

DETC2004-57520

AUTOMATED MODELING OF NONLINEAR MECHANISMS FOR VIRTUAL PROTOTYPING

Mark B. Colton

Department of Mechanical Engineering
University of Utah
Salt Lake City, Utah
colton@eng.utah.edu

John M. Hollerbach

School of Computing
University of Utah
Salt Lake City, Utah
jmh@cs.utah.edu

ABSTRACT

This paper addresses the problem of obtaining models of nonlinear mechanisms for use in virtual prototyping systems. The objective of the described modeling procedure is to more accurately represent the feel of complex mechanisms in force feedback applications. The model structure selected for this study is a 2nd-order impedance model, with mass, damping, stiffness, and coulomb friction that vary with position. An automated testbed system is described, as well as a segmented least-squares algorithm for estimating position-varying model parameters. Baseline results are obtained for a system comprised of a mass, dashpot, and nonlinear (hardening) spring.

1 INTRODUCTION

1.1 Motivation

Virtual prototyping allows engineers to design and test devices on the computer screen, without building physical prototypes that are costly in terms of both time and money. Recent research has focused on adding a sense of touch to virtual prototyping systems, thus allowing designers to feel, manipulate, and evaluate designs as a step in the design process (Hollerbach et al., 1996). This sense of touch is provided by a haptic interface, a robotic device that is manipulated by and provides force feedback to the user, the forces for which are calculated from the user's contact with the virtual device or mechanism (Nahvi et al., 1998). An example of this type of interaction is illustrated in Figure 1, in which a user manipulates and feels a virtual assembly using the Sarcos Dextrous Arm Master, a ten degree of freedom hydraulic exoskeleton (Jacobsen et al., 1990).

The advantages of haptic interaction with virtual mechanisms are numerous. For example, testing the feel (force-position profile) of a manual gearshift or turn signal switch, as well as their placement in the automobile, before constructing a physical mockup of the automobile, would greatly aid designers in making improvements to both the mechanical design of the gearshift or switch and the layout of the car interior.

The current approach to modeling the feel of complicated mechanisms is to use *a priori* engineering models to determine the relationship between how the mechanism is manipulated and the interaction forces generated. This approach is sometimes adequate for simple devices with well-understood dynamic forces. For more complicated mechanisms (nonlinear, with friction, backlash, etc.), there is a need for an experimental approach to obtaining the interaction models (Hollerbach, 2000). The model obtained from measurements on a physical device or mechanism could then be used to improve an existing virtual prototype by providing more realistic simulation forces, which would in turn improve the design process. The purpose of this paper is to propose an approach to obtaining

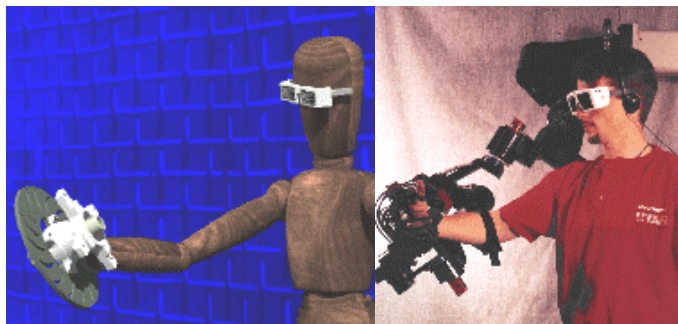


Figure 1: Haptic manipulation of a virtual assembly

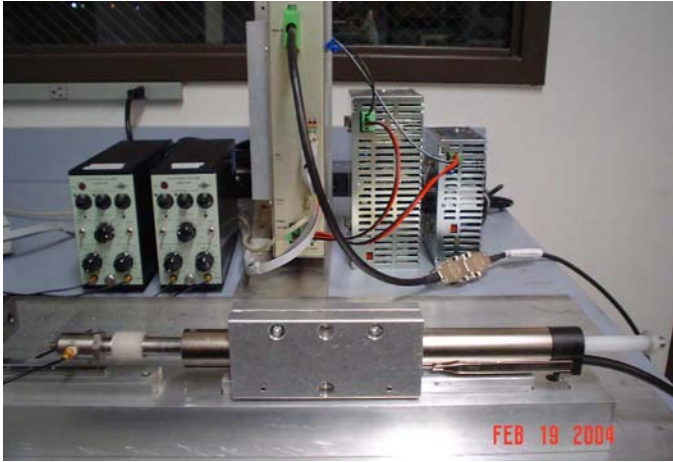


Figure 2: Photo of modeling testbed (controller not shown)

experimental models of mechanisms that can then be used for haptic simulation of virtual prototypes.

