Improved Multi-DOF Haptics With Spring Drive Amplifiers

Robert P. Wilson
Telerobotics Lab
Stanford University

Günter Niemeyer
Telerobotics Lab
Stanford University

Abstract—Spring drive amplifiers utilize the natural inductive properties of a DC motor to produce stronger position feedback than is achievable with traditional current amplifiers. Impedance-type haptic devices can leverage these amplifiers to render superior contacts with higher contact stiffness. Use of these amplifiers, however, requires the virtual environment to specify motion commands for each motor.

This work extends the recently developed 1-DOF integrated approach to multi-DOF applications. It presents a motion controller working with spring drive amplifiers, together producing the maximum achievable isotropic Cartesian stiffness. Isotropy is necessary to guarantee that haptic contact forces are rendered in the correct direction. The system is implemented on a PHANTOM 1.0 haptic device and analytic performance is verified by experiment.

I. INTRODUCTION

Kinesthetic force rendering on impedance-type haptic devices is characterized by excellent free space transparency, while simulation of rigid contact remains a challenge. This issue has been explored extensively and is generally attributed to the difficulty of producing high stiffness with digital springs while maintaining stability. Energy leaks caused by non-idealities in the digitization of state measurements and force commands ultimately limit achievable stiffness gains [1], [9], [3], [4].

Motivated by the importance of high stiffness over the full bandwidth of human haptic sensitivity for clear conveyance of rigidity [8], a method of stiff motion control for haptics was recently proposed[5], [10]. It uses load-compensated voltage drives for dc motors to utilize the analog inductive stiffness as a stiff, stable haptic coupling. This motion control approach requires a quasi-static virtual environment that is completely based on the PHANTOM 1.0 and concluding remarks are provided in Section VII.

II. SPRING DRIVE DESCRIPTION

In [5] the use of a voltage drive with resistive load compensation, which we will refer to as a spring drive for brevity, is shown to improve the haptic rendering of rigid contact when compared to standard current drive motor amplifiers. The key idea underlying this improvement is slowing the electrical dynamics and allowing the inductance to be utilized as a high-stiffness haptic coupling. This concept and the corresponding notation are covered briefly in this section.

The electrical dynamics of a typical brushed dc motor are

\[ e_A(t) = Ri(t) + L \frac{di(t)}{dt} + e_B(t) \]  

with coupling equations

\[ e_B(t) = k_T \theta(t) \]  

\[ \tau(t) = k_T i(t) \]  

where \( e_A \) is the applied voltage, \( e_B \) is the back-EMF, \( k_T \) is the torque/speed constant, \( \tau \) is the motor torque, \( i \) is the winding current, \( \theta \) is the rotor velocity, and \( R \) and \( L \) are the winding resistance and inductance, respectively. In order to pursue a haptic perspective on the effects of driving a motor with the spring drive, it is useful to interpret \( R \) and \( L \) as an equivalent mechanical spring \( K_L \)

\[ K_L = \frac{k_T^2}{L} \]  

and damper \( B_R \),

\[ B_R = \frac{k_T^2}{R} \]  

where \( K_L \) and \( B_R \) are connected in series. The spring drive approach couples the haptic device to the virtual environment through \( K_L \), which is available at all frequencies, inherently stable, and very stiff for small \( L \). It proposes to lock and eliminate \( B_R \) by canceling the winding resistance.

Resistance cancellation is accomplished by setting

\[ e_A = \tilde{R} i + e_W \]  

which effectively cancels the voltage drop across the winding resistance, while allowing additional voltage inputs \( e_W \). Implementation of (6) is accomplished by closing a positive current feedback loop with gain \( \tilde{R} \) around a voltage drive in analog circuitry.
Since the motor resistance $R$ varies with temperature and brush commutation, a residual uncanceled resistance
$$dR(T, \theta) = R(T, \theta) - \hat{R}, \quad (7)$$
will remain. It corresponds to the residual series damper $B_{dR} = \frac{k_T^2}{dR}$. \hspace{1cm} (8)

Fig. 1 shows the mapping of winding inductance $L$ and residual resistance $dR$ into the mechanical domain, leaving $\hat{R}$ to be canceled by (6). By the electromechanical coupling (2), the intermediate voltages $e_W$ and $e_S$ relate to the velocities $\dot{\theta}_W$ and $\dot{\theta}_S$, respectively; where $e_W$ is the applied node voltage between the series resistances $\tilde{R}$ and $dR$, $e_S$ is the node voltage between $R$ and $L$, $\dot{\theta}_W$ is the velocity of the setpoint of the mechanical equivalent series spring-damper, and $\dot{\theta}_S$ is the velocity of the connection point between the series spring and damper. With the inductance and residual resistance interpreted in the physical domain, voltage commands to the spring drive are equivalent to velocity commands to the set-point of the series spring-residual damper
$$e_W = k_T \dot{\theta}_W. \quad (9)$$

Therefore, the spring drive operates the dc motor as an approximate motion source. This is in contrast with the current drive motor amplifiers typically used in haptic devices, which speed up the electrical dynamics and operate the motor as a torque source. By recasting the motor as a motion source, it becomes necessary to design the digital controller as a motion controller, again in contrast to the impedance control typically paired with the current drive.

