Force Estimation Based Compliance Control of Harmonically Driven Manipulators

Leon M. Aksman[†], Craig R. Carignan[‡], and David L. Akin[†]

 F- Space Systems Laboratory (SSL)
 Department of Aerospace Engineering University of Maryland
 College Park, MD 20742
 Email: laksman@ssl.umd.edu,
 dakin@ssl.umd.edu

Abstract—The problem of estimating external forces exerted on a robotic manipulator with harmonic drive gearing without a force-torque sensor is considered. Manipulator dynamics together with motor current feedback is used to estimate external joint torques, which are transformed into estimated external end effector forces using knowledge of the manipulator's kinematics. Adaptive control is used to tune the parameters of the robot's modeled dynamics, while adaptive radial basis function (RBF) "neural" networks are used to learn the friction model. Compliance control is implemented on a two degree of freedom manipulator based on the force estimates. Results are compared to compliance control using a six-axis force-torque sensor mounted on the manipulator.

I. INTRODUCTION

THE most traditional robotic manipulator control schemes are those that attempt to control strictly position. Such schemes ultimately require that a manipulator track a time varying joint trajectory specified for each of its degrees of freedom. Position control is an intuitive and often effective means by which to accomplish tasks. Its major drawback is that a manipulator will attempt to track its desired trajectory even if that brings damage to itself and objects in its way.

As a result, force control schemes have been developed to deal with controlling interactions between the manipulator and its environment. Compliance control attempts to combine aspects of both position and force control by enforcing a mass-spring-damper relationship between external force and the manipulator's desired position, velocity and acceleration.

Robotic manipulators typically use force-torque sensors to realize force or compliance control; however, force-torque sensors have several well-known drawbacks in the form of their cost, size and the complexity they introduce into a manipulator's mechanical and electrical design. The latter point is due to the constraint that they be placed as close as possible to the end effector for best results. Another, less often considered drawback is that force-torque sensors saturate due to high water pressure, rendering them useless in deep-sea robotic sampling tasks. For such environments, a

 Imaging Science and Information Systems (ISIS) Center Department of Radiology Georgetown University Washington, DC 20057 Email: crc32@georgetown.edu

measure of safety can be provided using force estimation instead.

Harmonic drives are often used in manipulators due to their lack of backlash and the high gear ratio they enable. However, harmonic drives also suffer from difficult-tomodel friction, especially in a static situation. Static friction, also known as stiction, is a major source of error in force estimation due to the difficulty in modeling its behavior. This work will attempt to demonstrate that force estimation is possible in such systems despite the effects of stiction.

II. PREVIOUS WORK

Force estimation as applied to robotic manipulators has been a topic of interest since the early 1990s. The authors of [10] proposed a decoupled disturbance observer based approach. In [5], Hacksel and Salcudean presented a coupled-force observer based on accurate knowledge of a robot's dynamics. Both observer-based approaches demonstrated good results on direct drive manipulators with negligible friction dynamics.

More recently, dynamics learning has been used in force estimation. In [16], the authors used a neural network to learn the entire dynamical model of their 3 DOF haptic device offline. Their system contained little friction. In [17], the authors showed that force sensorless hybrid force/position control was possible in a geared, though not harmonically driven, manipulator. They used a simplified model of robot dynamics, consisting of a gravity term and learned friction terms. Adaptive neural networks were used for online friction learning though adaptation of the modeled dynamics was not performed.

In [14], motor current was used to estimate external forces for robots with harmonic drive gearing. The approach involved subtracting modeled dynamics from motor torque, assumed to be proportional to motor current, to form the estimated external torque. The estimated torque thus obtained contained significant unmodeled positiondependent friction. Filtering the estimated external torque in the position domain greatly improved the estimates. The filtering was done offline however, where the entire position history of the estimated external torque was known.

All of these techniques have relied on unchanging conditions in the manipulator's dynamics. In reality, factors such as end effector loading can significantly alter the parameters of the robot's modeled dynamics. In addition, because friction changes with loading and temperature, unmodeled dynamics such as friction that were learned and then frozen can prove to be inaccurate over extended running times. Both points lead to a desire to relearn the manipulator's dynamics at certain points in time chosen by either an operator or higher-level autonomy. The force estimation technique introduced in this work includes friction learning as well as adaptation to enable relearning during such changes in dynamics.