1.2 Related Work

Other researchers have investigated the idea of using measurements of physical systems to improve models for haptic simulations, although most have not focused on the problem of modeling general nonlinear mechanisms. Okamura et al. (2001) demonstrated using an instrumented probe to obtain vibration models of physical systems for haptic playback. Pai (2000) describes a facility capable of measuring shape, reflectance, deformation, and sound of physical devices. Measurements of surface texture using a wireless instrumented stylus have been

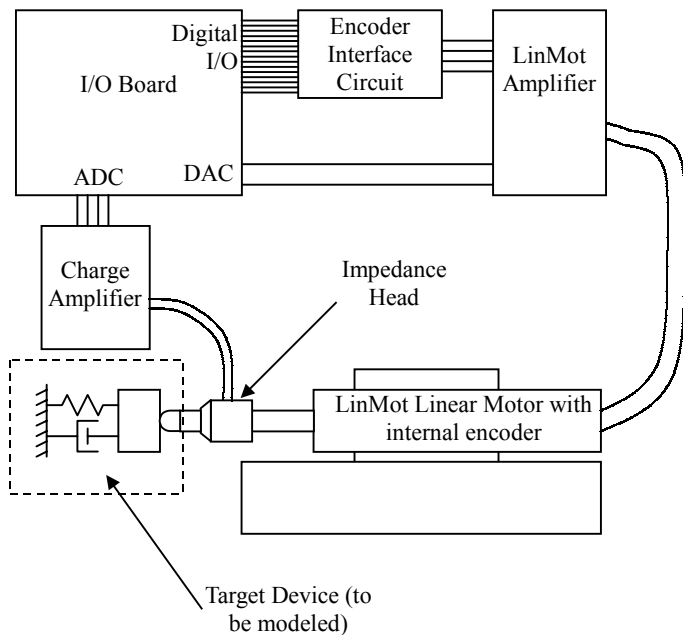


Figure 3: Schematic of modeling testbed

reported by Pai et al. (2003). MacLean et al. (1995) described an experimental setup used to obtain impedance models of physical mechanisms, and MacLean (1996) used this system to experimentally determine zero-order piecewise models (stiffness only) to represent the feel of common switches. More recently, Weir et al. (2004) described a method for characterizing the feel of switches based on physical measurements, but without obtaining parameterized models.

1.3 Approach

The objective of the research presented in this paper is to develop a method for obtaining models of nonlinear mechanisms that will be suitable for use in virtual prototyping haptic playback applications. The proposed approach is to use a generalized nonlinear mass/spring/damper/friction (2nd-order) model to approximate the feel of complex nonlinear mechanisms. A segmented least-squares algorithm is used to estimate the model parameters from measurements made using a testbed developed specifically for this purpose.

2 AN AUTOMATED MODELING TESTBED

2.1 Overview

Our ability to obtain nonlinear dynamic models of mechanical devices depends strongly on the capabilities of the system used to actuate and make measurements of the target device. The actuator must be capable of applying input signals of sufficient force, bandwidth, and range of motion to excite the dynamics of the target device, while the system's sensors must be capable of accurately measuring the response of the target device to the input signals. This requires that sensors have high bandwidth, linearity, and signal-to-noise ratio (SNR). The computer control system must be capable of operating at a rate high enough to maintain stable and accurate control of the actuator and sample the outputs of the sensors at a rate sufficient to encourage convergence of the parameter estimation algorithms.

With these factors in mind, a testbed system comprised of a linear motor, sensors, and controller was constructed, and is shown in Figure 2. A schematic of the testbed is shown in Figure 3. The testbed is capable of applying perturbation inputs to a variety of nonlinear mechanisms and recording the resulting responses.

