TOUCH CONTROL FOR A PLANAR THREE-LINK ROBOT ARM

A Thesis

Presented in Partial Fulfillment of the Requirements
for the Degree Master of Science

by

Jeffrey D. Donne, B.S.E.E.

The Ohio State University
1983

Reading Committee:

Prof. H. Hemami

Prof. U. Ozguner

Approved by

H. Hemami

Advisor
Department of Electrical Engineering

ACKNOWLEDGEMENTS

I would like to express my most sincere thanks to my advisor, Professor Hooshang Hemami, for the ideas to begin this thesis, and his guidance throughout the project. It has been an honor to work with him. I would also like to thank Professor Ümit Özgüner for reading this thesis, and also for his comments.

In addition, I would like to thank Emily Baird for the excellent typing of this thesis, and also for her patience while putting it all together.

This work was supported, in part, by the National Science Foundation under Grant Number ECS 820 1240.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (cont'd)

Chapter 1

INTRODUCTION

1.1 Overview

Robotic manipulators are being introduced into the workplace at an ever increasing rate. The present day industrial robot had origins in both numerically controlled machine tools and also in teleoperators. The teleoperator is a device which allows an operator to handle materials at a distance. Teleoperators are usually six-degree of freedom arms equipped with tongs for grasping. The teleoperator provides isolation from an object (such as a radioactive substance) but the problems associated with human error are still inherent within the system. The numerically controlled machine is used to do simple, repetitive tasks faster and more accurately than humans, but there is no interaction between the machine and its work. If an object is in the wrong location, the machine has difficulty adapting to the new position. Worse yet, if one desires to change to a new part, reprogramming is a difficult task. Today's industrial robots, although controlled by mini/micro-computers, are still basically positional machines. Because they are equipped with few or no external sensors, they cannot obtain vital information about the working environment. This thesis is an attempt to provide a manipulator with force sensing capability, and the ability to improve its performance based on this sensed information.

Manipulator performance is usually researched through the study of robot arm kinematics, dynamics, and control. Robot arm kinematics deals with the spatial configuration of the robot as a function of time. Robot arm dynamics deals with the mathematical formulation of the dynamic equations of robot arm motion. Robot arm control is concerned with maintaining the dynamic response of a manipulator in accordance with some pre-specified goals for system performance. One can use a similar analysis when studying human postural dynamics. For this reason, advances made in robotics also have applications toward understanding human locomotion, and vice versa. Thus, robotics research can be helpful in such areas as the development of prosthetic devices. Also, in understanding the control strategies necessary for human locomotion, one might be able to understand such crippling diseases as multiple sclerosis.

1.2 Literature Survey

The field of robotics is new and interdisciplinary , so it attracts many types of researchers from varying backgrounds. In order to give a consistent approach, there have been textbooks written which help to unify many theories and technologies. In [1], the author presents theories from the following fields: computer graphics, kinematics, dynamics, control, and programming. Much of the theory presented has already been applied in experimental work. Vukobratovic and colleagues [2, 3] have written a two volume set of books for robotics. The first deals with the dynamics of manipulation and motion. Variable structures of robots are studied as well as open and closed configurations. The second volume studies the control of robotic systems. The robotic system is studied as a redundant, multivariable, essentially nonlinear automatic control system. Since the robots are dynamically coupled, the control task itself is a dynamic task. Other texts have been written which also deal with the kinematics, dynamics, and control of manipulation robots [4, 5].

Some studies in the area of biped locomotion were done. In [6], Goddard studies the motion of a three-link biped model in the frontal plane as a constrained system with one or two feet on the ground. The transition from constrained to unconstrained motion is discussed. In [7], a three-link biped model was used to estimate internal feedback gains in the human body solely from remote and external measurements. A method was proposed by which postural stability and four motions of the biped could be realized: standing, sitting, bending, and squatting. This method was used to derive open loop and feedback torques. This work is significant since the validity of the results can be checked by making comparisons with actual human motion.

A computer graphics model was developed [8] to simulate the motion of a thirteen-link biped. In order to simulate different human movements, the assumption was made that human movements are stored in the brain as subroutines, and new movements can be learned by combining parts of old movements. Suggestions are made for advancement in such areas as robotics, prosthesis and even choreography. In a further study, Beck [9] focuses on a computer graphics model of the knee joint. Specific movements such as joint interaction and muscle contraction and relaxation are discussed.

Many mathematical studies have been done in the area of stabilization and placement of system poles. In [10], systems are discussed whose state-space is naturally partitioned into position and velocity coordinates. Planar linkage systems used as robot or biped models fall into this category. A strong controllability hypothesis results for these systems. Linear state variable feedback can be used to not only place the system poles, but also the corresponding eigenvectors. The resulting eigenstructures can be designed to stabilize and decouple systems in such a way that specified subspaces of the

state-space are invariant under the dynamics of the closed loop system. This particular type of eigenstructure assignment is also discussed in [11]. This method is applied by Bavarian [12] to control a constrained system. In this model, pre-compensation by an inverse plant is used to improve performance. In a recent study, Raibert and Craig [13] have discussed the hybrid position/force control of manipulators.

Manipulators must be capable of functioning at a number of different process operating points, which sometimes assume a rather wide range of values. If these values assume a range too wide, a normal fixed parameter system model is sometimes not capable of completing the required task. In such cases, a model is needed which is capable of changing its parameters during a specified task. Model reference adaptive control [14-17] has recently been used in robotic systems to solve this problem.

Control becomes difficult with manipulators if operating conditions change. When an object is being moved, its effective inertia usually changes along the trajectory. In [18], position and velocity control in a three-link manipulator is achieved by use of an adaptive controller at each joint.

In order for robotic systems to be effective, they must come in contact with the external environment. Therefore, force control is necessary to regulate the force exerted by the robotic system on its environment. This problem is discussed by Whitney in [19]. Small corrections are made in manipulator fine motions by incorporating a force-feedback model on a manipulator. Force/torque sensors are also used in [20]. In this paper a grasping mechanism is discussed which is very much like a human hand. In [21], a system is discussed which eases a problem in assembly operations such as mating parts. Stiff position controlled manipulators sometimes require compliance in which various coordinates are free to comply with external constraints.

A goal of many robotics researchers is to have robotic manipulators with human qualities. Nearly one third of the articles in the International Journal of Robotics Research are related to robot vision or vision sensors [22-27]. Tactile sensing has also been researched. A detailed study of the human hand [28] was done which incorporates touch control. Transfer functions for the force sensing mechanisms of the hand provide tactile sensing to the system. A survey was done [29] to assess the needs in state of the art automated tactile sensing. The survey determined that tactile sensing arrays are needed which are "skinlike" in nature, durable, and capable of transmitting a tactile image with high resolution. Drawbacks are in the area of data processing in software. Also mentioned was the problem of integrating taction and manipulation.

A step toward realizing a skinlike tactile sensor array was made by Hillis [30]. This sensor is a monolithic array of 256 tactile

sensors that fits approximately on the tip of a finger. The sensor is rugged, flexible, and has a skin-like texture. A test was done where the sensor was to discern between six test objects placed in front of it. It could confidently choose the correct object, but it would also identify other objects placed in front of it as one of the six test objects. Although not perfect, this sensor represents a major step in tactile sensing.

In the above examples, applications are made in both robotics and human modelling. These are just a few examples of the current research, and they also help to give insight into the areas that need to be addressed. Many unsolved problems remain for future research in robotics and human modelling.

1.3 Organization

Chapter 2 describes the three-link robot arm. The nonlinear equations of motion are given (a detailed derivation of these equations is in Appendix A) and control strategy is discussed. State variable feedback stabilizes the system. Constraint forces at the point of contact of the arm with the wall are calculated and shown to be a function of state and input. Sensing mechanisms are discussed, and force feedback is used to improve system response. In Chapter 3, the system is linearlized, and linear state feedback gains are calculated. Sensor dynamics are discussed, as well as the system with force feedback implemented. Chapter 4 gives the results of digital computer simulations used to verify the equation developed in the previous sections. Chapter 5 contains a summary and recommendations for further work. The programs used can be found in Appendix B. A fourth order Runge-Kutta integration routine is used on the system of nonlinear differential equations to calculate the state variables of the model.

Chapter 2

The Three-Link Planar Arm

2.1 Introduction

This chapter contains the theory and control strategy used to move the robot arm. Section 2.2 contains the equations of motion of the robot arm as well as the parameters used in this model. A control strategy for simultaneous stabilization and decoupling is presented in section 2.3, along with an inverse plant which generates control signals to move the arm along a specified trajectory.

Force sensing and characteristics of tactile sensors are discussed in section 2.4. These concepts are applied in 2.5, providing force feedback to the system. Block diagrams are given to specifically show the system dynamics.

2.2 Equations of Motion and Parameters

The robot arm shown in Figure 2.2.1 can be analyzed as a three-link planar inverted pendulum. This arm is chosen with parameters to match those of a robotic system designed by SANDHU Incorporated, called a RHINO $_R$ XR-1. The robot arm was built with the same operating technology as large industrial robots, and is used mainly for research and educational purposes in the field of robotics.

In this analysis, the robot arm is studied in the sagittal plane. The arm consists of three links, with the first link anchored to the ground by a pinjoint. The bottom of the second link is connected by a pin joint to the top of the first link. The third link is connected at its center of mass to the end of the second link. On the RHINO R XR-1, there are DC motors at each joint which provide the torques needed to move the arm. In this analysis, it is assumed that each torque generator is ideal, and capable of providing any amount of torque in zero time. Each joint is connected by a frictionless pin joint.

To aid in the discussion, a numbering system is introduced. The bottom link is named link 1, the middle link is named link 2, and the top link is named link 3. With each link, there is an associated torque u_i, joint J_i, and angle θ_i; i = 1, 2, 3. Angular position θ_i is measured at joint J_i clockwise with respect to the vertical. Each link is characterized by four parameters: length, mass, location of center

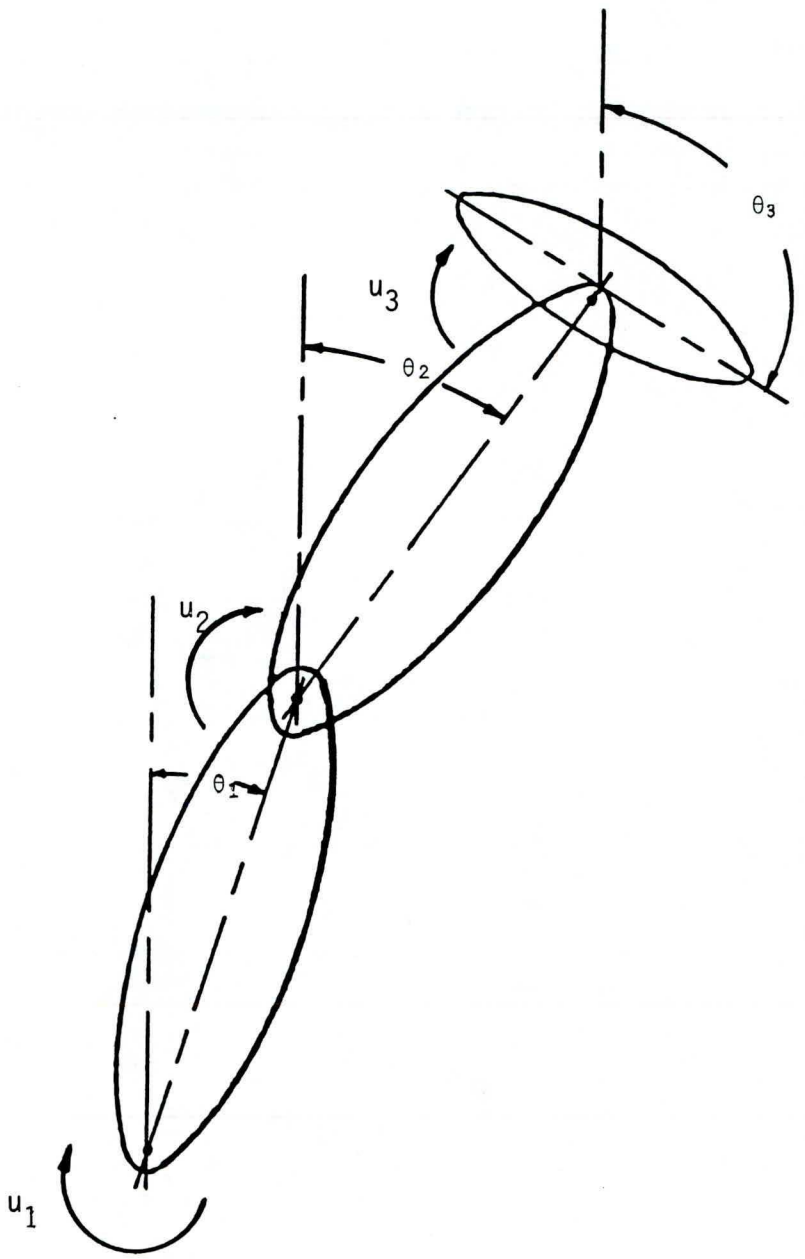


Figure 2.2.1 The Three-Link Robot Arm

of mass, and the moment of inertia about the center of mass. These parameters can be seen in Figure 2.2.2, where l_i is the length of link i, m_i is the mass of link i, k_i is the location of the center of mass of link i, and I_i is the moment of inertia about the center of mass of the ith link. The parameter values, which are estimates of actual link parameters [31], are given in Table 2.1.

The instantaneous position of the robot is specified by vector $\theta(t) = [\theta_1 \quad \theta_2 \quad \theta_3]^T$, and the velocity of the robot is vector $\dot{\theta}(t)$. In order to study contact of the robot arm with a surface, a vertical wall is assumed to exist at x = d. The arm leans against this wall with a force γ. A constraint equation which is a function of the angular position vector Θ governs this contact, and is given in Eq. 2.2.1.

$$P(\Theta) = l_1 \sin\theta_1 + l_2 \sin\theta_2 + k_3 \sin\theta_3 - d = 0 \qquad (2.2.1)$$

As stated earlier, the joint J_1 is constrained to stay fixed to the ground, and the arm is to be controlled to move such that link 3 moves along the wall with a contact force, γ. It is assumed that the wall is frictionless, so there is no tangential component of force at the point of contact on the wall.

The equations of motion for the arm (Appendix A) in matrix form are

$$J(\Theta)\ddot{\Theta} + B(\Theta)\dot{\Theta}^2 + F(\Theta) = CU - \frac{\partial P^T}{\partial \Theta} \gamma \qquad (2.2.2)$$

where

$$\ddot{\Theta} = [\, \ddot{\theta}_1 \quad \ddot{\theta}_2 \quad \ddot{\theta}_3 \,]^T$$

$$\dot{\Theta}^2 = [\, \dot{\theta}_1^2 \quad \dot{\theta}_2^2 \quad \dot{\theta}_3^2 \,]^T \qquad (2.2.3)$$

$$U = [\, u_1 \quad u_2 \quad u_3 \,]^T$$

The matrix C is a constant torque matrix,

$$C = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \qquad (2.2.4)$$

The force of constraint in the horizontal direction is γ. The matrices J, B, and F are nonlinear functions of Θ. The J matrix is of dimension (3 x 3) containing inertial terms, B is a (3 x 3) matrix containing coriolis terms, and F is of dimension, (3 x 1) containing gravitational

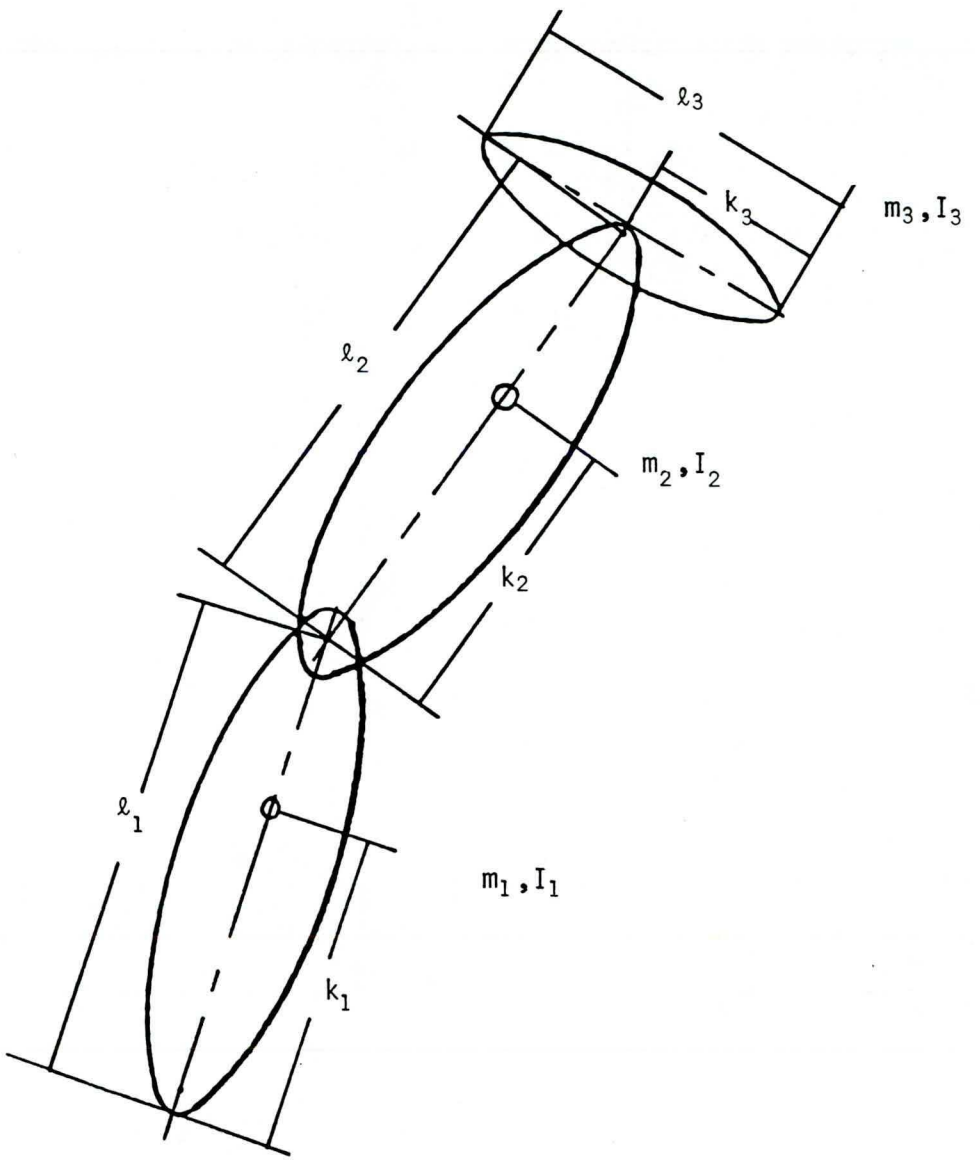


Figure 2.2.2 Parameters of the Robot Arm

Table 2.1

PARAMETERS OF THE ROBOT ARM

PARAMETER	VALUE	UNIT
l_1	0.25	m
l_2	0.25	m
l_3	0.10	m
k_1	0.15	m
k_2	0.15	m
k_3	0.05	m
m_1	1.0	kg
m_2	1.0	kg
m_3	1.0	kg
I_1	0.005	kg-m^2
I_2	0.005	kg-m^2
I_3	0.001	kg-m^2

$$\gamma = \left[\frac{\partial P}{\partial \Theta^T} J^{-1} \frac{\partial P^T}{\partial \Theta} \right]^{-1} \left[\left\{ \left(\frac{\partial}{\partial \Theta^T} \left(\frac{\partial P}{\partial \Theta^T} \right) \right) \dot{\Theta} \right\} \dot{\Theta} \right.$$

$$\left. - \frac{\partial P}{\partial \Theta^T} J^{-1} B \dot{\Theta}^2 - \frac{\partial P}{\partial \Theta^T} J^{-1} F + \frac{\partial P}{\partial \Theta^T} J^{-1} C U \right] \qquad (2.2.11)$$

It is apparent that γ is a function of both state X and input U. Figure 2.2.3 is a block diagram of the constrained system, obtained by combining Eqs. 2.2.6 and 2.2.9. The switch is closed for constrained motion, and open for unconstrained motion.