III. ADAPTATION AND LEARNING

The dynamics of an N degree of freedom (DOF) geared robotic manipulator are described by

$$\boldsymbol{\tau} + \boldsymbol{\tau}_{\text{ext}} = (\mathbf{H}(\mathbf{q}) + \mathbf{G}^{2}\mathbf{I}_{M}) \, \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \, \dot{\mathbf{q}} \\ + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}})$$
(1)

where τ is the N x 1 vector of commanded joint torques given by the control law (described below), τ_{ext} is the N x 1 vector of external torques and **q** is the N x 1 vector of actual joint angles; (**H**(**q**) + **G**²**I**_M) is the N x 1 vector of actual matrix, where **I**_M and **G** are the N x N diagonal matrices of rotor inertias and gear ratios corresponding to each DOF, respectively, and **H** is the N x N ungeared inertia matrix; **C** is the N x N Christoffel matrix of Coriolis and centripetal force terms [3] and **g** is the N x 1 gravity vector; **f** is the N x 1 vector describing the friction dynamics.

The following control law was used for adaptation and learning [13].

$$\begin{aligned} \boldsymbol{\tau} &= (\hat{\mathbf{H}}(\mathbf{q}) + \mathbf{G}^{2} \hat{\mathbf{I}}_{\mathrm{M}}) \ddot{\mathbf{q}}_{\mathrm{r}} + \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}_{\mathrm{r}} \\ &+ \hat{\mathbf{g}}(\mathbf{q}) + \hat{\mathbf{f}}_{\mathrm{v}}(\dot{\mathbf{q}}) - \mathbf{K}_{\mathrm{d}} \mathbf{s} \\ &= \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_{\mathrm{r}}, \ddot{\mathbf{q}}_{\mathrm{r}}) \hat{\mathbf{a}} + \hat{\mathbf{f}}_{\mathrm{v}}(\dot{\mathbf{q}}) - \mathbf{K}_{\mathrm{d}} \mathbf{s} \end{aligned} \tag{2}$$

The ' $^{\prime}$ symbol in (2) denotes estimates of terms. Y is an N x M matrix containing the known functions in the modeled dynamics that are parameterized by M constants, arranged in the M x 1 vector **a**. Several variables used in (2) are defined as follows

$$\dot{\mathbf{q}}_{\mathbf{r}} \equiv \dot{\mathbf{q}} - \Lambda (\mathbf{q} - \mathbf{q}_{\mathbf{d}}) \mathbf{s} \equiv (\dot{\mathbf{q}} - \dot{\mathbf{q}}_{\mathbf{d}}) + \Lambda (\mathbf{q} - \mathbf{q}_{\mathbf{d}})$$
(3)

where \mathbf{q}_d is the N x 1 vector of desired joint angles and Λ is a positive definite matrix, usually diagonal, revealing $-\mathbf{K}_d \mathbf{s}$ as the standard Proportional-Derivative (PD) term when $\Lambda = \mathbf{K}_d^{-1}\mathbf{K}_p$, and \mathbf{K}_p is the proportional gain.

The friction terms are implemented as adaptive RBF networks based on [7]. In [12], the approximation abilities of such functions were investigated and bounds on error were given based on the number of nodes in the network and the spacing between them. The basis function chosen for the neural network, \mathbf{y} , is the "hat" function, defined as

$$\mathbf{y}(\mathbf{x}) \equiv \begin{cases} 1 - |\mathbf{x}|, & |\mathbf{x}| < 1\\ 0, & \text{otherwise} \end{cases}$$
(4)

Using y, the RBF network is defined for the jth joint as

$$\hat{\mathbf{f}}_{\mathbf{v},\mathbf{j}}(\dot{\mathbf{q}}) = \sum_{\mathbf{k}=\mathbf{k}_{\text{MIN}}}^{\mathbf{k}-\mathbf{k}_{\text{MAX}}} \hat{\mathbf{c}}_{\mathbf{k},\mathbf{j}} \mathbf{y}(\mathbf{h}\dot{\mathbf{q}}_{\mathbf{j}} - \mathbf{k})$$
(5)

Each joint's network contains $(\mathbf{k}_{MAX} - \mathbf{k}_{MIN} + 1)$ total nodes. The parameter **h** determines the spacing between consecutive nodes. Node **k**'s center is located at $\mathbf{h}^{-1}\mathbf{k}$ and the node is zero outside of $\mathbf{h}^{-1}\mathbf{k} \pm \mathbf{h}^{-1}$. As a result just two nodes determine the network's output to an input velocity within its range. This choice of basis function reduces computation time in the both the coefficient update and output evaluation calculations.

