Proceedings of the 1996 IEEE
International Conference on Robotics and Automation
Minneapolis, Minnesota - April 1996

Identifying Mass Parameters for Gravity Compensation and Automatic Torque Sensor Calibration

Donghai Ma

MPB Technologies Inc.
151 Hymus Boulevard
Pointe Claire, Quebec H9R 1E9

John M. Hollerbach

Dept. of Computer Science
University of Utah
Salt Lake City, UT 84112

Abstract

This paper presents a new and simple procedure to identify link mass parameters used for gravity compensation and on-line automatic torque sensor calibration for a robot manipulator. The approach employs single-joint rotation and a recursive procedure that proceeds distally to proximally to identify composite mass moments. Experimental results are presented for the Sarcos Dextrous Arm.

1 Introduction

Gravity loading for all but the fastest robot motions dominates manipulator joint torque. For anthropomorphic-sized arms it has been shown that motions have to be completed in around a half second or less for inertial forces to be greater [6]. Even for fast motions, the use of a PD controller with gravity compensation apparently performs almost as well as a full feedforward controller that accommodates full inertial terms [1]. Figure 1 shows the block diagram for such a compensator.

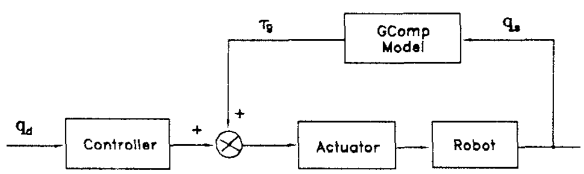

Figure 1: Gravity compensator diagram (state feedback is not shown here).

Gravity compensation (GComp) is particularly important when using a force-reflecting exoskeleton in teleoperation, such as the Sarcos Dextrous Arm Master. It is exceptionally burdensome for the operator to lift the weight of the master when attempting force reflection. Hence it is imperative that the weight of the master should be automatically compensated. It would be desirable to cancel inertial forces as well, but the typical small-motion tasks using a master mean that gravity loading dominates. Moreover, attempting to reduce or cancel inertia is known to be tricky [1], as instability easily results.

Figure 2: The Sarcos Dextrous Arm Master and Slave.

We propose a simple method for gravity compensation, which is a natural extension of earlier work on torque sensor calibration [11]. This torque sensor calibration method is the only published method that can calibrate manipulator joint torque sensors "in vivo" (without disassembly). This method employs single-joint rotation and fitting of sinusoidal models to the torque versus position readings. Identified are the torque sensor offsets and gains; as a by-product position sensor offsets and gains are also found. A feature of the method is attachment of a reference mass, where the kinematics of attachment (i.e., the moment arm) is not presumed known. No knowledge is assumed about the link mass properties either.

It turns out that after this torque sensor calibration, the same data can be employed to derive the gravity parameters which load the joints. This paper presents the procedure for this. In fact the torque sensor calibration and GComp procedures are tightly coupled in two ways. (1) It doesn't make sense to calibrate the gravity parameters unless one has calibrated torque sensors; the factory-specified torque sensor calibration was found to have drifted and recalibration is required. (2) Once the gravity parameters have been determined, the link masses may then be used as known loads to the joints to recalibrate the torque sensors in the field. This procedure would be more automatic and substantially simpler than our initial procedure [11], as the calibration mass is not required.

Accurate values of the mass parameters are typically unknown even to the manufacturers and methods need to be developed to identify them. Full inertial parameter estimation methods are by now a well-known part of the robotics literature [2, 8] in which joint torque is correlated with joint position, velocity, and acceleration. One drawback of these methods is the differentiation to obtain velocity and acceleration, which introduces noise and imprecision. Secondly, this approach will complicate the identification process if we only need to find out mass parameters, which is the case for GComp. The proposed approach, by identifying link mass parameters statically and separately, can avoid the noisy differentiation process and result in a simpler, more accurate identification procedure.

A straightforward way of identifying mass parameters separately is to follow inertia parameter estimation procedures, while setting all dynamic terms to zero. This procedure is employed by the manufacturer to implement GComp on the Sarcos Dextrous Arm [13]. Simultaneously multiple joint static rotation is required and care must be taken to choose the arm poses for better numerical accuracy. In comparison, our approach may be considered to have a more principled method of pose selection and of determination of identifiable parameter combinations.