2.3 Control Strategy

The robot arm is an unstable nonlinear system, and therefore a method of stabilization and control must be provided. In the three-link biped model of Hemami et al. [7], open loop torques are derived from records of a man performing the same actions as those desired by the three-link biped. This method however, is not practical for most industrial robots. They must do many different tasks that require large amounts of memory. Control is achieved in this model by linear state feedback. This section discusses pole placement and decoupling, as well as the control signals derived from an inverse plant model.

Linear State Feedback and Decoupling

For overall system stability, it is desired that all of the poles (eigenvalues) of the system be in the left half of the complex frequency plane. State variable feedback is used to place the system poles, and thus stabilize the system. Position and velocity feedback are generally used in linkage systems since these states are easily measured.

Marked improvement was demonstrated in the response of a five-link biped leg decoupling the motion of each link [32]. This method is both easy to apply and requires few computations. The necessary computations are matrix inversion and matrix multiplication.

The system is linearized for unconstrained motion ($\gamma = 0$). Substitution of the static values $\Theta = 0$, $\dot{\Theta} = 0$ into the equations of motion (2.2.2), and solving yields the bias torques for an arbitrary operating point.

$$U_b = C^{-1} F(\Theta_b) \qquad (2.3.1)$$

terms. Eq. 2.2.2 describes both the free motion of the arm, in which case $\gamma = 0$ and $P(\theta) \neq 0$ and the constrained motion where $\gamma \geq 0$ and $P(\theta) = 0$.

In order to study the robot arm under constrained motion, it is necessary to calculate the force of constraint, γ. The state equations are obtained by solving Eq. 2.2.2 for θ and by adding $\dot{\theta} = \dot{\theta}$ to obtain

$$\dot{\theta} = \dot{\theta}$$

$$\ddot{\theta} = J^{-1}(\theta)\left[-B(\theta)\dot{\theta}^2 - F(\theta) + CU - \frac{\partial P^T}{\partial \theta}\gamma\right] \qquad (2.2.5)$$

where the state is $X = [\theta \quad \dot{\theta}]^T$. Equation (2.2.5) can be compactly expressed as

$$\dot{X} = f(X, U, \gamma) \qquad (2.2.6)$$

The force of constraint γ can now be solved as a function of state and input. For this purpose, the constraint $P(\theta)$ (Eq. 2.2.1) is twice differentiated with respect to time, to obtain

$$\ddot{P}(\theta) = Q(X)\dot{X} = 0 \qquad (2.2.7)$$

Substituting X from Eq. 2.2.6 into Eq. 2.2.7 results in

$$Q(X)f(X, U, \gamma) = 0 \qquad (2.2.8)$$

Eq. 2.2.8 can now be solved for γ in terms of the state X and input U, to obtain

$$\gamma = \gamma(X, U) \qquad (2.2.8)$$

More explicitly, the solution for γ is obtained by the following steps. Eq. 2.2.7 when expanded is given below in Eq. 2.2.10.

$$\left\{\left(\frac{\partial}{\partial \theta^T}\left(\frac{\partial P}{\partial \theta^T}\right)\right)\dot{\theta}\right\}\dot{\theta} + \frac{\partial P}{\partial \theta^T}\ddot{\theta} = 0 \qquad (2.2.10)$$

Now, substitute $\ddot{\theta}$ from Eq. 2.2.5 into Eq. 2.2.10 to obtain the solution for γ.

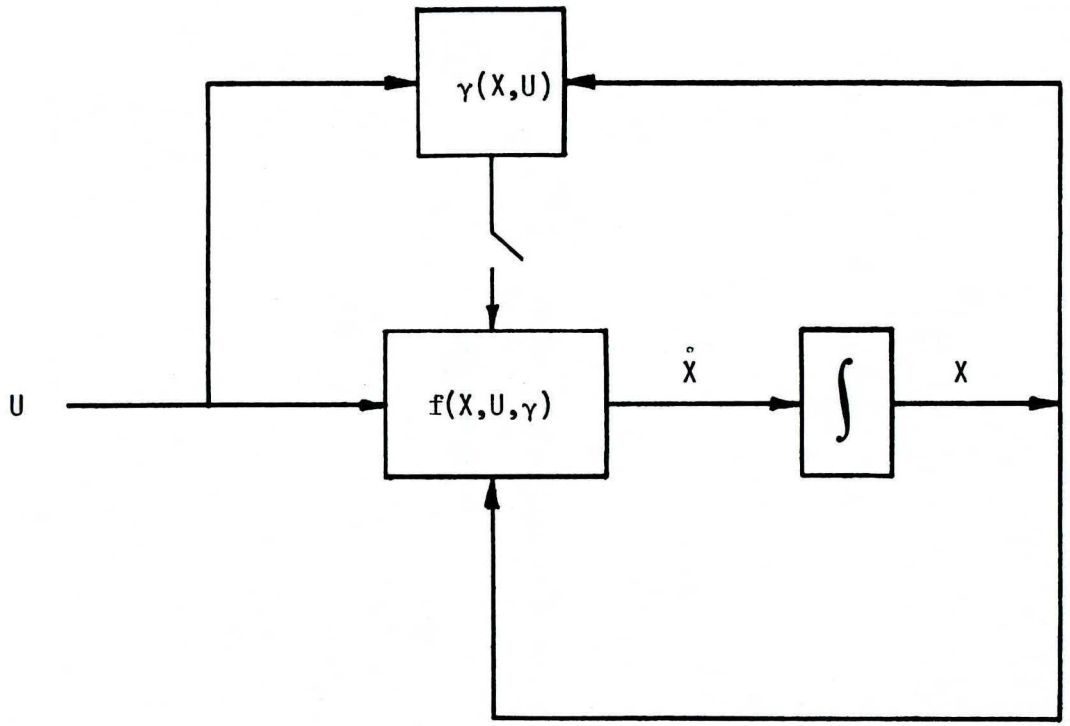


Figure 2.2.3 Block Diagram of the Dynamic System

To linearize Eq. 2.2.2, the following quantities are substituted:

$$\begin{aligned} \Theta &= \Theta_B + \Delta\Theta \\ \dot{\Theta} &= \dot{\Theta}_B + \Delta\dot{\Theta} \\ \ddot{\Theta} &= \ddot{\Theta}_B + \Delta\ddot{\Theta} \\ U &= U_B + \Delta U \end{aligned} \qquad (2.3.2)$$

Neglecting the powers of incremental variables, and subtracting the bias Eq. 2.3.1, the following linear equation results with the delta notation removed.

$$J(\Theta_B)\ddot{\Theta} + \frac{\partial F}{\partial \Theta}(\Theta_B)\,\dot{\Theta} = CU \qquad (2.3.3)$$

Let the desired poles for each link be λ_{i1}, λ_{i2}, i = 1, 2, 3. A decoupled system in matrix form describes the desired system behavior:

$$\ddot{\Theta} + M\dot{\Theta} + N\Theta = 0 \qquad (2.3.4)$$

The matrices M and N are diagonal matrices whose elements are found from the desired poles of each link:

$$\begin{aligned} m_{ii} &= -(\lambda_{i1} + \lambda_{i2}) \\ n_{ii} &= \lambda_{i1}\,\lambda_{i2} \end{aligned} \qquad (2.3.5)$$

Implementing state feedback in the linearized equations (2.3.3) yields the desired decoupled form. Substitution of the state feedback control law into Eq. 2.3.3 yields

$$J\ddot{\Theta} + \frac{\partial F}{\partial \Theta}\dot{\Theta} = C(U_b + K\Theta + G\dot{\Theta}) \qquad (2.3.6)$$

Algebraic manipulation of the above equation results in

$$\ddot{\Theta} + [-J^{-1}CG]\,\dot{\Theta} + [J^{-1}(\frac{\partial F}{\partial \Theta} - CK)]\Theta = J^{-1}CU_b \qquad (2.3.7)$$

Comparing coefficients with Eq. 2.3.4, the feedback gain matrices K and G can be calculated.

$$K = C^{-1}\left(JN + \frac{\partial F}{\partial \Theta}\right) \qquad (2.3.8)$$

$$G = -C^{-1}JM$$

In order to find these feedback gain matrices, it is necessary that the matrix C is invertible. This is true if there is a torque generator at each joint of the linkage system. The block diagram in Figure 2.2.3 is modified by adding feedback gains as shown in Figure 2.3.1. Thus, the system is stabilized about an operating point X_b.

Inverse Plant Model

In the preceding section, linear state feedback gains were derived using a linearized model of the system. Once these gains were calculated, they were put back into the nonlinear system, thus providing stability. However, if in addition to stability, a specific movement is desired, control inputs must be provided which will move the system to the desired operating point. The method of the inverse plant model is used to generate these control inputs.

Suppose that it is desired that the system track a pre-specified trajectory. This trajectory is a continuous function of angular position Θ, and has continuous first and second derivatives. An error signal is generated by subtracting the desired trajectory from the actual state.

$$\Theta_e = \Theta_d - \Theta$$

$$\dot{\Theta}_e = \dot{\Theta}_d - \dot{\Theta} \qquad (2.3.10)$$

This can be seen in Figure 2.3.2. To calculate the control inputs, the desired values for angular position, velocity, acceleration and constraint force are substituted into the equations of motion (2.2.2) and solved for U_{in}, the control inputs from the inverse plant. After making the substitutions stated above, the following equation results

$$J(\Theta_d)\ddot{\Theta}_d + B(\Theta_d)\dot{\Theta}_d^2 + F(\Theta_d) = CU_{in} - \frac{\partial P^T}{\partial \Theta}\gamma_d \qquad (2.3.10)$$

Solving for U_{in} yields

$$U_{in} = C^{-1}\left[J(\Theta_d)\ddot{\Theta}_d + B(\Theta_d)\,\dot{\Theta}_d^2 + F(\Theta_d) + \frac{\partial P^T}{\partial \Theta}\gamma_d\right] \qquad (2.3.11)$$

where

$$U_{in} = [u1_{in}\ \ u2_{in}\ \ u3_{in}] \qquad (2.3.12)$$

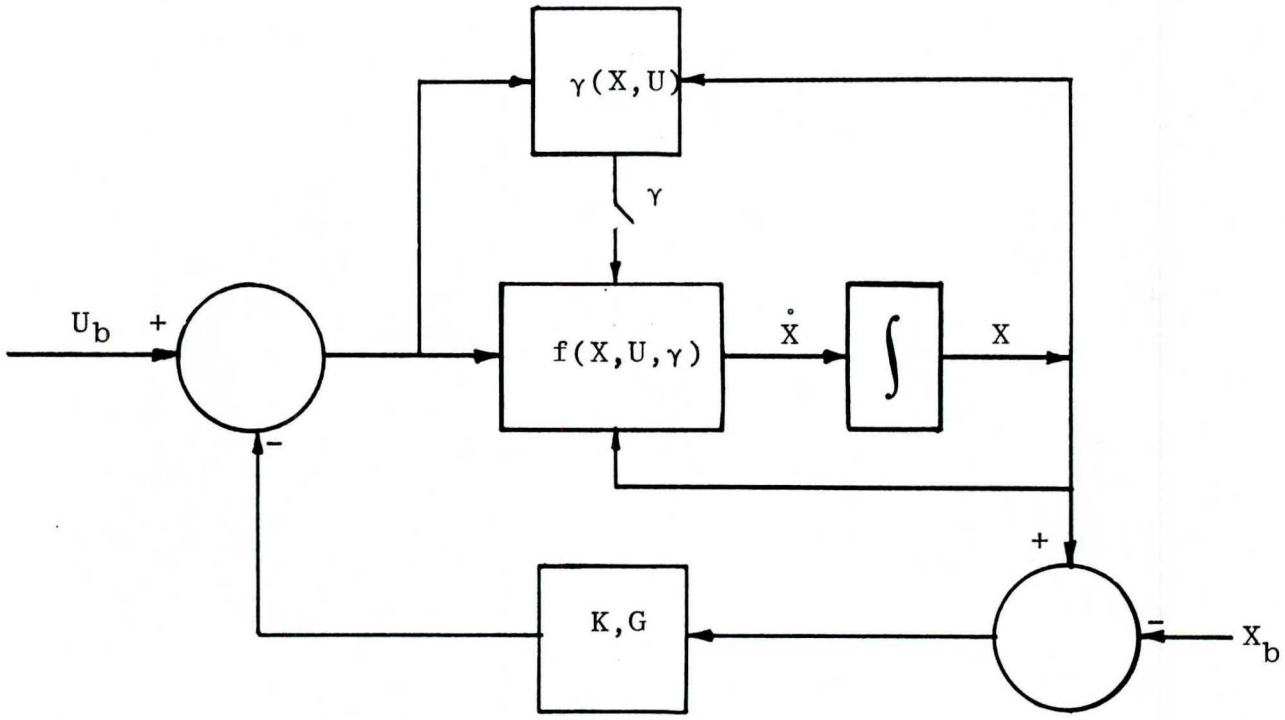


Figure 2.3.1 Block Diagram of System with State Feedback.

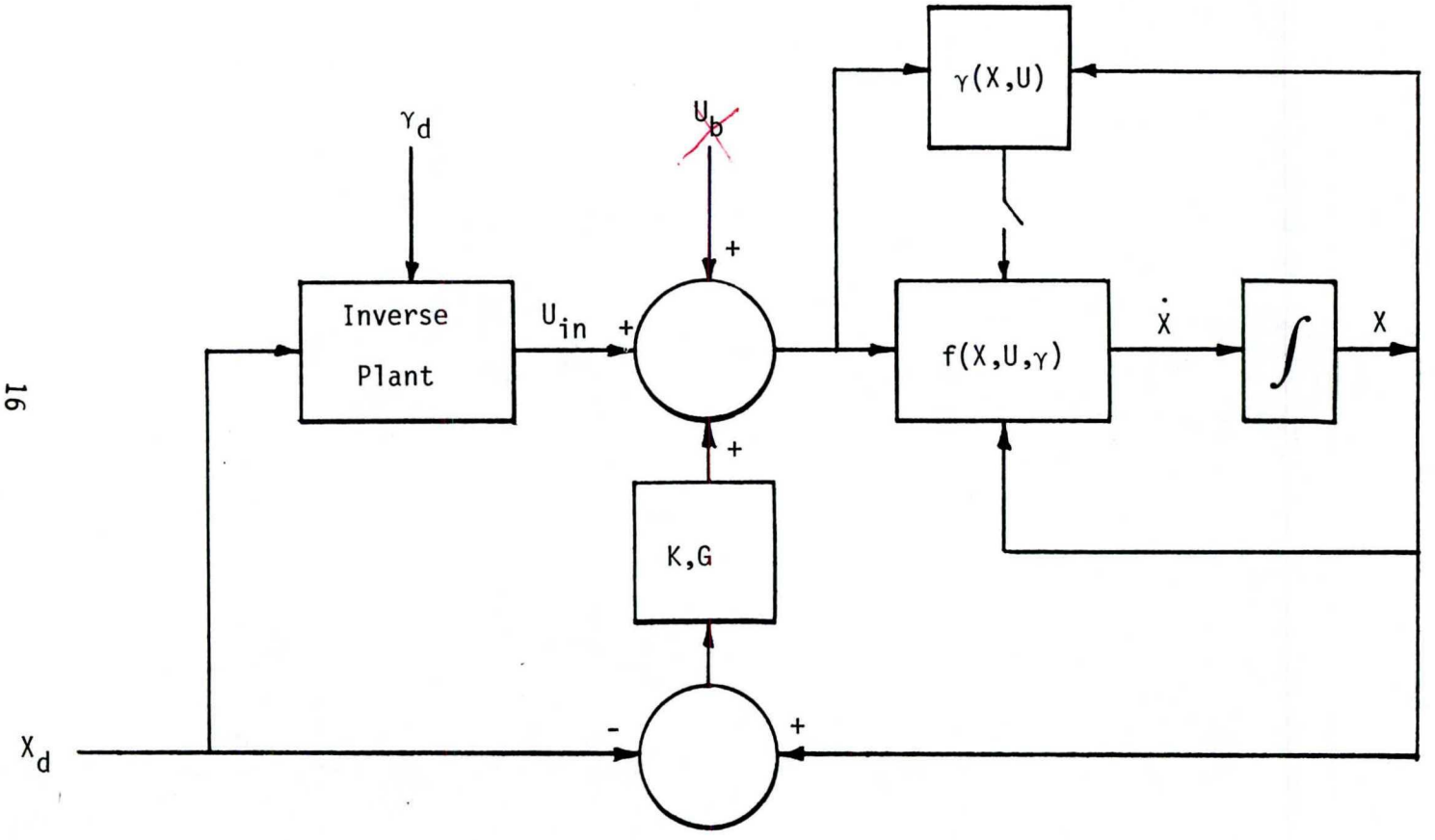


Figure 2.3.2 Block diagram of system including the inverse plant model

Each U_{iin} specifies the torque generated at the ith joint, J_i.

2.4 Force Sensing

Adding human-like sensing to a robotic manipulator is the real challenge in the robotics field. There has been a great deal of research in the area of visual sensing and pattern recognition [22-27, 33, 34]. However at the present, tactile sensing appears to be a more appropriate way to assess an environment. For one, there are far fewer data to be processed, which means that most complex processing can be performed in real time. Another factor is that data collection is much more readily controlled. Placement and pressure of a fingertip are controlled by a program, so one can go directly to the area at which a tactile image is desired.