2.2 Linear Motor

As mentioned previously, the primary considerations in selecting a linear actuator for the testbed were bandwidth, maximum force, and range of motion (stroke length). Secondary design considerations include modularity and ease of control, both aimed at increasing the versatility of the system. Other researchers have used voice coils in similar applications due to their modularity and high bandwidth (MacLean et al., 1995), but difficulties in finding voice coils with the proper combination of stroke length, maximum force, and cost

prohibited their use in this project.

Instead, a two-phase synchronous linear motor, manufactured by LinMot of Switzerland, was selected as the actuator. Its specifications are summarized in Table 1. The LinMot is a zero-backlash direct-drive actuator, and is self-contained and modular, making its incorporation into the testbed design much simpler. Control of the linear motor is achieved by providing an analog command voltage to the LinMot amplifier. The voltage corresponds directly with desired motor force, which is enforced through an internal force feedback loop in the amplifier.

Table 1: Linear Motor Specifications

Model	P01-23x160
Maximum Stroke	210 mm
Linear Stroke	70 mm
Maximum Acceleration	272 m/s ²
Maximum Velocity	3.4 m/s
Peak Force	60 N
Bandwidth (closed-loop)	~29 Hz

2.3 Sensors

Ideally, it would be possible to measure the position, velocity, and acceleration of the target mechanism, as well as the corresponding force needed to obtain that motion. In the present testbed, only three of those variables are directly measured: position, acceleration, and force. The velocity is obtained indirectly by differentiation of the position signal; future design iterations may include a velocity transducer.

The system position is measured using the internal LinMot incremental encoder, whose quadrature outputs are read using a custom built circuit that converts quadrature pulses to a 16-bit word that can be read by the control computer. The encoder has a resolution of 5 microns.

Acceleration and force are measured using a high-quality Bruel & Kjaer Type 8001 impedance head attached to the contact end of the LinMot slider. The impedance head outputs are conditioned by two Bruel & Kjaer Type 2635 charge amplifiers that amplify and lightly filter (1 KHz low-pass) the signals, which are then read by the controller. Details of the impedance head are given in Table 2.

Table 2: Impedance Head Specifications

Model	8001
Accelerometer Frequency Range	0.2-10000 Hz
Transverse Sensitivity	< 3%
Maximum Load in Tension	300 N
Maximum Load in Compression	2000 N

2.4 Controller

In the modeling and identification approach described in this paper, the inputs to the target device are position, velocity, and acceleration. The output is the force. The controller, therefore, is designed to cause the linear motor to follow a

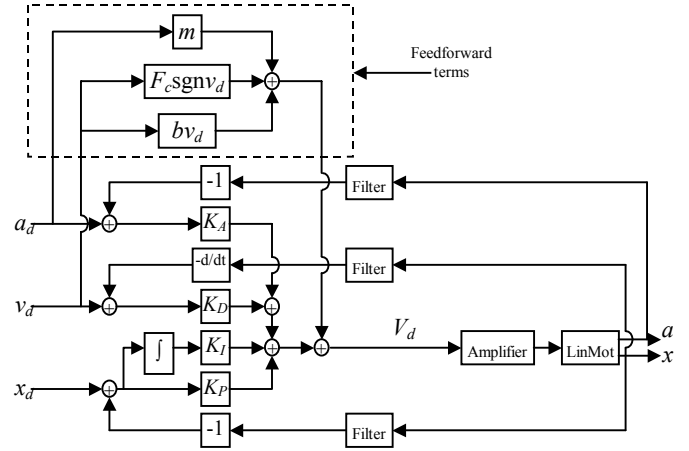


Figure 4: Modeling testbed control scheme

specified motion trajectory while measuring the resulting force exerted on the target device. This is accomplished using a PID feedback controller that uses the filtered encoder output and velocity (obtained by differentiation) as feedback signals. Filtering is accomplished digitally, with the cutoff frequency and filter order specified by the user at run time. Modifications to the basic PID scheme were implemented, including friction and inertia feedforward terms. Acceleration feedback was also found to improve the tracking capabilities of the system. The details of the control scheme are shown in Figure 4.