2 Notation and Background

We assume that the manipulator kinematics are known (perhaps through an earlier kinematic calibration procedure) and that the joint torque and position sensors have been calibrated [11]. Standard Denavit-Hartenberg (DH) parameters are employed; additional notation follows (Figure 3).

$\mathbf{p}_i^*$ is a vector from coordinate origin $i-1$ to coordinate origin i.

$\mathbf{s}_i$ is a vector from joint i to the link i center of gravity (COG).

${}^i\mathbf{s}_i$ expresses $\mathbf{s}_i$ with respect to (w.r.t.) link i coordinates. Similarly for other vectors.

${}^i\bar{\mathbf{s}}_i = \mathbf{R}_x(\alpha_i)^i\mathbf{s}_i$ expresses $\mathbf{s}_i$ w.r.t. a modified link i coordinate system, where after the rotation the modified $\bar{\mathbf{z}}_i$ axis is parallel to $\mathbf{z}_{i-1}$. Then

$$ {}^{i-1}\mathbf{s}_i = \mathbf{R}_z(\theta_i)^i\bar{\mathbf{s}}_i \tag{1}$$

Similarly for other vectors.

As will be seen, the z-component of ${}^i\bar{\mathbf{s}}_i$ does not contribute to joint torque.

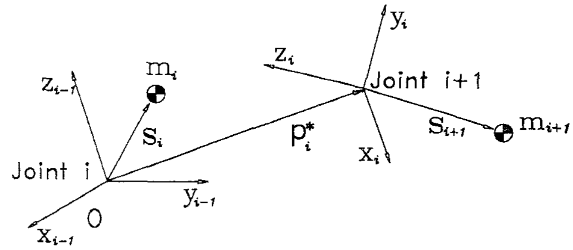

Figure 3: Link coordinate assignment and COG.

Next we extract pertinent developments from [11]. The composite mass moment $\bar{\mathcal{R}}_i$ about joint i considers all links from i to the most distal link n to form one rigid body; hence it is configuration-dependent. Express $\bar{\mathcal{R}}_i$ in the modified link i coordinates with the Euler notation:

$$ {}^i\bar{\mathcal{R}}_i = \begin{bmatrix} \mathcal{R}_i\, s\eta_i\, c\xi_i \\ \mathcal{R}_i\, s\eta_i\, s\xi_i \\ \mathcal{R}_i\, c\eta_i \end{bmatrix} \tag{2}$$

where $\mathcal{R}_i$ is the vector length and η_i and ξ_i are two angles that define the vector direction (Figure 4). The composite mass moment $\mathcal{R}_i$ influences joint gravity torque τ_i through:

$$ \tau_i = \mathbf{z}_{i-1}\cdot(\mathcal{R}_i \times \mathbf{g}) = ({}^{i-1}\mathbf{g}\times{}^{i-1}\mathbf{z}_{i-1})\cdot\mathbf{R}_z(\theta_i)^i\bar{\mathcal{R}}_i \tag{3}$$

where $\mathbf{R}_i = \mathbf{R}_z(\theta_i)\mathbf{R}_x(\alpha_i)$ defines the standard DH rotation matrix for joint i, and the gravity vector ${}^{i-1}\mathbf{g}$ is expressed in link $i-1$ coordinates and is presumed known. Expanding (3),

$$ \tau_i = A_i\ \sin(\theta_i + \Phi_i) \tag{4}$$

where we define

$$ \begin{aligned} A_i &= \mathcal{R}_i s\eta_i \sqrt{({}^{i-1}g_x)^2 + ({}^{i-1}g_y)^2} \\ \Phi_i &= \xi_i - \arctan({}^{i-1}g_y/{}^{i-1}g_x) \end{aligned} \tag{5}$$

Equation (4) reveals the sinusoidal relation for joint i between its gravity torque and joint position, assuming other joints are fixed. Experimentally we rotate joint i in steps for its full range while keeping other joints stationary. Joint i torque and position sensors are sampled for each step, to obtain data pairs $[\tau_i, \theta_i]$

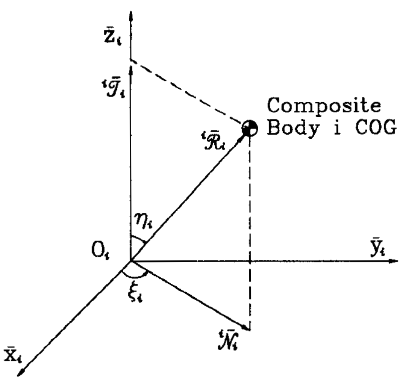

Figure 4: The Euler representation of the composite mass moment ${}^i\bar{\mathcal{R}}_i$.

on a sinusoidal curve. Ordinary least squares then fits the data pairs $[\tau_i, \theta_i]$ to (4). Parameters A_i and Φ_i are extracted from the sinusoidal curve fitting.

