PASSIVE HAPTIC CONTROL DURING TELEOPERATION

Benjamin Black ° Wayne J. Book °

*Intelligent Machine Dynamics Laboratory
Georgia Institute of Technology
Atlanta, Georgia USA

Abstract: The basis of haptic teleoperation focuses on augmenting a user’s ability to perform a task in a remote environment. Using an energetically passive haptic master assures that the energy can only be redirected or removed from the human-manipulator pair, guaranteeing the user’s safety. However using a passive master complicates the display of haptic forces due to the device’s limitations. The master device cannot be controlled arbitrarily, but instead can only resist the motion of the human user. This research focuses on the use of a planar passive haptic device actuated via three magneto-rheological (MR) brakes. This device is being used to teleoperate a linear motor over the internet via a UDP connection. Both the master and the slave are controlled using National Instruments (NI) hardware and LabVIEW software. The following will present and compare two actuation schemes applying traditional active teleoperation rules for to the control of the passive haptic system. The inadequacies of these schemes will be illustrated, showing the need for a more complex control scheme to produce haptic feedback in a passive system. Copyright 2006 IFAC

Keywords: haptic, control, passive, teleoperation

1. INTRODUCTION

Haptic devices are designed to provide physical feedback during human-machine or human-computer interaction. Devices can either be on a large scale using gross muscle movements or on a small scale providing tactile feedback (Zhu and Book, 2004). In either case, the devices are designed and controlled to improve the user’s ability to complete a task by providing feedback in the form of the sense of touch.

Haptic devices can be divided into two groups defined by the energetic nature of their actuators, either active or passive. Active haptic devices are actuated using components such as motors or pneumatics and comprise the largest portion of haptic devices. Conversely, passive haptic devices use energetically passive actuators, either energetically neutral such as the cobots of Colgate (Moore et al., 1999), or dissipative such as PTER by Book (Robert, 1994) (Swanson, 2003) (Book and Swanson, 2003) and the rehabilitation devices of Koyanagi (Koyanagi et al., 2002). Using passive actuators prevents the devices from adding energy to the system, all of the energy in the system is provided by the human operator. The passive nature of these devices complicates control due to the fact that they cannot arbitrarily move themselves (Gao and Book, 2005); however passivity guarantees the safety of the human user.

The second portion of the research presented focuses on remote control, or teleoperation. Whether the remote, slave device is in space (Yoon et al., 2001), in a hazardous area (Lee et al., 2004),
Fig. 1. The drawing shows the configuration of the planar passive haptic master. Note that joints A, B & E are actuated while joints C & D are not.

or in micro-scale (Boukhnifer et al., 2004), tele-operation provides virtual presence of a human operator. In addition, using a haptic master device in teleoperation produces tactile feedback to the user from the remote environment. Data fed back from the slave device can be used to calculate forces to be displayed to the human user. The following research explores the control difficulties of a passive haptic teleoperator and proposes two simple control schemes based on classic teleoperation principles.

2. HARDWARE & SOFTWARE

As with typical teleoperation systems, the hardware used for this research includes two separate systems, the master and its controller as well as the remote, or slave device and its controller. The two devices are controlled via National Instruments (NI) PXI computers, ruggedized systems with architecture designed specifically for input-output and control. A similar setup was used in previous research (Black and Book, 2005). The devices as well as the programming will be discussed briefly in the following.

2.1 Haptic Slave

The slave device used for the current research is an Anorad linear motor and amplifier, controlled by a “headless” NI PXI-8145 RT. The controller runs LabVIEW Real-Time on a 266MHz Pentium processor. Feedback from the motor’s Renishaw encoder is used along with a position setpoint received from the master device to achieve simple PD control. Over the slave workspace, the micron-resolution encoder provides 375,000 point resolution. The linear position of the motor is also sent back to the master controller to be used in force calculation.

Current research using a 1-Dimensional slave began as a stepping stone in setting up a teleoperation system, the plan being to move to a 2-D slave device. However, constraining the master to the 1-D workspace of the slave device produces interesting control difficulties that mirror previous passive haptic path-following research (Swanson, 2003)(Reed, 2003).

