforcing the virtual joint constraints are $k_j = 200\text{N/m/kg}$ and $b_j = 50\text{Ns/m/kg}$, respectively. The stiffness and damping of the virtual coupling are $\mathbf{K}_{vc} = (50\text{N/m} \quad 50\text{N/m} \quad 0.2\text{Nm/rad})^T$ and $\mathbf{B}_{vc} = (3\text{N/(m/s)} \quad 3\text{N/(m/s)} \quad 0.005\text{Nm/(rad/s)})^T$, respectively⁷. Trajectories are represented in task space coordinates, with x and y being coordinates in the plane of motion

⁷These values represent the maximum impedance of the virtual coupler for which the interaction is stable, i.e., for which chattering does not occur.

and θ being rotation about an axis perpendicular on the motion plane.

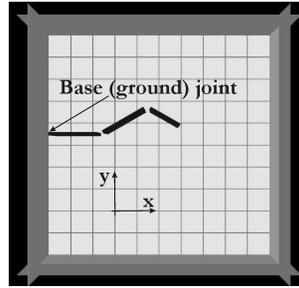


Fig. 9. Testbed virtual environment used to illustrate haptic manipulation of VMs from arbitrary user-selected links.

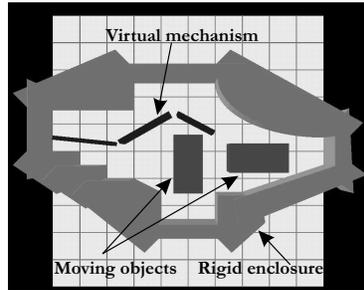


Fig. 10. Testbed virtual environment used to illustrate haptic manipulation of VMs within a dynamic virtual environment.

The first set of experiments contrasts three types of trajectories: (i) trajectories obtained via penalizing user’s motion along the directions resisted by the virtual joints using (18); (ii) trajectories obtained via controlling to zero user’s acceleration along these directions using the impedance controller in (11); and (iii) trajectories obtained via controlling user’s motion through virtual coupling. Since human users cannot apply the same wrenches during successive manipulations, a controlled constant wrench⁸ $\mathbf{F}_h = (0.4\text{N} \ 0\text{N} \ 0\text{Nm})^T$ represents the human operator in the first set of experiments. This constant wrench ensures the “same” user during all manipulations, and allows successive experimental trajectories to be compared to each other.

The trajectories of the user’s hand on the haptic device (“HD”) and of the user-selected operational point on the VM (“VM”) plotted in Fig. 11 correspond to manipulation from the middle link. The trajectories shown in Fig. 12 represent manipulation from the distal link. Note that the closed-loop dynamics are different when users are connected to the VM operational point via the various controllers. Therefore, the operational point on the VM follows a different trajectory in each case. Nevertheless, the experimental trajectories allow a qualitative comparison of the various controllers. In particular, Figs. 11(a) and 12(a) demonstrate that the control in (18) effectively opposes user’s motion along the directions resisted by the virtual joints regardless of the user-selected operational point. This is illustrated by the fact that the user’s hand trajectory follows the VM trajectory within the steady state error due to the limited stiffness of the penalties applied to users. In contrast,

⁸For the impedance type haptic interface employed in these experiments, a constant wrench represents a worst case scenario for stability [28].

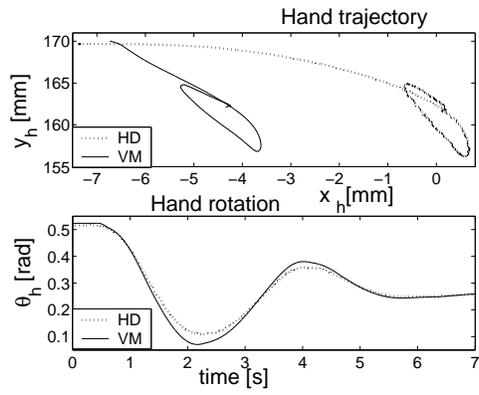
Figs. 11(c) and 12(c) show that the user's hand drifts from the VM operational point by tens of centimeters when the virtual joints are enforced via impedance control. This is because the impedance controller cannot oppose user-applied wrenches in the null space of the hand Jacobian, and the device moves freely under the user-applied wrenches in this space and quickly drifts from the VM operational point. Note also that the trajectories enforced via penalizing motion along the directions resisted by the virtual joints have smaller steady state error than the trajectories enforced via virtual coupling. The smaller steady state error demonstrates that users feel stiffer constraints via the proposed controller than via the virtual coupler. Furthermore, the haptic device quickly comes to rest under the proposed control (as it would if the linkage were real) and it oscillates around the equilibrium position when connected to the VM via virtual coupling. This demonstrates that the present approach results in more realistic haptic manipulation of VMs than the virtual coupler.