If one wants a robot arm which is like a human arm, then some type of grasping mechanism might be desired which compares to a human hand. A multi-fingered hand was made in [35] so that fine movements could be done without moving the entire arm. The human hand contains thousands of sensors which detect heat, pressure, vibration, and texture. It was found that the highest density of sensors is in the fingertip, and the lowest density is in the palm of the hand [36]. Because of the complexity of the human hand it is impossible to implement these same properties on a robotic hand, but it is possible to do a rough estimate. There are two main obstacles when using sensors: First, sensing mechanisms are still somewhat crude when presenting an image, and secondly, the large amount of data from sensors causes difficulty in data processing and decision making. However, it is not necessary to exactly model a human hand and arm for most robotic applications. Instead, one might provide the robot with sensors directly related to its job function, thus reducing the number of sensors and simplifying the arm.

There are many performance specifications of sensors which are usually highly dependent upon the application of the manipulator. One of these specifications is proximity sensing. It is necessary to determine location and orientation of an object to avoid damaging collisions when making contact. Another specification is contact sensing. For array type sensors, this refers to determination of a tactile image upon contact with an object. For single sensors it simply refers to binary sensing. The sensitivity of a sensor transducer is another important factor. One must apply enough force to an object so that it does not drop, but it must also not press too hard so as to damage the object. Therefore, a sensitivity range must be determined, again depending upon the particular application. Linearity and hysteresis of a device are qualities that are often mentioned together. Nonlinearity is tolerable since inverse compensation is easy if the device is stable. Hysteresis, on the other hand, is intolerable. Touch sensing devices should be stable, monotonic, and repeatable. Another

quality, time resolution, is sometimes discussed in two ways. First, in terms of the sensor response time, secondly in terms of overall control-loop response. Touch-transducer response time should be small compared to loop cycle time.

For this application, it is adequate to use a single analog sensor at the point of contact of the robot arm with the wall. It is desired to measure the force exerted by the arm on a flat wall.

The force sensor is described by a transfer function, as shown below in Eq. 2.4.1.

$$\gamma_p = H(s)\gamma \tag{2.4.1}$$

In this equation, γ is the actual force, calculated as a function of state and input and described by Eq. 2.2.11. The perceived contact force, γ_p is the force which is sensed. The transfer function H(s) provides a delay in the feedback loop, which is necessary for overall stability. Details of the transfer function are discussed in section 3.4.

2.5 Force Feedback

Once a sensor is implemented, a strategy must be determined to use the sensor measurements to aid in the control of a manipulator. Force feedback is used to control the force that the robot arm exerts on the wall. Since the force γ is a function of both states and inputs, it is necessary to alter one or both of these to change the force. The state of a system is usually specified, so the force must be controlled by changing the inputs to the system. These inputs are changed in such a way that the arm presses on the wall with a force comparable to some desired force γ_d.

One is usually concerned with moving the arm on a particular surface while moving along a specified trajectory. This regulatory process is initiated by the tactile sensing strategy in the previous section. The receptors of the sensing mechanism stimulate a signal which is transformed into some perceived contact force, γ_p. This force is compared to a desired force, γ_d and the error signal γ_e is fed to the controlling mechanism.

$$\gamma_e = -\gamma_p + \gamma_d \tag{2.5.1}$$

Since the inputs to the system are torque generators, the error signal γ_e is fed back as a torque through some gain factor. If one assumes that the torque to be fed back is a linear function of the difference between the perceived and desired contact force, then one can write

$$U_f = -G_1(\gamma_p - \gamma_d) \tag{2.5.2}$$

where G_1 is a (3 x 1) matrix of gains, and U_f is the feedback torque to be added to the overall torque as shown in Figure 2.5.1.

To calculate the gains G, the original equations of motion are used (Eq. 2.2.2), with the following exceptions.

$$\begin{aligned} U_T &= U + U_f \\ \gamma_T &= \gamma + \gamma_e \end{aligned} \tag{2.5.3}$$

In Eq. 2.5.3, U_T is the total torque and γ_T is the total force on the wall. The incremental torque and force from the force feedback loop are U_F and γ_e, respectively. If the static values $\ddot{\Theta} = 0$ and $\dot{\Theta} = 0$ and the bias values $\Theta = \Theta_b$ are substituted in Eq. 2.2.2, the following equation results after subtracting the bias equation.

$$C U_f = \frac{\partial P^T}{\partial \Theta} \Bigg|_{\substack{\gamma_e \\ \theta = \Theta_b}} \tag{2.5.4}$$

Solving for U_f,

$$U_f = C^{-1} \frac{\partial P^T}{\partial \Theta} \Bigg|_{\substack{\gamma_e \\ \theta = \Theta_b}} \tag{2.5.5}$$

Therefore, the feedback gains G_1 can be found as a function of an arbitrary operating point:

$$G_1 = C^{-1} \frac{\partial P^T}{\partial \Theta} \Bigg|_{\theta = \Theta_b} \tag{2.5.6}$$

The (3 x 1) matrix of gains G_1 is now determined. These gains are valid for excursions close to the operating point, Θ_b.

2.6 Summary

The equations of motion for a robot arm were derived for a rigid planar linkage system model. The evolution of the block diagram is seen as the complexity of the model is increased. The first block diagram shows the dynamic system and force calculations necessary for constrained motion.

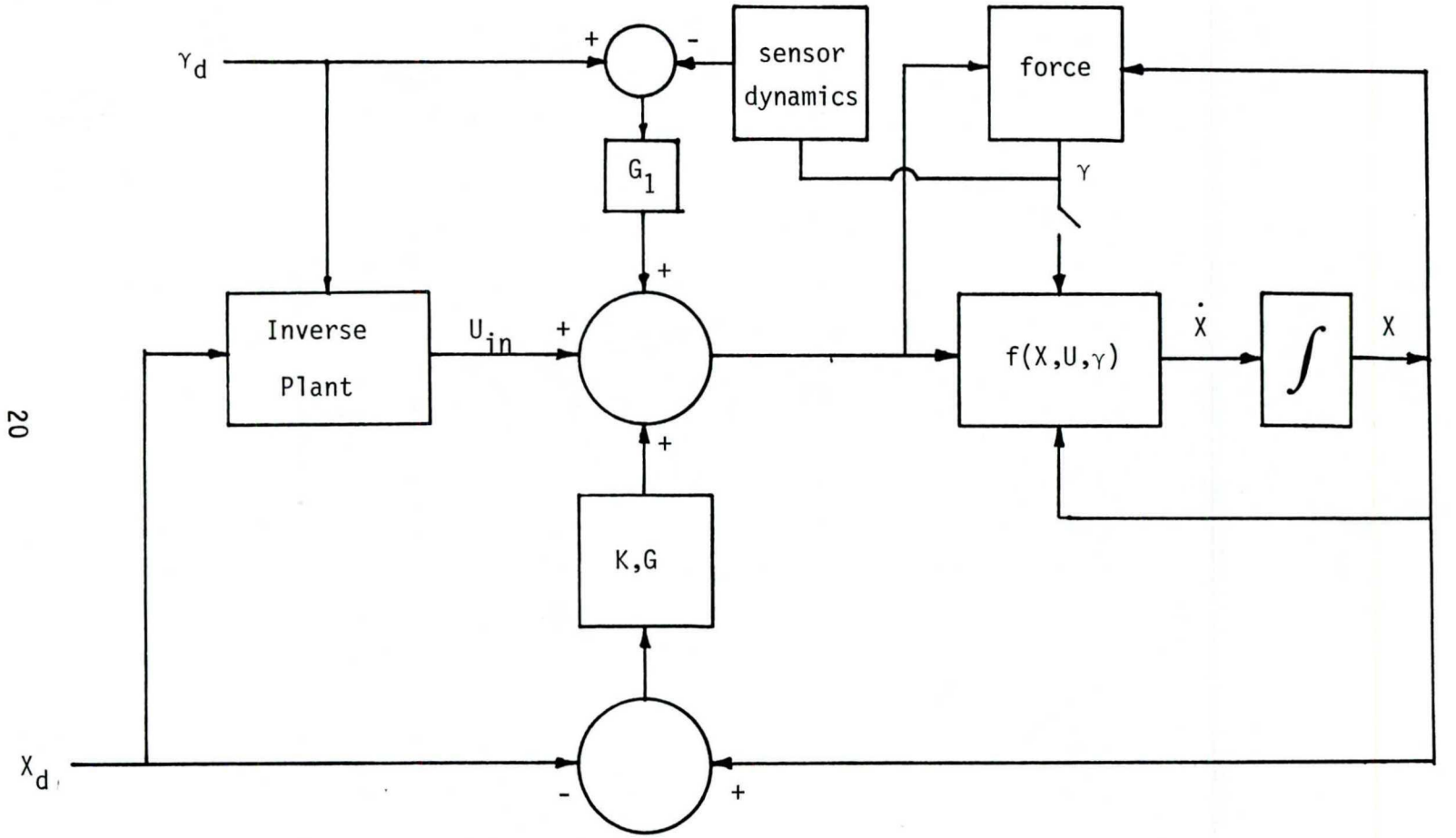


Figure 2.5.1 System with Force Feedback

As the chapter progresses, feedback is added to stabilize the system, and an inverse plant is used so that the system tracks a desired input. Finally, force sensing is discussed, and a force feedback strategy is derived to regulate the force on the wall.

Chapter 3

STABILITY

3.1 Introduction

In Chapter 3, the equations derived in Chapter 2 are used along with specific parameter values to obtain the numerical values to be used in the digital computer simulations. In section 3.2, the equations of motion are linearized for small excursions about an operating point. In section 3.3, the linear state-variable feedback gains are calculated using the linearized equations.

In section 3.4, sensor characteristics are discussed. Included in this discussion is the algorithm for moving from unconstrained to constrained motion. The sensor is equipped with a material which absorbs the shock of impact. Finally, in section 3.5, a force feedback loop is added to the system, and appropriate feedback parameters are derived.

3.2 Static Analysis - Linearization about an Operating Point

In section 2.2, the equations of motion for the robot arm are derived for both constrained and unconstrained motion. These equations are then linearized in section 2.3. The operating point is chosen so that the arm is situated in a vertical position, with the third link horizontal as shown in Figure 3.2.1. If the parameter values in Table 2.1 and the bias values

$$\theta_b = [\, 0 \quad 0 \quad \pi/2 \,]^T \tag{3.2.1}$$

$$\dot{\theta}_b = [\, 0 \quad 0 \quad 0 \,]^T$$

are substituted into Eq. 2.3.3, and $\gamma = 0$, the following matrix equation results.

$$\begin{bmatrix} .1525 & .1 & 0 \\ .1 & .09 & 0 \\ 0 & 0 & .001 \end{bmatrix} \ddot{\Theta} + \begin{bmatrix} -6.3765 & 0 & 0 \\ 0 & -3.924 & 0 \\ 0 & 0 & 0 \end{bmatrix} \Theta$$

$$= \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} U \qquad (3.2.2)$$

This equation is put into state space form:

$$\dot{X} = AX + BU \qquad (3.23)$$

$$X = [\Theta \quad \dot{\Theta}]^T$$

$$U = [u_1 \quad u_2 \quad u_3] \qquad (3.24)$$

The resulting state space equation is

$$\dot{X} = \left[\begin{array}{ccc|ccc} & & & 1 & 0 & 0 \\ & 0 & & 0 & 1 & 0 \\ & & & 0 & 0 & 1 \\ \hline 154.06 & -105.34 & 0 & & & \\ -171.18 & 160.65 & 0 & & 0 & \\ 0 & 0 & 0 & & & \end{array}\right] X$$

$$+ \left[\begin{array}{c} \\ 0 \\ \\ \hline 24.161 \\ -26.85 \\ 0 \end{array} \quad \begin{array}{c} \\ \\ \\ -57.01 \\ 67.785 \\ 0 \end{array} \quad \begin{array}{c} \\ \\ \\ 26.85 \\ -40.94 \\ 1000 \end{array}\right] U \qquad (3.2.5)$$

The poles of the linearized system are

$$\lambda = -17.029, -4.7991, 0, 0, 4.7991, 17.079.$$

Clearly, the above system is unstable. However, it can be stabilized since a simple calculation shows that controllability matrix has full rank:

$$\text{rank } [B \mid AB] = 6 \tag{3.2.6}$$

This is a system of controllability index two, and is the same type of system as those discussed by Ceranowicz, et al. [10].

3.3 State Feedback

State feedback is provided as a means of placing the system poles in the left half of the complex frequency plane, thus stabilizing the system. In section 2.3 the linearized equations are given, and a feedback scheme which simultaneously places the poles and decouples the system is discussed. In section 3.2, numerical values for the system parameters and bias positions are given in Eq. 3.2.2. The linearized system in state space form is given in Eq. 3.2.5. In this section, the state variable feedback gains are calculated. These will be used in all of the simulations of Chapter 4.

In section 3.2 the system was linearized about an operating point wich was chosen so that the arm stands vertical with the top link horizontal.

$$\begin{aligned} \theta_b &= [\, 0 \quad 0 \quad \pi/2 \,]^T \\ \dot{\theta}_b &= [\, 0 \quad 0 \quad 0 \,]^T \end{aligned} \tag{3.3.1}$$

With this choice of bias values, it was determined in section 2.3 that the gain matrices K and G can be calculated from Eq. 2.3.8. In the pole placement method used, it is first necessary to place the eigenvalues, so that the matrices M, and N can be calculated. The following poles were arbitrarily chosen:

$$\lambda_{11} = -2, \quad \lambda_{12} = -3, \quad \lambda_{21} = -4, \quad \lambda_{22} = -5, \quad \lambda_{31} = -6, \quad \lambda_{32} = -7$$

Now the diagonal matrices M and N from Eq. 2.3.4 are specified:

$$M = \begin{bmatrix} 5 & & \\ & 9 & \\ & & 13 \end{bmatrix} \qquad N = \begin{bmatrix} 6 & & \\ & 20 & \\ & & 42 \end{bmatrix} \tag{3.3.2}$$

From Eq. 2.3.8 K and G can be calculated

$$K = \begin{bmatrix} -7.8915 & -7.724 & -.042 \\ -0.6 & -5.724 & -.042 \\ 0 & 0 & -.042 \end{bmatrix}$$

$$G = \begin{bmatrix} -1.2625 & -1.71 & -0.013 \\ -0.5 & -0.81 & -0.013 \\ 0 & 0 & -0.013 \end{bmatrix} \tag{3.3.3}$$

For digital computer simulations, the nonlinear system equations are used with the linear state feedback discussed above. Therefore, the feedback is valid only for small excursions about the bias values.

3.4 Characteristics of Sensors

In order to model the effect of a sensor in a digital computer simulation, transfer functions must be derived so that appropriate equations can be written. These transfer functions must incorporate such characteristics of sensors as time resolution and sensitivity. The sensor must also be rugged in order to absorb contact forces when making the transition from unconstrained to constrained motion.

The time resolution of a sensor is affected by two parameters. First, touch-transducer response time, and secondly, time necessary for data-processing must be considered. As mentioned in section 2.4, it is important that sensor response time is small compared to the overall loop cycle time of the manipulator on which the sensor is incorporated.

The sensitivity of a sensor varies according to the application of the manipulator. For delicate operations, a sensor might need to be accurate enough to detect a change in mass on the order of a few grams, while for heavy machinery the sensor might not even respond to a mass less than a few kilograms. However, no matter what type of application, it is important to have a large dynamic range. In [29], it was reported that dynamic range is much more important than

linearity, since one can compensate for linearity during processing of data. For sensors today, a good response time would be logarithmic in nature with a dynamic range of about 1000:1.

In [28] transfer functions for the tactile sensing mechanisms in the human finger were derived. It was assumed that the sensor had a second order response characteristic. This response incorporates the delay which comes from the indentation of the skin which stimulates the neural response from the tactile sensors. In [29] it was reported that the amount of reaction time (whole-loop) it takes for a human to, say, push a lever as fast as possible after any stimulus, is about 150-200 ms on the average.

There are many transduction technologies available that can be used for tactile sensing. Some examples are resistive and conductive materials, semiconductors, piezoelectric transduction, capacitive sensing, and photoelectric transduction, to name a few. There is no one materials technology that simultaneously provides linear, stable, sensitive, rugged, and cheap sensing. However, many of these technologies are in use today and are performing relatively well.

These sensors have such a quick response that for the single force sensor implemented on this manipulator, a simple delay can be used for the transfer characteristic, the response is assumed ideal except for the delay element. Recall Eq. 2.4.1:

$$\gamma_p = H(s)\gamma \tag{3.4.1}$$

In this equation, γ is the actual force exerted on the wall, calculated as a function of state and input. The perceived contact force γ_p is that force which is sensed. The transfer function is a delay given by the following equation.

$$\gamma_p = \gamma(t - \tau) \tag{3.4.2}$$

The time delay τ used for this application is about 40 ms. This value could not be found explicitly in the literature, but it is assumed reasonable by researchers in sensor technologies.

A sensor must be rugged enough so that it is not damaged when contact is made with a surface or an object. Therefore, in order to make the transition from unconstrained to constrained motion, this sensor is equipped with an absorbing structure which is capable of absorbing the shock of impact. This material should be like the spongy part at the tip of a human finger. It is modelled as a spring and dashpot and is shown in Figure 3.4.1, and 3.4.2. The equation relating force and distance is Eq. 3.4.3, where ε is the distance from the sensor to the wall, and $\dot{\varepsilon}$ is the velocity in the X-direction of the tip of the manipulator.

$$F = \begin{cases} 0 & \text{IF} \quad \varepsilon > x_{eq} \\ -K\varepsilon - B\dot{\varepsilon} & \text{IF} \quad x_{com} < \varepsilon \leq x_{eq} \\ \gamma(X, U) & \text{IF} \quad 0 \leq \varepsilon \leq x_{com} \end{cases} \tag{3.4.3}$$

In the above equation, the force on the wall is zero if no contact is made. The force is calculated by using the second part of Eq. 3.4.3 when compressing the absorbing structure. In this application, values are chosen for x_{eq} and x_{com} to be x_{eq} = 1 cm and x_{com} = 0.5 cm. The absorbing structure cannot be compressed beyond x_{com}. The stiffness, K provides a restoring force as represented by a spring. Thus, if stretched, the spring tries to contract; if compressed, it tries to expand. The damping, or viscous friction, B characterizes the element that absorbs energy. The damping force is proportional to the relative velocity of the two ends of the dashpot. For this structure, K is chosen to be small since it is desired to mostly provide damping, with a small restoring force. The third part of Eq. 3.4.3 is the force on the wall calculated as a function of state and input as before.

As mentioned above, ε is the distance from the sensor at the manipulator tip to the wall (See Figure 3.4.1). The calculation for ε is given in Eq. 3.4.4, where x_{eq} is the size of the absorbing structure at rest.

$$\varepsilon = l_1 \sin\theta_1 + l_2 \sin\theta_2 + k_3 \sin\theta_3 - x_{eq} \tag{3.4.4}$$

Similarly, Eq. 3.4.4 can be differentiated with respect to time to yield the velocity in the K-direction of the manipulator tip, as given in Eq. 3.4.5.

$$\dot{\varepsilon} = l_1\dot{\theta}_1 \cos\theta_1 + l_2\dot{\theta}_2 \cos\theta_2 + K_3\ \dot{\theta}_3 \cos\theta_3 \tag{3.4.5}$$

The values for the stiffness, K and the damping, B can be determined as follows When the absorbing structure is fully compressed ($\varepsilon = X_{com}$, $\dot{\varepsilon} = 0$), the force should be equal to f_1, Newtons. Therefore, K can be calculated by substituting the above values for ε, and $\dot{\varepsilon}$ into Eq. 3.4.3 and solving for K as given below

$$K = f_1/x_{com} \tag{3.4.6}$$

The value for B can be chosen arbitrarily, depending on the amount of damping desired.