2.2 Haptic Master

Figure 1 illustrates the two degree of freedom (DOF) 4-link manipulator developed in previous research by Reed & Book (Reed, 2003) that is used as the haptic master. The two base joints, A & B as well as joint E are actuated using LORD magneto-rheological brakes (MRB-2107-3), the control of which achieves a time constant of approximately 7ms. The base angles, $\theta_A$ and $\theta_B$ in Fig. 1, are measured using Dynamics Research Quadrature Encoders that yield 278 counts per degree or approximately a 50,000 count resolution over the range of the workspace. The human user contacts the haptic master using a handle that is attached to an ATI force sensor at joint D.

Control of the master is achieved using a National Instruments (NI) PXI-8175 that operates under WindowsXP and runs on a PentiumIII 866MHz processor with 512MB of RAM. A PXI-6070E DAQ card along with a PXI-6711 analog output card provides the digital and analog I/O.

3. HAPTIC CONTROL

The basis for the control schemes presented here comes directly from classic haptic teleoperation. A virtual coupling, in this case a simple spring coupling, is used between the endpoint of the master and the endpoint of the slave. The haptic force to be displayed to the user is proportional to the difference between the position of the master and slave devices. Equation 1 defines $K$ as the virtual spring constant that sets the strength of the haptic coupling. Equation 2 represents the haptic force $\vec{f}_h$ as a magnitude in the direction of the unit vector $\vec{e}_h$. This magnitude will be used later in the control.

$$\vec{f}_h = K (p_s - p_m)$$

$$\vec{f}_h = K |p_s - p_m| \vec{e}_h$$

While the calculation of the haptic force comes from a simple spring equation, the display of that force proves to be complicated in a passive system. Locking a brake of the passive master will force the device to move along a single degree
Fig. 2. The three single degree of freedom paths created by locking brakes A, B & E as well as the possible forces generated by each brake of freedom circular path. At any point in the workspace of the master, there exist three such single DOF paths, the directions of which are shown in Fig. 2(a) for a specific configuration. The directions are labeled \( p_A, p_B & p_E \) to represent the paths created by locking brakes A, B & E and are guaranteed to be unique by physically preventing the device from entering a singular configuration. The force produced by a specific brake is perpendicular to the single DOF path at any given point and can point either toward the center of rotation or away from the center of rotation depending on the endpoint velocity. Figure 2(b) shows the direction of forces at the same device configuration as shown in Fig. 2(a). The forces are labeled \( f_A, f_B & f_E \) in the negative and positive direction to represent all possible directions of force that can be generated by the individual brakes A, B & E.

Since the device is guaranteed to remain passive due to its actuators, the torques generated by the brakes must oppose the motion of the joints. The forces at the endpoint must similarly oppose the endpoint velocity. Therefore the direction of the endpoint velocity defines the direction of endpoint force produced by each brake. The dashed vectors in Fig. 2(b) represent forces that cannot be generated. The region of \( \vec{v} \pm \frac{\pi}{2} \) defines the region of un-producible forces. This region is visible in Fig. 2(b) as the vertical line through the endpoint. From here forward, the discussion will deal with forces only when the input direction is known, and will be labeled only as \( f_A, f_B & f_E \) to represent the forces generated by actuating A, B & E. The following actuation schemes only apply if \( \vec{f}_h \) lies outside the region of un-producible forces. Note that as the velocity slows to zero, calculation of \( \vec{v} \) becomes indeterminate, and the control ceases to function properly.

![Fig. 2](image)

3.1 On-Off Control

In the simplest actuation scheme, the brake that produces a force closest in direction to \( \vec{f}_h \) is maximally actuated. The magnitude of \( f_a \) is ignored, and the brakes are actuated in a full-on / off manner. In Fig. 3(a), brake A would be actuated for the haptic force direction \( \vec{e}_{h1} \), and brake B would be actuated for force direction \( \vec{e}_{h2} \). Previous work has been done exploring the achievable results using this control scheme in a simple test (Black and Book, 2005).

3.2 Blended Control

A more complex control involves actuating multiple brakes at the same time. Again in Fig. 3(a) the calculated force is in the direction of \( \vec{e}_{h1} \), only brake A would be actuated. However, if the calculated haptic force direction is between two brake forces as with \( \vec{e}_{h2} \), the two brakes will be actuated in the following manner.