The controller described above is implemented on a PC equipped with a Measurement Computing PCI-DAS1602/16 I/O board, with two differential 16-bit analog input channels devoted to reading the impedance head outputs, one 16-bit analog output port that drives the LinMot amplifier, two 8-bit digital ports to read the encoder output, and one 8-bit digital port that controls various functions of the motor amplifier. The control software was written using Visual C++. The user interface allows control parameters and options to be specified at runtime. Filter coefficients are calculated at runtime by passing the filter order and cutoff frequency to MATLAB using the MATLAB Engine, a library used to transfer data and execute MATLAB commands from programs written in C or FORTRAN (Hanselman et al., 2001).

3 SEGMENTED MODELS OF NONLINEAR MECHANISMS

3.1 Overview

General mechanical devices exhibit a nonlinear relationship between the output force and the input position, velocity, and acceleration:

$$F = f(x, v, a, \theta_i) \quad (1)$$

where x is the input position, v the input velocity, a the input acceleration, F the resulting force, and θ_i the parameters of the particular nonlinear model described by Equation (1). If a user constrains a device to undergo motion described by x , v , and a , then the user will need to apply a force F to cause that motion. Of course there is an inverse view in which the user applies a force F , resulting in the motion described by x , v , and a . For the purposes of this paper, the motion variables are considered the inputs to the system, and the force the output.

The problem addressed in this paper is that of determining a suitable function f and estimating the model parameters θ_i based on experimental data obtained from the modeling testbed described in previous sections. One of the difficulties associated with this problem is that current nonlinear parameter estimation schemes require that the specific form of the nonlinear model be known *a priori*. So, for a system identification method to work for a device, the user would need to know in advance the basic internal structure of the mechanism to be modeled. The objective of this research is to demonstrate a more general method that allows a large class of devices to be modeled, without specifying the details of the model that is specific to a particular target device.

3.2 A General Model

A typical model of a mechanical system is shown in Figure 5. The model may be described by

$$F = ma + bv + k(x - x_e) + F_c \operatorname{sgn}(v) \quad (2)$$

where m is the mass, b the damping coefficient, k the spring constant, x_e the equilibrium position of the spring, and F_c the magnitude of the coulomb friction force. Comparing Equations (1) and (2), we see that m , b , k , x_e , and F_c are the parameters θ_i of the original model. The model can be made more general by allowing the parameters to vary with x :

$$F = m(x)a + b(x)v + k(x)(x - x_e(x)) + F_c(x) \operatorname{sgn}(v) \quad (3)$$

This describes a general nonlinear model that may be used to approximate single degree of freedom devices. Note that no mention has been made about *how* the parameters vary with x .

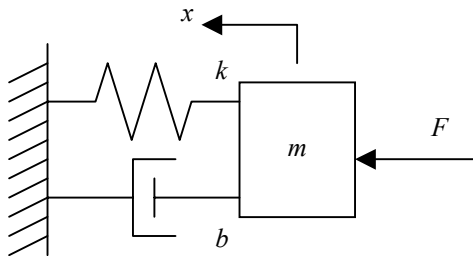


Figure 5: A simple mechanical model

This type of model is appealing for several reasons. First, most mechanical devices are actually comprised of elements having mass (or rotational inertia), damping, stiffness, and friction. By including nonlinear versions of these terms in the model, it is possible to approximate the behavior of the target device. Second, most people have an intuitive feel for the types of forces present in Equation (3). For a project such as the one described in this paper, where the objective is to model complex mechanisms so that they feel realistic to the user, the benefits of using models with intuitive appeal are obvious. Third, methods exist for estimating parameters appearing in models such as those described by Equation (3).

3.3 A Piecewise Linear Model

The parameter estimation problem is greatly simplified if certain restrictions are placed on the manner in which the parameters in Equation (3) are allowed to vary with x . One method for simplifying the problem is to segment the data in x , and assume piecewise constant parameters over the range of x in that segment. With this assumption in place, the variation of parameters can be described in a straightforward manner:

$$\begin{aligned} m(x) &= \sum_{j=1}^N m_j [h(x - x_j) - h(x - x_{j+1})] \\ b(x) &= \sum_{j=1}^N b_j [h(x - x_j) - h(x - x_{j+1})] \\ k(x) &= \sum_{j=1}^N k_j [h(x - x_j) - h(x - x_{j+1})] \\ x_e(x) &= \sum_{j=1}^N x_{ej} [h(x - x_j) - h(x - x_{j+1})] \\ F_c(x) &= \sum_{j=1}^N F_{cj} [h(x - x_j) - h(x - x_{j+1})] \end{aligned} \quad (4)$$