The second set of experiments illustrates manipulations from the distal link of the VM within the planar virtual environment depicted in Fig. 10. This virtual environment includes a rigid enclosure and two moving rectangular objects in addition to the VM. The device trajectories and the user-perceived wrenches plotted in Fig. 13 are obtained when the user is connected to the VM via the proposed approach. Those plotted in Fig. 14 are obtained when the user is connected to the VM via virtual coupling. Since users cannot apply the same wrenches in successive experiments, the trajectories in Figs. 13(a) and 14(a) cannot be compared to each other⁹. However, users operate the VM such that it collides with other virtual objects in each experiment. Therefore, the user-perceived wrenches in Figs. 13(b) and 14(b) can be qualitatively compared. The forces felt by users when the VM collides with other virtual objects are two orders of magnitude larger in Fig. 13(b) than in Fig. 14(b) (note the different vertical scales used in these figures). This confirms that the proposed controller does not low-pass filter the impulsive wrenches computed in the virtual environment. Hence, the experiments depicted in Figs. 13 and 14 validate that the proposed controller renders virtual joint constraints without degrading the transparency of the four channel teleoperation architecture (i.e., without low-pass filtering the impulsive contact wrenches).

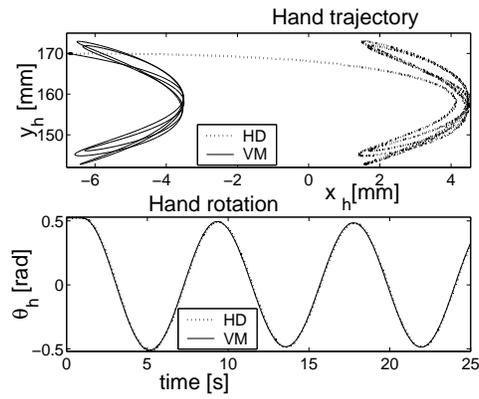
IX. CONCLUSION

This paper has proposed an approach for applying physically-based wrenches to users operating open-loop VMs. The approach allows manipulations of VMs with an arbitrary number of links, from any link, and through singularities. In this approach, a haptic device controller is designed as an integral part of a previously developed control architecture. The proposed controller enforces the VM joint constraints, and the existing architecture applies to users the VM inertia and the contact interactions between the VM and the virtual environment. The new controller acts as generalized springs and dampers that oppose users'

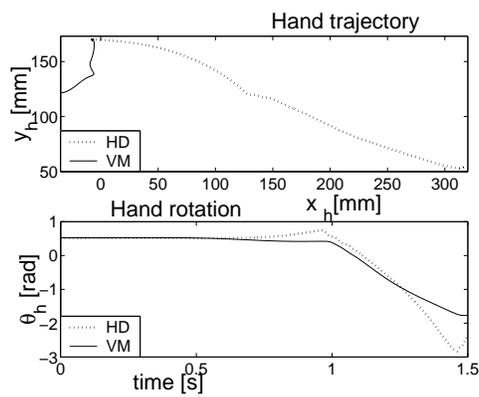
⁹Note also that the trajectories plotted in Figs. 13(a) and 14(a) do not represent the same interaction as the one depicted in Figs. 12 and 12 and, therefore, the device trajectory errors are different in the two sets of experiments.



(a) Joint constraints enforced via penalties.