III. MID-LEVEL DRIFT COMPENSATION

Before any high-level controller can be successfully implemented, the fact that a residual damper $B_{dR}$ exists between the commanded location $\theta_W$ and the output must be addressed. Without compensation, $B_{dR}$ will allow $\dot{\theta}_S$ and $\theta$ to drift unboundedly under low frequency external loads. In haptic applications this drift will degrade the simulation by making sustained rigid contacts feel like dampers. Therefore, position feedback with PD compensation is added digitally to combat drift and yields the control law
$$e_W = k_T \dot{\theta}_S + K_D (\dot{\theta}_S - \dot{\theta}) + K_P (\theta - \theta), \quad (10)$$
where $K_P$ and $K_D$ are the error gains. We assume a first order filter with cutoff frequency $\lambda$ on the differentiated velocity signal $\dot{\theta}$. This constitutes a mid-level controller interface between the analog inductive stiffness of the spring drive and the high-level motion controller described in the next section.

Implementation of this mid-level controller establishes a hybrid coupling between the user and the virtual environment consisting of the physical motor dynamics, analog electronics modifying those dynamics, and digital drift compensation. The combined physical and analog components comprise the series spring-damper shown in Fig. 1, while the digital component consists of the spring damper of (10) between $\theta$ and $\theta - \dot{\theta}$. Functionally, the analog stiffness $K_L$ dominates for high frequency deflections, while the digital mid-level controller provides low to mid frequency stiffness. The frequency-dependent stiffness transfer function for the complete coupling is found by substituting (10) into the resistance-canceled motor dynamics
$$e_W(t) = dR \dot{\theta}_W(t) + \frac{d\dot{\theta}_S(t)}{dt} + e_B(t) \quad (11)$$
and is
$$K_B(s) = \frac{\tau}{\dot{\theta}}(s) = - \frac{k_T}{s + \frac{dR}{s + \lambda}} \left[ \frac{((k_T + K_D + \frac{K_P}{\lambda} + \frac{k_T}{\lambda}) s + K_P}{\lambda} \right]. \quad (12)$$

For a detailed derivation and analysis of this result, see [11]. The magnitude Bode plot of (12) shown in Fig. 2 illustrates the combination of several factors that shape stiffness over frequency. Starting from the natural motor dynamics represented by the dotted curve, the spring drive lowers the corner frequency from $R/L$ to $dR/L$. The proportional term in the mid-level controller then pulls the low frequency rolloff caused by $B_{dR}$ up to a dc stiffness
$$K_{DC} = \frac{k_T K_P}{dR}. \quad (13)$$
while the derivative term boosts stiffness at midrange frequencies before the filter rolls it off to the high frequency stiffness
$$K_{HF} = K_L = \frac{k_T^2}{L}. \quad (14)$$

While not discussed here in detail it is worth noting that, $K_L$ may be increased by electrically reducing $L$ in addition to $R$. This can be accomplished by replacing the positive feedback gain $\hat{R}$ in (6) with lead compensation as described in [6].
A slight lag in filtering the velocity may cause the system to create slight non-zero forces, equivalent to an added mass. Given the low bandwidth of human actions compared to the speed of filtering, however, any added mass tends to fall far below perceptible levels.

Finally, compliant contact is achieved by setting the desired motion to a fraction of the user motion. The full dynamic range of the VE can be represented by

$$\dot{\theta}_{Sd} = \eta \dot{\theta} \quad \theta_{Sd} = \eta \theta + (1 - \eta) \theta_0,$$

where $0 \leq \eta \leq 1$, and $\eta = 1$ and $\eta = 1$ correspond to rigid contact and freespace, respectively. The low-frequency output stiffness is reduced by the factor $\eta$.

As a motion-based VE, no absolute force values are available, either explicitly via measurement or implicitly by commanding motor current. Output stiffneses, therefore, may no longer be programmed exactly, but must be specified as a fraction of the device's maximum achievable stiffness. Knowledge of $K_\theta(s)$, particularly $K_{DC}$, can be used to estimate values. Additional discussion of this high-level controller may be found in [11].