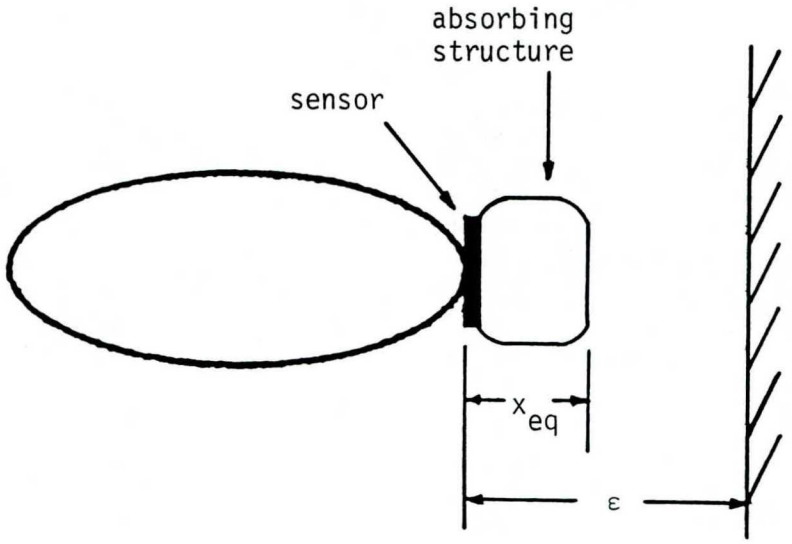


Figure 3.4.1 Link Three Equipped with Sensor and Absorbing Structure

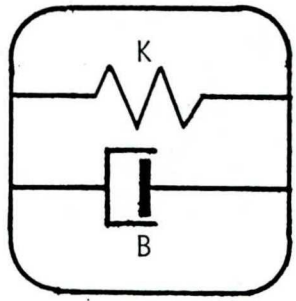


Figure 3.4.2 Absorbing Structure modelled as a Spring and Dashpot

3.5 System with Force Feedback

Force feedback is used in this system to provide fine control of the force exerted on the wall by the robot arm. The force is sensed by force sensor, and the perceived force is regulated about a desired force, γ_d. This desired force can be constant or time-varying.

The transfer functions of the previous section can be implemented in Eq. 2.5.2 to derive the appropriate feedback torques. Substituting Eq. 3.4.2 into Eq. 2.5.2, one obtains the following equation.

$$U_f = G_1[H(s)\gamma - \gamma_d] \tag{3.5.1}$$

The only unknown in this equation is the (3 x 1) vector G_1, representing the force feedback gains.

The feedback gains, G_1, can be derived using Eq. 2.5.6. Using the same operating point as in the previous sections:

$$\Theta_b = [\ 0 \quad 0 \quad \pi/2\ ]$$
$$\dot{\Theta}_b = [\ 0 \quad 0 \quad 0\ ] \tag{3.5.2}$$

One can make substitutions into Eq. 2.5.6. To get the following gain values.

$$G_1 = [\ 0.5 \quad 0.25 \quad 0] \tag{3.5.3}$$

If the system deviates from the bias values by too much, instability can result, so additional gain control is needed. One way to do this is by decreasing the above gains by a constant value.

$$G_1' = \alpha G_1 \tag{3.5.4}$$

In Eq. 3.5.4, α can be a scalar between 0 and 1, thus decreasing the gains.

A more desirable approach to regulating these gains is by using time-varying feedback. If the bias positions and velocities are time-varying, then the gains can be calculated on-line using Eq. 2.5.6.

$$G_1 = C^{-1} \frac{\partial P^T}{\partial \Theta} \Bigg|_{\Theta = \Theta_b(t)} \tag{3.5.5}$$

This is actually a more valid assumption, since one usually follows a time-varying reference trajectory. However, one must again be aware of

instabilities that might result from deviations too far from the bias values. These can be compensated for by multiplying by a scalar, α as mentioned above.

3.6 Summmary

In this chapter, the system was linearized and the system was simultaneously stabilized and decoupled by using a pole-placement routine. Sensors were discussed, along with the theory necessary for going from unconstrained to constrained motion. Finally, the system with force feedback was discussed. The feedback gains in the force feedback loop were adjusted so that instability did not arise.

Chapter 4

DIGITAL COMPUTER SIMULATIONS AND RESULTS

4.1 Introduction

In order to verify the results of the previous chapters, digital computer simulations of this robot system are carried out. Both constrained and unconstrained motion are studied. In section 4.2, free arm stability is demonstrated. In section 4.3, a similar simulation is done, but a wall at a distance d from the base of the arm is assumed, and the arm is constrained to keep contact with the wall at all time. A tracking problem is discussed in section 4.4, where the robot arm moves with a pre-specified periodic motion. In section 4.5, a force feedback loop is added to the system, showing improvements in system response. Finally, section 4.6 is a simulation of the arm touching the wall (i.e., going from unconstrained to constrained motion).

4.2 Free Arm Stability

A free arm simulation is used to check the validity of the original equations of motion. A bias point is chosen and the arm is offset from this bias point. The simulation is done to verify that the arm returns to the equilibrium position.

In this case, the bias point is chosen so that the bottom two links are vertical, and the top link is horizontal.

$$\theta_b = [\, 0 \quad 0 \quad 90^\circ \,]^T \tag{4.2.1}$$

The angular positions are given in degrees. The equilibrium velocities are chosen so that the system remains at rest after reaching equilibrium. The bias velocities are

$$\dot{\theta}_b = [\, 0 \quad 0 \quad 0 \,]^T \tag{4.2.2}$$

Initial conditions are chosen reasonably close to the bias values so that the system is not initially unstable. The initial conditions are

$$\theta_i = [10° \quad -15 \quad 70° \ ]^T$$
$$\dot{\theta}_i = [\ 0 \quad 0 \quad 0\ ]^T \qquad (4.2.3)$$

Using the above bias point and Eq. 2.3.1, the bias torques are found to be

$$U_b = [\ 0 \quad 0 \quad 0\ ]^T \qquad (4.2.4)$$

The results of this stability test are shown in figures 4.2.1 to 4.2.3.

In the plots of angular position (Figure 4.2.1), the system starts from the initial offset conditions, and settles to equilibrium in about 2 seconds. The velocities (Figure 4.2.2) start from rest and settle to the rest position ($\theta = 0$), after the system is in the equilibrium position ($\theta = \theta_b$). Since no torque is needed when the system is in equilibrium, the control U goes to zero. All of the graphs reach equilibrium in about two seconds, which is expected according to the choice of system poles.

4.3 Constrained Motion Stability

After verifying in the previous section that the feedback strategy is valid for unconstrained motion, a constraint is imposed, and a similar digital computer simulation is performed. The constraint is a wall at a distance x = d from the first joint of the robot arm. The arm is to move along the wall without violating the constraint.

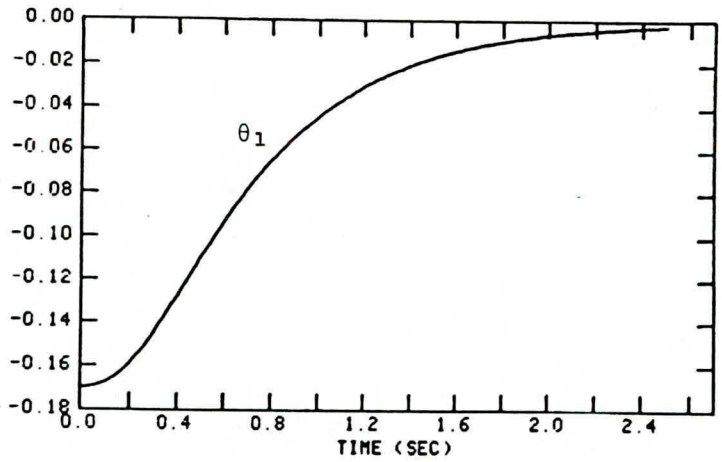
The equilibrium point is chosen to be the same as in the previous section:

$$\theta_b = [\ 0 \quad 0 \quad 90° \ ]^T$$
$$\dot{\theta}_b = [\ 0 \quad 0 \quad 0\ ]^T \qquad (4.3.1)$$

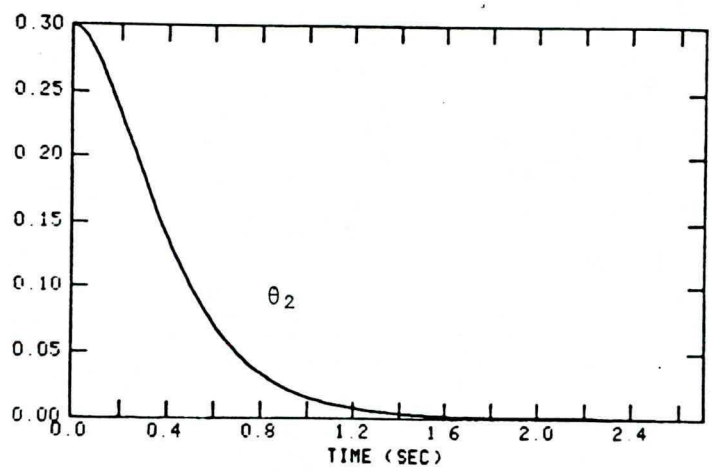
The constraint equation is also the same.

$$P(\theta) = l_1 \sin\theta_1 + l_2 \sin\theta_2 + K_3 \sin\theta_3 - d = 0 \qquad (4.3.2)$$

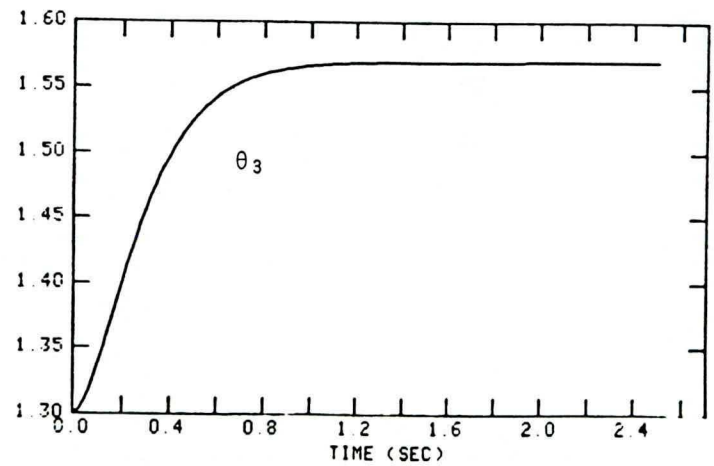
Initial conditions are chosen so that the arm does not initially violate the constraint. In addition, it is desired that the top link stay in the horizontal position throughout the simulation. The initial conditions are:



(a)

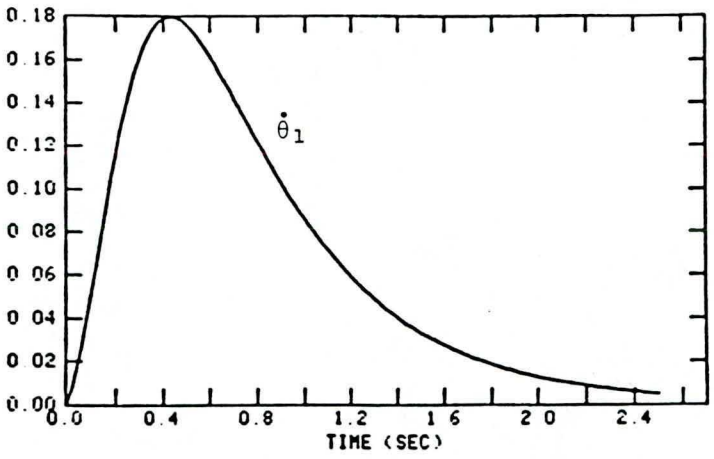


(b)

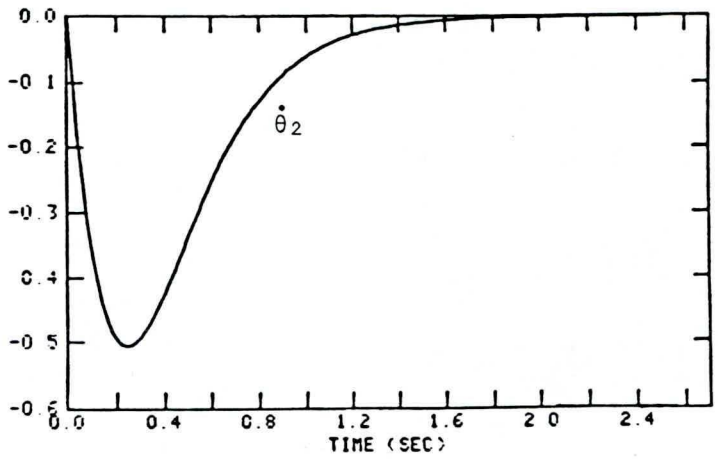


(c)

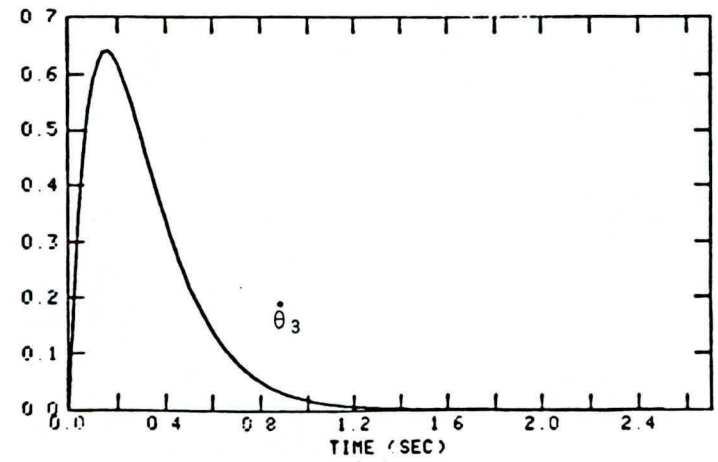
Figure 4.2.1 Angular Positions vs. Time



(a)

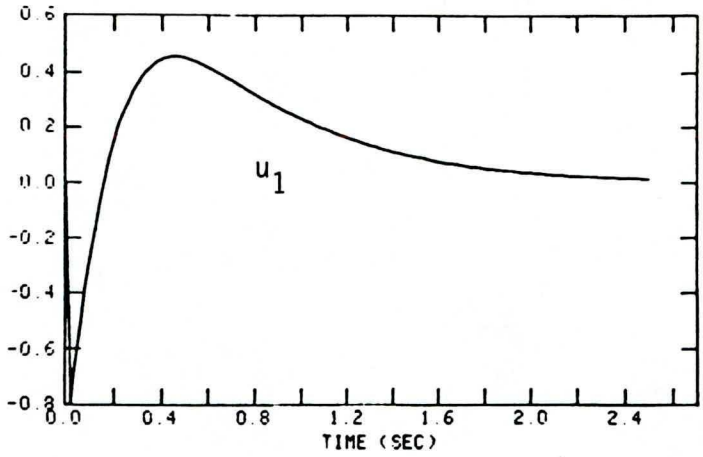


(b)

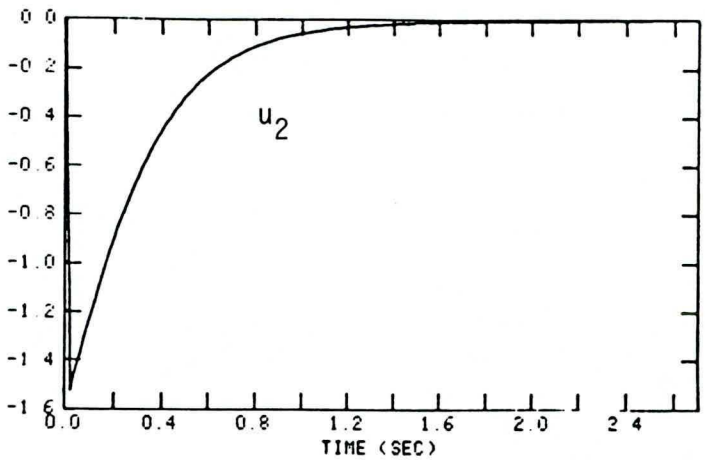


(c)

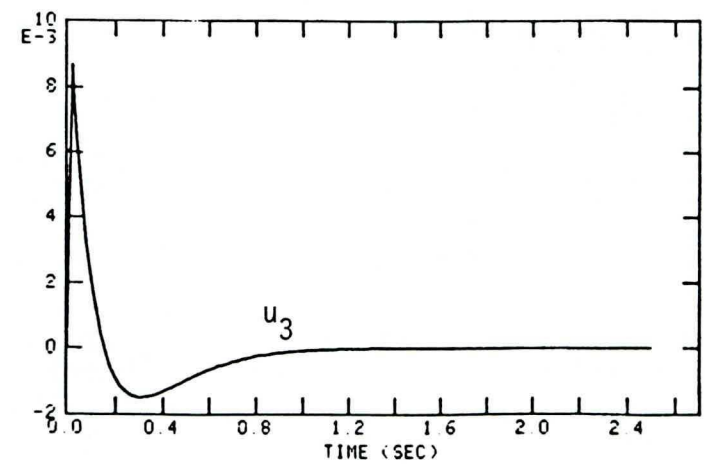
Figure 4.2.2 Angular Velocities (in rad/sec) vs. Time



(a)



(b)



(c)

Figure 4.2.3 Joint Torques (in N-m) vs. Time

$$\theta_i = [-10° \quad 10° \quad 90° \ ]^T$$
$$\dot{\theta}_i = [0 \quad 0 \quad 0 \ ]^T \qquad (4.3.3)$$

In addition to keeping the constraint, it is desired that the arm presses against the wall with a constant force of 10 newtons. The system at rest requires that bias torques are generated so that this constant force can be applied. To calculate these bias torques, a modified form of Eq. 2.3.1 is necessary; the bias force, γ_b must be included. For the system at rest, the velocities and accelerations must be equal to zero: $\dot{\theta} = 0$ and $\ddot{\theta} = 0$. After substituting these values into Eq. 2.2.2, in addition to the values for θ_b and γ_b, the following equation results, after some algebraic manipulation.

$$U_b = C^{-1} \left[F(\theta_b) + \frac{\partial P^T}{\partial \theta} \gamma_b\right] \qquad (4.3.3)$$

In this simulation, θ_b is given above in Eq. 4.3.1 and γ_b = 10N which yields

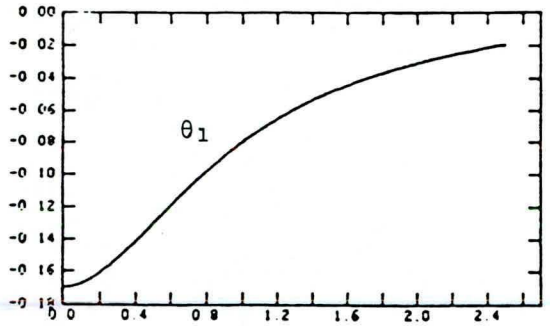
$$U_b = [5.0 \quad 2.5 \quad 0.0]^T \qquad (4.3.4)$$

This equation shows that no torque is needed at joint three since link three is pivoted at the center of mass.