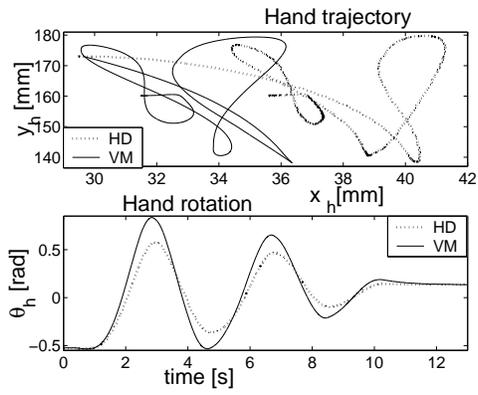


(b) Joint constraints enforced via virtual coupling.

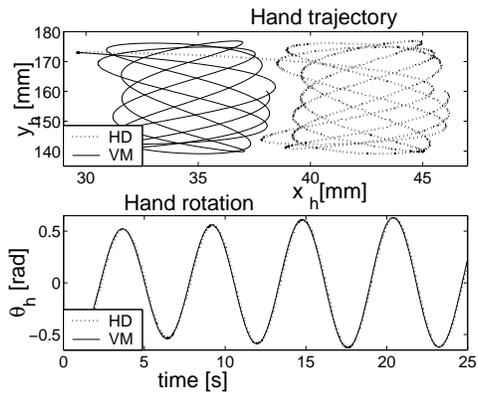


(c) Joint constraints enforced via impedance control.

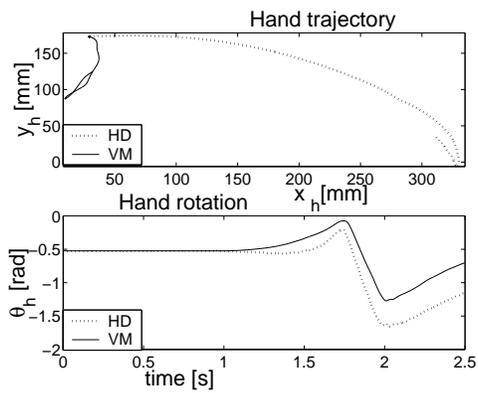
Fig. 11. Manipulation from the middle link of the planar VM shown in Fig. 9.



(a) Joint constraints enforced via penalties.

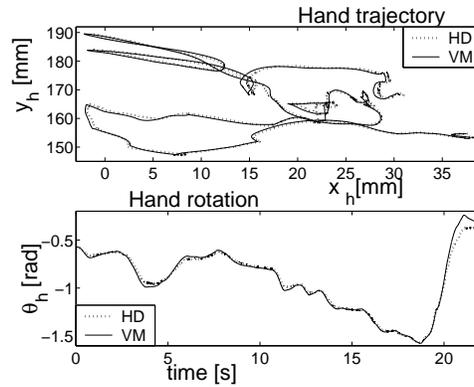


(b) Joint constraints enforced via virtual coupling.

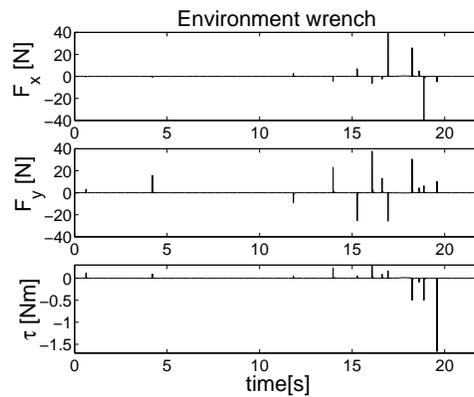


(c) Joint constraints enforced via impedance control.

Fig. 12. Manipulation from the distal link of the planar VM shown in Fig. 9.



(a) Linear ($x - y$) and angular (θ) trajectories.

