*International Journal of Precision Engineering and Manufacturing Vol. 6, No. 2, April 2005.*

# Control Strategy to Reduce Tracking Error by Impulsive Torques at the Joint

Chulho Yang#

School of Mechanical Engineering, Andong National University, Andong, South Korea, 760-749

## ABSTRACT

The study reported deals with investigating the feasibility of control strategy for a serial rigid link manipulator that applies impulsive torques at the joints. The strategy is illustrated for a planar three rigid link manipulator. An impulse-based concept which uses successive torque impulses on rigid link as the controller for motion correction was introduced. This control strategy was tested over the entire trajectory to demonstrate that the tracking error could be reduced effectively. The best condition for minimizing the tracking error with the least impulse input at each joint is investigated by considering one design and one operating parameter. The first was the damping in the system, and the second was the sampling time during operation. The results show that this approach can provide useful guidance for the design and control of robot manipulators that require minimum impulse feedback for accurate tracking.

**Key Words** : control strategy, impulse-based feedback controller, robot manipulator, tracking error, torque

## 1. Introduction

In servo control, two fundamental problems are the point-to-point control problem and the tracking problem. The point-to-point control deals with movement of the control object from one point to another point. Therefore the transient path to the final point is not important in point-to-point control. If the plant outputs are to follow a class of desired trajectories, e.g., all polynomials up to certain order, the problem is referred as a servomechanism problem; if the desired trajectory is a particularly prescribed function of time, the problem is called a tracking problem. In tracking control, the control object must be moved along the desired trajectory.

This study is concerned with the control of tracking of the manipulator. In practice, the motions of all commercially available manipulators are preprogrammed. However, if the load is unknown or variable, the preprogrammed forces and torques will be incorrect and the manipulators will fail to follow the desired trajectory efficiently. The joints of the robot manipulator are so highly coupled that control of this kind of problem is a demanding task. So, it is essential to search for an adequate control strategy.

Many researches proposed control strategy such as computed torque method,[1,2,3] adaptive control,[4,5,6] sliding mode control,[7,8] command shaping method[9] and resolved-acceleration control.[10] The computed torque method has been shown to compensate for manipulator nonlinearities. The required input torque is computed as a function of a desired joint acceleration, velocities, and position and actual counterparts. However, the validity of this method is questionable when there are significant differences between the computed torque model parameters and the actual robot dynamic parameters.[11] This situation exists with any manipulator with significant unmodelled joint friction or unknown payload mass. Hypersensitivity to modeling error is a drawback of any feedforward controller.

Uebel et al.[12] investigated the improved computed torque method. The standard computed torque is not directly applicable to the manipulators in industry due to

Manuscript received: July 9, 2004
Accepted: December 17, 2004
# Corresponding Author: Email: cyang@andong.ac.kr
*Tel: +82-54-820-6159, Fax: +82-54-820-6159*

the dynamics introduced by the joint drive systems. So, they propose a new approach to overcome this problem by combining a globally computed torque algorithm with local torque controllers at each joint. The former provides the torque commands for all joints, while the latter regulates the actual torque output of the joint drive system. They concluded that the proposed controller improved tracking performance over a conventional PD controller.

In the control system design of robot manipulators, robustness considerations are important.[13, 14] In order to improve the robustness, a PID control system has been introduced. In the usual PID control system, high powered actuators are required so as to minimize the effect of robot dynamics. Recent equipment in PID control systems has become large, and the overall system is not flexible enough to permit changes of control strategy. Therefore, many control design methodologies have been proposed.

A manipulator's inertia is derived from the inertias of the links, which are highly coupled and non linear, and the inertias of the joints, which are linear and uncoupled. Because of the high gear ratios of the most industrial manipulators, the joint inertia tends to dominate the inertia terms seen by the manipulator's actuators, and as a result, most industrial robots are adequately controlled by PD controllers.

For small gear ratios, however, link inertia terms dominate, resulting in a high level of coupling between joints, and a manipulator inertia which will vary widely over its workspace. In this case, a linear, joint-space-based control, such as PD control, is inadequate. For instance, to ensure that the robot never overshoots its target, damping in PD controllers must be set sufficiently high to ensure the stability.