For calculation of the actuation, the control scheme uses the variables shown in Fig. 3(b). Assume \( \vec{f}_h \) lies between two brake forces, shown in this figure as \( f_1 \) and \( f_2 \), the magnitudes of which are calculated with Eqn. 3, represent a unit actuation of a given brake, and are dependent on both the kinematics and user input.

\[
\begin{bmatrix}
\dot{x}
\dot{y}
\end{bmatrix}^T = \begin{bmatrix}
\tau_\alpha
\tau_\beta
\end{bmatrix}^T [J_{\alpha\beta}]^{-1} + \begin{bmatrix}
\tau_\gamma
\tau_\epsilon
\end{bmatrix}^T [J_{\gamma\epsilon}]^{-1} \tag{3}
\]

The Jacobian can be broken down into sections such that \( \alpha, \beta, \gamma \) or \( \epsilon \) corresponds to joint A,B,C or E. For example, \( J_{AB} \) would relate torques at joints A & B to endpoint forces. Through that, the endpoint forces can be related to any combination of the joint torques. By sequentially setting each brake torque to a unit strength and the other three torques to zero, then calculating the endpoint force, the force vector created by a unit actuation is found. This is done for the brakes A, B &
E and used in the control as \( f_1 \) and \( f_2 \) from Fig. 3(b). As a side note, the friction torque as well as any actuation at joint C is ignored. A square version of the Jacobian is always invertible due to the fact that the device is mechanically limited from entering singular configurations. The control uses the angle of the two resultant forces on either side of \( f_h \), denoted by \( \theta_1 \) or \( \theta_2 \). The variable \( \theta_h \) defines the direction of \( f_h \). The brakes are actuated so that the resultant endpoint force matches the direction of \( f_h \). To assure this, the tangent of the linear combination of \( f_1 \) and \( f_2 \) must match the tangent of \( f_h \). This yields Eqn. 4

\[
\sin(\theta_h) = \frac{af_1 \sin(\theta_1) + bf_2 \sin(\theta_2)}{af_1 \cos(\theta_1) + bf_2 \cos(\theta_2)}\]

(4)

Solving for \( a \) and \( b \) yields Eqs. 5 & 6.

\[
a = \frac{\cos(\theta_h) \sin(\theta_2) - \sin(\theta_h) \cos(\theta_2)}{f_1(\cos(\theta_1) \sin(\theta_2) - \sin(\theta_1) \cos(\theta_2))}
\]

(5)

\[
b = \frac{\sin(\theta_h) \cos(\theta_1) - \cos(\theta_h) \sin(\theta_1)}{f_2(\cos(\theta_1) \sin(\theta_2) - \sin(\theta_1) \cos(\theta_2))}
\]

(6)

The output force will match the direction of \( f_h \), if the ratio of brake actuation, defined as variables \( V_1 \) & \( V_2 \), matches the ratio of \( a \) to \( b \). Normalizing the total actuation of brakes 1 & 2 to the magnitude of \( f_h \) from Eqn. 2 yields Eqn. 8.

\[
\frac{a}{b} = \frac{V_1}{V_2}
\]

(7)

\[
\sqrt{V_1^2 + V_2^2} = K |\hat{p}_s - \hat{p}_m| = |f_h|
\]

(8)

The following brake actuation would result:

\[
V_1 = \frac{a|f_h|}{\sqrt{a^2 + b^2}}
\]

(9)

\[
V_2 = \frac{b|f_h|}{\sqrt{a^2 + b^2}}
\]

(10)

As long as \( f_h \) lies between two brake forces, the control algorithm can produce actuation to match the direction of the calculated force. The magnitude of the output force will match \( f_h \), limited only by the output of the brakes.

4. CONTROL VERIFICATION

A testbed has been designed to produce a repeatable environment over which the control schemes can be evaluated (Black and Book, 2005). A constant force input is created using a mass attached to the handle of the master via a string and assuming a quasi-static situation. The On-Off control, the Blended control with three K values, as well as an uncontrolled case are tested. For each of the five controls, the test is run starting in 2 different positions. Each starting position test is run for three directions of input force. All of the tests are run three times so that averages can be taken. Figure 4 illustrates the workspaces of the master and the slave as well as the starting positions and applied force directions. The test is ended either when the master stops moving (the brake forces balance the input force) or when the master reaches the virtual constraint at 60% of the slave’s workspace.