In these equations, N is the number of segments into which the range of x is divided; m_j , b_j , k_j , x_{ej} , and F_{cj} are the parameter values in segment j ; and $h(x - x_j)$ is the heaviside function:

$$h(x - x_j) = \begin{cases} 0, & x < x_j \\ 1, & x \geq x_j \end{cases} \quad (5)$$

As used in Equation (4), this function serves the purpose of determining which parameter values are active in the current segment by activating the j^{th} parameters when x_j (the lower limit of the j^{th} segment) is reached, and deactivating the j^{th} parameters when x_{j+1} (the upper limit of the j^{th} segment) is reached. Figure 6 illustrates the piecewise variation of the parameter m .

To improve the ability of the model described by Equations (3) and (4) to accurately represent physical devices, it is also

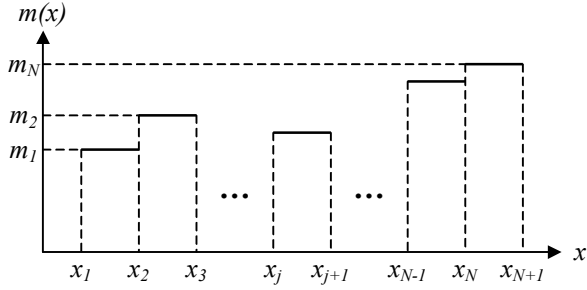


Figure 6: Piecewise segmentation of mass parameter

useful to introduce a dependence of the parameters on the direction of motion of the device, i.e., allow each segment to take on two parameter values: one for positive travel (positive v), and one for negative travel (negative v). This modification improves the model's ability to convey the feeling of backlash and hysteresis, as well as accounting for directional variations in the parameters. With this additional assumption in place, the force in the j^{th} segment can be written as

$$\begin{aligned} F_j &= m_j a_j + b_j v_j + k_j (x_j - x_{e_j}) + F_{c_j} \\ &= m_j a_j + b_j v_j + k_j x_j + (F_{c_j} - k_j x_{e_j}) \end{aligned} \quad (6)$$

This is valid for one direction, and another identical equation (with different parameter values) for the other direction.

3.4 Least-Squares Parameter Estimates

Estimation of these piecewise model parameters lends itself well to solution by ordinary least-squares. Ordinary least-squares, which provides the basis for many parameter estimation schemes, is a robust and simple matrix method of solving for the model parameter values from experimental data (Ljung, 1999). To apply this method to the current problem, Equation (6) must be rewritten in a vector-matrix form:

$$F_j = [a_j \quad v_j \quad x_j \quad 1] \begin{bmatrix} m_j \\ b_j \\ k_j \\ F_{c_j} - k_j x_{e_j} \end{bmatrix} = X_j \Theta_j \quad (7)$$

where F_j , a_j , v_j , and x_j are column vectors of experimental data belonging to the j^{th} segment. The parameter estimates are then found using the well-known result (Hsia, 1977):

$$\Theta_j = (X_j^T X_j)^{-1} X_j^T F_j \quad (8)$$

3.5 Parameter Estimation Algorithm

The algorithm for estimating the nonlinear model parameters may be summarized as follows:

