N90-29833

Model Based Manipulator Control

Lyman J. Petrosky
Westinghouse Advanced Energy Systems Division
Madison, PA, 15663

Irving J. Oppenheim
Departments of Architecture and Civil Engineering
Carnegie Mellon University
Pittsburgh, PA, 15213

1. INTRODUCTION

This research effort has its origins in an experimental study[1] of dynamically stable manipulation. The interest in dynamically stable systems was driven by the objective of high vertical reach, for which human balance was the inspiration, and the objective of planning inertially favorable trajectories for force and payload demands, for which human (animal) efficiency was also the general inspiration. A double inverted pendulum system was constructed as the experimental system for this mission, and the research effort led to activities in non-linear control methods, in trajectory planning (still to be completed), and in the use of *model based control*. The findings from that last task form the main emphasis of this paper. Sections 2, 3 and 4 herein are drawn in large part from a recent workshop paper [5] paper; we then discuss in sections 5 and 6 two general areas by which this work is pertinent to space tele/robotics.

The design of a control system for manipulators is a formidable task due to the complexity of the nonlinear coupled dynamics. The goal is the calculation of actuator torques which will cause the manipulator to follow any desired trajectory. In a broad sense, two basic categories of control design are found in the literature. The first contains the robust control methods in which the control is able to overpower the system's nonlinear coupled dynamics. The second contains the *model-based* control (MBC) methods in which many of the system nonlinearities are calculated using a systems dynamic model and the nonlinear system forces are then canceled by actuation forces. Recent advances in computational hardware have made it possible to evaluate in real time the equations of motion of robotic manipulators. Khosla [1]was the first to demonstrate the feasibility of real time MBC using an inexpensive computer system for control of a six degree of freedom manipulator, the CMU Direct Drive Arm II. The requirements for applying MBC can be satisfied for many manipulators of practical interest to space applications. Basically, the system must be amenable to mathematical modeling, and the mathematical model and the control law must be evaluated in real time.

[1]The project was sponsored by the Department of Energy, Advanced Reactor and Nuclear System Technology Support, Program NE-85-001, under contract DE-AC02-85NE37947, *Dynamic Stability for Robot Vertical Reach and Payload.* We are indebted to Clint Bastin of DOE for his particular interest and support, and are most grateful to Westinghouse AES for their cooperation in accommodating Mr. Petrosky's residence at Carnegie-Mellon. We further wish to acknowledge the contributions of Professors I. Shimoyama and J. Bielak, and graduate student Eric Hoffman. *Disclaimer: The view, opinions, and/or findings contained in this report are those of the authors and should not be construed as an official U. S. Department of Energy or Carnegie-Mellon University position, policy or decision, unless designated by other documentation.*

2. CONTROL APPROACH

2.1. Computed-Torque Control

Computed-torque [2] control is a *model-based* control scheme which strives to use the complete dynamic model of a manipulator to achieve dynamic decoupling of all the joints using nonlinear feedback. The dynamic model of the manipulator is described by the system equations of motion which can be derived from Lagrangian mechanics:

$$\sum_{j=1}^{N} D_{ij}\,\ddot{q}_j + \sum_{j=1}^{N}\sum_{k=1}^{N} C_{jk}(i)\,\dot{q}_j\,\dot{q}_k + g_i = \tau_i \qquad \text{for } i = 1, \ldots, N. \tag{1}$$

where the q are the joint coordinates. The τ_i are the externally applied joint actuation torques/forces. The inertial D_{ij}, centrifugal and Coriolis $C_{jk}(i)$, and gravitational g_i coefficients of the closed-form dynamic robot model in Equation 1 are functions of the instantaneous joint positions q_i and the constant kinematic, dynamic and gravity manipulator parameters. The kinetic energy gives rise to the inertial and centrifugal and Coriolis torques/forces, while the potential energy leads to the gravitational torques/forces. Actuator dynamics can be incorporated in the dynamic robot model by additions to the Lagrangian energy function.