Since gravity is known, we can then find $\mathcal{R}_i s\eta_i$ from A_i and ξ_i from Φ_i to obtain the x, y components of ${}^i\bar{\mathcal{R}}_i$ in (2). Consequently, define the identifiable x,y component ${}^i\bar{\mathcal{N}}_i$ of ${}^i\bar{\mathcal{R}}_i$:

$$ {}^i\bar{\mathcal{N}}_i = \begin{bmatrix} \mathcal{R}_i\, s\eta_i\, c\xi_i \\ \mathcal{R}_i\, s\eta_i\, s\xi_i \\ 0 \end{bmatrix} \qquad (6) $$

The z component, defined as ${}^i\bar{\mathcal{T}}_i$ such that ${}^i\bar{\mathcal{R}}_i = {}^i\bar{\mathcal{N}}_i + {}^i\bar{\mathcal{T}}_i$ (Figure 4), cannot be found at this point and does not influence joint i torque because it is parallel to the rotation axis. However, it may influence more proximal joint torques. Expression (3) can be simplified as:

$$ \tau_i = ({}^{i-1}\mathbf{g} \times {}^{i-1}\mathbf{z}_{i-1}) \cdot \mathbf{R}_z(\theta_i) {}^i\bar{\mathcal{N}}_i \qquad (7) $$

which emphasizes the need to know the normal component ${}^i\bar{\mathcal{N}}_i$ of the composite mass moment ${}^i\bar{\mathcal{R}}_i$; there is no need to identify the z component ${}^i\bar{\mathcal{T}}_i$.

3 Gravity Torque Model

This section develops a recursive procedure to calculate ${}^i\bar{\mathcal{N}}_i$. This procedure is equivalent to specialization of previous research on the identifiable combinations of all inertial parameters [9, 10, 12] to just the gravity components, but we present our own derivation for greater clarity in this particular context. The recursive relation among composite mass moments $\mathcal{R}_i$ is:

$$ \begin{aligned} \mathcal{R}_n &= m_n \mathbf{s}_n \\ \mathcal{R}_{n-1} &= m_{n-1}\mathbf{s}_{n-1} + m_n \mathbf{p}^*_{n-1} + \mathcal{R}_n \\ &\cdots \\ \mathcal{R}_i &= m_i \mathbf{s}_i + (\textstyle\sum_{j=i+1}^n m_j)\mathbf{p}^*_i + \mathcal{R}_{i+1} \end{aligned} \qquad (8) $$

To identify these $\mathcal{R}_i$, we recast (8) into quantities that for joint i are (1) already known or computable, (2) identifiable by joint i rotation, or (3) not identifiable but passed down as a constant to joint $i-1$.

This derivation is facilitated by defining $\mathbf{r}_i$ as the mass moment component of $\mathcal{R}_i$ acting at joint i that is constant, i.e., does not depend on the distal joint angles from i to n. At the end, ${}^n\bar{\mathcal{R}}_n$ and ${}^n\bar{\mathbf{r}}_n$ are the same:

$$ {}^n\bar{\mathcal{R}}_n = m_n {}^n\bar{\mathbf{s}}_n = {}^n\bar{\mathbf{r}}_n \qquad (9) $$

As before, we separate from ${}^n\bar{\mathbf{r}}_n$ the x, y component ${}^n\bar{\mathbf{n}}_n$ which influences joint n torque, from the z component ${}^n\bar{\mathbf{t}}_n$ which does not:

$$ {}^n\bar{\mathbf{r}}_n = {}^n\bar{\mathbf{t}}_n + {}^n\bar{\mathbf{n}}_n \qquad (10) $$

In this case, ${}^n\bar{\mathcal{N}}_n = {}^n\bar{\mathbf{n}}_n$ and ${}^n\bar{\mathcal{T}}_n = {}^n\bar{\mathbf{t}}_n$, but not for $i < n$. Thus ${}^n\bar{\mathbf{n}}_n$ can be identified completely from joint n rotation from the discussion in Section 2. But ${}^n\bar{\mathbf{t}}_n$ is unknown and will be passed down to link $n-1$ to be identified in linear combination with other mass moment components that influence joint torque τ_{n-1}.

