5 Conclusions

We developed a unified actuator model capable of describing both the SPM and the finger joint. This allows the same equations to apply to both devices. We analyzed the dynamics of the SPM and showed how the finger joint is just a special case of an SPM with constraints. We implemented a PID controller for the SPM and finger joint, and a number of algorithms to exploit the benefits of closed loop control.

References

[1] B. B. Bederson, R. S. Wallace, and E. L. Schwartz, *Two miniature pantilt devices*, in IEEE International Conference on Robotics and Automation, May 1992.

[2] ——, *Control and design of the spherical pointing motor*, in 1993 IEEE International Conference on Robotics and Automation, May 1993.

[3] R. S. Wallace, *Miniature direct drive rotary actuators*, in 1993 International Symposium on Robotics Research, October 1993.

[4] ——, *Miniature direct drive rotary actuators*, Robotics and Autonomous Systems, 48 (1993), p. 76.

3.2 Finger Joint

Since the finger joint does not have an A coil, there is no steady state current. The finger joint likewise does not pan or have a C coil, so no C coil control current is necessary. Hence we propose a simpler PID control current for the finger joint:

$$i_B = b_1(\theta - \theta_0) + b_2 \int (\theta - \theta_0)dt + b_3\theta'$$

4 Experiments

We performed experiments using the SPM and finger joint designed to test the capabilities and versatility of the devices as well as finding optimal control constants for the PID controller.

For brevity, we omit the dicussion of feedback experiments using the SPM, and focus instead on the simpler case of the finger joint. The maximum average velocity measured so far with the joint is about 2800°/ *sec*, with accelerations in the range of 2800°/ sec^2. The rotor inertia is approximately 100 $gm - cm^2$.

Figure 5 illustrates the effect of controlling the finger joint with a PD and a PID controller. The data suggest that up to 20 point-to-point moves per second, with less than 0.03° error are possible.

4.1 Record and Playback Trajectories

We implemented a form of teach-move algorithm. First the user grasps the finger joint rotor and rotates it in an arbitrary manner while the microcontroller records the sequence positions. Then the user's movements can be played back by using the recorded sequence of positions to adjust the set point for the PID controller.

4.2 Tactile Textures

The user moving the rotor can be given the impression of different tactile textures by adjusting the forces on the limb based on the limb's position. A simple "notched" texture can be obtained by dividing the rotor's range into several discrete bins. The set point is then dynamically set to always be the center of the bin nearest to the current location of the rotor. The tactile sensation experience as the user pushes the limb toward a bin boundary is of increased resistence followed by a pull in the opposite direction as the boundary is crossed. The tactile sensation is similar to flipping a light switch.

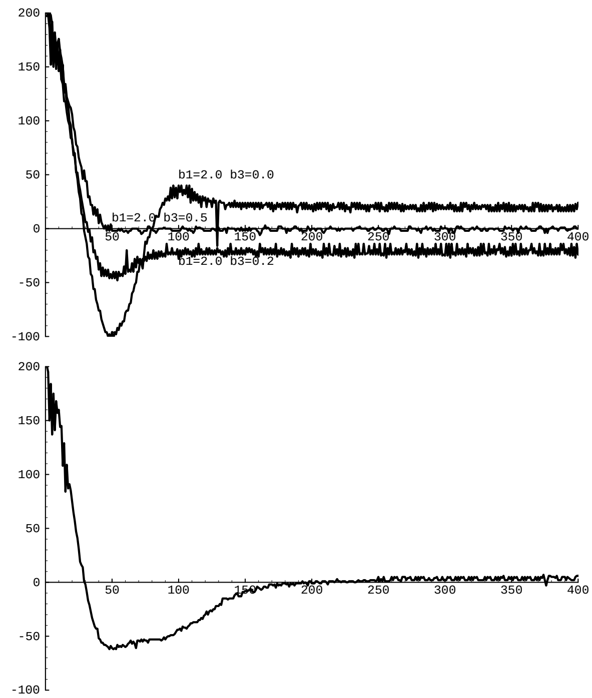

Figure 5: Plot shows finger joint angle error as a function of time in response to a series of step changes in set point. The vertical axis is sensor measurements and there are approximately 33 sensor units per degree. Top: values of b1 and b3 for three PD controllers. In all cases, the actuator converged to withing one degree of the desired position. In the best case, convergence occured within 50 ms. Bottom: Including an integral term, the PID controller reaches the desired rotor angle with 0.03 degrees, but converges somewhat more slowly.