The *Computed-torque* algorithm begins with a calculation of the required torque to be applied to each of the joints (in vector notation):

$$\begin{aligned} \tau &= \tilde{\mathbf{D}}\mathbf{u} + \tilde{\mathbf{H}} + \tilde{\mathbf{g}} \\ \tilde{H}_i &\equiv \dot{\mathbf{q}}^T \tilde{\mathbf{C}}(i)\,\dot{\mathbf{q}} \end{aligned} \tag{2}$$

where $\mathbf{u}$ is the commanded joint accelerations. The "~" indicates that these matrices are calculated from the system model based on estimated system parameters. The resulting dynamic equations for the closed-loop system are:

$$\ddot{\mathbf{q}} = \mathbf{u} - \mathbf{D}^{-1}\left\{[\mathbf{D} - \tilde{\mathbf{D}}]\mathbf{u} + [\mathbf{H} - \tilde{\mathbf{H}}] + [\mathbf{g} - \tilde{\mathbf{g}}]\right\} \tag{3}$$

If the system dynamic parameters are known exactly, then $\tilde{\mathbf{D}} = \mathbf{D}$, $\tilde{\mathbf{H}} = \mathbf{H}$, and $\tilde{\mathbf{g}} = \mathbf{g}$, then the closed loop system is described by:

$$\ddot{\mathbf{q}} = \mathbf{u} \tag{4}$$

which is the equation for a set of decoupled second order integrators. This completes the formulation of the modeling and feed forward decoupling functions of the algorithm.

The feedback control law for the commanded joint acceleration u_i is formulated to incorporate the error feedback signal and the reference signal. After decoupling, each joint acts as a second order integrator, therefore the control law is given the form:

$$u_i = \ddot{q}_{id} - 2\zeta\omega(\dot{q}_i - \dot{q}_{id}) - \omega^2(q_i - q_{id}) \tag{5}$$

which causes each joint to act as a second order damped oscillator with natural frequency ω and damping ratio ζ. The form of the equation causes the joint to track the desired joint values q_{id}, $\dot{q}_{id}$, and $\ddot{q}_{id}$.

The computed-torque control defined above is based on the assumptions that the system model is accurate and that all joints are actuated. In our experimental effort the dynamic parameters of the manipulator were manually measured to provide an accurate system model. We assume in all simulations that the dynamic model is accurate. Our experimental system, the double inverted pendulum depicted in Figure 1, does not conform to the second assumption (in that all joints are not actuated) and therefore the algorithm was extended as described in the next section.

2.2. Application to Balancing

Consider first the simplest balancing problem, the planar single inverted pendulum. Balancing is a fourth order control problem with a single input and in the context of this article is equivalent to controlling two manipulator joints with a single actuator. The presumption of the computed torque algorithm, that all joints are actuated, does not apply. However, a suitable control law was found by Petrosky [3]; that method, called *hierarchical partitioning*, is directly applicable to the balancing problem, is robust, and can be integrated with MBC. The balancing problem is partitioned into two second order subsystems, tilt and position. The input signal, base position acceleration, has a component driving the tilt subsystem. The tilt in turn is considered as the input to the position subsystem. This cascaded pair of subsystems is then controlled by a pair of control laws of the form of Equation 5 with the tilt subsystem given a faster time constant. By removing internal variables from the cascaded system, a nonlinear balancing control law is obtained for the manipulator base position variable. This is combined with the computed-torque control for the actuated joints to complete the manipulator control algorithm.

2.3. Determination of Applied Forces

Indirect determination of applied forces (*i.e.* without the use of load sensors) is accomplished by comparison of the manipulator mathematical model and the observed manipulator behavior. A simple example of this is the algorithm for payload determination for the balancing manipulator. Payload estimation can be performed for a balancing manipulator on-line in real time. Consider the equation of motion for pivoting about the base of the dynamically balanced manipulator:

$$\tau_i = \sum_{j=1}^{N} D_{ij}\ddot{q}_j + \sum_{j=1}^{N}\sum_{k=1}^{N} C_{jk}(i)\,\dot{q}_j\dot{q}_k + g_i \qquad \textit{for } i = \textit{base rotation} \tag{6}$$

The base joint of a balanced manipulator is not actuated, therefore $\tau_i = 0$. However, if the payload value is incorrect, then this equation will evaluate to a non-zero value of τ_i when the observed values of the joint variables are entered. The difference indicates the value of the payload which is given by:

$$\Delta P = -\left(\frac{\partial \tau_i}{\partial P}\right)^{-1} \tau_i \tag{7}$$

where ΔP is the difference between the actual payload and the current estimated value. Under ideal conditions this equation would yield the correct payload value in a single sample; however, the accuracy of the values for $\ddot{q}_j$ can be exceedingly poor if obtained by double differentiation of position measurements. This was the case in the experimental system, but the problem was overcome by the use of a parameter estimator.