V. MOTION CONTROL OF HAPTIC DEVICES: MULTI-DOF

In extending the above approach to multi-DOF haptics, we recognize the application's requirements. Though we wish to display the maximum achievable stiffness, force directions need to be rendered accurately to convey proper geometric surface properties. For example, forces should always fall perpendicular to frictionless surfaces. To support this requirement, perceived output stiffnesses need to be spatially uniform.

Given an n-DOF device, one drift-compensated spring motor is used to drive each joint motor. Thus, each joint independently replicates the one-DOF system described in Section III, acting as a joint motion source with a stiffness described by (12). We assume here that all joints exhibit the same stiffness, though relative scaling between joints could be incorporated if necessary. We define a desired Cartesian location $\bar{x}_{Sd}$ and velocity $\dot{\bar{x}}_{Sd}$, and collect the individual joint values into a joint position vector $\bar{q}$ and velocity vector $\dot{\bar{q}}$ with equivalent desired position $\bar{q}_{Sd}$ and velocity $\dot{\bar{q}}_{Sd}$.

$$\bar{q}_{Sd} = \text{invkin}(\bar{x}_{Sd})$$

$$\dot{\bar{q}}_{Sd} = \bar{J}^{-1}\dot{\bar{x}}_{Sd}$$

obtained by inverting the mechanism's kinematics. This will produce joint torques

$$\bar{\tau} = K_q \Delta \bar{q}$$

via a diagonal joint stiffness matrix

$$K_q = K_\theta(s)I,$$

which maps to a Cartesian stiffness of

$$K_x^{\text{inv}} = \bar{J}^{-T}K_q\bar{J}^{-1} = K_\theta(s)(JJ^T)^{-1}$$

resembling that

$$\bar{F} = K_x^{\text{inv}}\Delta \bar{x}$$

$$\bar{\tau} = \bar{J}^T\bar{F}$$

$$\Delta \bar{x} \approx \bar{J}\Delta \bar{q}.$$
We denote the stiffness matrix with 'inv' to indicate the joint values were set by an inverse Jacobian kinematic algorithm.

Unfortunately, this Cartesian stiffness is non-spherical such that forces are not necessarily generated along displacement vectors. Fig. 3 depicts such a stiffness ellipsoid for a simple 2-DOF case. Proper haptic rendering of force requires that the multi-DOF motion controller reshapè this ellipsoid into a sphere.

We instead propose to calculate

\[ \dot{q}_{sd} = \dot{q} + \alpha J^T (\ddot{x}_{sd} - \dot{x}) \]  

(23)

and approximate the set-point derivative as

\[ \ddot{q}_{sd} = \dot{q} + \alpha J^T (\ddot{x}_{sd} - \dot{x}), \]  

(24)

where we ignore the Jacobian's derivative. By construction this simplification only affects forces rendered at high velocities and high frequencies with minimal impact on user perception as discussed at the end of this section. This delivers a Cartesian force \( \ddot{F} \) of

\[ \ddot{F} = J^{-T} K_q \alpha J^T \Delta \ddot{x} \]  

(25)

and the Cartesian stiffness

\[ K_x^{\text{trans}} = \alpha J^{-T} K_q J^T. \]  

(26)

Since \( K_q \) is the scaled identity matrix, \( K_x^{\text{trans}} \) collapses to the diagonal

\[ K_x^{\text{trans}} = \alpha K_\phi(s) I. \]  

(27)

Effectively, the transpose algorithm (23) has reshaped the ellipsoid into a sphere and restored the force directions to parallel any deflections.

Having resolved the force direction problem, there is now the issue of what value to select for \( \alpha \). To address this, first recall that each joint is tuned for a maximum joint stiffness \( K_\phi(s) \). Also note that locking each joint results in the Cartesian stiffness \( K_x^{\text{inv}} \). Thus, \( K_x^{\text{inv}} \) represents the maximum achievable Cartesian stiffness for a given configuration, limited by the stability of each joint. Using the Jacobian transpose solution to the inverse kinematics has reshaped \( K_x^{\text{inv}} \) into the Cartesian stiffness ball \( K_x^{\text{trans}} \), which must now be appropriately scaled by \( \alpha \). Intuitively, \( K_x^{\text{trans}} \) cannot be scaled arbitrarily large, as this would allow the effective Cartesian stiffness to be increased without bound. Indeed, if the magnitude of \( K_x^{\text{trans}} \) exceeds that of \( K_x^{\text{inv}} \) in any direction, we would be asking for a stiffness beyond the stable maximum. Doing this would effectively increase the gains of one or more joint controllers and compromise their stability. Therefore, taking \( K_x^{\text{inv}} \) as the upper bound to maintain stability, the eigenvalues of \( K_x^{\text{trans}} \) must satisfy

\[ \lambda_{\text{max}}(K_x^{\text{trans}}) \leq \lambda_{\text{min}}(K_x^{\text{inv}}). \]  

