Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, MN, April 24-27, 1996.

The Precise Control Of Manipulators With Joint Friction: A Base Force/Torque Sensor Method

Guillaume Morel and Steven Dubowsky
Massachusetts Institute of Technology, Dept. of Mechanical Engineering
Cambridge, MA, 02139, USA

Abstract

Joint friction is a major problem in accurately controlling robot position during manipulator tasks involving small and slow motions. Previous research in this field suggests the use of either complex modeling and identification techniques, or expensive and delicate torque sensors that must be integrated into the manipulator. This paper proposes a simple, cost-effective method for compensating the effect of joint friction, which utilizes a six-axis force/torque sensor mounted on the maniupulator'sbase.

From the base wrench measurements, the joint torques are estimated and fed back through a torque controller, that virtually eliminates friction and gravity effects. With such high-quality torque control, a simple PD position controller is sufficient to provide high precision motion control even at very low speed and small motions.

Theoretical and practical aspects of the torque estimation are first discussed. Next, the control design and tuning is shown. Experimental results for an industrial Puma manipulator, with high Coulomb friction in its gear trains, show the effectiveness of the method. The achieved precision is substantially greater than for conventional methods and approaches the resolution of the Puma's encoders.

1 Introduction

In many new applications of robotic manipulators, such as surgery or micro assembly, the manipulator end-effector must be controlled very accurately during small, slow motions. The precision required is difficult to achieve with currently available systems, due to nonlinear joint friction, which can lead to stick-slip motions, static positioning errors, or limit cycle oscillations.

Previous techniques developed to deal with this problem can be classified in three categories: model based compensation, torque pulse generation, and torque feedback control. In the first, a model is used to compute an estimate of the friction torque, which is provided to the actuator controller. The friction model can be used either in feedforward compensation control [1,2], or in feedback compensation control [3]. A very accurate model is needed in this method, as there is no measurement of the friction in the joint. Such precise models can be adaptively identified [3], but they still must account for many nonlinear phenomena such as Coulomb friction, dependency on joint position, influence of changes in load and temperature, nonbackdriveability, etc. As a result of this complexity, the modeling, identification, and adaptation aspects of the model-based compensation method are not fully solved, thus making these techniques difficult to implement in practice.

The second friction compensation method computes the width and magnitude of a torque pulse necessary to provide a small joint displacement. The computation can use either an explicit model [1] or simple rules of qualitative reasoning [4]. This approach appears to be more practicable than model based compensation, and usually a few pulses are sufficient to accurately reach the desired position in spite of Coulomb friction. However, the pulse generation method is limited to applications for which the trajectory to reach the final position is not important, since only finite displacements are controlled.

The third technique is based on a joint torque control loop. The torque applied to the manipulator joint is sensed and fed back in a joint torque loop. This method has produced the best experimental results we have found in the literature for joint friction compensation. In experiments involving manipulators with high friction gear trains, this technique has reduced the effective friction torque by up to 97 % [5,6]. In addition, the method does not require any friction model and is very robust with respect to changes in load or friction torque magnitude. Unfortunately, most commercially available manipulators are not equipped with joint torque sensors. Installing them in the joints of an existing manipulator would be very difficult. Also, manipulators designed to include such sensors have a number of practical problems. For example, introducing flexures instrumented with strain gages in the joint adds structural flexibilities and decreases the overall performances of the manipulator [7]. Substantial nonlinearities in the sensor output can result from the complex loading on the sensor by a joint gear train. Finally, individual joint sensors are expensive, add to wiring complexity, and are subject to damage due to manipulator vibrations or overloads.

This paper proposes a new approach to deal with joint friction in manipulators performing fine motions, that overcomes the difficulties of the three methods discussed above. The method uses a six axis force/torque sensor mounted between the manipulator and its base (see Figure 1). Joint torques are estimated from the measurements provided by this sensor. The estimation process uses Newton-Euler equations of successive bodies. The estimated torques are used in joint torque loops as is done with direct torque measurements. A position control loop encloses the torque controller and provides it with desired torques computed from measured position errors.