3. EXPERIMENTAL SYSTEM

The experimental manipulator is a planar double inverted pendulum as depicted in Figure 1; the system is presumed to traverse an approximately level surface, and requires constant active balancing motions. In its plane of motion there are three degrees-of-freedom: translation of the base position, q_1, rotation of the lower arm with respect to the vertical, q_2, and rotation of the upper arm with respect to the lower arm, q_3. It is q_2 which is not directly controlled in this system. The manipulator has a servo driven wheeled base, a hinged connection (free rotation) to the lower arm section, an elbow joint which is servo driven, the upper arm, and an electro-magnet pickup at the tip. It is constructed primarily of aluminum and has a total weight of 13 kg; the tip of the manipulator can reach a height of 1.8 meters in an erect stance.

The wheeled base and the elbow joint are driven by Aerotek servos rated at 1.3 N-m peak torque. The elbow joint has a chain reduction ratio of 57.6:1 and the drive wheels have a chain reduction of 4.8:1. The chain reduced servo arrangement was chosen over direct drive to save weight, and over gear-reduced or harmonic drive to mitigate costly damage in the event of a severe floor collision.

The sensors utilized for manipulator control are:

- Inclination RVDT - a rotary differential transformer measures the angle between the floor surface (via a feeler) and the lower arm.
- Motor Encoders - each servo has an optical encoder of 500 counts per revolution which runs a hardware counter read by the parallel interface board.

The control system hardware consists of a Motorola M68000 based single board computer as the master CPU, a Marinco Array Processor Board (APB), an Analog to Digital Converter (ADC) 32 channel input board, a Digital to Analog (DAC) 4 channel output board, a 96 line Parallel Input/Output (PIO) Interface board, and a terminal. The Marinco APB, with an instruction cycle of 125 *ns*, is used to perform the calculation intensive operations required to implement MBC. The board has fixed point multiplier and addition hardware which are used for floating point operations. The floating point addition or multiplication routines execute in approximately 1 μs. Negation requires 125 *ns*. Computation of the sine/cosine pair requires 15 μs. Additional routines perform data type conversion and other functions required to format sensor data.

Manipulator trajectory calculations are handled by the M68000 CPU on a time sharing basis. In operation a timer interrupts the CPU at each sampling interval. The CPU copies the sensor data to the APB memory and initiates APB execution. The APB formats the data, does scaling operations, performs the trigonometric functions, and then calculates the inverse dynamics. The formatted output data is ready in less than 0.5 *ms*. Data needed for control are returned to the CPU, which outputs them to the DAC's. Cycle time is sufficiently fast for the control algorithm and dynamic model to be evaluated at a sampling frequency in excess of 1000 Hz. However, 100 Hz appeared to be more than adequate for the experimental system.

4. EXPERIMENTAL RESULTS

The experimental manipulator was fully reliable in maintaining balance for long periods while performing a variety of tasks. The base moves approximately ± 3 mm to maintain balance and the tilt varies by ± 0.0063 radian. This motion does not indicate a flaw in the balancing algorithm, but rather the motion results from being at the limit of tilt resolution of the RVDT sensor used with the floor feeler; the RVDT signal variation corresponds to the magnitude of a single digital count. Because the base dimension of the experimental system is zero, it is physically impossible for the manipulator to balance without some minor motions.

The manipulator proved very resistant to upset; its recovery ability appears to exceed that of a human under similar magnitude disturbances. Figure 2 records the transient response of the manipulator to a severe impact applied 0.3 seconds into the record. The manipulator moved forward in order to balance, translating 0.75 meters, and then quickly returned to its original base position. Rotation through a range of 0.25 radians is recorded for the lower arm. The manipulator was also extremely forgiving (compliant) of collision. The manipulator would bounce lightly off an obstacle and come to rest simply leaning against it. When commanded to back away from the obstacle, the manipulator would resume balancing as soon as contact was broken.