To reflect this idea, for link $n-1$ the mass moment relation (8) is modified as follows:

$$ \begin{aligned} {}^{n-1}\bar{\mathcal{R}}_{n-1} &= m_{n-1}{}^{n-1}\bar{\mathbf{s}}_{n-1} + m_n {}^{n-1}\bar{\mathbf{p}}^*_{n-1} \\ &\quad + {}^{n-1}\bar{\mathbf{t}}_n + {}^{n-1}\bar{\mathbf{n}}_n \\ &= {}^{n-1}\bar{\mathbf{r}}_{n-1} + {}^{n-1}\bar{\mathbf{n}}_n \\ &= {}^{n-1}\bar{\mathbf{t}}_{n-1} + {}^{n-1}\bar{\mathbf{n}}_{n-1} + {}^{n-1}\bar{\mathbf{n}}_n \end{aligned} \qquad (11) $$

where the constant mass moment portion ${}^{n-1}\bar{\mathbf{r}}_{n-1}$ for link $n-1$,

$$ {}^{n-1}\bar{\mathbf{r}}_{n-1} = m_{n-1}{}^{n-1}\bar{\mathbf{s}}_{n-1} + m_n {}^{n-1}\bar{\mathbf{p}}^*_{n-1} + {}^{n-1}\bar{\mathbf{t}}_n \qquad (12) $$

has been broken into x, y components ${}^{n-1}\bar{\mathbf{n}}_{n-1}$ and z component ${}^{n-1}\bar{\mathbf{t}}_{n-1}$. Note that ${}^{n-1}\bar{\mathbf{r}}_{n-1}$ consists of components from both link $n-1$ and n, all of which are constant in link $n-1$ coordinates and independent of the joint n position. This definition should be contrasted with the traditional definition of the mass moment $m_{n-1}\mathbf{s}_{n-1}$ for link $n-1$, which depends only on the mass distribution of its link. In (11) ${}^{n-1}\bar{\mathbf{t}}_n$ was passed down as an unknown when identifying link n components, but here appears as a constant in linear combination with other constant components. We also have kept ${}^{n-1}\bar{\mathbf{n}}_n$ separated; it is completely known, but depends on θ_n and must be recomputed for different arm poses.

When rotating joint $n-1$ by itself to generate a sinusoidal curve, we identify the x, y components ${}^{n-1}\bar{\mathcal{N}}_{n-1}$ of ${}^{n-1}\bar{\mathcal{R}}_{n-1}$ in (11):

$$ {}^{n-1}\bar{\mathcal{N}}_{n-1} = {}^{n-1}\bar{\mathbf{n}}_{n-1} + \mathrm{diag}(1,1,0){}^{n-1}\bar{\mathbf{n}}_n \qquad (13) $$

where the square matrix diag$(1,1,0)$ expresses the operation of copying just the x, y components of known vector ${}^{n-1}\bar{\mathbf{n}}_n$. Therefore we can find ${}^{n-1}\bar{\mathbf{n}}_{n-1}$. As before, the z component ${}^{n-1}\bar{\mathbf{t}}_{n-1}$ is unidentifiable and is passed down to the next link $n-2$, where it similarly appears as a constant to be identified in linear combination with other parameters in the mass moment of link $n-2$.

We now complete the derivation recursively. Suppose for link $i+1$ we have determined the composite mass moment in the following form:

$$ {}^{i+1}\bar{\mathcal{R}}_{i+1} = {}^{i+1}\bar{\mathbf{t}}_{i+1} + \sum_{j=i+1}^{n} {}^{i+1}\bar{\mathbf{n}}_j \tag{14} $$