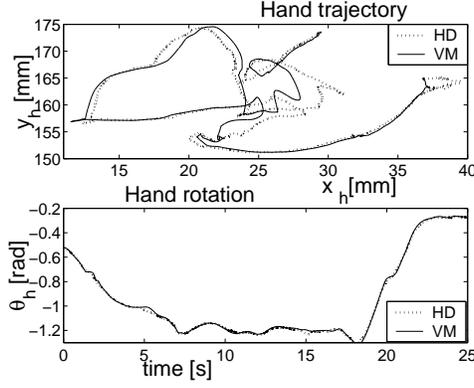


(b) Wrenches applied on users. Note that the wrenches rendering the VM impacts with the virtual environment are two orders of magnitude larger than and, thus, can easily be distinguished from other user-perceived wrenches.

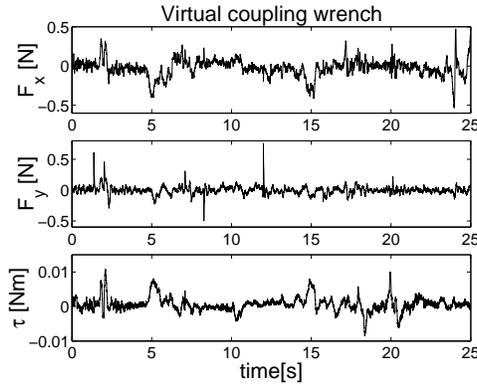
Fig. 13. Manipulation from the distal link of the planar VM shown in Fig. 10 rendered via the proposed approach.

motion along the directions resisted by the virtual joints. These directions are orthogonal to the directions along which users feel contact wrenches. Therefore, the new controller enforces joint constraints and simultaneously allows users to perceive contact interactions with high frequency components, such as the impacts between the VM and the virtual environment. Experimental results also illustrate that users can feel stiffer joint constraints when connected to the VM operational point via the proposed approach than when connected via virtual coupling.

Future work will investigate the extension of the proposed approach to virtual mechanisms with arbitrary topologies, including multiple closed loops, and the development of controllers with nonlinear damping for increasing the perceived rigidity of the virtual joint constraints.



(a) Linear ($x - y$) and angular (θ) trajectories.



(b) Wrenches applied on users. Note that the wrenches rendering the VM impacts with the virtual environment are the same order of magnitude as and, thus, cannot easily be distinguished from other user-perceived wrenches.

Fig. 14. Manipulation from the distal link of the planar VM shown in Fig. 10 rendered via virtual coupling.

APPENDIX: EQUALITY OF THE NULL SPACES OF Λ_h^{-1} AND \mathbf{J}_h^T

This section shows that the null spaces of Λ_h^{-1} and \mathbf{J}_h^T are equal by showing that they have the same basis. The proof is by construction and uses the SVD of Λ_h^{-1} and \mathbf{J}_h^T .

Let the SVD of \mathbf{J}_h^T be given by:

$$\mathbf{J}_{h_{d \times 6}}^T = \mathbf{V}_{d \times d} \Sigma_{d \times 6} \mathbf{U}_{6 \times 6}^T, \quad (24)$$

and \mathbf{J}_h^T have rank r . Then:

$$\Sigma_{d \times 6} = \begin{bmatrix} \Sigma'_{r \times r} & \mathbf{0}_{r \times (6-r)} \\ \mathbf{0}_{(d-r) \times r} & \mathbf{0}_{(d-r) \times (6-r)} \end{bmatrix}, \quad (25)$$

where $\Sigma'_{r \times r}$ is the r -dimensional diagonal matrix having the r non-zero singular values of \mathbf{J}_h^T on its main diagonal. Furthermore,

the last $6 - r$ columns in $\mathbf{U}_{6 \times 6}^T$ provide a basis for the null space of \mathbf{J}_h^T .

Substituting (25) into (6), $\mathbf{\Lambda}_h^{-1}$ can be computed by:

$$\mathbf{\Lambda}_{h_{6 \times 6}}^{-1} = \mathbf{U}_{6 \times 6} \mathbf{\Sigma}_{6 \times d} \mathbf{V}_{d \times d}^T \mathbf{D}_{d \times d}^{-1} \mathbf{V}_{d \times d} \mathbf{\Sigma}_{d \times 6}^T \mathbf{U}_{6 \times 6}^T. \quad (26)$$