Figure 3 records the transient response of the manipulator under the application and removal of a payload at the tip, with the upper arm near the horizontal; the payload was 0.811 kg, and the tip position was offset horizontally by 0.8 meters from base position. The time histories of q_1 and q_2 reflect the payload applied at 5 seconds, removed at 13 seconds, and applied again at 19 seconds. The presence, magnitude, and location of the payload was determined indirectly as discussed in section 2.3; the information was used to adapt the control scheme by updating the sytem model. Figure 3 shows the trace of this payload estimation process, which is noteworthy for its accuracy. In this manner it was possible to adapt to large payloads, demonstrated experimentally with ease up to 3.2 kg, or 25% of the total system weight. A payload estimation record from ongoing balancing in the absence of payload (not shown) demonstrates a typical noise level of ± 26 gm, which is only 0.2% of the system mass.

Another experiment demonstrated the successful development and control of lateral force through the motion of the system masses. A chain connected the manipulator to a heavy mass on a rough table, and the manipulator was used to pull the mass against the force of friction through some target distance. The manipulator developed a lateral force through the movement of its mass center to a point behind its wheel axis; the system them maintained control through the motion ensuing as the lateral force exceeded the friction force, in much the same way that a human would pull a heavy weight accross a floor. Another experiment demonstrated the pickup of the 3.2 kg payload from the floor to an overhead height of 1.8m. The vertical force required to raise the mass was generated by placing the manipulator system masses at great eccentricity to the payload; this effect, and subsequent control of the system, closely resembled a weightlifter's *clean-and-jerk*.

5. APPLICATION OF MODEL BASED CONTROL TO SYSTEMS WITH FLEXIBLE LINKS

MBC has potential space tele-robotic application for manipulators with flexible links. In principle, information available from the on line system model can be utilized to adjust controller gains to the current manipulator configuration. We observe (but do not discuss further) that joint-flexible manipulators, in which flexibility effects are confined to revolute joints, would be controllable in all configurations. We direct our attention at manipulators characterized by linear elastic link bending effects, and presume in our discussion that lumped parameter modelling can apply. Such manipulators are difficult to control because there are many additional system degrees-of-freedom (the "deformation variables" introduced in modelling the flexibility effects) and because some flexural modes may be poorly coupled in the inputs. In this section we develop specialized equations of motion and discuss the potential for the application of MBC using modal decomposition.

Flexible manipulators undergo quasi-periodic oscillations due to elastic deformation. These vibrations develop in response to actuated motions and disturbances. Small vibrations of this type normally decompose into orthogonal modes. This holds true for a manipulator only if it is not undergoing gross motion. As a result of the nonlinear manipulator dynamics, oscillations in the structure exhibit cross terms which negate modal orthogonality. This effect can be deduced from the equations of motion. Equation 1, the manipulator equations of motion, can be expanded for a manipulator with flexibility; deleting summation symbols for purposes of clarity, it becomes:

$$\tau_i = D_{ij}\,\ddot{q}_j + C_{jk}(i)\,\dot{q}_j\,\dot{q}_k + g_i + K_{ij}\,q_j \qquad \textit{for } i = 1, \ldots, M. \tag{8}$$

where q also includes required deformation degrees of freedom. Consider a decomposition of the q into a vibration component, δq, plus an equilibrium trajectory component, q. Substituting into Equation 8 and segregating the terms for vibration yields:

$$\begin{aligned} \tau_i + \delta\tau_i = {} & \\ & D_{ij}\,\ddot{q}_j + C_{jk}(i)\,\dot{q}_j\,\dot{q}_k + g_i + K_{ij}\,q_j \\ & + D_{ij}\,\delta\ddot{q}_j + 2C_{jk}(i)\,\dot{q}_j\,\delta\dot{q}_k + C_{jk}(i)\,\delta\dot{q}_j\,\delta\dot{q}_k \\ & + K_{ij}\,\delta q_j \qquad \textit{for } i = 1, \ldots, M. \end{aligned} \tag{9}$$