The objective of this study is to investigate the feasibility of a proposed impulse-based controller. The new control technique used in this study is to optimize the tracking performance by minimizing the errors with minimum joint impulse input. The optimized system parameters considered in this study are the sampling time and the damping coefficient.

This paper is constructed as follows. In the section 2, the dynamic model of manipulators and control algorithm proposed are summarized. Results of simulation in focus to the effects of design variable and operating variable are given in section 3. The summary and discussion of this study is given in section 4.

## 2. System Model

A crucial stage in the analysis, design, and simulation of a manipulator is to get an accurate mathematical representation of the system dynamics. In this section the dynamic equations are discussed which model a rigid-link serial manipulator having revolute joints. A complete description of this topic can be found in Thomas and Tesar. [15]

### 2.1 Inertia Driving Torque

The equations derived in this section depict the components of the dynamic model formulation which deal with the kinetic energy and inertial loads of the manipulator in motion. If a manipulator is in motion, forces have been applied to the manipulator in order to accelerate it from rest and overcome the system's inertia. This inertia can be treated as an opposing acceleration force which will be referred to as the inertial load. Here, the manipulator dynamics are analyzed by using Lagrange's equation of motion. By using this equation, the inertial torque acting at joint n can be expressed as

$$T_n^i = \frac{d}{dt}\left(\frac{\partial K.E.}{\partial \dot{\theta}_n}\right) - \frac{\partial K.E.}{\partial \theta_n} \tag{1}$$

where K.E. is the total kinetic energy of the system. To apply Lagrange's equation to the manipulator it is necessary to have an expression for the total kinetic energy of the manipulator. This is found in equation (2) by adding whole kinetic energies of the individual links:

$$K.E. = \sum_{j=1}^{n} \frac{1}{2}\left\{\left(M_{jk} V_{cj}^T V_{Cj} + W_{jk}^T \left[\Pi_{jk}\right] W_{jk}\right)\right\} \tag{2}$$

where $M_{jk}$ is the mass of the link **jk** , $V_{Cj}$ is the translational velocity of the mass centroid of link **jk** , and $W_{jk}$ is the angular velocity of link **jk** . The symbol $\left[\Pi_{jk}\right]$ represents the inertia tensor of link **jk** about its centroid and is given by using the fixed reference system. It changes with manipulator orientation. For the case of 2-D manipulator, the right-most term of equation reduces to

$$\omega_{jk}^{T}[\Pi_{jk}]\omega_{jk} = \omega_{z,jk}^{2} I_{zz,jk} \tag{3}$$

where $\omega_{z,jk}$ is the z-component and also only non-zero component of $\omega_{jk}$. The scalar $I_{zz,jk}$ is the inertia of link **jk** about a line parallel to the z-axis and passing through the link's centroid. Equation (2) can be rewritten in terms of first-order influence coefficients[15, 16] and the joint velocities

$$\begin{aligned} K.E. &= \frac{1}{2}\sum_{j=1}^{N}\{M_{jk}\dot{\Theta}^{T}[G_{cj}]^{T}[G_{cj}]\dot{\Theta} + \dot{\Theta}^{T}[G_{jk}]^{T}[\Pi_{jk}][G_{jk}]\dot{\Theta}\} \\ &= \frac{1}{2}\dot{\Theta}^{T}[I^{*}]\dot{\Theta} \end{aligned} \tag{4}$$

Equation (4) defines the equivalent inertia matrix $[I^{*}]$ which indicates the effective system mass as seen at the inputs. The scalar component in row m, column n of this matrix of inertia coefficients is given by the general equation.

$$[I^{*}]_{m,n} = \sum_{j=1}^{N}\{M_{jk}[G_{cj}]_{m}^{T}[G_{cj}]_{n} + [G_{jk}]_{m}^{T}[\Pi_{jk}][G_{jk}]_{n}\} \tag{5}$$