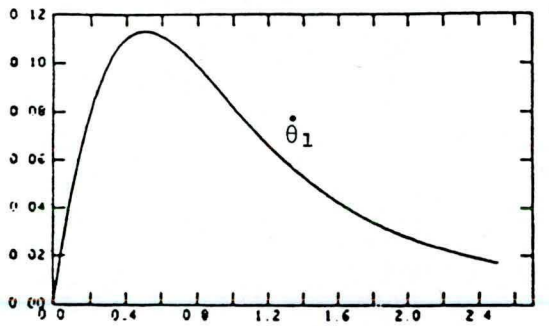
The results of the digital computer simulation are shown in Figures 4.3.1 to 4.3.2. The positions and velocities start from the initial values and return to the equilibrium values this time in about 2.5 seconds (See figures 4.3.1 a-c and 4.3.1 d-f, respectively). The control law is again state variable feedback; the controls are shown in Figure 4.3.2 a-c, where it is apparent that the inputs settle to the bias values, U_b. Figure 4.3.2d is a graph of the force, γ vs. time. The force settles to the constant value of 10 newtons, but it starts at about 9.2 newtons at the beginning of the simulation. In some industrial applications of robotics, it might be desired that the force stay within a closer range of values to γ_b. This problem of tightening the control on the force is discussed in a later section. Figure 4.3.2e is a plot of the constraint equation (2.2.1). Notice that the magnitude of the values is on the order of 10^{-9}, so the constraint is not violated.

4.4 Periodic Movement

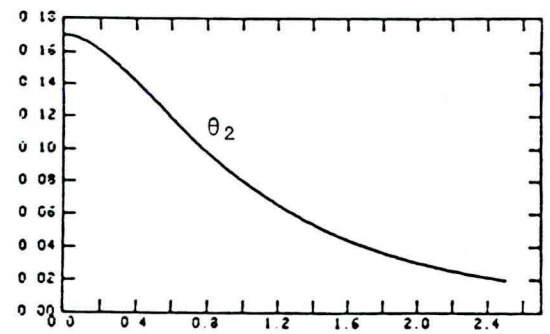
In the next simulation, it is desired that the system tracks a specified trajectory. The robot arm is again constrained to movement along the wall. However, in this case, values for θb and $\dot{\theta}_b$ are chosen which are



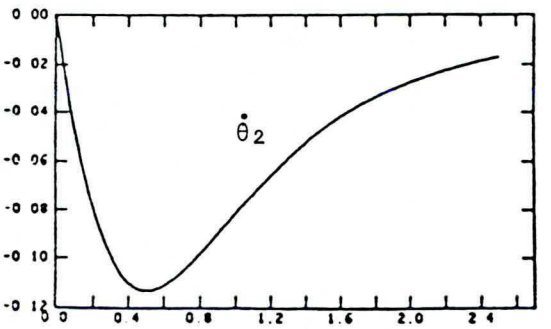
(a)



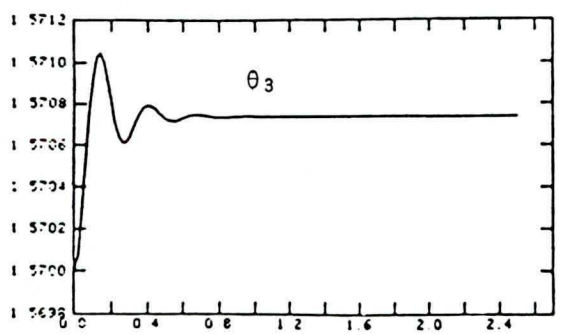
(d)



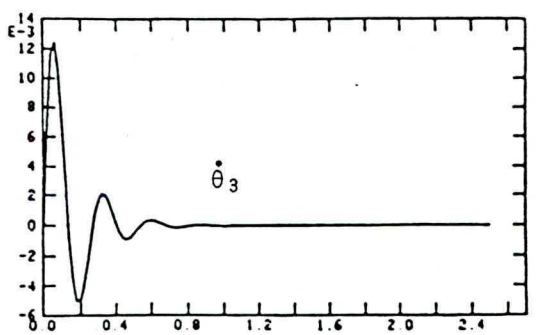
(b)



(e)

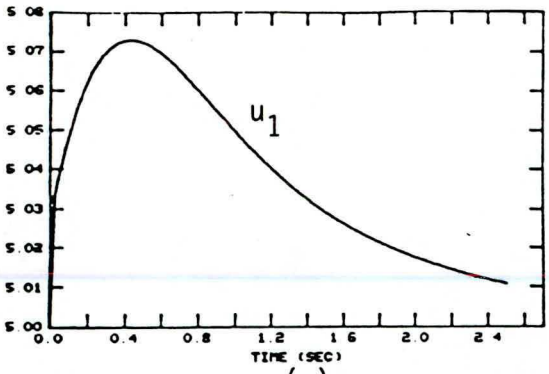


(c)

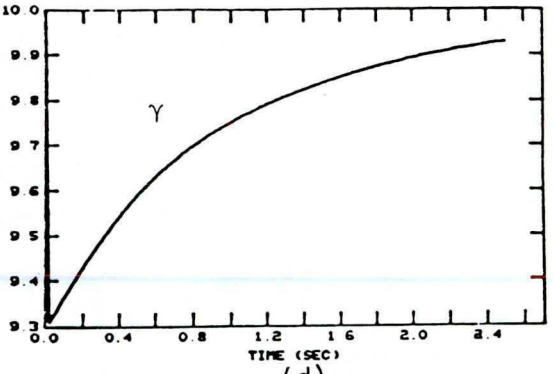


(f)

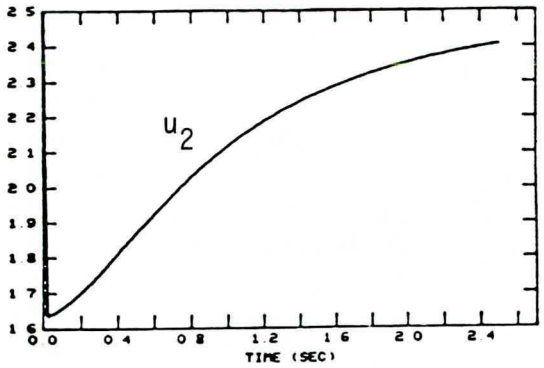
Figure 4.3.1 Positions (in rad) and Velocities (in rad/sec) vs. Time



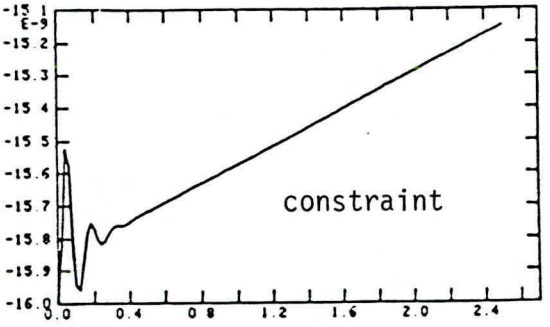
(a)



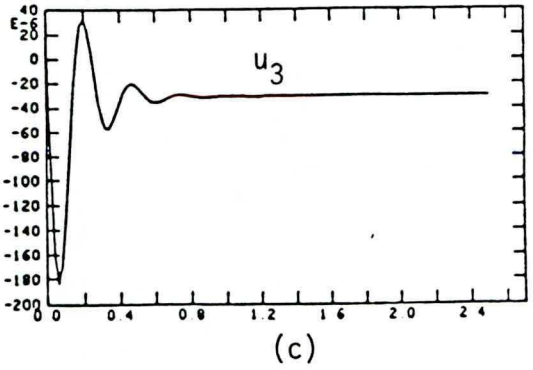
(d)



(b)



(e)



(c)

Figure 4.3.2 Torques (in N-m) force (in N), and constraint vs. Time

not constant.

The system is to track a sinusoidal trajectory, where the point of contact of the arm with the wall moves with simple harmonic motion over a vertical distance of about twelve centimeters. This is accomplished by letting θ_1 vary sinusoidally from 0° to -40°] (0 to -0.7 rad). Similarly, θ_2 moves through an angle equal in magnitude but opposite in sign to θ_1. Link three stays in the horizontal position such that θ_3 = 90° ($\pi/2$ radians). The position of the arm at the highest and lowest points on the trajectory is shown in Figures 4.4.1 and 4.4.2, respectively.

The time-varying bias trajectories are given in equations 4.4.1 to 4.4.3. The bias velocity $\dot{\theta}_b$ is the first time derivative of the position θ_b, and the bias acceleration $\ddot{\theta}_b$ is the second derivative of the position with respect to time.

$$\begin{bmatrix} \theta_{b1} \\ \theta_{b2} \\ \theta_{b3} \end{bmatrix} = \begin{bmatrix} -0.70 * [.5 - .5 \cos(\frac{2\pi t}{T})] \\ -0.70 * [.5 - .5 \cos(\frac{\pi t}{T})] \\ \frac{\pi}{2} \end{bmatrix} \qquad (4.4.1)$$

$$\begin{bmatrix} \dot{\theta}_{b1} \\ \dot{\theta}_{b2} \\ \dot{\theta}_{b3} \end{bmatrix} = \begin{bmatrix} -0.70 \ (\frac{2\pi}{T})(0.5 \sin(\frac{2\pi t}{T})) \\ -0.70 \ (\frac{2\pi}{T})(0.5 \sin(\frac{2\pi t}{T})) \\ 0 \end{bmatrix} \qquad (4.4.2)$$

$$\begin{bmatrix} \ddot{\theta}_{b1} \\ \ddot{\theta}_{b2} \\ \ddot{\theta}_{b3} \end{bmatrix} = \begin{bmatrix} -0.70 \ (\frac{2\pi}{T})^2(0.5 \cos(\frac{2\pi t}{T})) \\ -0.70 \ (\frac{2\pi}{T})^2(0.5 \cos(\frac{2\pi t}{T})) \\ 0 \end{bmatrix} \qquad (4.4.3)$$

In the above equations θ_b is measured in radians, $\dot{\theta}_b$ is measured in rad/sec$_2$. The period T is chosen to be two seconds for these simulations.

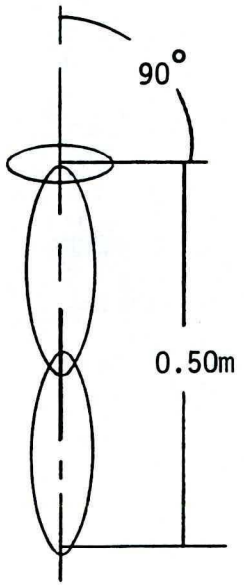


Figure 4.4.1 Robot Arm Fully Extended

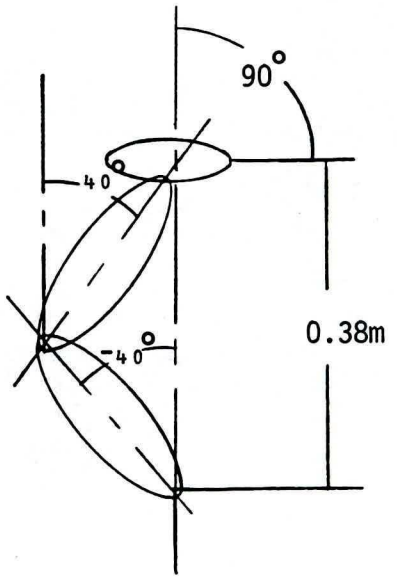


Figure 4.4.2 Robot Arm Fully Contracted

In order to implement this periodic movement, the bias equations (4.4.1) and (4.4.2) are subtracted from the actual values of Θ and $\dot{\Theta}$ to generate an error signal. The error signal is then multiplied by the feedback gains to apply the state-variable feedback. The above two bias equations are used along with (4.4.3) to generate the appropriate control signals for the inverse plant pre-compensator. At each time step, the bias values Θ_b, $\dot{\Theta}_b$, $\ddot{\Theta}_b$ are substituted into Eq. 2.3.11 to solve for the control signals of the inverse plant as shown below.

$$U_{in} = C^{-1}[J(\Theta_b)\ddot{\Theta}_b + B(\Theta_b)\dot{\Theta}_b^2 + F(\Theta_b) - \frac{\partial P^T}{\partial \Theta}\Big|_{\Theta = \Theta_b} \gamma_b \,] \qquad (4.4.4)$$

The results of the digital computer simulation are shown in Figures 4.4.3 and 4.4.4. In Figures 4.4.3a and 4.4.3b, the graphs of θ_1 and θ_2 are superimposed on the graphs of the reference positions. Similarly, the graphs of $\dot{\theta}_1$ and $\dot{\theta}_2$ are superimposed on the corresponding bias velocity graphs. The position and velocity graphs track the reference, but with phase lag and overshoot. This may or may not be a problem, depending upon the specific application. If the arm is painting a wall, then lag and overshoot are acceptable. If closer following of the reference is desired, then some changes must be made.

The value of the force γ on the wall varies about the bias value (γ_b = 10 N), as shown in Figure 4.4.4d. The constraint, as shown in Figure 4.4.4 e is maintained. The force on the wall will be controlled by force feedback in the next section, thus providing a smaller range of excursion of the force about the bias value.

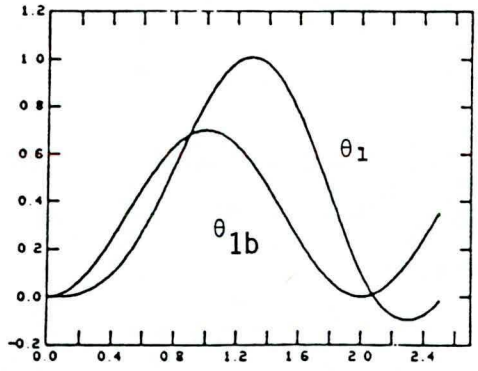
The phase lag and overshoot can be handled in a number of ways. One way is to vary some parameters of the system. To change system parameters, one can either move the system poles and speed up the system, or one can use adaptive control. A simulation was performed where the angles varied sinusoidally from 0° to 10°, and the system tracked nearly perfectly. From this simulation, one can infer that the methods are valid for small excursions about the operating point.

Another way to decrease phase lag and overshoot is to use dynamic state feedback, accomplished by implementation of a servo compensator.

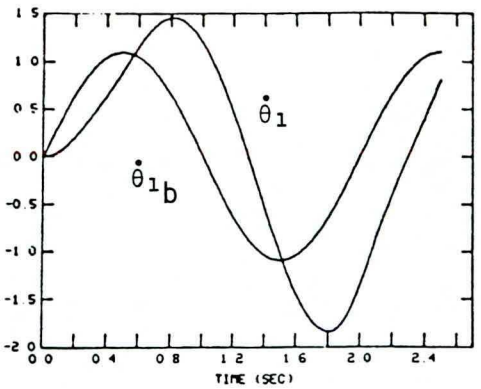
4.5 Effect of Force Feedback

It was shown in the previous section that the robot arm is able to track a desired trajectory, but the force exerted on the wall varies around a constant reference force. In this section, force feedback is used to keep the force on the wall closer to the reference force.

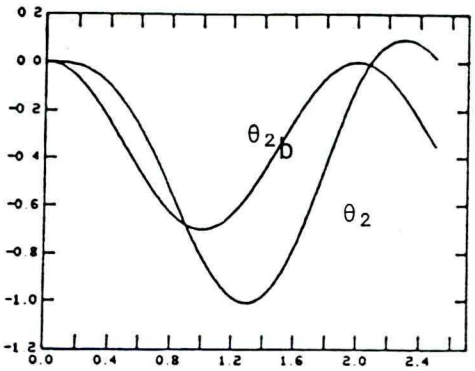
All of the simulations done in this section track the same



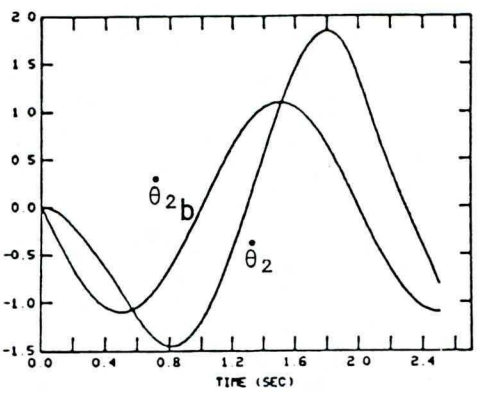
(a)



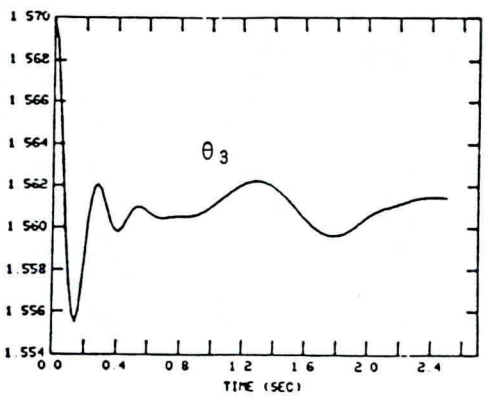
(d)



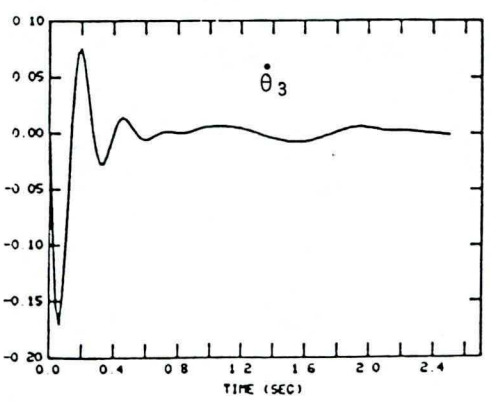
(b)



(e)

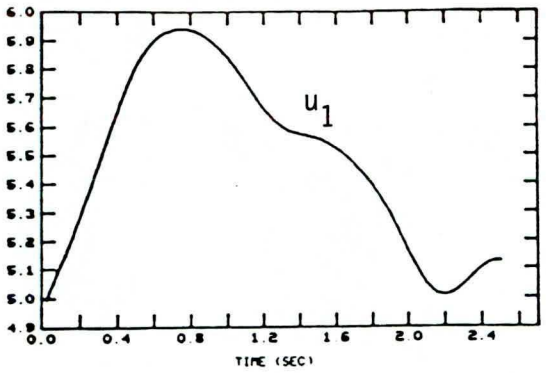


(c)