Because the equilibrium trajectory portion of the equation by definition satisfies Equation 8, the remaining terms for the vibration component yield the governing equation of motion for vibrations:

$$\begin{aligned} \delta\tau_i = {} & D_{ij}\,\delta\ddot{q}_j + 2C_{jk}(i)\,\dot{q}_j\,\delta\dot{q}_k \\ & + C_{jk}(i)\,\delta\dot{q}_j\,\delta\dot{q}_k + K_{ij}\,\delta q_j \qquad \textit{for } i = 1, \ldots, M. \end{aligned} \tag{10}$$

We see that velocity cross terms exist if the manipulator is in motion. If the amplitude of vibration is small the equation linearizes. The D, C, and K are constant and the $C_{jk}(i)\,\delta\dot{q}_j\,\delta\dot{q}_k$ term is ignored. The free vibration (i.e. $\delta\tau_i = 0$) portion of the manipulator motion forms a linear dynamic system:

$$0 = D_{ij}\,\delta\ddot{q}_j + B_{ij}\,\delta\dot{q}_k + K_{ij}\,\delta q_j \qquad \textit{for } i = 1, \ldots, M. \quad (11)$$

$$\textit{where } B_{ij} = \sum_{j=1}^{M} 2\, C_{jk}(i)\, \dot{q}_j$$

The B matrix appears in the role of a damping term, however due to its form no vibrational energy is lost, only exchanged among the modes. If the manipulator is stationary, $\dot{q}_j = 0$, then it behaves like an undamped multiple degree of freedom elastic structure. To achieve stable control it is necessary to use the system inputs to add damping to the vibration equation.

In principle, such a manipulator would remain amenable to mathematical modeling. The computational burden of calculating a manipulator stiffness matrix is low compared to calculating the dynamic parameters, except that for the link flexible manipulator the entire system has more degrees of freedom. However, a valid control scheme which utilizes the model of the flexible manipulator is significantly more complex that for its rigid counterpart. Nonlinear decoupling such as achieved by the Computed-Torque method *cannot* be anticipated in most cases for flexible systems. Considering next modern control theory methods for pole placement in Multiple-Input Multiple-Output (MIMO) systems, since the system model in MBC can be continuously updated for the current manipulator configuration, MIMO pole placement control would have available at all times a model to linearize for control feedback gain calculation. However, preliminary evaluation of MIMO pole placement indicates that the methods involve numerous matrix inversions, and would not be suited to online implementation using current microprocessors.

An alternate control scheme to discuss is modal decomposition. Presumably, free vibration mode shapes can be calculated based on the system model. Once calculated, modal decomposition of the system dynamic equations and determination of input gains would be straightforward. It appears feasible to determine control gains by specifying the required modal damping matrix and calculating the resulting required actuator inputs. The calculation burden for this control scheme is high because of the eigenvector calculation, but appears to be within the capability of current technology. If proven feasible, this method represents an excellent solution to the flexible manipulator problem.

6. TRAJECTORY PLANNING FOR UTILIZATION OF INERTIAL EFFECTS

Trajectory planning utilizing inertial effects promises efficiencies of great significance to space applications. The payload experiments described at the conclusion of section 4 exemplify these efficiencies at an informal level. More broadly, in this category of trajectory planning one would find minimum energy paths, minimum energy-density paths, minimum time paths, minimum torque paths, and so on. One would also find paths which represent favorable matches between actuator capacities and task requirements. This work is the doctoral research objective of the first author [4] and is currently under investigation. Its pursuit is supported by the MBC capabilities described herein, but is not a direct extension of them. Therefore this brief section is less prescriptive and more descriptive than the discussion of MBC for flexible manipulator control.

Optimal control can solve certain of these problems, such as the minimum time path, and mathematical approaches exist which (under restrictions) can solve others, such as the geodesic for the minimum energy path. Our interest is in approximate approaches which can be framed more generally, and which can be calculated on-line, though not necessarily in real-time. A number of approaches are being studied, including evaluation of different abstractions for use as objective functions, and various mappings of inertial space from which approximate paths might be determined.

7. CONCLUSIONS

The feasibility of utilizing real time Model Based Control (MBC) for robotic manipulators has been demonstrated. The experimental results demonstrate the effectiveness of the control approach, balancing, and of the payload estimation/adaptation algorithm developed for this effort. The mathematical modeling of dynamics inherent in MBC permit the control system to perform functions that are impossible with conventional non-model based methods. These capabilities include:

- Stable control at all speeds of operation;
- Operations requiring dynamic stability such as balancing;
- Detection and monitoring of applied forces without the use of load sensors;
- Manipulator "safing" via detection of abnormal loads;
- Control of flexible manipulators.