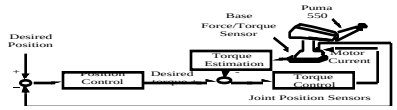


Figure 1: High precision control structure using a base force/torque sensor

2 Joint torque estimation

2.1 Theoretical issues

In this section, the basic dynamic equations used in the torque estimation process are developed.

Consider a manipulator mounted on a base force/torque sensor. The wrench W_b exerted by the manipulator on its support can be expressed as the sum of two wrenches:

$$W_b = W_g + W_d \tag{1}$$

where W_g is the wrench due to the gravity and W_d is the dynamic wrench due to the motions of the manipulator. It should be noted that the base sensor measures wrenches that corresponds only to forces and torques effectively transmitted to the manipulators links. Hence transmission friction does not appear in the measured base wrench.

The first step in the estimation process is to compensate for the gravity component W_g in order to estimate the dynamic component W_d. The gravity wrench is compensated for using the following model [8] :

$$W_d = W_b - W_g = W_b - \begin{pmatrix} F_g = \sum_{i=1}^{n} m_i g \\ M_g^{O_s} = \sum_{i=1}^{n} O_s G_i \times m_i g \end{pmatrix} \tag{2}$$

Where F_g and $M_g^{O_s}$ are the gravity force and moment at the center of the sensor O_s, respectively, m_i and G_i are the mass and the center of mass of link i, respectively.

In the following analysis, the gravity compensated wrench (W_d) is propagated through the successive bodies of the manipulator. This results in estimated joint torques that do not include the joint gravity component.

The Newton Euler equations of the first i links are, after gravity compensation:

$$\begin{cases} W_{0\to 1} = -W_d \\ W_{1\to 2} = W_{0\to 1} - W_{dyn_1} \\ \vdots \\ \vdots \\ W_{i\to i+1} = W_{i-1\to i} - W_{dyn_i} \end{cases} \tag{3}$$

where $W_{i\to i+1}$ is the wrench exerted by the link i on the link i+1 and W_{dyn_i} is the dynamic wrench for the link i. W_{dyn_i} can be expressed at any point A in terms of the acceleration $\dot{V}_{G_i}$ of G_i, the angular acceleration $\dot{\omega}_i$ and the angular velocity ω_i:

$$W_{dyn_i} = \begin{pmatrix} F_{dyn_i} = m_i \dot{V}_{G_i} \\ M^A_{dyn_i} = I_i\dot{\omega}_i + \omega_i \times I_i\omega_i + G_iA \times m_i \dot{V}_{G_i} \end{pmatrix} \tag{4}$$

where I_i is the inertia tensor of link i at G_i.

Summing the equations (3) yields:

$$W_{i\to i+1} = -W_d - \sum_{j=1}^{i} W_{dyn_j} \tag{5}$$

Given this wrench, the torque in joint i+1 is obtained by projecting the moment vector at O_i along z_i (Figure 2) :

$$\tau_{i+1} = -z_i^t \left[M_d^{O_i} + \sum_{j=1}^{i} \left(I_j\dot{\omega}_j + \omega_j \times I_j\omega_j + O_iG_j \times m_j \dot{V}_{G_j} \right) \right] \tag{6}$$

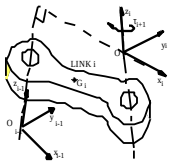


Figure 2 : Body-fixed coordinate frames for link i

2.2 Application to the Puma 550

The general equations presented above can be significantly simplified when applied to a particular manipulator. Here we consider a Puma 550 manipulator mounted on a base force/torque sensor, as shown in Figure 3.

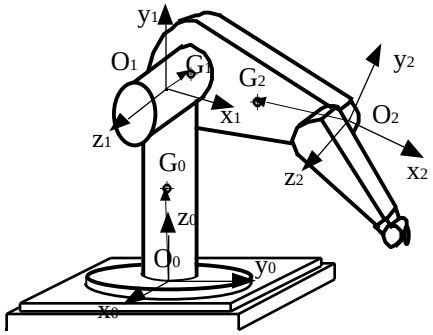


Figure 3 : Puma 550 link frames, attached following the Denavit-Hartenberg notation.

It has been shown experimentally that the gravity wrench can be efficiently and accurately estimated by developing equation (2) as a function of joint angles [9].