The estimates of \mathbf{a} and $\mathbf{c}_{\mathbf{k},\mathbf{j}}$ are updated as follows

$$\hat{\mathbf{a}} = -\Gamma \mathbf{Y}^{\mathrm{T}} \mathbf{s}_{\Delta}$$
$$\hat{\mathbf{c}}_{\mathbf{k},\mathbf{j}} = -\gamma \mathbf{y} (\mathbf{h} \dot{\mathbf{q}}_{\mathbf{j}} - \mathbf{k}) \mathbf{s}_{\Delta \mathbf{j}}$$
(6)

where Γ is an M x M matrix of learning gains for the adaptation and γ is the scalar learning gain for the neural networks. The adaptation laws use \mathbf{s}_{Δ} , a modified version of the tracking error s. The components of this modified error measure are

$$\mathbf{s}_{\Delta \mathbf{j}} = \mathbf{s}_{\mathbf{j}} - \varphi \operatorname{sat}(\mathbf{s}_{\mathbf{j}}/\varphi), \quad \mathbf{j} = 1, \dots, \mathbf{N}$$
 (7)

where sat() is the saturation function defined as follows

$$\operatorname{sat}(\mathbf{x}) \equiv \begin{cases} \mathbf{x}, & |\mathbf{x}| < 1\\ 1, & |\mathbf{x}| \ge 1 \end{cases}$$
(8)

Subtracting the scaled saturation function from s forms a deadzone that assures that the adaptation does not cause instability by trying to achieve perfect s = 0 tracking. The value of φ chosen was 0.001 in training.

IV. Force Estimation

Equations (2), (4), and (6) determine the adaptation plus friction learning control law used during the training phase of the force estimation process. An important point is that during training, zero external force is assumed to be acting on the manipulator. After the training phase, the control law stops updating its estimates of **a** and $C_{k,j}$ and the system goes into estimation mode. In practice, switching control laws between training mode and estimation mode can be implemented by setting a flag to enable the update law (6). If the updates were to continue in estimation mode, the friction learning neural networks would learn the joint torques needed to overcome the external torque in addition to the actual friction torque of the system, causing incorrect estimation.

To form the estimate of external torque acting on the manipulator's joints, equation (1) is solved for τ_{ext} using the estimate of the manipulator's dynamics formed in training mode using control law (2). The estimate becomes

$$\begin{aligned} \hat{\boldsymbol{\tau}}_{\text{ext}} &= (\hat{\mathbf{H}}(\mathbf{q}) + \mathbf{G}^2 \hat{\mathbf{I}}_{\text{M}}) \ddot{\mathbf{q}} + \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} \\ &+ \hat{\mathbf{g}}(\mathbf{q}) + \hat{\mathbf{f}}_{\text{v}}(\dot{\mathbf{q}}) - \boldsymbol{\tau}_{\text{actual}}. \end{aligned} \tag{9}$$

Instead of the command torque τ , the measured motor current, i_m , was converted to torque using knowledge of the gear ratios, contained in G and the motor constants, contained in the diagonal matrix K_m , forming

$$\boldsymbol{\tau}_{\text{actual}} = \mathbf{G} \, \mathbf{K}_{\text{m}} \, \mathbf{i}_{\text{m}} \tag{10}$$

Using knowledge of the manipulator's kinematics, namely the Jacobian matrix, the external torque estimate can be converted to estimated external force/moment at the manipulator's end effector (referred to simply as estimated force elsewhere in this paper).

$$\hat{\mathbf{f}}_{\text{ext}} = (\mathbf{J}^{\mathrm{T}})^{-1} \hat{\boldsymbol{\tau}}_{\text{ext}}$$
(11)

In the case of N < 6 or N > 6 the inverse of the transpose Jacobian is replaced with either the right or left pseudoinverse respectively [11]. Note that the resulting estimated force is in the same reference frame as the Jacobian matrix. Equations (9) and (11) together describe how to use the dynamical model, learned using the control law in equation (2), together with motor current feedback and kinematics knowledge to estimate force.

Note that the motor current contains high frequency noise due to the motor driver switching frequency and the encoder differencing used in joint velocity calculation that is injected into the commanded torque via the PD term. The noise was reduced by a digital low pass filter implemented on the torque estimates generated by (9), before applying (11). The filter chosen was a fifth order elliptic filter [9] with a passband of 20 Hz. The same filter was applied to the twice differenced encoder measurements to form the actual acceleration used in (7). The filters effectively limited the bandwidth of the the force estimates to 20 Hz.