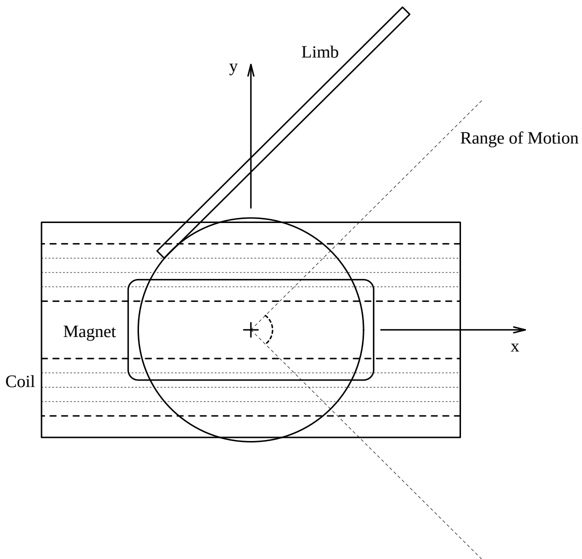

Figure 4: finger joint shown in home position with coordinate system. The finger joint is free to rotate in the xy plane between -45 and 45 degrees from the home position.

2.2 Finger Joint

The finger joint is essentially a special case of an SPM where the A and C coils are removed and the finger joint limb is restricted to rotate only in the θ or tilt direction. This is equivalent to restricting $\phi = 0$.

Reevaluating the limb direction restricting $\phi = 0$ yields

$$\begin{aligned} (P_x, P_y, P_z) &= (1,0,0) \cdot [\mathrm{rot}_Z(\theta)] \\ &= (\cos\theta, \sin\theta, 0) \end{aligned}$$

Similarly by reevaluating q, v, and p, we find that the proposed (q, v, p) coordinates are unnecessary for describing the finger joint position since

$$\begin{aligned} q &= \theta \\ v &= \pi/2 - \theta \\ p &= \pi/2 \end{aligned}$$

To find the differential equations for the finger joint, we take note of the fact that the finger joint does not have an A or C coil (hence $i_A = i_C = 0$) and that we may simplify the equations based on the assumption that $\phi = 0$. These simplifications allow the three equations to be combined into one equation:

$$\theta'' = (K/I) i_B \cos\theta.$$

3 Feedback Control

3.1 SPM

We now attempt to address the question of what current should be applied to the coils in order to achieve a certain desired limb position? First we look to the steady state solution where we ignore transitory response. This is equivalent to asking: for what coil currents is the overall torque on the limb equal to zero at the desired limb position? In other words, given a desired limb position (q_0, v_0, p_0), for what currents (i_A, i_B, i_C) is $\boldsymbol{\tau} = 0$? By solving for $\boldsymbol{\tau} = 0$, the following two relations are found:

$$\begin{aligned} i_B &= \frac{\cos v_0}{\cos q_0} i_A \\ i_C &= \frac{\cos p_0}{\cos q_0} i_A \end{aligned}$$

Now we must consider the dynamic response of the SPM. Given the current position of the limb (θ, ϕ), move it to the desired target position (θ_0, ϕ_0). A simple though naive solution would be to use the steady state solution to find the currents needed to reach the target position and then to apply this current to the coils. This solution does not work particularly well since when the limb moves from the initial position to the target position, the limb behaves like a pendulum, and instead of stopping at the target position, it oscillates around that position until friction forces finally bring the limb to a halt.

A better control force would take into account that as the limb moves toward the target position, it picks up momentum that must be negated as the limb approaches the target position so that the limb does not overshoot. One of the best-known controllers used in practice is the PID controller.

We propose the following PID functions for our control currents:

$$i_B = b_1(v - v_0) + b_2 \int (v - v_0) dt + b_3 v' + \frac{\cos v_0}{\cos q_0} i_A$$

$$i_C = c_1(p - p_0) + c_2 \int (p - p_0) dt + c_3 p' + \frac{\cos p_0}{\cos q_0} i_A$$

This formulation has a number of desirable characteristics. For example, it is linear and therefore easy to compute and handle analytically. Also the angular deviation and velocity terms are conducive to linearization assumptions as they will decay to zero as the limb approaches the target position.

Each of the proposed functions for i_B and i_C contains three constants. The relations among these constants will determine the effective speed and accuracy of the device.

limb. In this coordinate system the position of the limb is described by specifying the angle that the limb makes with each of the coordinate axes.