This work directly demonstrates the first two capabilities and indicates the feasibility of the additional capabilities. The control of flexible manipulators is a particularly important potential application because this problem has proven very difficult to solve. This technology also supports our work on trajectory planning for favorable utilization of inertial forces.

8. REFERENCES

[1] Khosla, P.K.
Real-Time Control and Identification of Direct-Drive Manipulators.
PhD thesis, Carnegie-Mellon University, 1986.

[2] Markiewicz, B.R.
Analysis of the Computed-Torque Drive Method and Comparison with the Conventional Position Servo for a Computer-Controlled Manipulator.
Technical Memorandum 33-601, Jet Propulsion Laboratory, Pasadena, CA, March, 1973.

[3] Petrosky, L.J.
Problem Substructuring Applied to the Design of a Controller for the Stabilization of a Double Inverted Pendulum.
Master's thesis, Carnegie-Mellon University, 1986.

[4] Petrosky, L.J.
Utilization of Dynamic Forces in Robotic Manipulation (Dissertation Proposal).
PhD thesis, Carnegie-Mellon University, 1987.

[5] Petrosky, L.J., and Oppenheim, I.J.
Application of Model Based Control to Robotic Manipulators.
In S. Griffin (editor), *Second Annual Workshop on Space Operations Automation and Robotics (SOAR '88)*, pages 487-493. NASA Conference Publication 3019, July, 1988.