Furthermore, to estimate the torques at the first three joints of the Puma, we use the following assumptions :

- W_b is measured directly in O_1 (see Figure 3).
- The center of mass G_1 is on the z_1 axis
- Off-diagonal terms in the inertia tensors I_1 and I_2, expressed in bases (x_1,y_1,z_1) and (x_2,y_2,z_2) respectively, can be neglected.

These assumptions yield :

$$\tau_1 = -z_0{}^t M_d^{O_1} = -(0,0,1)^t M_d^{O_1} \quad (7)$$

$$\tau_2 = -z_1{}^t M_d^{O_1} = -(s_1 x_0, -c_1 y_0, 0)^t M_d^{O_1} \quad (8)$$

$$\begin{aligned}\tau_3 = \tau_2 &- a_2 (s_2 c_1, c_2 c_1, c_2)^t F_d \\ &+ \left[A_1 c_2 + A_2 s_2\right]\ddot{q}_1 + A_3 \ddot{q}_2 + A_4 \dot{q}_2^2 \\ &+ \left[A_5 c_2{}^2 + A_6 s_2{}^2 + A_7 c_2 s_2\right]\dot{q}_1^2\end{aligned} \quad (9)$$

where a_2, A_1 to A_7 are constant scalar values depending on masses, inertia and lengths of the two first joints (see appendix A), and (s_i, c_i) stands for ($\sin(q_i)$, $\cos(q_i)$).

2.3 Implementation

The torque estimation requires knowledge of joint positions, velocities and accelerations. Joint positions are precisely measured with optical incremental encoders. In addition, a Digital Signal Processor board acquires the encoder data and performs differentiations and filtering, at a sampling rate of 2500Hz, to compute the velocities and accelerations. By experiment, estimation of the position derivatives using this hardware has appeared to be sufficiently fast and precise, since neither noise nor delay corrupt the torque estimation process.

Knowledge of mass and inertia properties is also required in the estimation process. In the implementation of the algorithm, we have used values identified using the base force/torque sensor [9].

It should be noted that the torque estimation equations for the Puma 550 are not computationally intensive. Using a single 68020 VME board supporting VxWorks, a 300Hz sampling frequency was achieved to measure the base wrench, compensate for the gravity (Equation 2), compute the torques (Equations 7,8 and 9) and the torque control loop presented in the next section.

3 Torque control

3.1 Open loop results

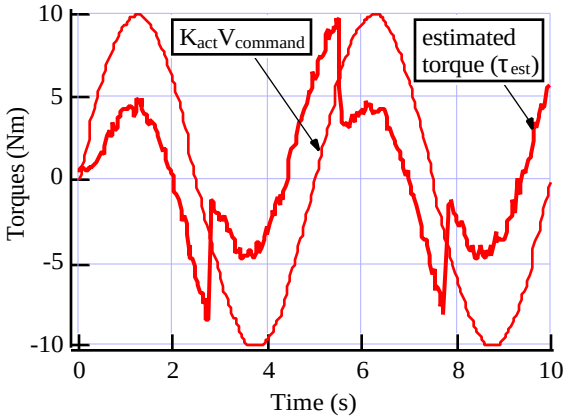
Open loop experiments have been conducted to provide a relevant model for the torque control design, and to evaluate the accuracy and the validity of the torque estimation process. They consisted of applying a given voltage to the input of the power amplifiers and simultaneously estimating the torques at the joints with the base sensing method. From the experimental results, a very simple model of the Puma actuators has been derived.

The amplifiers, actuators and transmissions can be modeled as a linear term K_{act}, with a disturbance torque τ_{dist} that accounts for unmodeled nonlinear effects. The torque τ_{load} provided by the actuator to the joint is:

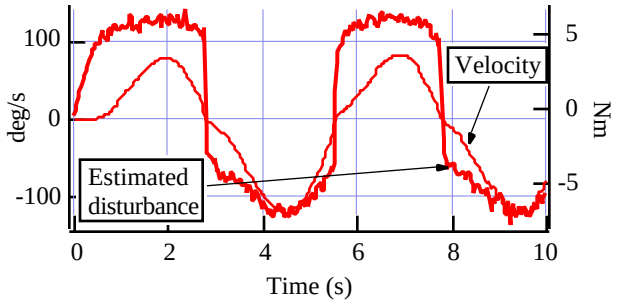
$$\tau_{load} = K_{act} V_{command} + \tau_{dist} \quad (10)$$

where $V_{command}$ is the controller voltage output.