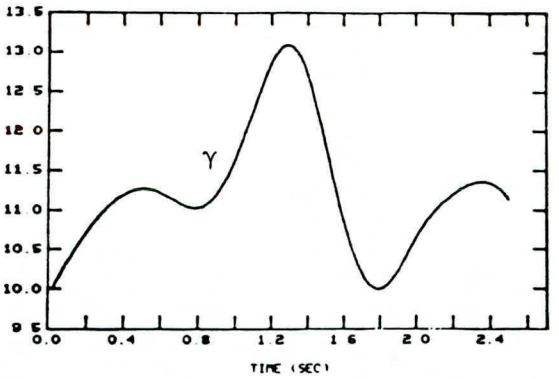


(f)

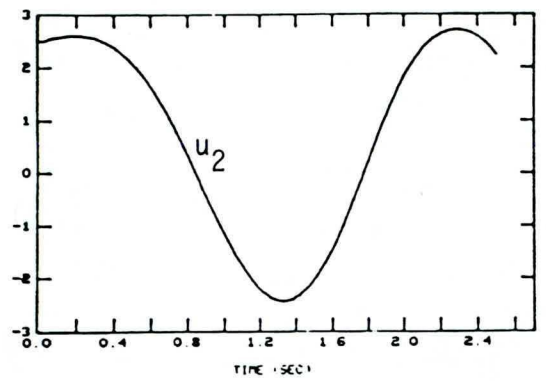
Figure 4.4.3 Angular Positions (in rad) and Velocities (in rad/sec) vs. Time



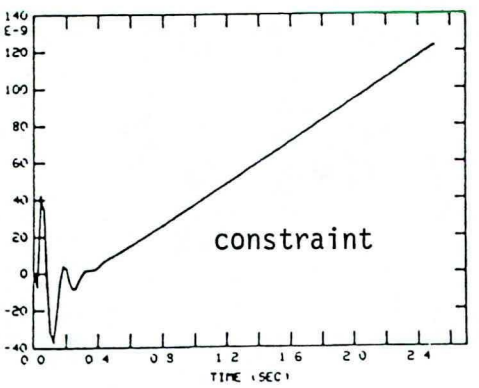
(a)



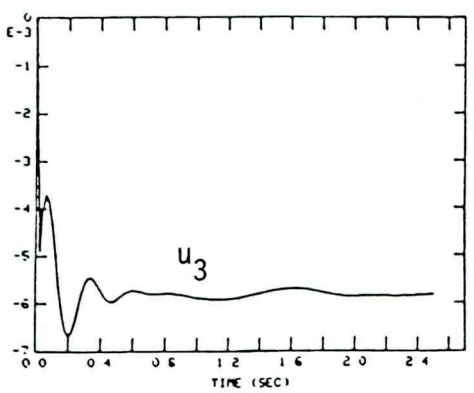
(d)



(b)



(e)



(c)

Figure 4.4.4 Torques (in N-m), force (in N), and constraint vs. Time

sinusoidal motion as in the previous section. Improvements and modifications are made by implementing a force feedback loop. In this loop, a transfer function is needed which models the force sensing device, and feedback gains need to be calculated so that the appropriate control signals can be derived.

In the first simulation, the force sensor is modelled as a delay element. The amount of delay was chosen to be a 40 ms delay. Although the size of the delay depends on both the type of material used for the sensor and the speed of the processor, we can assume that this value is reasonable. In the digital simulations, the steps are taken in T = 20 ms increments. The discrete time equations for the sensor can be written as in Eq. 4.5.1.

$$\gamma_e(kT) = \gamma((k-2)T) - \gamma_d \tag{4.5.1}$$

This error signal is multiplied by a (3x1) gain matrix, and the appropriate control signals are derived. However, one must first calculate the gain values. Using Eq. 3.5.3, gain values can be calculated based on the following static values:

$$\dot{\Theta} = 0,\ \ddot{\Theta} = 0,\ \Theta = \left[\, 0 \quad 0 \quad \frac{\pi}{2} \,\right]^T \tag{4.5.2}$$

These gains are valid for small excursions about an operating point. In the simulations done in this section, a sinusoidal trajectory is desired. This type of reference trajectory deviates from the static reference values by too much, causing instability. Therefore, the gain matrix G_1 is multiplied by a constant $\alpha = .2$ so that the gains in Eq. 4.5.3 result.

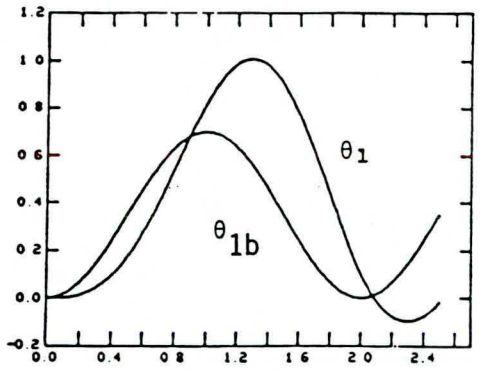
$$G_1 = [0.1 \quad 0.05 \quad 0.0]^T \tag{4.5.3}$$

Using these gains, the control signals for the force feedback loop can be derived and are given in Eq. 4.5.4.

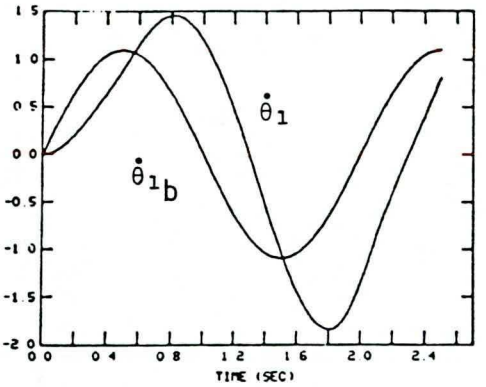
$$U_{FFB} = G_1 \gamma_e \tag{4.5.4}$$

The simulation is shown in Figures 4.5.1 and 4.5.2. Notice the decrease in the excursion of the force on the wall, γ. The bias force is again 10N.

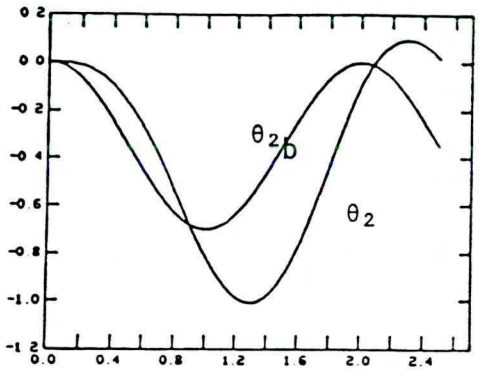
The next simulation uses the same transfer function for the sensor, but different feedback gains. In this section, time varying feedback is used, as described by Eq. 3.5.5. The reference positions are the same as those described by Eq. 4.4.1. A simulation was done using these gains resulting instability. The gains were then multiplied



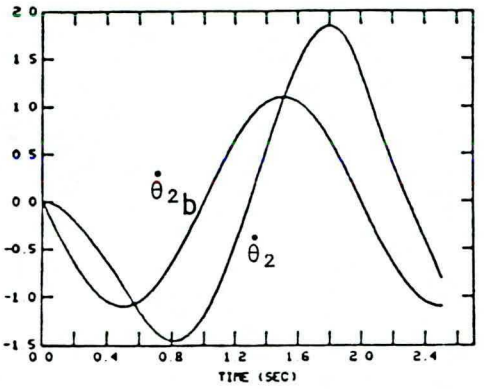
(a)



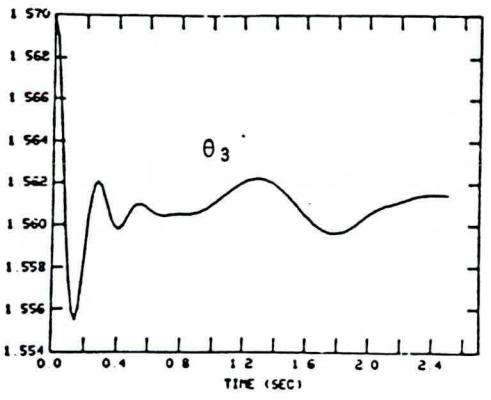
(d)



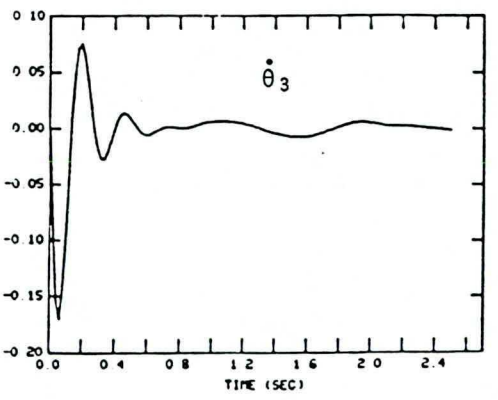
(b)



(e)

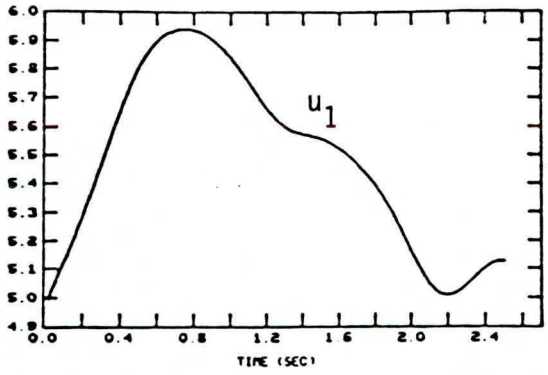


(c)

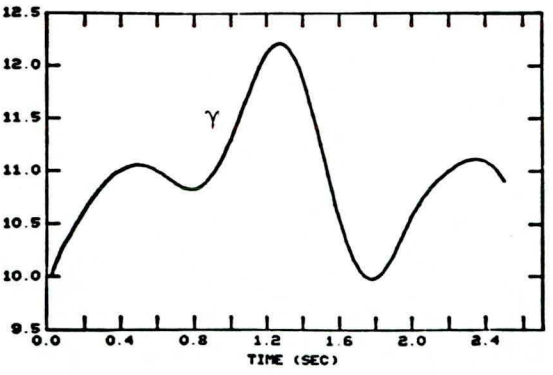


(f)

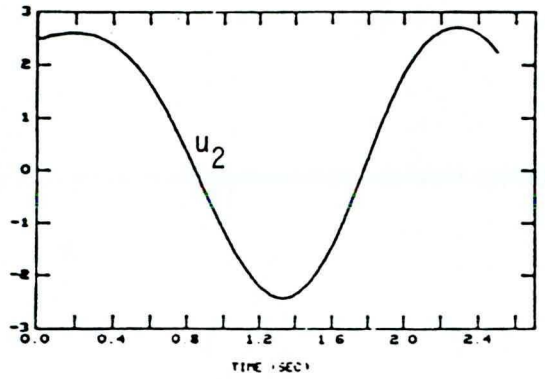
Figure 4.5.1 Angular Positions (in rad) and Velocities (in rad/sec) vs. Time



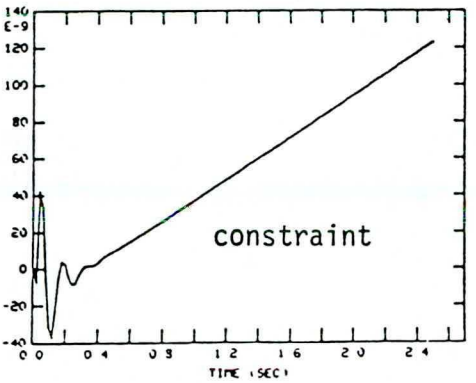
(a)



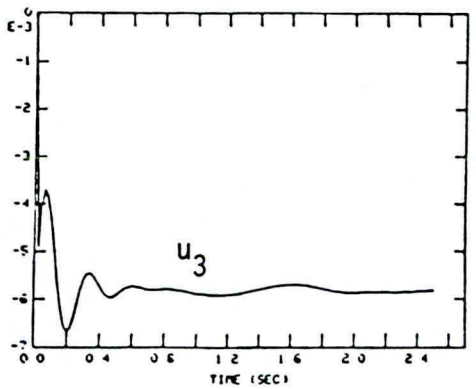
(d)



(b)



(e)



(c)

Figure 4.5.2 Torques (in N-m), force (in N), and constraint vs. Time

by α = 0.5, and the resulting simulation is shown in Figures 4.5.3 and 4.5.4.

Notice once again that the maximum value of γ has now decreased to about 11.6 Newtons. One can infer that tighter restrictions could be made on the excursions of the force if the trajectories were followed more closely.

4.6 Touching the Wall

In this section, the problem of moving from unconstrained to constrained motion is explored. The robot arm is to start from a rest position away from the wall. It will then move toward the wall and re-establish equilibrium after making contact with the wall. To avoid damage to the arm, a structure is mounted on the tip of the manipulator which absorbs the force of impact.

The arm will start at the following initial position.

$$\Theta = [\ -10° \quad -10° \quad 80°\ ] \tag{4.6.1}$$

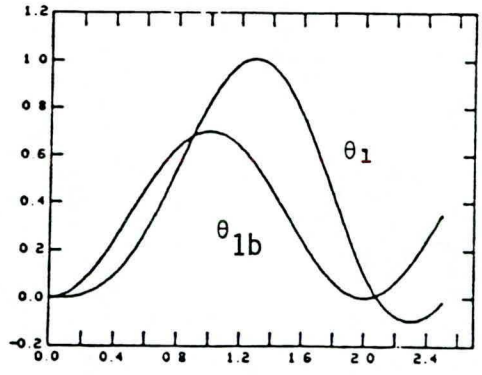
The positions Θ follow a specific trajectory and the control signals to follow this trajectory are given by the inverse plant as in the previous section. The force exerted on the wall is calculated the same way as in section 3.4. It is calculated as follows:

$$F = \begin{cases} 0 & \text{if} \quad \varepsilon < 0 \\ -K\varepsilon - B\dot{\varepsilon} & \text{if} \quad x_{com} < \varepsilon \leq x_{eq} \\ \gamma(X, U) & \text{if} \quad \varepsilon \leq x_{com} \end{cases} \tag{4.6.2}$$

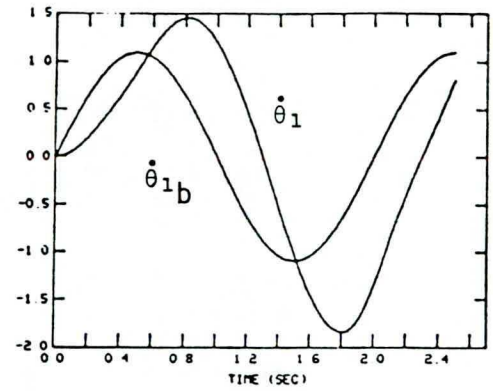
In the above equation x_{eq} is the length of the absorbing structure at rest, and x_{com} is the length when fully compressed. In this simulation, x_{eq} = 10 mm and x_{com} = 5 mm. If these values along with K = 100, and B = 50 are substituted into Eq. 4.6.2 the following equation results.

$$F = \begin{cases} 0 & \text{if} \quad \varepsilon < 0 \\ -100\ \varepsilon - 50\ \dot{\varepsilon} & \text{if} \quad 5\ \text{mm} < \varepsilon \leq 10\ \text{mm} \\ \gamma(X, U) & \text{if} \quad \varepsilon \leq 5\ \text{mm} \end{cases} \tag{4.6.3}$$

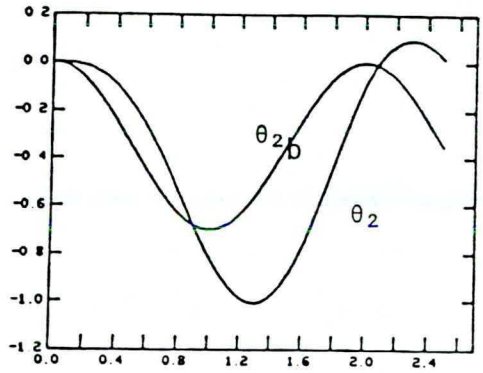
The reference trajectories for Θ can be described as follows. The simulation starts with the system at rest. At t = 0.2 sec, the reference is a ramp function until contact is made with the wall. The reference then switches to the final equilibrium position, $\Theta = [\ -0.01 \quad -0.01 \quad \pi/2\ ]^2$, where all angles are measured in radians.