This expression shows that the equivalent inertia matrix depends only on the manipulator's mass properties and the corresponding first-order influence coefficients. From the form of equation (5) and since $[\Pi_{jk}]$ is symmetric for each link, $[I^{*}]$ is also symmetric matrix. The expression given in equation (4) for the total kinetic energy can be substituted into Lagrange's equation of motion to obtain the desired inertia torque $T_{n}^{i}$. Solving equation (4) step by step,

$$\frac{\partial K.E.}{\partial \dot{\Theta}_{n}} = \dot{\Theta}^{T}[I^{*}]_{n} \tag{6}$$

where $[I^{*}]_{n}$ is the n-$_{th}$ column of the inertia matrix $[I^{*}]$. Equation (4) takes on this simplified form only because of the symmetry of $[I^{*}]$. Equation (6) is differentiated with respect to time using the chain rule. The result is in equation (7).

$$\frac{d}{dt}\left(\frac{\partial K.E.}{\partial \dot{\Theta}_{n}}\right) = \ddot{\Theta}^{T}[I^{*}]_{n} + \dot{\Theta}^{T}\sum_{m=1}^{N}\{\frac{\partial}{\partial \Theta_{m}}([I^{*}]_{n})\dot{\Theta}_{m}\} \tag{7}$$

Defining a matrix $[B_{n}]$ such that the m-$_{th}$ column of $[B_{n}]$ is equal to the partial derivatives of $[I^{*}]_{n}$ with respect to $\Theta_{m}$, equation (7) can be rewritten as

$$\frac{d}{dt}\left(\frac{\partial K.E.}{\partial \dot{\Theta}_{n}}\right) = \ddot{\Theta}^{T}[I^{*}]_{n} + \dot{\Theta}^{T}[B_{n}]\dot{\Theta} \tag{8}$$

Equation (8) is substituted into Lagrange's equation to obtain following equation

$$T_{n}^{i} = \ddot{\Theta}^{T}[I^{*}]_{n} + \dot{\Theta}^{T}[B_{n}]\dot{\Theta} - \dot{\Theta}^{T}\{\frac{1}{2}\frac{\partial}{\partial \Theta_{n}}([I^{*}])\}\dot{\Theta} \tag{9}$$

or

$$T_{n}^{i} = \ddot{\Theta}^{T}[I^{*}]_{n} + \dot{\Theta}^{T}[P_{n}]\dot{\Theta} \tag{10}$$

where the inertia power matrix $[P_{n}]$ is defined by

$$[P_{n}] = [B_{n}] - \frac{1}{2}\frac{\partial}{\partial \Theta_{n}}([I^{*}]) \tag{11}$$

Equation (10) can be rewritten in an equivalent form as

$$T_{n}^{i} = \ddot{\Theta}^{T}[I^{*}]_{n} + \dot{\Theta}^{T}[P_{n}^{*}]\dot{\Theta} \tag{12}$$

This relationship is equivalent because the inertia power modeling matrix $[P_{n}^{*}]$ is defined to have the same symmetric part as the inertia power matrix $[P_{n}]$.
The l,m-$_{th}$ component of $[P_{n}^{*}]$ can be expressed in terms of the first and second-order influence coefficients [15, 16] as

$$\begin{aligned} [P_{n}^{*}]_{l,m} = \sum_{j=1}^{N}\{&M_{jk}[H_{cj}]_{l,m}^{T}[G_{cj}]_{n} + [H_{jk}]_{l,m}^{T}[\Pi_{jk}][G_{jk}]_{n} + [G_{jk}]_{l}^{T} \\ &[\Pi_{jk}]([G_{jk}]_{n} \times [G_{jk}]_{m})\} \end{aligned} \tag{13}$$

From equation (5) and (13) it is apparent that both $[I^{*}]$ and $[P_{n}^{*}]$ can be readily evaluated using the influence coefficients defined previously.[15, 16] These modeling matrices describe the equivalent inertia properties of the serial manipulator in a given configuration and depend only on the configuration of the manipulator, $[I^{*}]$ and $[P_{n}^{*}]$ depend only on the position, geometry, and mass distribution of the manipulator's links. Thus, the expression given for $T_{n}^{i}$ in equation (10) clearly separates the controllable input dynamics, $\dot{\Theta}_{n}$ and $\ddot{\Theta}_{n}$, from configuration dependent terms or system constants.