Figure 4, which contains an open loop result for the first joint, shows the validity of the model. The base sensor estimated torque reproduces the input voltage sine wave with a disturbing torque whose sign is changed when the velocity sign changes (4-a). This disturbance torque appears to be mostly a Coulomb friction term (4-b)



a: Open loop torque response



b: Friction characteristics

Figure 4 : Open loop experiment for joint 1

Also, note that the estimated joint torque has very low noise, due to the quality of the force sensor and its electronics.

3.2 Closed loop control design

As shown in Figure 4, open loop experimental results exhibit very large Coulomb friction. In very fine motion applications, friction will be much larger in magnitude than the dynamic torque applied to the load. Hence, a high DC gain in the torque controller is required to compensate for this static disturbance.

Considering this, the torque control law implemented is an integral controller with feedforward compensation:

$$V_{command} = \frac{1}{K_{act}}\left[\tau_{des} + K_{int}\int_0^t (\tau_{des} - \tau_{est})\right] \quad (11)$$

where τ_{des} and τ_{est} are the desired and the actual (i.e. base-sensed) torques, respectively.

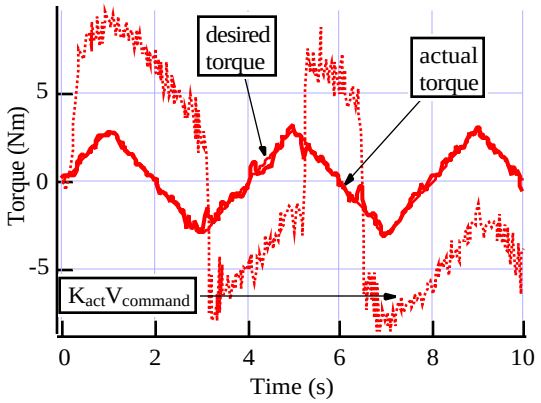
Linear analysis of an experimentally derived model has suggested that an integral compensator provides the best performance in this type of design [10]. It achieves low-pass filtering and zero steady state error, whereas a proportional compensator could introduce instability, and a

derivative compensator is ineffective and difficult to implement. While this study also suggests that a feedforward compensator should not be used in conjunction with integral control, our experimental work (with a real nonlinear system) shows some improvement in the torque control performance when a feedforward term is used.

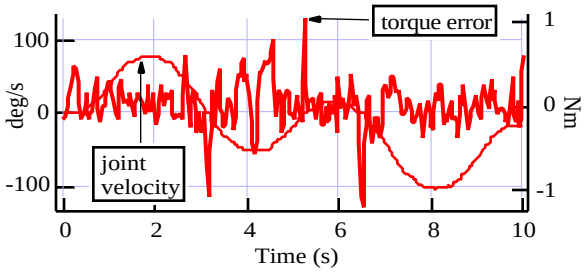
The control gain K_{int} was tuned to 75% of the value that caused experimental structural oscillations.

3.3 Experimental results

Figure 5 demonstrates the effectiveness of base sensed torque control for the first joint of the Puma 550.



a: Torque response



b: Torque error

Figure 5: Joint 1 torque control experimental results

In this example, the desired torque is a triangular function with a maximum value of 3 Nm, while the dry friction is more than 5 Nm. Without torque feedback, the actual torque applied to the link would simply be zero, as the friction would be larger than the motor torque. However, with torque feedback, the experimental results show that the actual torque remains very closed to its desired value (Figure 5-a). The torque controlled motor must produce nearly 8 Nm to obtain the next 3 Nm required by the command. Figure (5-b) shows that when the sign of the velocity changes, involving large torque disturbance, the torque error peak remains small (±1 Nm, i.e. 20% only of the Coulomb friction) and is quickly compensated for.