(28)

Substituting (27) and (21), this condition simplifies to

\[ \alpha \leq \lambda_{\text{min}}[(J J^T)^{-1}], \]  

(29)

where \( \lambda_{\text{min}}[(J J^T)^{-1}] \) may be computed on the fly to maximize \( K_x^{\text{trans}} \) independently for each configuration, or computed over the entire workspace offline to select the global minimum for a consistent \( K_x^{\text{trans}} \) at all configurations.

Fig. 3 illustrates the ellipses \( K_x^{\text{inv}} \) and \( K_x^{\text{trans}} \) with \( \alpha = \lambda_{\text{min}}[(J J^T)^{-1}] \) for a PHANTOM constrained to 2-DOF by locking its base motor. This configuration will be revisited experimentally in the next section.

The stiffness shape compensation performed by this multi-DOF VE does not have infinite bandwidth. As a result \( K_x^{\text{inv}} \) will still exist at high frequencies, and impulsive force vectors may be directed incorrectly. Since these direction discrepancies exist only at high frequency, however, the user will likely be unable to detect them kinesthetically, due to a low perceptual bandwidth of 20 Hz to 30 Hz [7]. Tactile detection of the discrepancy may fare better, but [2] indicates that the direction discrimination threshold is only about 25°, determined for perception of low frequency forces. Informal experience with a multi-DOF implementation on a PHANTOM 1.0 suggests that these discrepancies are not perceptible, and certainly not disruptive to the haptic simulation, however, further investigation of human sensitivity to impulse force direction may be interesting.

Similar to the one-DOF case, compliant surfaces and free space may be obtained by setting

\[ \ddot{x}_{sd} = \eta \ddot{x} \]

(30)

with \( 0 \leq \eta \leq 1 \).

VI. EXPERIMENTAL IMPLEMENTATION

The multi-dof motion control approach described above was implemented on a PHANTOM 1.0 haptic device in order to confirm the generation of correct force vectors. Evaluation of force direction was accomplished by mounting a six-axis force sensor to the end effector of the PHANTOM and then fixing both the PHANTOM base and the sen-
sor to ground. This effectively locked the PHANTOM into a fixed configuration, holding all configuration-dependent variables constant. Fig. 4 is a photo of the setup, which features the same configuration as shown in Fig. 3. With the PHANTOM locked, virtual surfaces were then specified so that the device was already in a penetration state and the resulting force direction was recorded.

The results in Fig. 5 compare the realized reaction forces under both the Jacobian inverse and proposed Jacobian transpose control for a horizontal wall placed above the device. Here, the device is represented by the black circle, and the dotted line indicates the correct reaction force vector. As expected, the Jacobian transpose produces an accurately directed force while Jacobian inverse control exhibits significant deviation.

Fig. 6 reflects the results of repeating this experiment for additional virtual surfaces. These surfaces are specified tangentially around the unit circle in the 2-D plane of Fig. 5 with a radial spacing of 15°. The plot displays the angular error between the measured and correct force directions in each case. As expected, Jacobian transpose control provides low errors. Additionally, it is not well correlated with the angle of the surface normal, which suggests the corresponding Cartesian stiffness is circular. Jacobian inverse control, on the other hand, produces large errors that are strongly and periodically correlated with the normal direction, indicative of an elliptical Cartesian stiffness.

VII. CONCLUSION

Generalization of one-DOF motion control of impedance-type haptic devices to multiple degrees of freedom is achieved through the use of Jacobian transpose control. The system acts to regulate the Cartesian stiffness to a uniform sphere. The resulting haptic simulation produces accurately directed force vectors in response to motion constraints, while taking advantage of the high analog stiffness of load-compensated voltage motor drivers.

This spring drive approach contrasts with traditional haptic impedance control by setting a maximum stiffness in the low-level actuator and using high-level control to scale the rendered stiffness between freespace and this maximum. While the loss of force calculations in the virtual environment prevents the coding of absolute virtual stiffnesses, this limitation is traded off by the improved perceptual performance.

We hope extending the spring drive’s superior performance
to a commercial multi-DOF haptic device will enhance future haptic applications and improve the user’s experiences.

VIII. ACKNOWLEDGMENTS

This material is based upon work supported under a National Science Foundation Graduate Research Fellowship.

REFERENCES