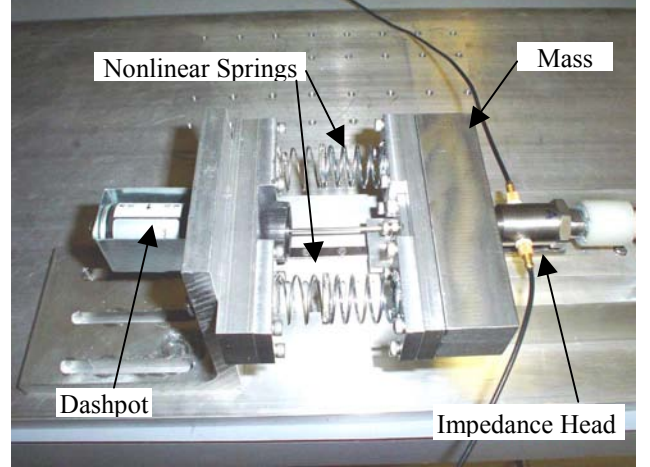


Figure 7: Nonlinear test system

- Specify N , the number of segments over the range of x
- Determine the segment endpoints, x_j , for $j = 1 \dots N$
- For each segment j , find all data points (force, position, velocity, and acceleration) that lie in that segment ($x_{j+1} > x > x_j$)
- For each segment j , group the data into positive and negative direction data
- For each segment j , apply Equation (8) separately to the data corresponding to each direction
- The estimated mass, damping, stiffness, and coulomb force/spring offset force for the j^{th} segment are then the elements of the vector Θ_j

4 TEST SYSTEM DESIGN AND CALIBRATION

4.1 Overview

Before applying the modeling algorithm to general mechanisms (such as a turn signal switch or other devices with nonlinear dynamics), it was desired to first test and verify the effectiveness of the algorithm on a much simpler nonlinear device. The test system was designed to be closely modeled by Equation (3) and Figure 5, with the spring constant and damping parameters varying nonlinearly with position and direction. The resulting system, comprised of a constant mass, two nonlinear springs, and a dashpot, is shown in Figure 7. The details of the individual components will now be given.

4.2 Springs

A nonlinear (hardening) spring element was created by connecting in series a relatively stiff spring (of stiffness k_s) and a relatively compliant spring (of stiffness k_c), where $k_s > k_c$. For small deflections ($x < x_T$), the effective stiffness of the system is given by the series combination of the two stiffnesses. The nonlinearity arises when the compliant spring is completely compressed, thus increasing the overall stiffness to that of the stiff spring alone. This may be summarized as follows:

$$k = \begin{cases} \frac{k_s k_c}{k_s + k_c}, & x < x_T \\ k_s, & x > x_T \end{cases} \quad (9)$$

In this equation x_T is the transition deflection at which the compliant spring is completely compressed. Two of these spring systems were connected in parallel (as shown in Figure 8), and then placed on an Instron load frame to obtain a measure of the actual stiffness vs. deflection. The results are shown in Figure 9. The upper and lower bounds are based on the spring manufacturer's published stiffness tolerances of $\pm 10\%$. Note that the calculated stiffness lies within the predicted bounds. An additional increase in stiffness occurs after $x = 20$ mm. This is attributed to the partial compression of the stiff spring.

4.3 Dashpots

Damping for the system is provided by a dashpot manufactured by the Airpot Corporation. To estimate the numerical value of the damping, the dashpot was placed vertically with a known mass suspended from the dashpot connecting rod. The mass was released, and the ensuing motion recorded using an Optotrak (optical position measurement system manufactured by Northern Digital Inc.). The damping coefficient was then estimated from the terminal velocity calculated from the Optotrak data. Four trials were performed, with each result varying by less than 1% from the mean value of 11.5 kg/s.

5 RESULTS

The testbed and parameter estimation algorithm described in previous sections were used to estimate parameters on the nonlinear test system comprised of a mass, nonlinear springs, and dashpots. Figure 7 shows the connection between the test system and the impedance head. The testbed applied a 1.5 Hz sinusoid to the test system for 100 seconds, with a sampling rate