To convert the limb position to this coordinate system, we use the the dot product to find the cosine of the angle between the limb and each axis. For example, to find the cosine of the angle between the limb and the x axis, we take the dot product of the x axis (1,0,0) and the limb position $(\cos\theta\cos\phi, \sin\theta, -\cos\theta\sin\phi)$. Similarly for the y and z axis angles. We call the deflection angles from the x, y, and z axes q, v, and p respectively. This coordinate system allows us to express the limb position (P_x, P_y, P_z) as $(\cos q, \cos v, \cos p)$ where

$$(\cos q, \cos v, \cos p) = (\cos\theta\cos\phi, \sin\theta, -\cos\theta\sin\phi)$$

The torque on the limb arises from the basic electromagnetic principle that a rotating magnet inside a magnetic field experiences a torque that will tend to twist the magnet into alignment with the field. Given a coil with magnetic dipole moment $\boldsymbol{\mu}$ and a permanent magnet with field $\mathbf{B}$, the torque $\boldsymbol{\tau}$ exerted on the magnet is the cross product

$$\boldsymbol{\tau} = \mathbf{B} \times \boldsymbol{\mu}$$

where $\boldsymbol{\mu}$ is the magnetic dipole moment of the coil having direction perpendicular to the plane of the coil and magnitude

$$|\boldsymbol{\mu}| = NiA$$

where N is the number of windings in the coil, i is the current in the wire, and A is the area enclosed by the coil.

Let us examine $\boldsymbol{\tau}_A$ the torque due to coil A. The orientation of the limb (and hence the permanent magnet) is $(\cos q, \cos v, \cos p)$, and coil A has magnetic dipole moment $\boldsymbol{\mu} = N_A i_A A_A(1,0,0)$. We use here the constant $K = N_A A_A = N_B A_B = N_C A_C$.

$$\begin{aligned} \boldsymbol{\tau}_A &= K i_A(\cos q, \cos v, \cos p) \times (1,0,0) \\ &= K i_A(0, \cos p, -\cos v) \end{aligned}$$

We calculate the torques due to coils B and C similarly.

$$\boldsymbol{\tau}_B = K i_B(-\cos p, 0, \cos q)$$

$$\boldsymbol{\tau}_C = K i_C(\cos v, -\cos q, 0)$$

We can now write the total torque on the limb due to the coils as

$$\boldsymbol{\tau} = \boldsymbol{\tau}_A + \boldsymbol{\tau}_B + \boldsymbol{\tau}_C$$

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} -Ki_B\cos p + Ki_C\cos v \\ Ki_A\cos p - Ki_C\cos q \\ -Ki_A\cos v + Ki_B\cos q \end{bmatrix} \qquad (1)$$

Torque is defined as a function of the second derivative with respect to time of the angle around the axis of rotation. In our model, we have chosen the axes of rotation to be the x, y, and z axes. However, we have not defined variables to represent the angles around the axes. Hence there is no simple relation between the torque axes and the second derivatives of our angles q, v, and p. We must find the torque axes that correspond to each of our variables in order to express the torques in terms of second derivatives of q, v, and p.

We have defined v as the angle between the limb and the y axis. Since torque affecting v moves the limb within the plane defined by the limb and the y axis, the direction of the torque vector affecting v is perpendicular to this plane. The direction of this vector can be found by taking the cross product between the y axis $(0,1,0)$ and the limb position $\mathbf{P}$. If we take the torque vector $\boldsymbol{\tau} = (\tau_x, \tau_y, \tau_z)$ from (1) and project it onto this direction, the result will be $\boldsymbol{\tau}_v$ the torque vector affecting v:

$$\boldsymbol{\tau}_v = \left(\frac{(0,1,0)\times\mathbf{P}}{|(0,1,0)\times\mathbf{P}|}\cdot\boldsymbol{\tau}\right)\frac{(0,1,0)\times\mathbf{P}}{|(0,1,0)\times\mathbf{P}|}$$

We perform similar operations to find $\boldsymbol{\tau}_q$ and $\boldsymbol{\tau}_p$ to obtain the results:

$$\boldsymbol{\tau}_q = \left(-\frac{\cos p}{\sin q}\tau_y + \frac{\cos v}{\sin q}\tau_z\right)\left(0, -\frac{\cos p}{\sin q}, \frac{\cos v}{\sin q}\right)$$

$$\boldsymbol{\tau}_v = \left(\frac{\cos p}{\sin v}\tau_x - \frac{\cos q}{\sin v}\tau_z\right)\left(\frac{\cos p}{\sin v}, 0, -\frac{\cos q}{\sin v}\right)$$

$$\boldsymbol{\tau}_p = \left(-\frac{\cos v}{\sin p}\tau_x + \frac{\cos q}{\sin p}\tau_y\right)\left(-\frac{\cos v}{\sin p}, \frac{\cos q}{\sin p}, 0\right)$$