V. COMPLIANCE CONTROL

The compliance controller chosen was the admittance control scheme used in [4], a variation of the "position-based impedance control" scheme introduced by [8]. Conceptually, admittance controllers accept sensed external force and react by modifying the manipulator's desired trajectory

$$\mathbf{f}_{\text{ext}} = \mathbf{M}_{\text{s}} \tilde{\mathbf{x}} + \mathbf{C}_{\text{s}} \tilde{\mathbf{x}} + \mathbf{K}_{\text{s}} \tilde{\mathbf{x}}$$
(12)

where $\mathbf{\tilde{x}} \equiv \mathbf{x}_d - \mathbf{x}_n$, \mathbf{x}_n is the nominal desired Cartesian position, and \mathbf{x}_d is the modified desired Cartesian position due to the enforcement of the desired system's impedance. The desired impedance is specified by the mass, spring, and damping matrices \mathbf{M}_s , \mathbf{C}_s , and \mathbf{K}_s respectively. Because the modified desired trajectory is formed in Cartesian space rather than joint space, the following numerical inverse kinematics algorithm, the iterative Jacobian pseudo-inverse method [15], was used:

$$\mathbf{q}_{d}(\mathbf{k}+1) = \mathbf{q}_{d}(\mathbf{k}) + \mathbf{J}(\mathbf{q}_{d}(\mathbf{k}))^{-1}(\mathbf{x}_{d} - \mathbf{f}(\mathbf{q}_{d}(\mathbf{k})))$$
(13)

where **k** is the iteration number and **J**⁻¹ is the inverse Jacobian, which is replaced by the pseudo-inverse for N \neq 6. The algorithm consists of iterating **k** until the forward kinematics of the estimated **q**_d yields the desired Cartesian position **x**_d at each time step.

The corresponding joint velocity and acceleration are computed numerically as follows

$$\dot{\mathbf{q}}_{\mathbf{d}}[\mathbf{n}] = \frac{(\mathbf{q}_{\mathbf{d}}[\mathbf{n}] - \mathbf{q}_{\mathbf{d}}[\mathbf{n} - \mathbf{1}])}{\Delta \mathbf{t}}$$
(14)

$$\ddot{\mathbf{q}}_{\mathbf{d}}[\mathbf{n}] = \frac{(\dot{\mathbf{q}}_{\mathbf{d}}[\mathbf{n}] - \dot{\mathbf{q}}_{\mathbf{d}}[\mathbf{n}-\mathbf{1}])}{\Delta \mathbf{t}}$$
(15)

where Δt is the sample period, and **n** is the time sample index. In training mode the modified desired trajectory due to the compliance control was not used; instead, the desired trajectory was specified directly in joint space. The desired trajectory may also be specified in Cartesian space and converted into joint space. During estimation mode the desired trajectory was still specified in joint space though it was converted into Cartesian space for the compliance controller's sake as follows

$$\mathbf{x}_{\mathbf{n}} = \mathbf{L}(\mathbf{q}_{\mathbf{n}}) \tag{16}$$

$$\dot{\mathbf{x}}_{\mathbf{n}} = \mathbf{J} \dot{\mathbf{q}}_{\mathbf{n}} \tag{17}$$

$$\ddot{\mathbf{x}}_{\mathbf{n}} = \frac{\mathbf{d}\,\mathbf{x}_{\mathbf{n}}}{\mathbf{d}t} \tag{18}$$

where \mathbf{L} is the manipulator's forward kinematics. Numerical differentiation was used to calculate (18) as in (14) and (15).

A block diagram of the compliance and position controllers is shown in Fig. 4. Note that the control law (2) is unaware of the modification to its desired trajectory by the compliance control stage just as the compliance control stage is unaware of the use of force estimation instead of actual force sensing. This last point can be made explicit by substituting $\hat{\mathbf{f}}_{ext}$ from (11) instead of \mathbf{f}_{ext} in (12).

VI. EXPERIMENTAL SETUP

The force estimation based compliance control scheme was implemented on a two DOF pitch-roll manipulator developed at the Space Systems Laboratory of the University of Maryland [1]. Each DOF is harmonically driven, where the pitch and roll gear ratios are 161 and 160, respectively. A 100+ hour run-in test was performed on both DOFs that helped reduce the friction in the gearing. The incremental quadrature encoders embedded in the manipulator each had 36,000 counts per revolution before gearing. The manipulator's motors are both brushless and driven by Advanced Motion Control (AMC) Models B15A8 and B30A8 brushless motor drivers. The AMC drivers provide motor current information that was directly used in (10). A JR3 force-torque sensor was mounted near the end effector as shown in Fig. 1.