For example, the expression (11) for joint $n-1$ has this form, which serves as the first step of the induction. All components of ${}^{i+1}\bar{\mathbf{n}}_j$ have been identified or can be computed up to this point. The component ${}^{i+1}\bar{\mathbf{t}}_{i+1}$ is unidentifiable here, and is passed down to link i:

$$ \begin{aligned} {}^{i}\bar{\mathcal{R}}_i &= m_i {}^{i}\bar{\mathbf{s}}_i + (\sum_{j=i+1}^{n} m_j){}^{i}\bar{\mathbf{p}}_i^* + {}^{i}\bar{\mathcal{R}}_{i+1} \qquad (15) \\ &= m_i {}^{i}\bar{\mathbf{s}}_i + (\sum_{j=i+1}^{n} m_j){}^{i}\bar{\mathbf{p}}_i^* + {}^{i}\bar{\mathbf{t}}_{i+1} + \sum_{j=i+1}^{n} {}^{i}\bar{\mathbf{n}}_j \\ &= {}^{i}\bar{\mathbf{t}}_i + {}^{i}\bar{\mathbf{n}}_i + \sum_{j=i+1}^{n} {}^{i}\bar{\mathbf{n}}_j \end{aligned} $$

where again we identify the x, y components ${}^{i}\bar{\mathbf{n}}_i$ and the z component ${}^{i}\bar{\mathbf{t}}_i$ of the constant link i mass moment component ${}^{i}\bar{\mathbf{r}}_i$:

$$ {}^{i}\bar{\mathbf{r}}_i = m_i {}^{i}\bar{\mathbf{s}}_i + (\sum_{j=i+1}^{n} m_j){}^{i}\bar{\mathbf{p}}_i^* + {}^{i}\bar{\mathbf{t}}_{i+1} \tag{16} $$

Again, we identify through single-joint rotation of joint i the x, y components ${}^{i}\bar{\mathcal{N}}_i$ of ${}^{i}\bar{\mathcal{R}}_i$ in (16):

$$ {}^{i}\bar{\mathcal{N}}_i = {}^{i}\bar{\mathbf{n}}_i + \mathrm{diag}(1,1,0)(\sum_{j=i+1}^{n} {}^{i}\bar{\mathbf{n}}_j) \tag{17} $$

Therefore we can find ${}^{i}\bar{\mathbf{n}}_i$, since all terms to its right are identifiable or computable. This completes the induction.

After identifying ${}^{i}\bar{\mathbf{n}}_i$ of link i, for i=n,...,1, we are ready to calculate the gravity torque on each joint for any configuration of the arm, by equation (7) and

$$ {}^{i}\bar{\mathcal{N}}_i = \mathrm{diag}(1,1,0)\sum_{j=i}^{n} {}^{i}\bar{\mathbf{n}}_j \tag{18} $$

4 Identification Algorithm

We briefly recapitulate the procedure to identify mass moment components and compute gravity torques. Begin at the last joint, and repeat Steps 1 to 4 below recursively from $i = n$ to $i = 1$.

Step 1: Sample joint torque and position sensor readings during static rotation of joint i. In our case, this process has been done in [11] when we calibrated torque sensor offsets.

Step 2: Fit torque-position data to the sinusoidal model (4) and identify ${}^{i}\bar{\mathcal{N}}_i$ from (6).

Step 3: Compute ${}^{i}\bar{\mathbf{n}}_i$ from (17).

Step 4: Calculate the gravity torque (7), where ${}^{i}\bar{\mathcal{N}}_i$ is obtained from (18).

link i	$m_i{}^{i}s_{ix}$	$m_i{}^{i}s_{iy}$	$m_i{}^{i}s_{iz}$
1	0.1116	0.0485	-0.6305
2	-0.1063	0.3289	0.5566
3	0.1256	0.2997	0.8100
4	0.0437	0.2184	-0.0692
5	0.0376	-0.0057	0.1491
6	-0.0217	-0.0620	0.0149
7	0.1132	-0.0594	0.8770

Table 1: Assumed mass moment parameters (traditional definition) used in simulation.

link i	${}^{i}\bar{n}_{ix}$	${}^{i}\bar{n}_{iy}$	${}^{i}\bar{n}_{iz}$
1	0.1116	-0.9594	0
2	-0.1063	-4.6877	0
3	0.1256	-1.0284	0
4	0.0437	1.5219	0
5	0.0376	0.0871	0
6	-0.0217	0.6686	0
7	0.1132	-0.0597	0

Table 2: The identified x,y component of the link mass moment in simulation.

5 Simulations

Simulations were first conducted to verify the identification algorithm. The simulation is done in Matlab^{TM}, using the Robotics Toolbox from Corke [3] and the System Identification Toolbox from Mathworks. Arbitrarily assumed mass moment parameters (Table 1) are used to generate gravity torque for each joint when the link rotates about the joint. These mass moments $m_i{}^i\mathbf{s}_i$ are the traditional intrinsic link i mass moments (Figure 3).