Then we use our expressions from (1) for τ_x, τ_y, and τ_z to solve for the magnitudes of $\boldsymbol{\tau}_q$, $\boldsymbol{\tau}_v$, and $\boldsymbol{\tau}_p$. We can now use the fact that $\tau_q = Iq''$, $\tau_v = Iv''$, and $\tau_p = Ip''$, where I is the moment of inertia of the limb. This allows the equations to be stated fully in terms of q, v and p as

$$q'' = (K/I)((i_C\cos p + i_B\cos v)\cot q - i_A\sin q)$$

$$v'' = (K/I)((i_C\cos p + i_A\cos q)\cot v - i_B\sin v)$$

$$p'' = (K/I)((i_B\cos v + i_A\cos q)\cot p - i_C\sin p)$$

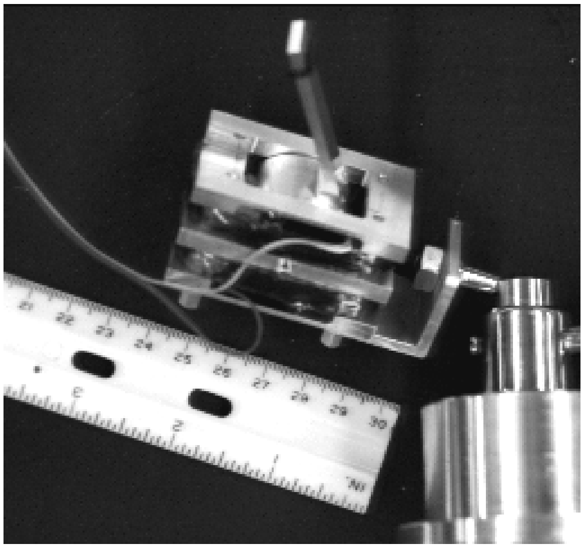

Figure 2: The direct drive finger joint has a 90 degree workspace. The finger joint rotor contains a cylindrical permanent magnet which is rotated by applying currents to the surrounding coils.

ance, holding two Helmholtz coil pairs. The gimbal is contructed with high precision ball bearings. On the backplate of the SPM are four Hall effect sensors, the A/D converter and the driver chip. In a companion paper, we present further details about the construction of the SPM[3].

1.2 Finger Joint

The finger joint (see figure 2) rotor consists of a cylindrical magnet holder and a link arm. The rotor magnet is oriented so that the maximum torque occurs when the finger limb is positioned exactly at 45 degrees. The backplate of the SPM includes two Hall effect sensors, the A/D converter and the current driver chip. Further details about the construction of the finger joint can be found in [3].

2 Actuator Dynamics

We define coordinate systems for the SPM and finger joint that allow convenient expression of the limb's position and of the torques on the limb. See figure 3.

2.1 SPM

We define a right handed coordinate system with the origin lying at the common intersection point of the rotor's two axes of rotation. The rotor limb is free to pivot around this point with two degrees of freedom. The coordinate system is aligned so that the z axis corresponds to the tilt axis and the y axis matches the pan axis, and the x axis extends along the limb in it's home position.

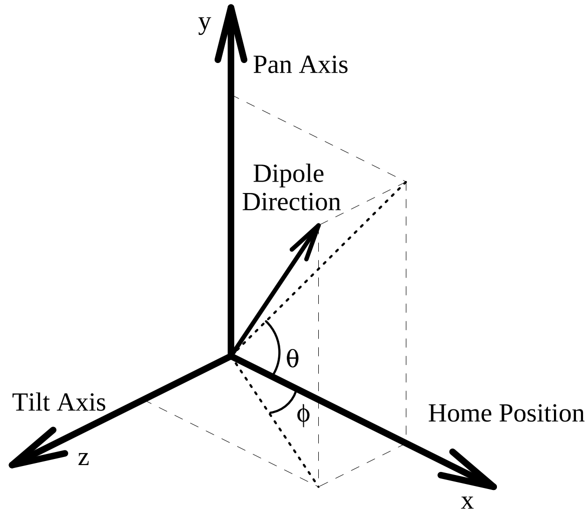

Figure 3: SPM and finger joint coordinate system showing tilt and pan angle.

Surrounding the rotor limb are three sets of coils (or two sets of coils and one permanent magnet). The "A" coils or "home position" coils wrap around the x axis and lie in the yz plane, the "B" coils or "tilt" coils wrap around the y axis and lie in the xz plane, and the "C" coils or "pan" coils wrap around the z axis and lies in the xy plane. The A coils may be optionally replaced by a permanent magnet.