\mathbf{D}^{-1} is symmetric positive definite and can be reduced to the diagonal form \mathbf{A} through a suitable rotation \mathbf{R} :

$$\mathbf{A} = \begin{pmatrix} a_1 & 0 & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & a_d \end{pmatrix} = \mathbf{R} \mathbf{D}^{-1} \mathbf{R}^T. \quad (27)$$

Moreover, $\mathbf{Y} = \mathbf{R} \mathbf{V} = \begin{pmatrix} \mathbf{y}_1 & \dots & \mathbf{y}_d \end{pmatrix}$ is orthogonal (since $\mathbf{V}^T \mathbf{R}^T \mathbf{R} \mathbf{V} = \mathbf{I}$), and:

$$\begin{aligned} \mathbf{V}^T \mathbf{D}^{-1} \mathbf{V} &= \mathbf{V}^T \mathbf{R}^T \mathbf{A} \mathbf{R} \mathbf{V} = \\ &= \begin{pmatrix} \mathbf{y}_1^T \\ \vdots \\ \mathbf{y}_d^T \end{pmatrix} \begin{pmatrix} a_1 & 0 & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & a_d \end{pmatrix} \begin{pmatrix} \mathbf{y}_1 & \dots & \mathbf{y}_d \end{pmatrix} = \\ &= \begin{pmatrix} \mathbf{y}_1^T \\ \vdots \\ \mathbf{y}_d^T \end{pmatrix} \begin{pmatrix} a_1 \mathbf{y}_1 & \dots & a_d \mathbf{y}_d \end{pmatrix} = \mathbf{A}. \end{aligned} \quad (28)$$

From (26), (28) and (25), it follows that:

$$\begin{aligned} \mathbf{\Sigma}_{6 \times d} \mathbf{V}_{d \times d}^T \mathbf{D}_{d \times d}^{-1} \mathbf{V}_{d \times d} \mathbf{\Sigma}_{d \times 6}^T &= \\ &= \begin{bmatrix} \mathbf{\Sigma}'_{r \times r} \mathbf{A}_{r \times r} \mathbf{\Sigma}'_{r \times r} & \mathbf{0}_{r \times (d-r)} \\ \mathbf{0}_{(d-r) \times r} & \mathbf{0}_{(d-r) \times (d-r)} \end{bmatrix}. \end{aligned} \quad (29)$$

In (29), $\mathbf{A}_{r \times r}$ is the diagonal matrix having a_1, \dots, a_r on its main diagonal, and:

$$\mathbf{\Sigma}' \mathbf{A} \mathbf{\Sigma}' = \begin{bmatrix} a_1 \sigma_1^2 & \dots & 0 \\ \vdots & & \\ 0 & \dots & a_r \sigma_r^2 \end{bmatrix}. \quad (30)$$

After substitution from (30), (26) becomes:

$$\begin{aligned}\Lambda_{h_{6 \times 6}}^{-1} &= \mathbf{U}_{6 \times 6} \begin{bmatrix} \Sigma'_{r \times r} \mathbf{A}_{r_r \times r_r} \Sigma'_{r \times r} & \mathbf{0}_{r \times (d-r)} \\ \mathbf{0}_{(d-r) \times r} & \mathbf{0}_{(d-r) \times (d-r)} \end{bmatrix} \mathbf{U}_{6 \times 6}^T = \\ &= \mathbf{U}_{6 \times 6} \Sigma_{\Lambda_{6 \times 6}} \mathbf{U}_{6 \times 6}^T,\end{aligned}\tag{31}$$

with $\mathbf{U}_{6 \times 6}$ orthogonal, and $\Sigma_{\Lambda_{6 \times 6}}$ diagonal and having the first r diagonal elements non-zero. In other words, (31) gives the SVD of Λ_h^{-1} and the last $6 - r$ columns in $\mathbf{U}_{6 \times 6}^T$ provide a basis for the null space of Λ_h^{-1} .