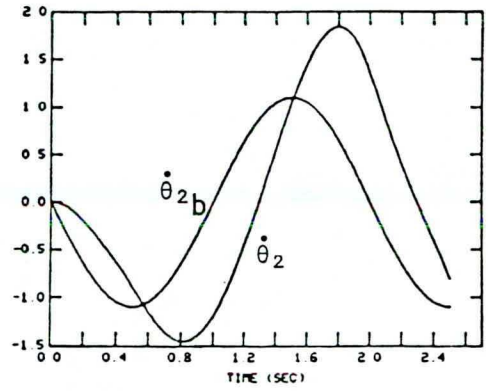
(a)



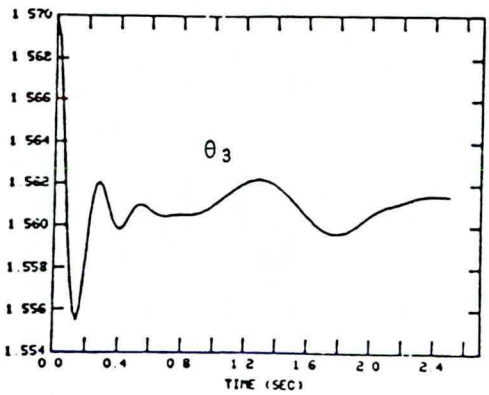
(d)



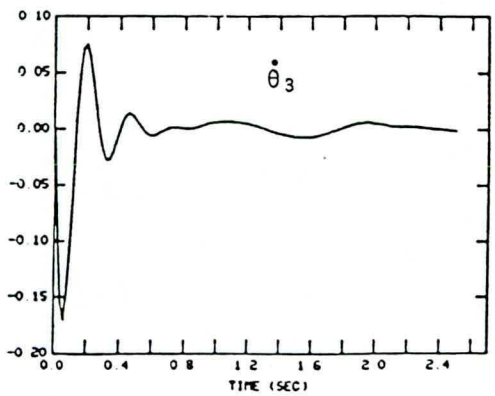
(b)



(e)

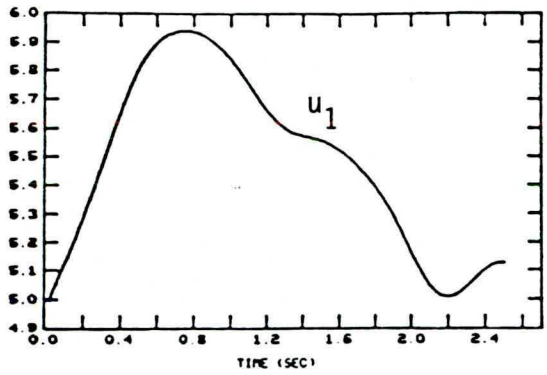


(c)

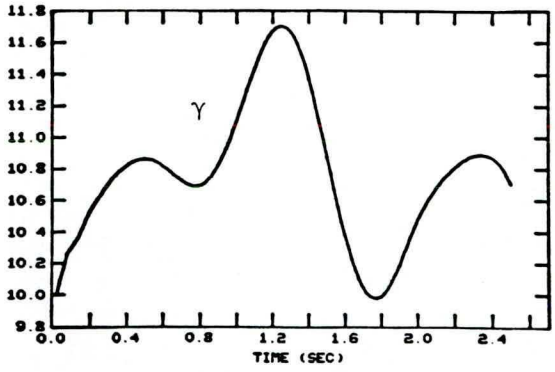


(f)

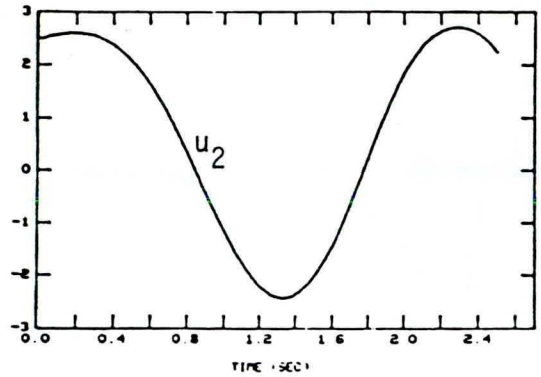
Figure 4.5.3 Angular Positions (in rad) and Velocities (in rad/sec) vs. Time



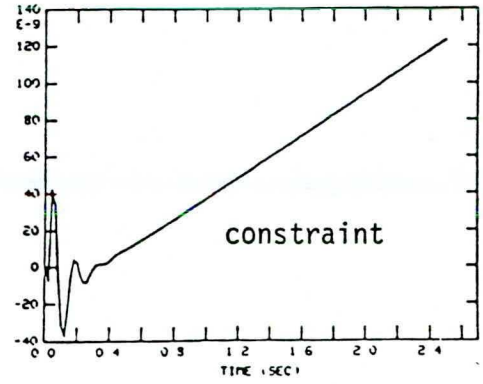
(a)



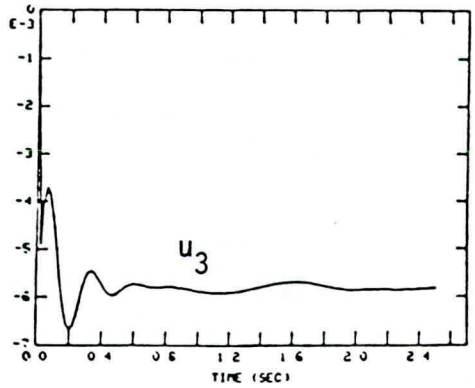
(d)



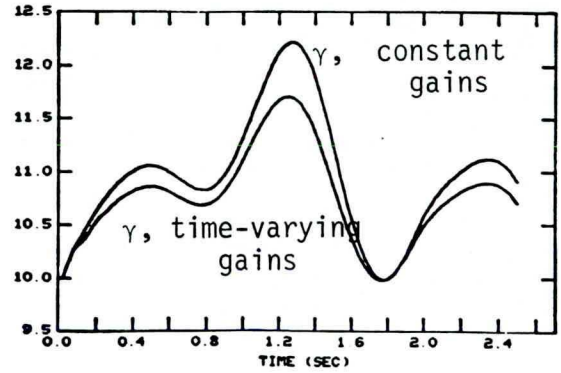
(b)



(e)



(c)



(f)

Figure 4.5.4 Torques (in N-m), force (in N), and constraint vs. Time

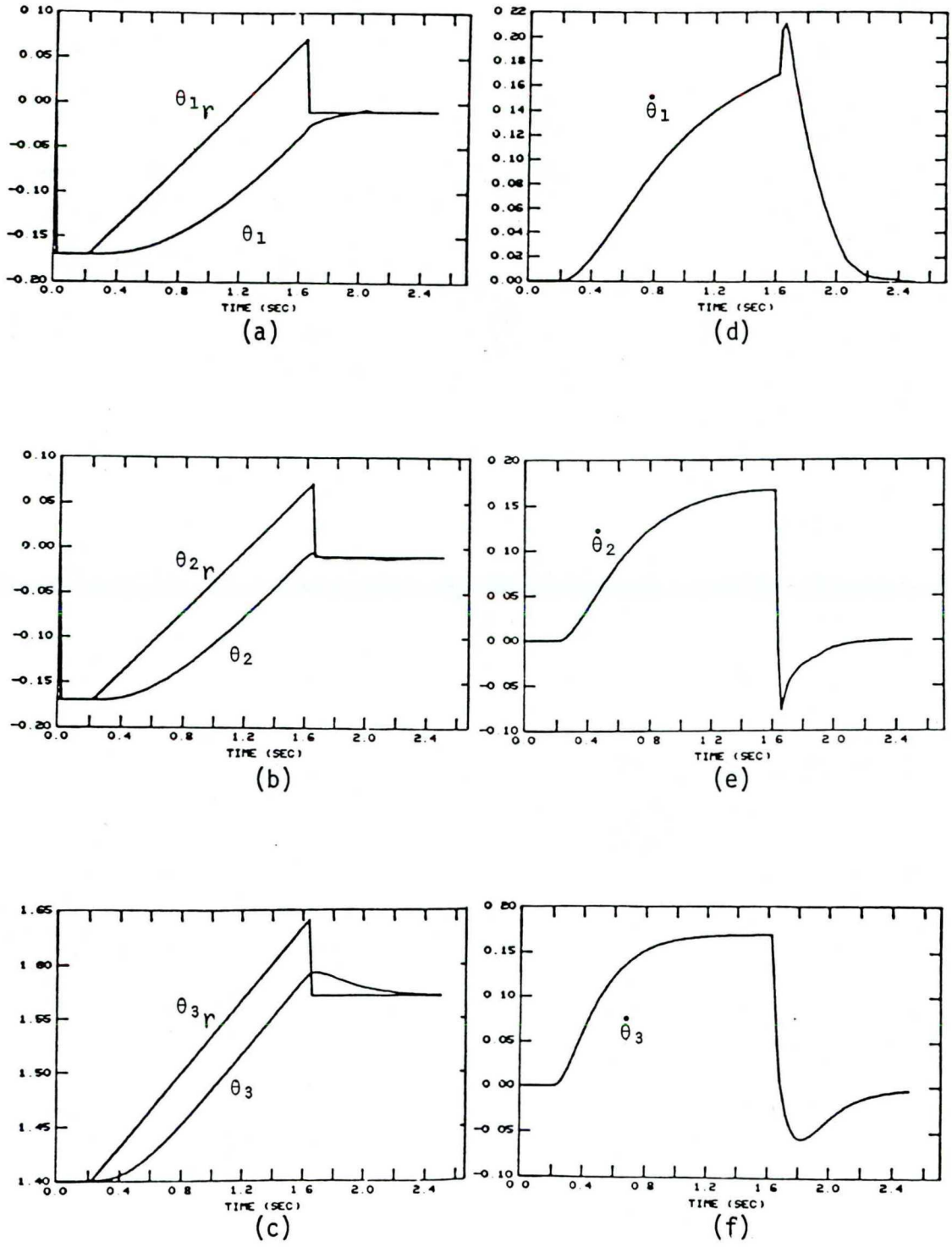
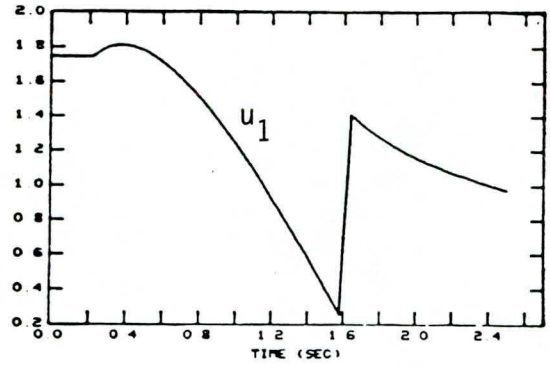
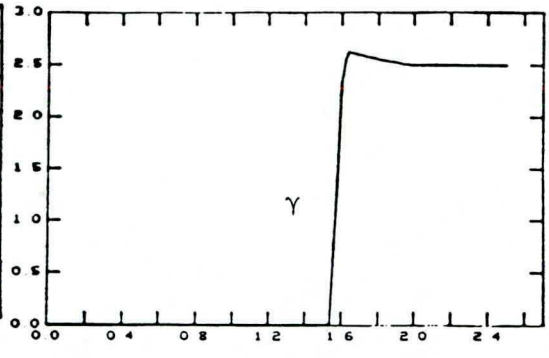


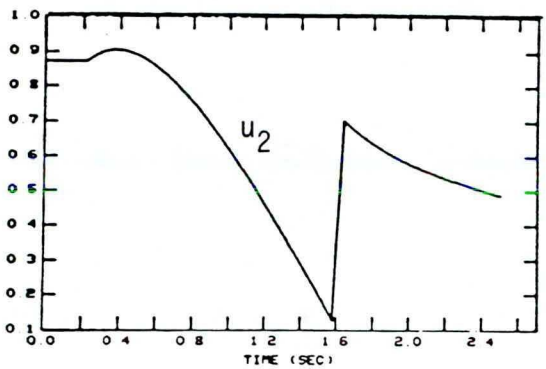
Figure 4.6.1 Positions (in rad) and Velocities (in rad/sec) vs. Time



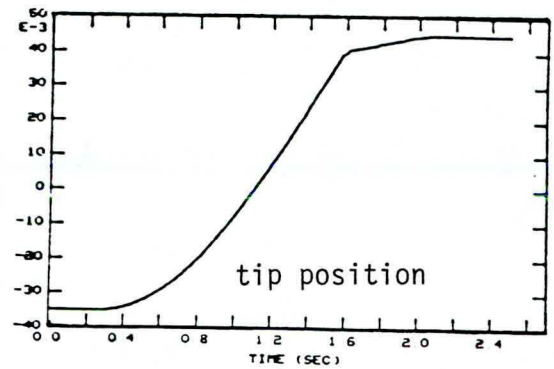
(a)



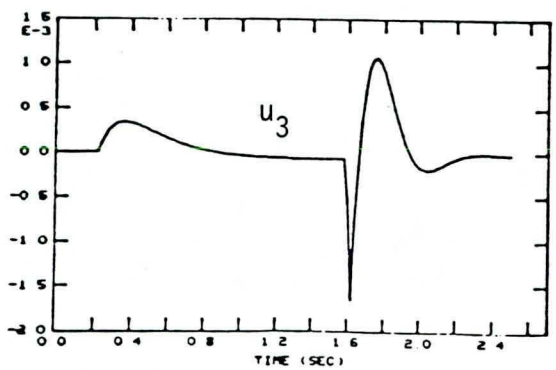
(d)



(b)



(e)



(c)

Figure 4.6.2 Torques (in N-m), Force (in N), and position of tip from the wall (in m)

The simulations are shown in Figures 4.6.1 and 4.6.2. The positions Θ are shown with their reference trajectories. They each lag the reference, but end at their equilibrium values. The graphs of the velocities Θ all tend toward zero at the end of the simulation. The force γ, shown in Figure 4.6.2d, is zero while no contact is made, and immediately jumps to about 2.5N after making contact. The system stays at rest with the force remaining at 2.5N. A graph of ε vs. time is included in Figure 4.6.2e. Notice that after making contact with the wall, the system remains at the position where ε = 5 mm, which is the value for x_{com} as described previously.

4.7 Summary

In this chapter, the results of the previous chapters were shown to be valid through the use of digital computer simulations. The system was shown to be stable during both free motion and constrained motion. Also, the arm was able to move from unconstrained to constrained motion in the same simulation. A tracking problem was discussed, and force feedback helped to provide a fine control on the force exerted on a wall while tracking a desired motion.

Chapter 5

SUMMARY AND RECOMMENDATIONS

5.1 Summary

A planar three-link robot arm was presented and modelled in the sagittal plane. In Chapter 2, the equations of motion were derived. A control strategy was derived which helped to stabilize and decouple the system. In addition, a control strategy was derived so that the system could track a desired motion. The robot arm was equipped with a force sensor to provide force feedback and to enable fine control on the force exerted on a wall.

In Chapter 3, explicit feedback gains were derived based on poles which were placed to stabilize the system. The control necessary for moving from unconstrained to constrained motion was also discussed.

Chapter 4 verified all of the results of the previous chapters through digital computer simulations. Both unconstrained and constrained motion were shown, as well as a tracking system. Force feedback was used effectively to provide a smaller excursion of the force exerted on a wall.

5.2 Recommendations

Many areas remain to be addressed in the robotics industry. This thesis is a step toward understanding robots, and also lays the basis for further work in this area. This thesis can be expanded in many ways.

In this thesis, constrained motion was done where the robot arm moved along a flat frictionless surface. It would be interesting to do constrained motion along a curved surface. Or, one could simulate frictional forces on the surface. This is perhaps not a difficult problem, due to the way this particular arm is built. Since the third link is pivoted at the center of gravity, frictional forces could be controlled by the torque generator for the third link, u_3.

Another problem that could be addressed is the possibility of using two of these arms at the same time. The arms would have to be coordinated in order to work in tandem. One example is picking up an

object. Another problem might be to equip to robot arm with a grasping mechanism. Or, alternatively, to use a more complicated tactile sensor.

This thesis is also significant because there is a hardware model available so that one can compare experimental results with theoretical results. It is difficult to do this with biped motion, because feedback gains, torques, etc. are difficult to derive from human motion. However, in robotics, one can verify the theory presented herein.

There are many areas to be explored, and this work is a starting place for future research in this area.

APPENDIX A

Detailed Derivation of the Equations of Motion

The robot arm can be analyzed as a three-link inverted pendulum in the sagittal plane, (Figure A1). All of the links are connected by frictionless pin joints, and the bottom link is connected to the ground. Each link L_i ($i = 1, 2, 3$) is characterized by four parameters: its mass m_i, its moment of inertia, I_i about the center of mass, the distance l_i from joint i to joint I + 1, and the distance from joint i to the center of mass k_i. These parameters are shown in Figure A2. The angle θ_i of each link is measured clockwise from the vertical. A vertical wall is situated at $x = d$. The arm leans against this wall with a force γ.

To derive the equations of motion, the robot can be divided into three free body diagrams. Each link can be represented as in Figure A3. Let F_i and G_i be the reaction forces acting on link i at pin joint i. Let x_i and y_i be the horizontal and vertical translation of the center of mass of link i, and let g be the gravitational acceleration: $g = 9.81\ m/sec^2$.

The sum of the horizontal forces is equal to the product of mass times horizontal acceleration of the center of mass.

$$\begin{aligned} F_1 - F_2 &= m_1\ddot{x}_1 \\ F_2 - F_3 &= m_2\ddot{x}_2 \\ F_3 - \gamma &= m_3\ddot{x}_3 \end{aligned} \qquad \text{(A1)}$$

The sum of the vertical forces equals the product of mass and vertical acceleration of the center of mass.

$$\begin{aligned} G_1 - m_1 g - G_2 &= m_1\ddot{y}_1 \\ G_2 - m_2 g - G_3 &= m_2\ddot{y}_2 \\ G_3 - m_3 g - G_2 &= m_3\ddot{y}_3 \end{aligned} \qquad \text{(A2)}$$

In addition, the algebraic sum of clockwise torques exerted on each link is equal to the product of moment of inertia and angular acceleration.