In Figure 6, experimental results for the Puma's second joint are shown for a of a 3 Hz, ±10 Nm input. This square wave input torque creates a triangular velocity wave with a constant sign. Therefore no Coulomb friction compensation is observed in the motor command. The rise time is small (about 20 ms).

Also, the gravity is automatically compensated for by the torque loop. Since the joint position is almost constant, the motor torque $K_{act}V_{command}$ mainly consists of the sum of a constant gravity compensation and the desired torque. Note that with the base sensor approach, the gravity is compensated for only once, from the base measured wrench, whereas using direct torque sensing methods, it would be required to provide a gravity joint torque compensation model for each joint of the manipulator.

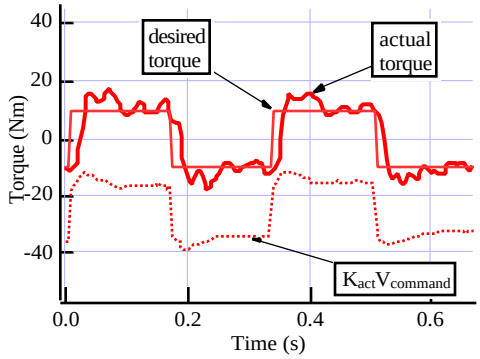


Figure 6 : Joint 2 torque control results

4 Position control with torque feedback

4.1 Controller design

In some experiments that have been performed with the base measured torque feedback method, the manipulator had a zero desired torque, and external forces were applied to the end-effector. In this case, the robot behaves virtually as a frictionless and free-floating device. With such quality performance, it is easy to obtain precise position control using a simple PD loop enclosing the torque controller. The final controller is (see Figure 7):

$$V_{command} = \frac{1}{K_{act}}\left[\tau_{des} + K_{int}\int_0^t(\tau_{des} - \tau_{est})\right] \quad (12)$$

$$\text{with} : \tau_{des} = K_p(q_d - q) + K_d(\dot{q}_d - \dot{q}) \quad (13)$$

Where K_p and K_d are the proportional and derivative diagonal gain matrices, respectively.

Note that because the torque control loop eliminates any significant frictional effects, the position control tuning is very straightforward and corresponds to a linear second order system.

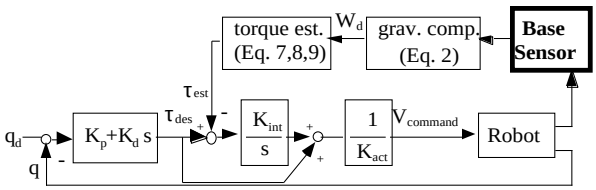
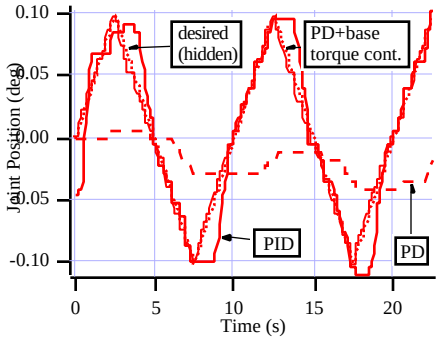


Figure 7: The precise position control scheme with base sensed torque feedback

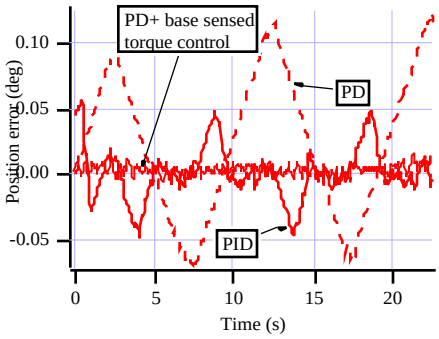
The robustness and effectiveness of the base sensed torque control method is illustrated in the following experimental results.

4.2 Joint space experimental results

The task considered is to move the joint 1 very slowly, tracking a triangular wave. The magnitude of the desired motion is +/-0.1 degrees, with a period of 10 seconds. This corresponds to a desired velocity of 7 encoder counts per second.



a: Tracking performance



b: Position error

Figure 8: Precise position control results

The base sensed torque feedback control method (Figure 7) has been experimentally compared with conventional PD and PID controllers for this task. For these three controllers, the proportional and derivative positions gains have been tuned to provide a bandwidth of 5 Hz and a damping ratio of 0.5. The integral gain in the PID control has been selected to be quite high, equal to 80% of the smallest value exhibiting instability.