A National Instruments NI PCI-6025e data acquisition (DAQ) board was used to communicate between the computer and the manipulator. The control program was written in C and run on a Dell Dimension 8400 PC with a 3.6 GHz processor and 1 GB of RAM. The computer was running distributed TimeSys real-time Linux kernel 2.6.16.9. The DAQ board was used with a Comedi driver, which is a set of open source Linux drivers for various commercial DAQ boards. The computer communicated with the force-

torque sensor through a JR3 PCI board using a Linux driver written by Mario Plats at UJI (Spain). The fully integrated system is shown in Fig. 2.



Fig. 1. The manipulator with frame definitions. Frame 1 is the pitch DOF frame, frame 2 is the roll DOF frame. World frame origin is the same as frame 1's, both at the center of joint 1. JR3 force-torque sensor shown is used to confirm force estimates.



Fig. 2. System integration.

VII. EXPERIMENTAL RESULTS

Initial experiments were conducted to compare the force estimates against readings from a force-torque sensor. Because the manipulator is a 2 DOF non-planar type, the 3 x 1 estimated force vector has only two true directions of estimation, which vary according to the manipulator's configuration. All force estimates were 3 x 1 vectors generated using the translational part of the Jacobian for (11) and (13). To compare the estimated force with the force-torque sensor's readings, the sensor's 3 DOF reading was limited to the manipulator's two degrees of freedom using the following transformation:

 $\mathbf{f}_{ext} = \mathbf{J} (\mathbf{J}^{T} \mathbf{J})^{-1} \mathbf{J}^{T} \mathbf{\bar{f}}_{ext}$ (19) where $\mathbf{\bar{f}}_{ext}$ is the sensor's 3 x 1 force vector and $\mathbf{J} (\mathbf{J}^{T} \mathbf{J})^{-1} \mathbf{J}^{T}$ is a 3 x 3 matrix with rank at most 2. It is the result of transforming the force vector to joint torque using the transpose Jacobian, then using (11) with the right pseudo-inverse of the transpose Jacobian because N < 6. Note that before (19) was applied it, the force-torque sensor's readings were transformed from the sensor frame, which depends on both joint positions, to the world frame shown in Fig. 1. All force estimates are made in the world frame as well.

Thanks to the real-time kernel, the position control and compliance control frequencies were both 3 kHz, though the compliance controller may be run at a lower rate in more complex systems. The high running rate of the position control reduced time delay and improved velocity estimates, allowing high PD gains to be used during estimation mode. Lower PD gains were used in training mode as they provided the adaptation and friction learning with larger and cleaner error signals, enabling better learning using (6). High PD gains in estimation mode were found to be the most effective method of reducing stiction during zero commanded joint velocity. Gains were experimentally chosen to be as high as possible without causing instability. Other methods of reducing stiction were attempted including: adding a dithering signal to the commanded torque, dithering the desired trajectory, and modifying the velocity signal in the low velocity regime [6]. These approaches caused undesirable amounts of motor chatter near zero commanded joint velocity.

The desired trajectory used during training mode consisted of sinusoidal desired joint positions, velocities and accelerations. Training was performed for ten minutes before switching to estimation mode. In estimation mode a threshold of 1 Nm was placed on the estimated external torques before the application of (11).

Estimates in Fig. 3 are shown for when the manipulator was stationary in the joint configuration shown in Fig. 1. Compliance was not enabled. Despite being in the stiction regime, the force errors in all three world frame directions were usually 5 N (15% of maximum sensed force) or less during contact though there were spikes of up to 8 N (21% of maximum sensed force).

Estimates in Fig. 5 are shown for when the compliance was enabled and the compliance values were

$$M_s = 667 I_3$$

 $C_s = 333 I_3$
 $K_s = 40 I_3$

where I_3 is the 3 x 3 identity matrix. The damping ratio is approximately 1 using these values. Force errors were usually 10 N (26% of maximum sensed force) or less during contact though there were spikes of up to 15 N (39% of maximum sensed force).