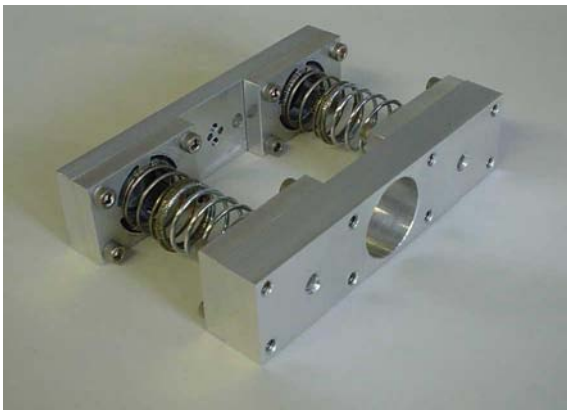


Figure 8: Nonlinear spring assembly

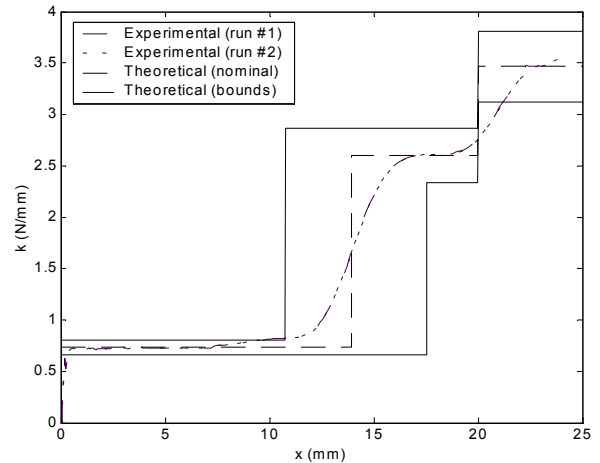


Figure 9: Spring stiffness from load frame measurements

of 4000 Hz. The data was analyzed offline using the segmentation algorithm with eight segments. The results are shown in Figure 10. The upper plot shows the estimated mass vs. position, with the actual value of 581 grams represented by the solid line, and the positive and negative estimates represented by the dashed and dotted lines, respectively. The second plot shows the estimated damping, with the same line types representing the actual, positive, and negative data. The third plot shows the segmented estimates of the system stiffness. The solid line is a segmented version of the stiffness obtained from the load frame (compare with Figure 9). The transition region is apparent in this plot. The last plot is the sum of coulomb friction and offset spring force, which are inseparable in this estimation algorithm, as is apparent from Equation (7).

6 CONCLUSIONS

6.1 Discussion

In general, the parameter estimates obtained using the testbed and segmented least-squares algorithm agreed closely with the values obtained using other methods. The mass estimates were especially impressive, considering the general difficulty in obtaining good estimates of this parameter due to noise, sensor drift, and other issues usually associated with the measurement of acceleration. MacLean (1996) cited these factors in not obtaining estimates of mass in her dynamic models. The success in the present study was likely due to the quality of the impedance head, as well as the robustness of least-squares methods. The spring stiffness estimates were equally accurate, and faithfully tracked the nonlinear transition region shown in Figure 9. The estimate of damping, on the other hand, did not closely match the value obtained using the OptoTrak system. This may be due to the noisy velocity signal, which was obtained by differentiating the position signal. Finally, the force comprised of coulomb and spring offset terms, though having no actual values with which to be compared,

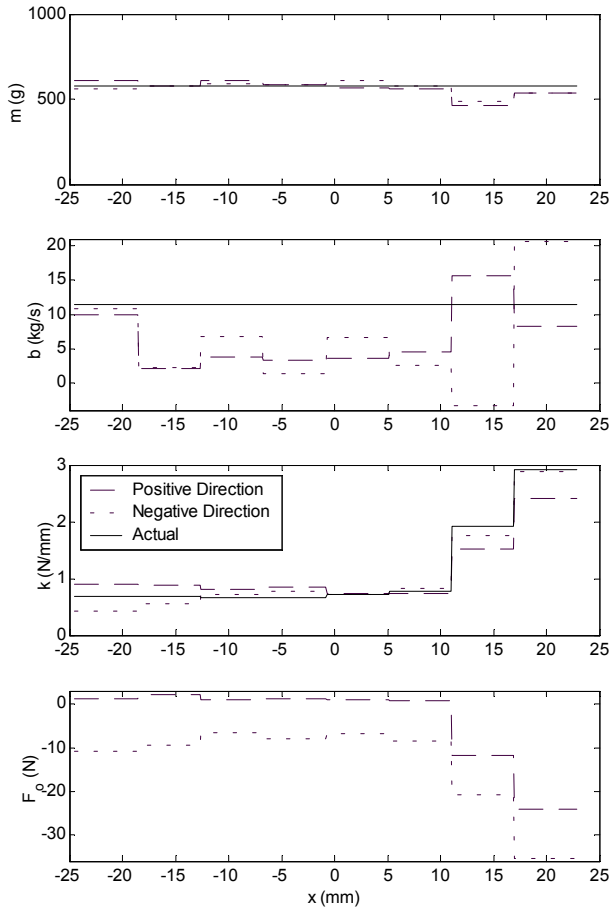


Figure 10: Parameter estimates for nonlinear test system

exhibited the expected trend, with the magnitude of the force increasing with increasing stiffness.