Figure 8 displays the improved performance provided by the base sensed torque feedback. Conventional PD control leads to almost no motion, due to dry friction. The PID controller performs much better, and provides a zero steady state positioning error. However, when the sign of the velocity changes, the position integral compensator requires a long time (2.5 s) to compensate for the friction disturbance, resulting in lack of positioning precision. On the other hand, the base sensed torque feedback control method compensates rapidly for the Coulomb friction at velocity sign changes (~50 ms) and the position error remains close to zero during the task.

Table 1 quantitatively summarizes the performances of the three controllers. The results of base sensed torque control show that the resolution of the encoder is reached. An encoder count corresponds to a 0.0058 degree angle, and thus that the Root Mean Square error (0.0042 deg) is less than one encoder count throughout the entire task.

Controller	Max. Error (deg)	Root Mean Square error (deg)	Integral Square error (deg^2 s)
PD	0.12	0.059	$7.7\ 10^{-2}$
PID	0.056	0.020	$9.1\ 10^{-3}$
PD+base sensed torque control	0.012	0.0042	$4.0\ 10^{-4}$

Table I: Summary of position control performances.

4.3 Cartesian space experimental results

Cartesian space motion tasks require the end-effector to track a desired trajectory. From this desired path, the desired trajectories of the first three joints (the wrist joints are locked during these experiments) are computed off-line using inverse kinematics. Thus, the control scheme is unchanged (Figure 7).

For very fine motion tasks, the estimation process is simplified. It has been experimentally found that the precision performance is not affected by assuming the following:

1 W_g is assumed to be constant, and set equal to the initial static wrench measured with the base sensor.

2 The dynamic terms in equations (7) to (9) are neglected.

Hence the experimental results shown thereafter have been obtained with a controller that does not require any knowledge of the robot's mass properties.

The desired end-effector trajectory is a circle with a 350μm radius. The robot configuration is selected such that the corresponding joint displacements are maximized. In such a configuration (Figure 9), the maximum magnitude of the joint motions is 0.1 degrees.

To verify the end-effector positioning performance, an external position sensor was used. This 2D photodetector measures the position of a spot of light created by a laser mounted on the robot's end-effector (Fig. 9).

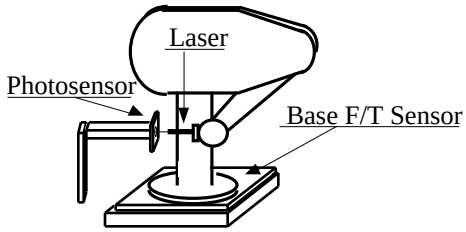


Figure 9: Setup for Cartesian space experiments

Figure 10 shows the Cartesian tracking results in the sensor coordinate frame. It should be noted that, since the motion is cyclic, the sign of the velocity changes at least

once in all the three joints during the motion. This results in large frictional disturbances. In spite of these perturbations, the precision remains excellent: the maximum absolute position error is less than 30µm.

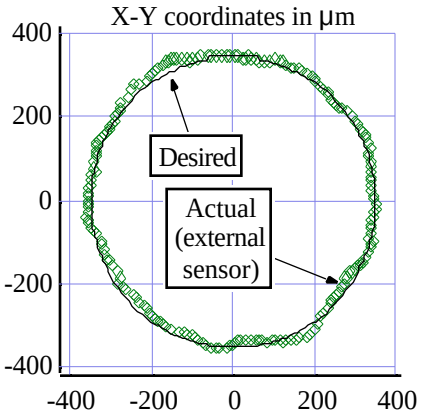


Figure 10: Cartesian tracking results

5 Summary and conclusions

In this paper, we have presented a new method to compensate for joint friction in fine motion control of manipulators. Previous methods require either a complex modeling and identification process or expensive and delicate sensors that must be designed into the equipment.