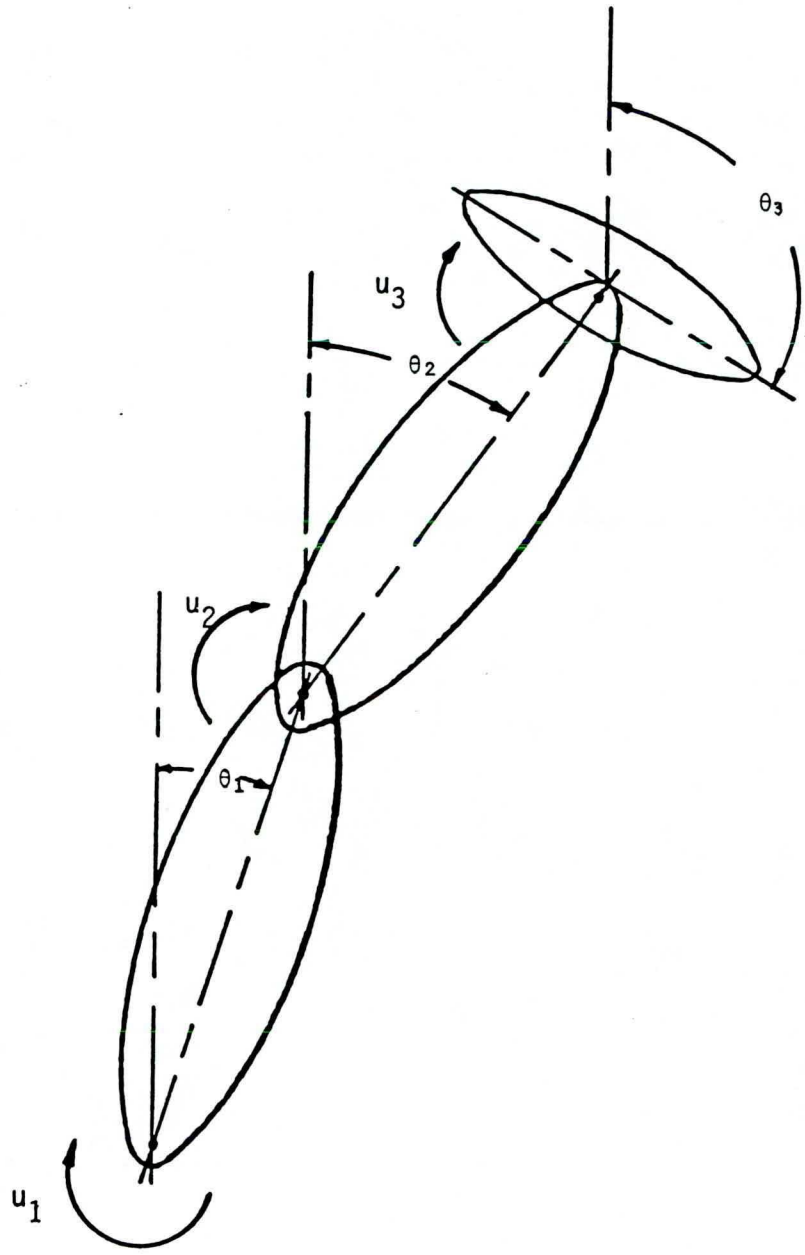


Figure A1 The Three-Link Robot Arm

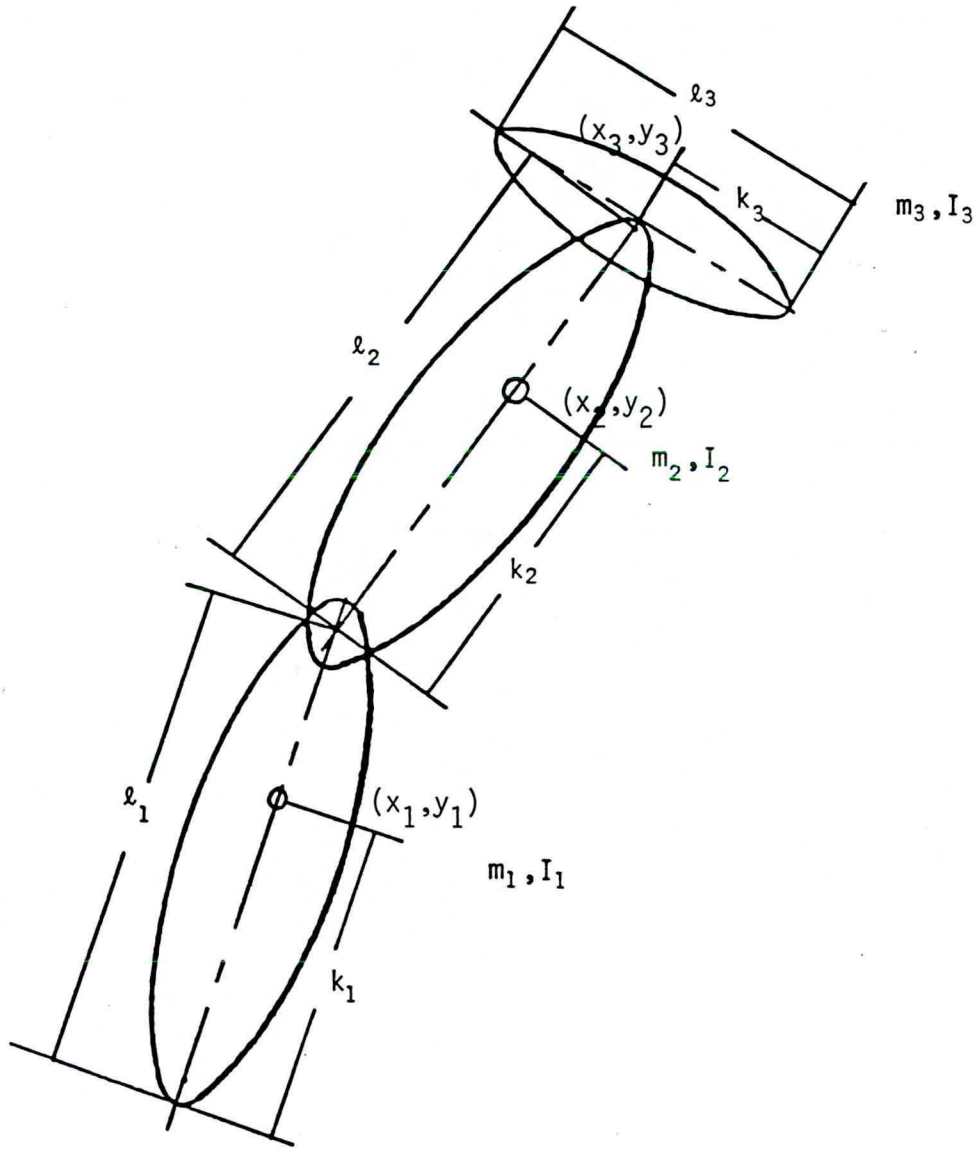
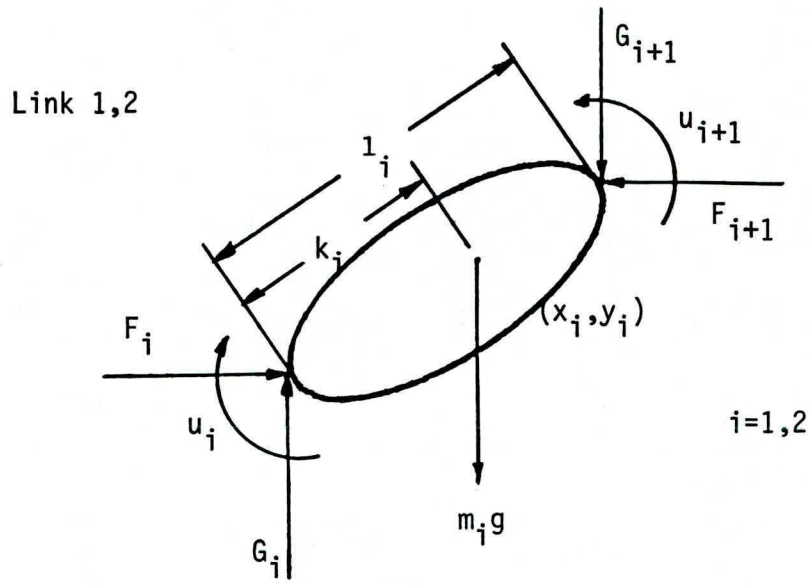


Figure A2 Parameters of the Robot Arm



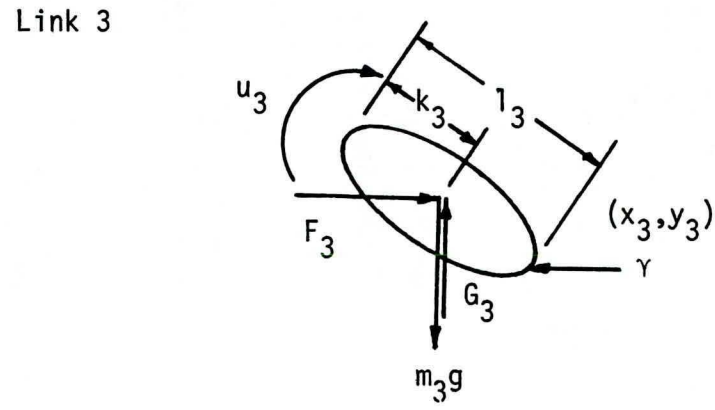


Figure A3 Free Body Diagrams of Each Link

$$I_1\ddot{\theta}_1 = G_2(l_1 - k_1)\sin\theta_1 + G_1k_1\sin\theta_1 - F_1k_1\cos\theta_1$$
$$- F_2(l_1 - k_1)\cos\theta_1 + u_1 - u_2$$

$$I_2\ddot{\theta}_2 = G_2k_2\sin\theta_2 + G_3(l_2 - k_2)\sin\theta_2 - F_2k_2\cos\theta_2$$
$$- F_3(l_2 - k_2)\cos\theta_2 + u_2 - u_3$$

$$I_3\ddot{\theta}_3 = u_3 - \gamma k_3\cos\theta_3 \qquad \text{(A3)}$$

There are six constraint equations which are derived from Figure A2.

$$x_1 = k_1\sin\theta_1$$
$$y_1 = k_1\cos\theta_1$$

$$x_2 = k_2\sin\theta_2 + l_1\sin\theta_1$$
$$y_2 = k_2\cos\theta_2 + l_1\cos\theta_1$$

$$x_3 = l_2\sin\theta_2 + l_1\sin\theta_1$$
$$y_3 = l_2\cos\theta_2 + l_1\cos\theta_1 \qquad \text{(A4)}$$

Equations (A1) and (A2) are solved to yield force equations in terms of mass times acceleration and γ, which is assumed known.

$$f_i = \sum_{k=1}^{3} m_k\ddot{x}_k + \gamma \qquad i = 1, 2, 3$$

$$g_i = \sum_{k=1}^{3} m_k\ddot{y}_k \qquad i = 1, 2, 3 \qquad \text{(A5)}$$

The constraint equations are twice differentiated and solved for $\ddot{x}_i$, $\ddot{y}_i$ and substituted into the force equations, (A5). These new force equations are then substituted into Eq. A3 the torque equations, eliminating all internal forces except γ, the constraint force.

After simplification, the following equations are obtained.

$$
\begin{aligned}
I\ddot{\theta}_1 &= (-mk_1^2 - m_1 l_2^2 - m_3 l_1^2)\ddot{\theta}_1 \\
&\quad - \ddot{\theta}_2[(m_2 l_1 k_2 + m_3 l_1 l_2)\cos(\theta_1 - \theta_2)] \\
&\quad - \dot{\theta}_2^2[(m_2 l_1 k_2 + m_3 l_1 l_2)\sin(\theta_1 - \theta_2)] \\
&\quad + [(m_2 + m_3) g l_1 + m_1 k_1 g]\sin\theta_1 - l_1 \gamma \cos\theta_1 + u_1 - u_2 \\
I\ddot{\theta}_2 &= -\ddot{\theta}_1[(l_1 l_2 m_3 + l_1 k_2 m_2)\cos(\theta_1 - \theta_2)] \\
&\quad + \ddot{\theta}_2[-l_2^2 m_3 - k_2^2 m_2] \\
&\quad + \dot{\theta}_1^2[(l_1 k_2 m_2 + l_1 l_2 m_3)\sin(\theta_1 - \theta_2)] \\
&\quad + (l_2 m_3 g + k_2 m_2 g)\sin\theta_2 - l_2 \gamma \cos\theta_2 + u_2 - u_3 \\
I\ddot{\theta}_3 &= -\gamma k_3 \cos\theta_3 + u_3
\end{aligned}
\tag{A6}
$$

Matrix notation provides ease of algebraic manipulation for the equations of motion. Expressing these equations in matrix form yields:

$$
J(\Theta)\ddot{\Theta} + B(\Theta)\dot{\Theta}^2 + F(\Theta) = CU + D(\Theta)\gamma \tag{A7}
$$

The elements of each matrix are given below:

$$
\begin{aligned}
j_{11} &= I_1 + m_1 k_1^2 + m_2 l_1^2 + m_3 l_1^2 \\
j_{12} &= (m_2 l_1 k_2 + m_3 l_1 l_2)\cos(\theta_1 - \theta_2) \\
j_{13} &= 0 \\
j_{21} &= (m_2 l_1 k_2 + m_3 l_1 l_2)\cos(\theta_1 - \theta_2) \\
j_{22} &= I_2 + m_2 k_2^2 + m_3 l_2^2 \\
j_{23} &= 0 \\
j_{31} &= 0 \\
j_{32} &= 0 \\
j_{33} &= I_3 \\
b_{11} &= 0 \\
b_{12} &= -(m_2 l_1 k_2 + m_3 l_1 l_2)\sin(\theta_1 - \theta_2)
\end{aligned}
$$

$$b_{13} = 0$$

$$b21 = -(m_2 l_1 k_2 + m_3 l_1 l_2) \sin(\theta_1 - \theta_2)$$

$$b_{22} = b_{23} = b_{31} = b_{32} = 0$$

$$f_1 = -[(m_2 + m_3) l_1 + m_1 k_1] g \sin\theta_1$$

$$f_2 = -[m_2 k_2 + m_3 l_2] g \sin\theta_2$$

$$f_3 = 0$$

$$d_1 = -l_1 \cos\theta_1$$

$$d_2 = -l_2 \cos\theta_2$$

$$d_3 = -k_3 \cos\theta_3$$

$$C = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\ddot{\Theta} = [\, \ddot{\theta}_1 \quad \ddot{\theta}_2 \quad \ddot{\theta}_3 \,]^T$$

$$\dot{\theta}^2 = [\, \dot{\theta}_1^2 \quad \dot{\theta}_2^2 \quad \dot{\theta}_3^2 \,]^T$$

$$U = [\, u_1 \quad u_2 \quad u_3 \,]^T$$

Note that the contact constraint equation is

$$P(\Theta) = -(l_1 \sin\theta_1 + l_2 \sin\theta_2 + k3 \sin\theta_3) - d \qquad \text{(A8)}$$

Differentiating Eq. A8 with respect to position θ, gives

$$-\frac{\partial P^T}{\partial \Theta} = D(\Theta) \qquad \text{(A9)}$$

Substituting Eq. A9 into A7, we obtain the final form of the equations of motion.

$$J(\Theta)\ddot{\Theta} + B(\Theta)\dot{\Theta}^2 + F(\Theta) = CU - \frac{\partial P^T}{\partial \Theta} \gamma \qquad \text{(A10)}$$

Equation A10 is convenient for two reasons. It is a compact form for writing the equations of motion, which simplifies computations. Also it can be used for both constrained and unconstrained motion, where the latter is the case where $\gamma = 0$.

REFERENCES

[1] Paul, Richard P., Robot Manipulators, The M.I.T. Press, Cambridge, Massachusetts, 1981.

[2] Vukobratovic, M., Potkonjak, V., Dynamics of Manipulation Robots, Springer-Verlag, Berling, Heidelberg, New York, 1982.

[3] Vukobratovic, M., Stokle, D., Control of Manipulation Robots, Springer-Verlag, Berlin, Heidelberg, New York, 1982.

[4] Albus, J.S., Brains, Behavior, and Robotics, McGraw Hill Co., New York, 1981.

[5] Brady, M., Hollerbach, J.M., Johnson, T.L., Lozano-Perez, T. and Mason, M.T., Robot Motion: Planning and Control, The M.I.T. Press, Cambridge, Massachusetts, 1982

[6] Goddard, R.E., Hemami, H., Weimer, F.C., "Biped Sidestep in the Frontal Plane," IEEE Trans. on Auto. Control, Vol. AC-28, No. 2, February 1983.

[7] Hemami, H., Jaswa, V.C., "On a Three-Link Model of the Dynamics of Standing Up and Sitting Down," IEEE Trans. on Systems, Man, and Cybernetics, Vol. SMC-8, No. 2, February 1978.

[8] Kish, L.R., A Computer Model and Display of Articulated Human Movement, M.S. Thesis, The Ohio State University, Columbus, Ohio.

[9] Beck, R.P., A Computer Graphics System for the Study of Human Mechanics, M.S. Thesis, The Ohio State University, Columbus, Ohio.

[10] Ceranowicz, A.Z., Wyman, B.F., and Hemami, H., "Control of Constrained Systems of Controllability Index Two," IEEE Trans. on Automatic Control, Vol. AC-25, No.6, Dec. 1980, pp. 1102-1111.

[11] Moore, B.C., "On the Flexibility Offered by State Feedback in Multivariable Systems Beyond Closed Loop Eigenvalue Assignment," IEEE Transactions on Auto. Control, Vol. AC- 21 No. 5, October 1976, pp. 689-692.

[12] Bavarian, B., Wyman, B.F., Hemami, H., "Control of the Constrained Planar Simple Inverted Pendulum," International Journal of Control, 1983, Vol. 37, No. 4, pp. 741-753.

[13] Raibert, M.H. and Craig, J.J., "Hybrid Positions Force Control of Manipulators," J. Dynamic Systems, Measurement, Control, 102, June 1981, pp. 126-133.

[14] Dubowsky, S., DesForges, D.T., "The Application Model-Referenced Adaptive Control to Robotic Manipulators," Journal of Dynamic Systems, Measurement, and Control, September 1979, Vol. 101, pp. 193-200.

[15] Narendra, K.S., Valavani, L.S., "Direct and Indirect Model Reference Adaptive Control," Automatica, Vol. 15, pp. 653-664, 1979.

[16] Fever, A., Morse, S., "Adaptive Control of Single-Input, Single-Output Linear Systems," IEEE Trans. on Automatic Control, Vol. AC-23, 1978, pp. 557-569.

[17] Narendra, K.S., Valavani, L.S., "Stable Adaptive Controller Design," Direct Control," IEEE Trans. on Automatic Control, Vol. 23, 1978, pp. 570-583.

[18] Koivo, A.J., Guo, T.H., "Adaptive Linear Controller for Robotic Manipulators," IEEE Trans. on Automatic Control, Vol. AC-28, February 1983, pp. 162-170.

[19] Whitney D.E., "Force Feedback Control of Manipulator Fine Motions," Joint Automatic Control Conference - 1976, pp. 687-698.

[20] Bejczy, A.K., "Issues in Advanced Automation for Manipulator Control," Joint Automatic Control Conference - 1976, pp. 700-711.

[21] Parr, R., Shimano, B., "Compliance and Control," Joint Automatic Control Conference - 1976, pp. 694-699.

[22] Bolles, R.C. and Cain, R.A., "Recognizing and Locating Partially Visible Objects," The International Journal of Robotics Research, Vol. 1, No. 3, Fall 1982, pp. 57-82.

[23] Cvetkovic, V. and Vukobratovic, M., "Development of an Optical Distance Sensor for Robots," The Int. J. of Robotics Research, Vol. 1, No. 4, Winter 1982, pp. 3-14.

[24] Pipitone, F.J., Marshall, T.G., "A Wide-Field Scanning Triangulation Rangefinder for Machine Vision," The Int. J. of Robotics Research, Vol. 2, No. 1, Spring 1983, pp. 39-49.

[25] Haraliek, R.M., Caffey, T.J. and Watson, L.T., "The Topographic Primal Sketch," The Int. J. of Robotics Research, Vol. 2, No. 1, Spring 1983, pp. 50-72.

[26] Binford, T.O., "Survey of Model-Based Image Analysis Systems," The Int. J. of Robotics Research, Vol. 1, No. 1, Spring 1982, pp. 18-64.

[27] Lavin, M.A., and Lieberman, L.I., "AML/U: An Industrial Machine Vision Programming System," The Int. J. of Robotics Research, Vol. 1, No. 3, Fall 1982, pp. 42-56.

[28] Buchner, A.J., Finger Movement in the Sagital Plane and the Mechanisms of Touch Control, M.S. Thesis, The Ohio State University, Columbus, Ohio, March 1983.

[29] Harmon, L.D., "Automated Tactile Sensing," Int. Journal of Robotics Research, Vol. 1, No. 2, Summer, 1982, pp. 3-32.

[30] Hillis, W.D., "A High-Resolution Imaging Touch Sensor," Int. Journal of Robotics Research, Vol. 1, No. 2, Summer 1982, pp. 33-44.

[31] Sandhu, Inc., Specifications Sheet for RHINO XR-1.

[32] Hemami, H., Robinson, C.S., Ceranowicz, A.Z., "Stability of Planar Biped Models by Simultaneous Pole Assignment and Decoupling," International Journal of Systems Science, 1980, Vol. 11, No. 1, pp. 65-75.

[33] Gonzalez, R.C., Safabakhsh, R., "Computer Vision Techniques for Industrial Applications and Robot Control," Computer, Vol. 15, No. 12, pp. 17-33.

[34] Fu, King-Sun, "Pattern Recognition for Automatic Visual Inspection," Computer, Vol. 15, No. 12, pp. 34-41.

[35] Salisbury, J.K., Craig, J.J., "Articulated Hands: Force Control and Kinematic Issues," Int. Journal of Robotics Research, Vol. 1, No. 1, Spring 1982, pp. 4-17.

[36] Johansson, R.S., and Vallbo, A.B., "Spatial Properties of the Population of Mechanoreceptive Units in the Glabrous Skin of the Human Hand," Brain Research, 1984 (1980), pp. 353-366.