REFERENCES

- [1] A. Nahvi, D. Nelson, J. Hollerbach, and D. Johnson, "Haptic Manipulation of Virtual Mechanisms from Mechanical CAD Designs," in *Proc. IEEE Int. Conf. Robot. Autom.*, Leuven, Belgium, May 1998, pp. 375–380.
- [2] D. Ruspini and O. Khatib, "Haptic Display for Human Interaction with Virtual Dynamic Environments," *J. Robot. Syst.*, vol. 18, no. 2, pp. 769–783, 2001.
- [3] W. Son, K. Kim, N. Amato, and J. Trinkle, "Interactive Dynamic Simulation using Haptic Interaction," in *Proc. IEEE Int. Conf. Robot. Autom.*, San Francisco, CA, May 2000, pp. 145–150.
- [4] D. Ruspini and O. Khatib, "Dynamic Models for Haptic Rendering Systems," in *Adv. Robot Kinematics: ARK98*, Strobl/Salzburg, Austria, 1998, pp. 523–532.
- [5] —, "Collision/Contact Models for Dynamic Simulation and Haptic Interaction," in *Proc. 9th Int. Symp. Robot. Res.*, Snowbird, UT, 1999, pp. 185–195.
- [6] O. Khatib, O. Brock, K.-S. Chang, D. Ruspini, L. Sentis, and S. Viji, "Human-Centered Robotics and Interactive Haptic Simulation," *Int. J. Robot. Res.*, vol. 23, no. 2, pp. 167–178, 2004.
- [7] W. Son, K. Kim, N. Amato, and J. Trinkle, "A Generalized Framework for Interactive Dynamic Simulation for MultiRigid Bodies," *IEEE Trans. Syst. Man Cyber.*, vol. 34, no. 2, pp. 912–924, April 2004.
- [8] D. Constantinescu, S. Salcudean, and E. Croft, "Haptic Rendering of Rigid Contacts using Impulsive and Penalty Forces," *IEEE Trans. Robot.*, vol. 21, no. 3, pp. 309–323, June 2005.
- [9] R. Featherstone, *Robot Dynamics Algorithms*. Boston:Kluwer, 1987.
- [10] A. Jain, "Unified Formulation of Dynamics for Serial Multibody Systems," *J. Guid. Contr. Dyn.*, vol. 14, no. 3, pp. 531–542, 1991.
- [11] K. Lilly and D. Orin, "Efficient Dynamic Simulation of Multiple Chain Robotic Mechanisms," *Trans AMSE J. Dyn. Syst. Meas. Contr.*, vol. 116, pp. 223–231, 1994.
- [12] K.-S. Chang and O. Khatib, "Operational Space Dynamics: Efficient Algorithms for Modeling and Control of Branching Mechanisms," in *Proc. IEEE Int. Conf. Robot. Autom.*, San Francisco, CA, 2000, pp. 850–856.
- [13] O. Khatib, "A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation," *IEEE Trans. Robot. Autom.*, vol. 3, no. 1, pp. 43–53, 1987.
- [14] —, "Inertial Properties in Robotic Manipulation: An Object-Level Framework," *Int. J. Robot. Res.*, vol. 13, no. 1, pp. 19–36, 1995.
- [15] D. Stewart and J. Trinkle, "An Implicit Time-Stepping Scheme for Rigid Body Dynamics with Inelastic Collisions and Coulomb Friction," *Int. J. Num. Meth. Eng.*, vol. 39, no. 15, pp. 2673–2691, 1996.
- [16] B. Mirtich and J. Canny, "Impulse-based Dynamic Simulation," in *Workshop Algorithm. Found. Robot.*, K. Goldberg, P. Halperin, J.-C. Latombe, and R. Wilson, Eds. A.K.Peters, Boston, MA., 1994, pp. 407–418.