Our method is more practical. The hardware design is a 6 axis force/torque sensor mounted at the base of the manipulator. The sensor is external to the robot and hence can be easily mounted under existing manipulators.

A torque estimation algorithm, as well as a controller design, have been presented in the paper. No friction model is required during any stage of the development. In addition, for very fine motion applications, the method does not require any knowledge of the mass properties.

The experimental results show a very substantial enhancement of the manipulator capabilities. At the joint level, the precision reaches encoder's resolution. At the end-effector, during very slow displacements, the position error remains smaller than 30µm.

Based on these promising results, extensions of the Base Force/Torque Sensor approach are currently being investigated for delicate tasks that require both fine motions and very small interaction forces.

6 Acknowledgments

Authors want to thank Electricité de France, Direction des Etudes et Recherches, Ensembles de Production - SDM - P29, Chatou, France for the financial support they provided for this work.

7 References

[1] B Armstrong, Control Of Machines With Friction, Kluwer Academic Publishers, Boston, USA, 1991

[2] M.R Popovic, K.B. Shimoga and A.A. Goldenberg, Model Based Compensation Of Friction In Direct Drive Robotic Arms, J. of Studies in Informatics and Control, Vol. 3, No 1., pp. 75-88, March 1994.

[3] C. Canudas de Wit, Adaptive Control Of Partially Known Systems, Elsevier, Boston, USA, 1988.

[4] M.R. Popovic, D.M. Gorinevsky and A.A. Goldenberg, Accurate Positioning Of Devices With Nonlinear Friction Using Fuzzy Logic Pulse Controller, Int. Symposium of Experimental Robotics, ISER' 95, *preprints*, pp. 206-211.

[5] J.Y.S. Luh, W.B. Fisher and R.P. Paul, Joint Torque Control By Direct Feedback For Industrial Robots, IEEE Trans. on Automatic Control, vol. 28, No 1, Feb. 1983.

[6] L.E. Pfeffer, O. Khatib and J. Hake, Joint Torque Sensory Feedback Of A PUMA Manipulator, IEEE Trans. on Robotics and Automation, vol. 5, No 4, pp. 418-425, 1989

[7] Hake J.C. and Farah J., Design Of A Joint Torque Sensor For The Unimation PUMA 500 Arm, Final Report, ME210, University of Stanford, CA, 1984

[8] H. West, E. Papadopoulos, S. Dubowsky and H. Chean, A Method For Estimating The Mass Properties Of A Manipulator By Measuring The Reaction Moment At Its Base, Proc. IEEE Int. Conf. on Robotics and Automation, pp , 1989

[9] T. Corrigan and S. Dubowsky, Emulating Micro-Gravity In Laboratory Studies Of Space Robots, Proc. ASME Mechanisms Conf., pp. , 1994

[10] R. Volpe and P. Khosla, An Analysis Of Manipulator Force Control Strategies Applied To An Experimentally Derived Model, Proc. IEEE/ RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1989-1997, 1992

Appendix A

The expressions for the constant parameters used in the torque estimation for the Puma 550 (Eq.. 7,8,9) are:

$$A_1 = m_2 r_{2_y}\left(r_{2_z} + d_2\right)$$
$$A_2 = m_2 r_{2_x}\left(r_{2_z} + d_2\right) - m_1 a_2 r_{1_z}$$
$$A_3 = -I_{2_{zz}} - \left(r_{2_x}{}^2 + r_{2_x} a_2 + r_{2_y}{}^2\right) m_2$$
$$A_4 = -m_2 a_2 r_{2_y}$$
$$A_5 = -m_2 r_{2_y}(a_2 + r_{2_x})$$
$$A_6 = A_4 - A_5$$

where a_2 and d_2 are the Denavit-Hartenberg parameters describing the transformation between the frames 1 and 2, $O_2G_2 = r_2 = r_{2_x} x_2 + r_{2_y} y_2 + r_{2_z} z_2$ (see Figure 3) and

$$I_2 = \begin{pmatrix} I_{2_{xx}} & 0 & 0 \\ 0 & I_{2_{yy}} & 0 \\ 0 & 0 & I_{2_{zz}} \end{pmatrix}$$ in the base (x_2, y_2, z_2).