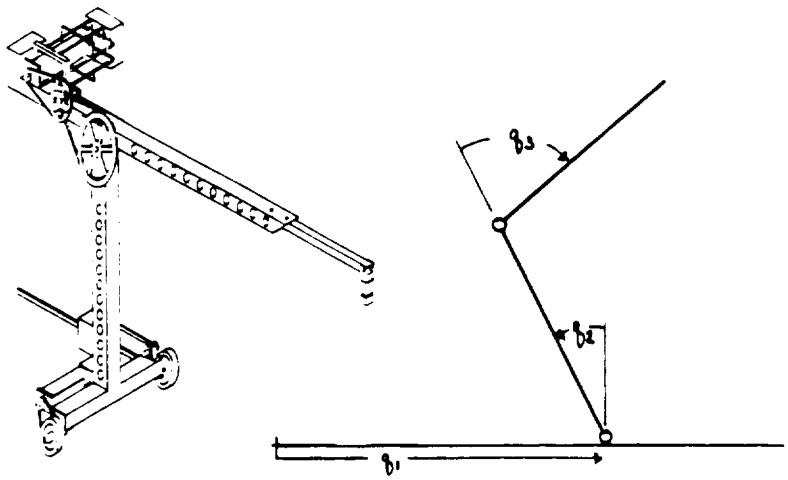


Figure 1. Experimental Manipulator

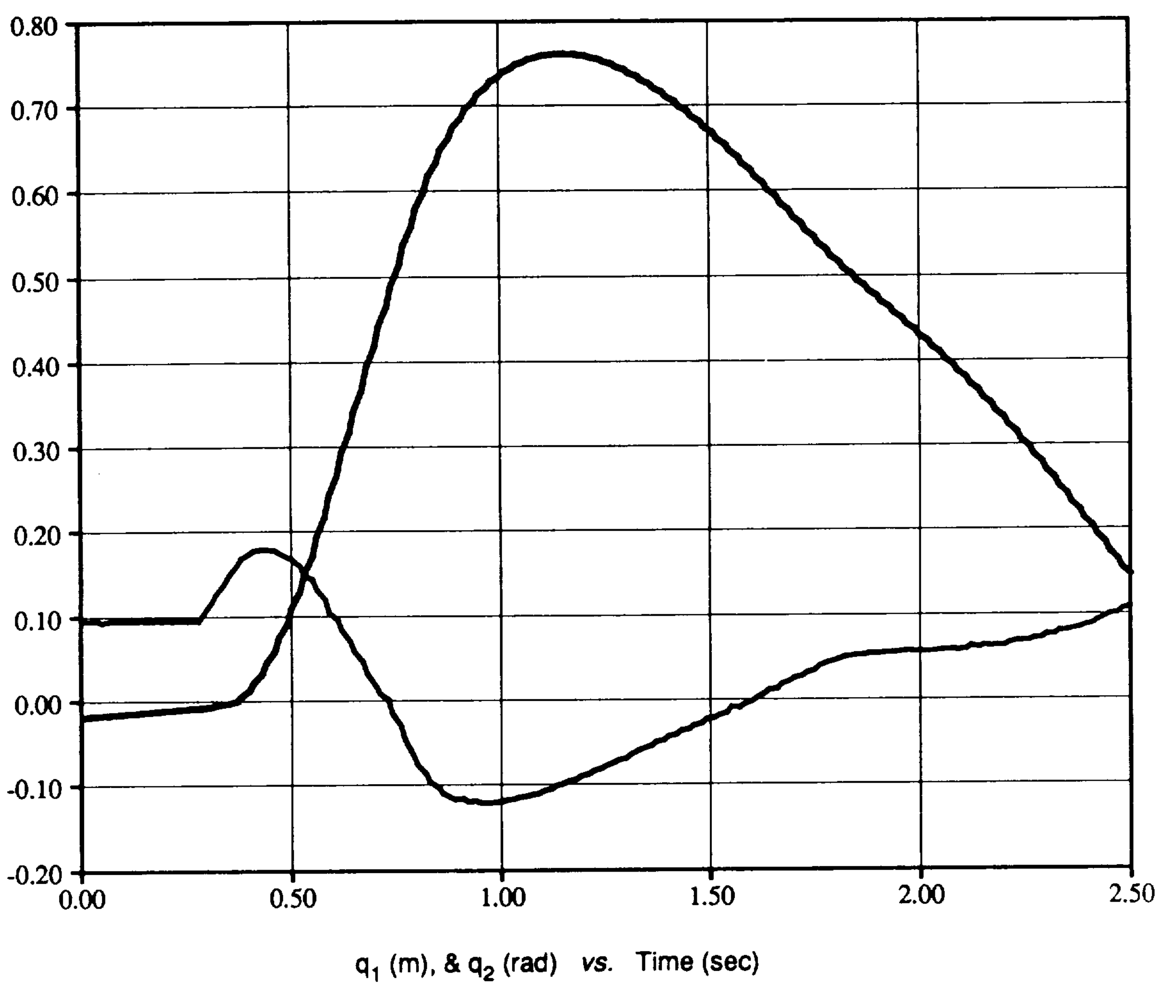
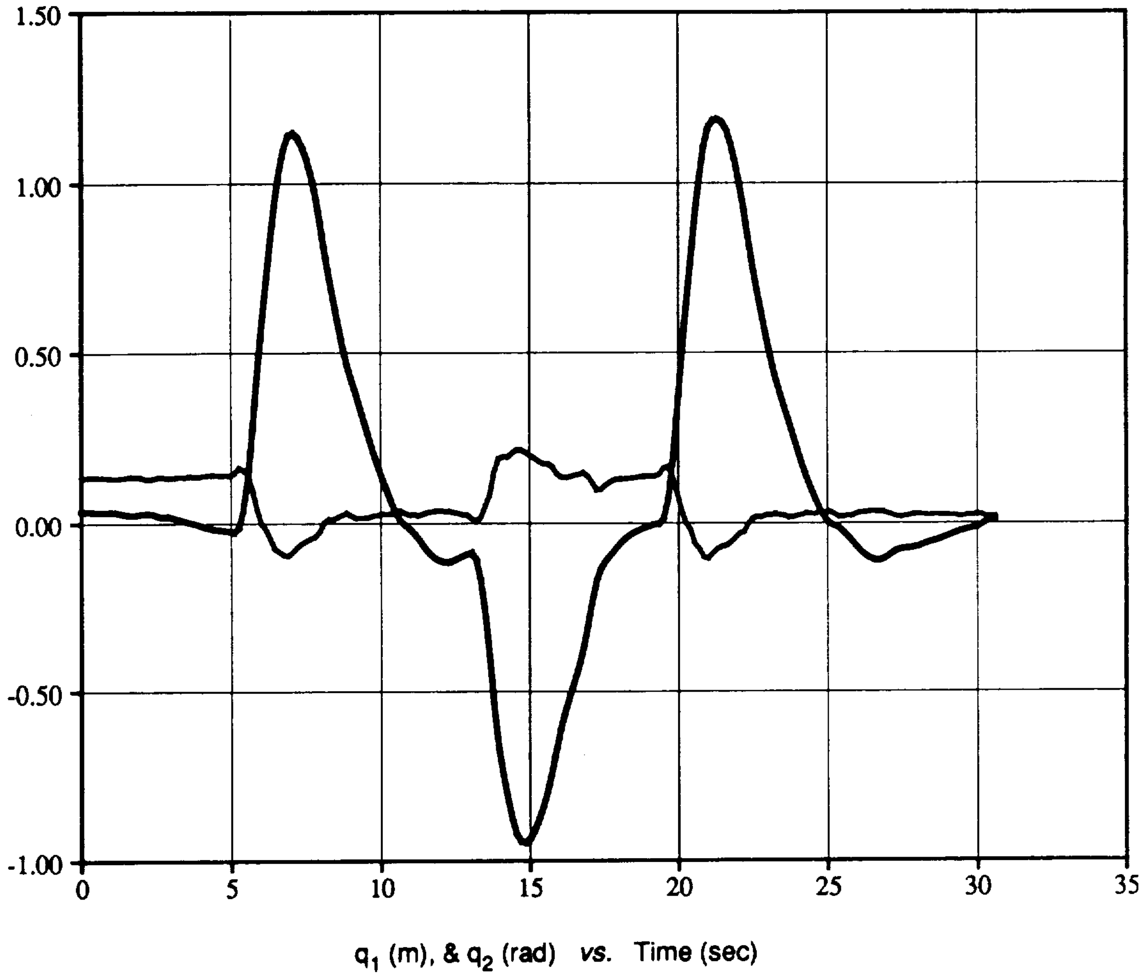