The programs for the master and slave are written in LabVIEW 7.1 graphical programming language. Pre-built LabVIEW blocks, and formula nodes are used whenever possible to simplify coding. The master control program utilizes a prioritized multi-loop structure that allows the user-interface and the control to run in separate threads at different priorities. The master GUI updates at 10Hz to alleviate strain from the system. The communication loop between the master and the slave operates at 33Hz and forces the control calculations in the master to run at the same rate. In the slave the PD loop operates at 333Hz (updating the setpoint every tenth iteration) and is non-rigorously tuned for a small amount of overshoot. Tuning of the PID loop as well as the high loop rate guarantees stability of the slave, and the passive nature of the master guarantees its stability.

5. RESULTS

The tests have been run as described above. Figure 5 shows the x-position and y-position of both the master and the slave for a representative test using the Blended control with gain of \( K = 100 \). Note that the slave lags the master in the x-direction and that the constant y-position of the slave corresponds to the mapping of its 1-DOF workspace.

To begin looking at the data, it is important to identify the characteristics of the results that
Table 1. Results from described test

<table>
<thead>
<tr>
<th>Control</th>
<th>Position Difference</th>
<th>Angle Error</th>
<th>Time to Finish</th>
<th>Trials Unsatisfied</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Control</td>
<td>21.54%</td>
<td>N/A</td>
<td>1.53 sec</td>
<td>0</td>
</tr>
<tr>
<td>On-Off Control</td>
<td>3.32%</td>
<td>0.68 rad</td>
<td>11.46 sec</td>
<td>0</td>
</tr>
<tr>
<td>Blend K=10</td>
<td>19.67%</td>
<td>0.53 rad</td>
<td>1.67 sec</td>
<td>0</td>
</tr>
<tr>
<td>Blend K=100</td>
<td>6.84%</td>
<td>0.54 rad</td>
<td>7.34 sec</td>
<td>0</td>
</tr>
<tr>
<td>Blend K=500</td>
<td>1.80%</td>
<td>0.48 rad</td>
<td>N/A sec</td>
<td>18</td>
</tr>
</tbody>
</table>

Fig. 5. Representative data of master and slave position over time of test

Fig. 6. Results of tests with no control

Fig. 7. Results of tests with On-Off control

Fig. 8. Results of tests with Mixed control and a gain of $K = 10$

Fig. 9. Results of tests with Mixed control and a gain of $K = 100$

Fig. 10. Data for actuated force direction, $\theta_{error}$ is used to judge the ability of the control to produce a force that matches the direction of $f_h$. This metric allows a judgment to be made about the control's ability to reproduce the virtual spring force during the teleoperation task. Finally, the time to finish the task is calculated. If the control completely stopped the motion of the master before the end of the task, that was noted as well. Recall that the control becomes indeterminite as $v$ goes to zero.

Table 1 shows these metrics averaged over the 18 trials. The metrics ignore the effect of the slave’s contact with the virtual constraint, and in doing so more closely resemble metrics for judging path following devices. Figures 6-10 show the data for
Fig. 10. Results of tests with Mixed control and a gain of $K = 500$

the position and angle errors for all tests. Note that the time axes for each graph are different.

While On-Off control provides low position error, it increases task completion time and therefore increases user workload. The Mixed control improves the task completion time while only slightly increasing the position error. Furthermore, Blended control allows better replication of $\mathbf{f}_h$, as it provides a better ability to generate force in an arbitrary direction (demonstrated by the lower values of $\theta_{error}$). The Blended control with $K=100$ provides a balance between position tracking, angle error and speed.

6. CONCLUSIONS

Position tracking is achieved at the expense of task completion time, yielding a higher workload to the user and more dissipated energy. The results show expected trends; the Mixed control with a low gain behaves like the uncontrolled case, and the Mixed control with a high gain behaves like On-Off control due to the saturation of the brakes. The best control would match the time to completion time of an un-controlled case with very low error.

The work presented here focuses on applying classic teleoperation schemes to passive haptic teleoperation. The next phases of research include re-evaluating the way in which $\mathbf{f}_h$ is calculated. Improvements will include the addition of a viscous force component to help prevent overshoot as well as a dynamic component to improve transparency.

7. ACKNOWLEDGMENTS

The authors would like to thank National Instruments’ Academic Programs for a generous grant of hardware, software, and student funding.

REFERENCES