We describe the position of the rotor limb by describing the direction the limb is pointing in spherical coordinates. In the home position, the direction of the limb is (1,0,0). We define the tilt angle, θ, as clockwise rotation of the limb around the z axis and the pan angle, ϕ, as counterclockwise rotation of the limb around the y axis. We can express the limb direction in our Cartesian space by rotating the home position vector (1,0,0) first by θ around the z axis and then by ϕ around the y axis. The limb direction (P_x, P_y, P_z) in terms of θ and ϕ is

$$\begin{aligned}(P_x, P_y, P_z) &= (1,0,0)\cdot[\mathrm{rot}_Z(\theta)]\cdot[\mathrm{rot}_Y(\phi)] \\ &= (\cos\theta\cos\phi, \sin\theta, -\cos\theta\sin\phi)\end{aligned}$$

This spherical coordinate system is convenient for specifying the limb position, however, it will become helpful to introduce an additional coordinate system to simplify the equations describing the motion of the

Feedback Control of Miniature Direct Drive Devices*

David Max Richard S. Wallace

Courant Institute of
Mathematical Sciences
New York University
New York, NY

Abstract

We discuss dynamics and control of miniature direct drive actuators, specifically for the two axis spherical pointing motor and for the one axis direct drive finger joint. These actuators can move both accurately and at high speed, however the capabilities of these devices cannot be fully exploited using open loop control techniques. We derive an ideal PID feedback control scheme. Our initial experiments indicate that PID feedback control for the SPM and finger joint is highly feasible and a significant improvement over open loop methods.

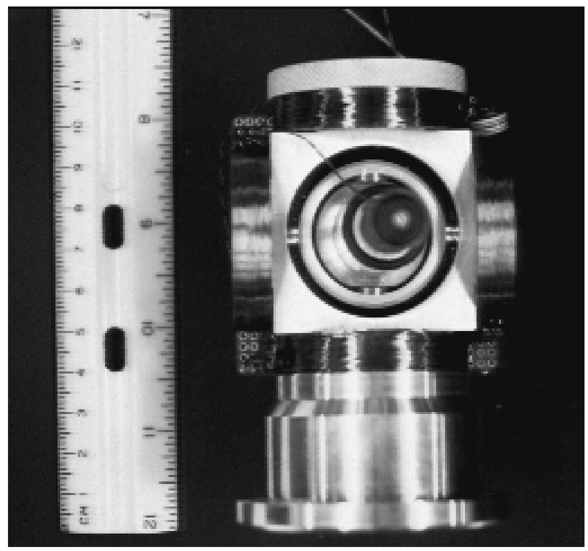

Figure 1: The Spherical Pointing Motor is machined to a high tolerance and has a 75 degree conical workspace. The rotor limb contains a cylindrical permanent magnet which is moved by applying currents to the horizontal and vertical sets of coils.

1 Introduction

Closed loop feedback control of the direct drive finger joint (finger joint) spherical pointing motor (SPM) is necessary in order to realize the full potential speed and accuracy of these devices. This paper reports our early results on using Hall effect sensors to measure the angular position of the finger joint and SPM rotor and also to infer the instantaneous velocity and acceleration measurements necessary for a PID controller. Detailed descriptions of the actuators as well as historical background material and full citations may be found in [1] [2] [3] [4].

In our devices, we use Hall effect sensors to sense the position of the rotor magnet. Computer control of the devices is via the Motorola M68332 microcontroller. A 12 bit serial A/D converter (MAX188) is used to read the Hall sensors. Currents are applied to the coils in the form of Pulse Width Modulation (PWM) signals passed through an integrated power driver chip (L293).

1.1 Spherical Pointing Motor (SPM)

The SPM (see figure 1) rotor limb is a cylindrical magnet rotating on as gimbal in such a way that the magnet axis pans and tilts around a central point of rotation. Our high precision SPM has a machined aluminum frame made to one thousandth inch toler-

* This report is published as New York University Courant Institute Computer Science Technical Report No. XXX. This research was supported in part by a grant from the National Science Foundation, number CDA-9018673, and by a grant from the NYU Arts and Science Technology Transfer Fund, and by an equipment donations from *Motorola Advanced Microcontroller Division*. Please address correspondence to Richard S. Wallace, Courant Institute of Mathematical Sciences, New York University, 251 Mercer St., New York, NY 10012. rsw@cs.nyu.edu