The algorithm described in Section 4 was employed to identify the x, y components ${}^i\bar{\mathbf{n}}_i$ of the composite constant link mass moments ${}^i\mathbf{r}_i$. The identified parameters are shown in Table 2. The results in general are not comparable because of the different definitions of $m_i{}^i\mathbf{s}_i$ and ${}^i\bar{\mathbf{n}}_i$, although for the Sarcos Dextrous Arm it turns out that $m_i{}^i s_{ix} = {}^i\bar{n}_{ix}$ because all skew angles are $\alpha_i = \pi/2$. The y-components though are different. To verify the identification result, we computed the gravity torque from the identified parameters. Results agree with those computed from the assumed parameters.

6 Experiments

Experiments were then performed on the Sarcos Dextrous Arm Slave (Figure 5). The Sarcos Dextrous Arm Slave is an advanced hydraulic manipulator with 7-DOF revolute joints and a 3-DOF, 3-fingered gripper (Figure 2). Each joint is instrumented with high precision position and torque sensors, whose readings are accessible from the user level computer via a low level joint controller and mid level computer [13]. Work has been done to calibrate both the kinematic model and the joint sensor parameters [5, 7].

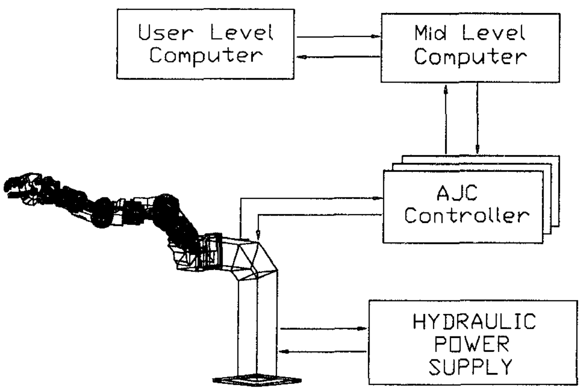

Figure 5: Experimental setup.

We applied the identification procedures of Section 4 to the data we sampled during the torque sensor offset calibration in [11]. We fixed joints that are to be stationary while another rotates, by applying a bias torque to push and hold these joints against the mechanical stops. The identified parameters are given in Table 3 for the first 6 joints. We verified the identification results by comparing the model output torque with joint torque sensor reading when the Arm is at the same pose. The result is shown in Figure 6, and shows reasonable agreement.

link i	${}^i\bar{n}_{ix}$	${}^i\bar{n}_{iy}$	${}^i\bar{n}_{iz}$
1	0.7022	0.4967	0
2	-0.6180	6.6570	0
3	0.2125	-0.1663	0
4	-0.2865	-3.0014	0
5	0.0213	-0.0207	0
6	0.0276	-0.0417	0

Table 3: Experimental results of the mass moment identification.

7 Discussion

This paper has presented a novel, simple and robust algorithm to identify the link mass parameters statically. Our algorithm doesn't explicitly identify the traditional mass moment for each link. Instead, it identifies all components contributing to the gravity torque on one joint. The constant mass moment portion of a proximal link may contain components of the more distal links. The algorithm assumes that the robot joint torque and position sensors have previously been calibrated [11]. In our case, the experimental data is the same as from the torque sensor offset calibration; hence no new experiment is needed. In fact, one can argue that torque sensor calibration is an immediate prerequisite to gravity parameter determination, in order to achieve the most accurate results.

The algorithm rotates joints singly, and fits a sinusoidal curve to the resulting data. Simulations and experiments verify the method. Following the identification of the robot link mass moment, one may build a gravity compensator to cancel out gravity disturbance for teleoperation and motion tracking. Moreover, a gravity torque observer based on the identified parameters can be used to implement fully automatic on-line torque sensor calibration.