-
- [17] D. Ruspini, K. Koralov, and O. Khatib, "The Haptic Display of Complex Graphical Environments," in *Proc. SIGGRAPH 97*, Los Angeles, CA, 1997, pp. 345–352.
- [18] J. Brown and J. Colgate, "Physics-based Approach to Haptic Display," in *Proc. Topic. Workshop Virt. Real. Int. Symp. Meas. Contr. Robot.*, vol. 1, Houston, TX, 1994, pp. 101–106.
- [19] R. Adams and B. Hannaford, "Stable Haptic Interaction with Virtual Environments," *IEEE Trans. Robot. Autom.*, vol. 15, no. 3, pp. 465–474, June 1999.
- [20] M. Sirouspour, S. DiMaio, S. Salcudean, P. Abolmaesumi, and C. Jones, "Haptic Interface Control – Design Issues and Experiments with a Planar Device," in *Proc. IEEE Int. Conf. Robot. Autom.*, San Francisco, Ca, 2000, pp. 789–794.
- [21] D. Constantinescu, S. Salcudean, and E. Croft, "Local Models of Interaction for Realistic manipulation of Rigid Virtual Worlds," *Int. J. Robot. Res.*, vol. 24, no. 10, pp. 789–804, October 2005.
- [22] L. Joly and C. Andriot, "Imposing Motion Constraints to a Force Reflecting Telerobot through Real-Time Simulation of a Virtual Mechanism," in *Proc. IEEE Int. Conf. Robot. Autom.*, Nagoya, Japan, 1995, pp. 357–363.
- [23] G. Luecke, J. Edwards, and B. Miller, "Virtual Cooperating Manipulator Control for Haptic Interaction with NURBS Surfaces," in *Proc. IEEE Int. Conf. Robot. Autom.*, Albuquerque, New Mexico, 1997, pp. 112–117.
- [24] G. Luecke and N. Zafer, "Constraint Coupling of Force and Motion Between Real and Virtual Mechanisms," in *SPIE Proc. Mobile Robot XV Telemat. Telepres. Technol. VI*, vol. 4195, Boston, MA, 2001, pp. 331–339.
- [25] N. Hogan, "Impedance Control: An Approach to Manipulation. Theory, Implementaion, Application," *Trans. ASME J. Dyn. Syst. Meas. Contr.*, vol. 107, pp. 1–24, March 1985.
- [26] D. Lawrence, "Stability and Transparency in Bilateral Teleoperation," *IEEE Trans. Robot. Autom.*, vol. 9, no. 5, pp. 624–637, 1993.
- [27] J. Baumgarte, "Stabilization of Constraints and Integrals of Motion in Dynamical Syatems," *Comp. Meth. Appl. Mech. Eng.*, vol. 1, pp. 1–16, 1972.
- [28] B. Hannaford and R. Anderson, "Experimental and Simulation Studies of Hard Contact Force Reflecting Teleoperation," in *Proc. IEEE Int. Conf. Robot. Autom.*, Scottsdale, AZ, May 1988, pp. 24–29.

LIST OF FIGURES

1	Mechanical representation (excluding the communication delay and force scaling) of one dimensional haptic interaction within a virtual environment via the four channel teleoperation controller.	4
2	Example manipulations of a planar VM where the virtual joints restrict the instantaneous body motion at the user-selected operational point.	7
3	Penalty wrench constraining users to the configuration manifold of the VM that they manipulate.	10
4	Simulated manipulations of a three-links planar VM. The initial VM position is shown in black. VM positions during manipulation are shown in grey.	11
5	Simulink diagram of the haptic manipulation of a three-links planar VM with motion unrestricted by other virtual objects.	12
6	Simulated planar manipulation of a three-links planar VM held from the COM of the middle link.	14
7	Simulated planar manipulation of a three-links planar VM held from the COM of the distal link.	15
8	Decoupling of the force control loop from the virtual environment (VE) simulation through a local model of rigid body interaction.	15
9	Testbed virtual environment used to illustrate haptic manipulation of VMs from arbitrary user-selected links. . .	17
10	Testbed virtual environment used to illustrate haptic manipulation of VMs within a dynamic virtual environment. . .	17
11	Manipulation from the middle link of the planar VM shown in Fig. 9.	19
12	Manipulation from the distal link of the planar VM shown in Fig. 9.	20
13	Manipulation from the distal link of the planar VM shown in Fig. 10 rendered via the proposed approach. . . .	21
14	Manipulation from the distal link of the planar VM shown in Fig. 10 rendered via virtual coupling.	22

LIST OF TABLES

I	Parameters of the three-link planar VM operated by the user in the simulations.	11
---	---	----