VIII. CONCLUSION

This work represents an initial attempt to use force estimates instead of direct force sensing for compliance control in harmonically-driven manipulators. The force estimation scheme presented relies on adaptation to tune the parameters of the modeled dynamics as well as learning of unmodeled dynamics such as friction using RBF neural networks. High PD gains were used to reduce stiction during zero commanded joint velocity. When compliance control based on force estimates was enabled, noise in the estimates (due to unmodeled stiction, motor current, and velocity estimation) led to a noisy commanded trajectory about zero velocity. This lead to increased estimation errors from repeated zero velocity crossings, during which friction is hardest to model.

ACKNOWLEDGMENTS

Authors are grateful to Robert Sanner for discussions and software, Mike Liszka and Mike Lochner for CAD work, and Advanced Motion Controls (AMC) for hardware donation.

REFERENCES

- D. L. Akin, "MORPHbots: Miniature Self-Reconfiguring Modular Robotics for space operations," USU/AIAA Small Satellite Conference, Logan, UT, Aug. 2004.
- [2] L. M. Aksman, "Force estimation based compliance control of a two link harmonically driven robotic manipulator," MS Thesis, University of Maryland, College Park, 2006.
- [3] J.J. Craig, J.J. Introduction to Robotics, 3rd Ed., Pearson-Prentice Hall, Upper Saddle River, NJ, 2005.
- [4] P. Guion, "Impedance control with adaptive friction for a two link robotic manipulator," MS Thesis, University of Maryland, College Park, 2003.
- [5] P. J. Hacksel and S. J. Salcudean, "Estimation of environmental forces and rigid-body velocities using observers," *Proc. of IEEE Conf. on Robotics and Automation*, 1994, pp. 931-936.
- [6] J. P. Hauschild, G. Heppler, and J. McPhee, "Friction compensation of harmonic drive actuators," *6th International Conf. on Dynamics and Control of Systems and Structures in Space*, Liguria, Italy, Jul. 2004, pp.683-692.
- [7] K. C. Liu, "Experimental evaluation of adaptive neurocontrollers for a prototype space robotic manipulator", AIAA, Aerospace Sciences Meeting and Exhibit, 35th, Reno, NV, Jan. 1997.

- [8] J. Maples and J. Becker, "Experiments in force control of robotic manipulators", *Proc. of IEEE Conf. on Robotics and Automation*, Apr. 1986.
- [9] S. K. Mitra, *Digital Signal Processing: A Computer Based Approach*, 2nd Ed., McGraw-Hill, Jul. 2001.
- [10] T. Murakami, R. Nakamura, F. Yu, K. Ohnishi, "Force sensorless impedance control by disturbance observer," *Record of the Power Conversion Conf.*, Apr. 1993 pp.352 - 357.
- [11] P.N. Sabes, "Linear algebraic equations, SVD, and the pseudoinverse," http://keck.uscf.edu/~sabes/Docs/SVDnotes.pdf, Oct. 2001.
- [12] R. M. Sanner, J. J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Transactions on Neural Networks*, Vol. 3, No. 6, Nov. 1992, pp. 837-863.
- [13] R. M. Sanner, J. J. E. Slotine, "Stable adaptive control of robot manipulators using "neural" networks," *Neural Computation*, Vol. 7, No. 4, Nov. 1995.
- [14] J. W. L. Simpson, C. D. Cook, and L. Zheng, "Sensorless force estimation for robots with friction," *Proc. of 2002 Australasian Conf.* on Robotics and Automation, Nov. 2002.
- [15] L. Sciavicco and B. Siciliano, Modeling and Control of Robot Manipulators, 2nd Ed., Springer-Verlag, New York, 2000.
- [16] A.C. Smith, K. Hashtrudi-Zaad, "Application of neural networks in inverse dynamics based contact force estimation," *Proc. of IEEE Conf. on Control Applications*, Aug. 2005, pp. 1021-1026.
- [17] V. Zahn, R. Maass, M. Dapper, R. Eckmiller, "Learning friction estimation for sensorless force/position control in industrial manipulator," *Proc. of IEEE Conf. on Robotics and Automation*, May 1999, pp. 2780-2785.



Fig. 3. Force estimates versus force/torque sensor data, transformed using (16), in all three world frame directions when manipulator is stationary in joint configuration shown in Fig. 1. External forces were applied by hand on the end effector. Compliance control was turned off.



Adaptive, Friction Learning Position Controller

Fig. 4. Block diagram of adaptive, friction learning position controller, force estimation and compliance control based on the force estimates.



Fig. 5. Force estimates versus force-torque sensor data in all three world frame directions when manipulator nominal position is stationary in joint configuration shown in Fig. 1. Modified desired joint position is shown for both joints due to the activated compliance control.