6.2 Future Work

The immediate application of this work will be to obtain models of actual physical devices and use the models in haptic simulations for virtual prototyping. An interesting class of target devices includes switches and knobs. Work is under way to model and simulate an automotive turn signal switch. This device was chosen for its highly nonlinear input-output relationship, as well as for its familiarity (most people know what it feels like to move a turn signal switch).

Additional work must be done on improving the modeling procedure. Current work is aimed at developing smart segmentation schemes, rather than arbitrarily selecting the number of segments in advance. Recursive least-squares algorithms are also being investigated for generating nonlinear models without segmenting the model parameters.

ACKNOWLEDGMENTS

This work was supported by NSF Grant DMI 9978603.

REFERENCES

- Hanselman, D., and Littlefield, B., "Mastering MATLAB 6," Prentice Hall, Upper Saddle River, NJ, 2001.
- Hollerbach, J.M., Cohen, E., Thompson, W.B., and Jacobsen, S.C., "Rapid virtual prototyping of mechanical assemblies," *Proc. 1996 NSF Design and Manufacturing Grantees Conf.*, Albuquerque, NM, Jan. 2-5, 1996, pp. 477-478.
- Hollerbach, J.M., Cohen, E., Thompson, W.B., Freier, R., Johnson, D.E., Nahvi, A., Nelson, D.D., and Thompson, T.V., "Haptic interfacing for virtual prototyping of mechanical CAD designs," *Proc. 1997 ASME Design for Manufacturing Symposium*, Sacramento, CA, Sept. 14-17, 1997, CDROM proceedings.
- Hollerbach, J.M., "Some current issues in haptics research," *Proc. IEEE Intl. Conf. Robotics and Automation*, San Francisco, April 24-28, 2000, pp. 757-762.
- Hsia, T.C., "System identification: least-squares methods," Lexington Books, Lexington, MA, 1977.
- Jacobsen, S.C., Smith, F.M., Iversen, E.K., and Backman, D.K., "High performance, high dexterity, force reflective teleoperator," *Proc. 38th Conf. Remote Systems Technology*, Washington DC, Nov. 1990, pp. 180-185.
- Ljung, L., "System identification: theory for the user," Prentice Hall PTR, Upper Saddle River, NJ, 1999.
- MacLean, K.E., and Durfee, W.K., "An apparatus to study the emulation of haptic feedback," *Proc. ASME Dynamic Systems and Control Division*, pp. 615-621, 1995.
- MacLean, K.E., "The 'haptic camera': a technique for characterizing and playing back haptic properties of real environments," *Proc. ASME Dynamic Systems and Control Division*, 1996.
- Nahvi, A., Nelson, D.D., Hollerbach, J.M., and Johnson, D.E., "Haptic manipulation of virtual mechanisms from mechanical CAD designs," *Proc. IEEE Intl. Conf. Robotics & Automation*, Leuven, Belgium, May 16-21, 1998, pp. 2566-2573.
- Okamura, A.M., Dennerlein, J.T., and Cutkosky, M.R., "Reality-based models for vibration feedback in virtual environments," *ASME/IEEE Trans. on Mechatronics*, Focused Section on Haptic Devices and Applications, Vol. 6, No. 3, 2001, pp. 245-252.
- Pai, D.K., "Robotics in reality-based modeling," *Robotics Research: the Ninth International Symposium*, J. Hollerbach and D. Koditschek (Eds), Springer-Verlag, London, 2000.
- Pai, D.K., and Rizun, P., "The What: a wireless haptic texture sensor," *Proc. 11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Mar. 22-23, 2003, pp. 3-9.
- Weir, D.W., Peshkin, M., Colgate, J.E., Buttolo, P., Rankin, J., and Johnston, M., "The haptic profile: capturing the feel of switches," *Proc. 12th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Mar. 27-28, 2004, pp. 186-193