Figure 2. Manipulator Response to Impact



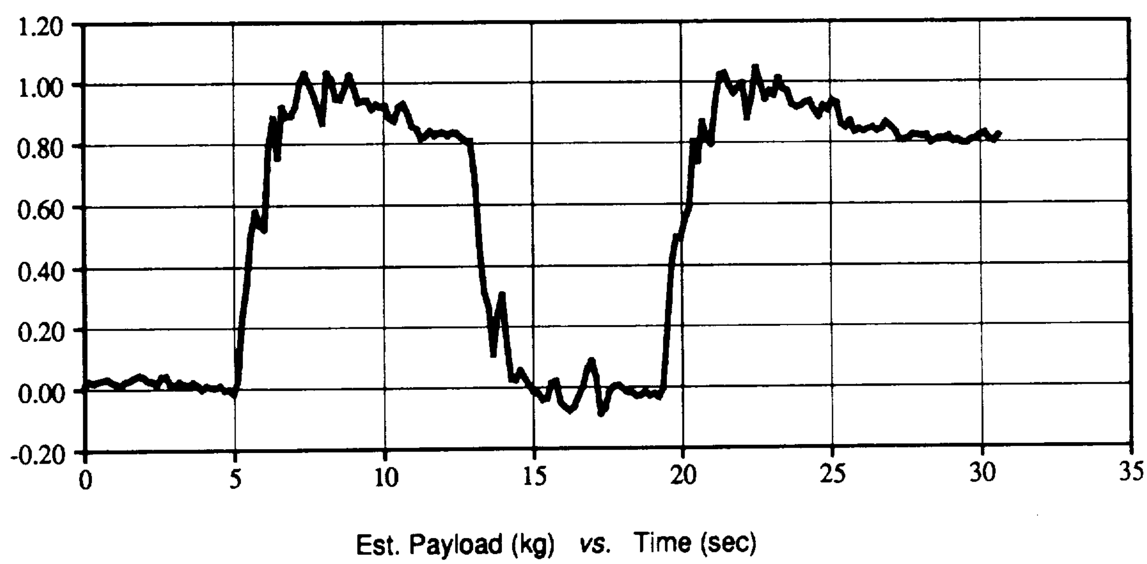


Figure 3. Manipulator Response to Application of Payload at Tip