Our procedure identifies the mass parameters for each link independently. The advantage of this approach to others, such as those requiring all robot joints to move simultaneously, is that it subdivides the general problem into a set of problems of lower complexity, thus achieving good stability and numerical precision. A disadvantage is that identification errors of distal links will propagate to proximal links. This can be overcome by re-identifying the parameters using all joint data simultaneously and a global optimization procedure, starting with nominal values the parameters identified from the proposed recursive

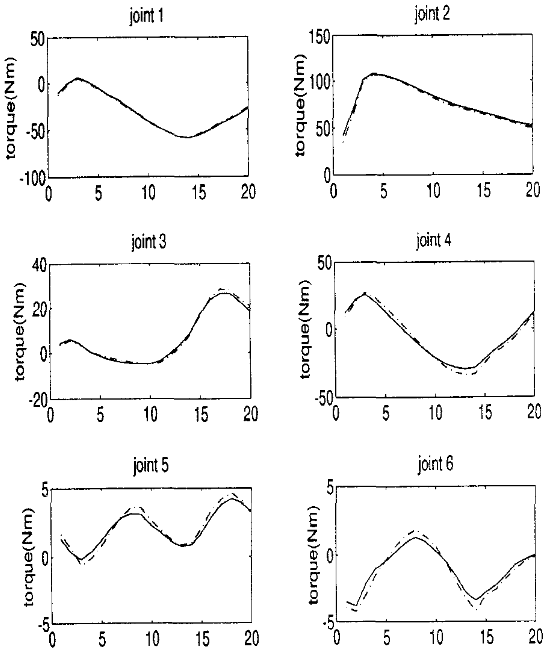

Figure 6: Experimental verification of the identified GComp model. The x axis is the pose number and the y axis is the gravity torque. Solid lines come from the identified model while dashed lines come from the joint torque sensor.

method. Although conceptually straightforward, we have not yet implemented this re-identification.

Acknowledgements

Support for this research was provided by PRECARN under the Telerobotics Development Systems project, by the NSERC NCE IRIS II Project AMD-5, and by NSF Grant MIP-9508588.

References

[1] An, C.H., Atkeson, C.G., and Hollerbach, J.M., *Model-Based Control of a Robot Manipulator.* Cambridge, MA: MIT Press, 1988.

[2] Atkeson, C.G., An, C.H., and Hollerbach, J.M., "Estimation of inertial parameters of manipulator links and loads," *Int. J. Robotics Research*, vol. 5, no. 3, pp. 101-119, 1986.

[3] Corke, Peter I., *Robotics Toolbox for use with MATLAB.* Australia: Internet: ftp.mathworks.com, 1994.

[4] Gautier, M., and Khalil, W., "Direct calculation of minimum set of inertial parameters of serial robots," *IEEE Trans. Automation and Robotics*, vol. 6, pp. 368-373, 1990.

[5] Giugovaz, L., and Hollerbach, J.M., "Closed-loop kinematic calibration of the Sarcos Dextrous Arm," in *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, Munich, Germany, pp. 329-334, Sept. 12-16, 1994.

[6] Hollerbach, J.M., "Dynamics," *Robot Motion: Planning and Control*, edited by Brady, J.M., Hollerbach, J.M., Johnson, T.L., Lozano-Perez, T., and Mason, M.T.. Cambridge, Mass.: MIT Press, pp. 51-72, 1982.

[7] Hollerbach, J.M., Giugovaz, L., Buehler, M., and Xu, Y., "Screw axis measurement for kinematic calibration of the Sarcos Dextrous Arm," in *Proc. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, Yokohama, pp. 1617-1621, July 26-30, 1993.

[8] Khosla, P.K., Kanade, T., "Parameter identification of robot dynamics," in *Proc. IEEE Intl. Conf. on Decision and Control*, pp. 1754-1760, 1985.

[9] Khalil, W., and Kleinfinger, J.F., "Minimum operations and minimum parameters of the dynamic models of tree structure robots," *IEEE J. Robotics & Automation*, vol. RA-3, no. 6, pp. 517-526, 1987.

[10] Lin, S.-K., "Minimal linear combinations of the inertia parameters of a manipulator," *IEEE Trans. Robotics and Automation*, vol. 11, pp. 360-373, 1995.

[11] Ma, D., Hollerbach, J.M., and Xu, Y., "Gravity based autonomous calibration of robot manipulators," in *Proc. IEEE Intl. Conf. Robotics & Automation*, San Diego, pp. 2763-2768, 1994.

[12] Mayeda, H., Yoshida, K., and Osuka, K., "Base parameters of manipulator dynamic models," *IEEE Trans. Automation and Robotics*, vol. 6, pp. 312-321, 1990.

[13] Sarcos Reseach Corporation, *Dextrous Teleoperation System Manual.* Salt Lake City, Utah: Sarcos Research Corporation, 1993.