## 2.2 Equivalent Input Torques Due to Applied Loads and Gravity

The end-effector of a manipulator usually experiences some general loading during the execution of a task including as external loads and gravity forces. It is necessary then to find the effective torque acting at each actuator joint by using the various applied external loads. Eq. (14) describes effective torque, acting at each joint n:

$$\mathbf{T}_n^L = \sum_{p=1}^{p} \mathbf{f}_p^T \{\mathbf{G}_p\}_n + \sum_{j=1}^{\grave{} } \mathbf{m}_{jk}^T [\mathbf{G}_{jk}]_n \qquad (14)$$

where $\mathbf{f}_p^T$ represents an external force acting at point p and $\mathbf{m}_{jk}$ represents an external torque acting on the link **jk** . As before, $[\mathbf{G}_p]_n$ and $[\mathbf{G}_{jk}]_n$ are the n-$_{th}$ columns of the first-order influence coefficient matrices. [15]

## 2.3 Controlling Equation of Motion

The dynamic model formulation used in this study enables all system forces and torques to reduce as equivalent effects at the generalized inputs. The controlling dynamic equations of motion can be expressed as a torque balance at each input because all load and motion characteristics are represented at the inputs. In general, there are four torque terms to be considered at input location n:

1) Equivalent torque due to applied loads; $\mathbf{T}_n^L$

2) Equivalent torque due to other system components such as springs and dash pots (viscous damping); $\mathbf{T}_n^s$

3) Torque required to accelerate the system inertia; $\mathbf{T}_n^i$

4) Driving torque supplied by the actuator; $\mathbf{T}_n^A$

Eq. (15) describes the generalized inertia torque which should be supplied by all of the torque-generating system components.

$$\mathbf{T}_n^i = \mathbf{T}_n^A + \mathbf{T}_n^L + \mathbf{T}_n^s \qquad (15)$$

Rearranging this equation, the dynamic analysis form gives the torque which the actuator must provide to achieve a prescribed manipulator motions as

$$\mathbf{T}_n^A = \mathbf{T}_n^i - \mathbf{T}_n^L - \mathbf{T}_n^s \qquad (16)$$

The inertia torque term has two distinct components. These constitute an acceleration-related term, $\mathbf{T}^{ia}$ , and a velocity related term $\mathbf{T}^{iv}$ . Considering the equations of motion for all N inputs simultaneously, it is possible to rewrite equation (15) as a vector equation

$$\mathbf{T}^{ia} + \mathbf{T}^{iv} = \mathbf{T}_n^A + \mathbf{T}_n^L + \mathbf{T}_n^s \qquad (17)$$

where each torque vector is composed of the scalar torque values for all N generalized inputs. The generalized inertia torque has been decomposed into two distinct vectors in this equation. Equation (17) can be rearranged to give the following equation.

$$\mathbf{T}^{ia} = [\mathbf{I}^*]\ddot{\Theta} = \mathbf{T}_n^A + \mathbf{T}_n^L + \mathbf{T}_n^s - \mathbf{T}^{iv} \qquad (18)$$

For non-singular $[\mathbf{I}^*]$ , we could find a joint angular acceleration vector by changing equation (18) into a inverse configuration. Consequently, the joint angular acceleration can be expressed as:

$$\ddot{\Theta} = [\mathbf{I}^*]^{-1}(\mathbf{T}_n^A + \mathbf{T}_n^L + \mathbf{T}_n^s - \mathbf{T}^{iv}) \qquad (19)$$

This equation provides an explicit means of computing the joint accelerations from knowledge of applied forces, actuator torques, joint position, and joint velocities. This equation will be integrated via Runge-Kutta integration scheme to obtain joint position. The equations given in this section provide an accurate description of the dynamics of an ideal N-degree of freedom serial manipulator having revolute joints. In this study the frictional torques was omitted in order to simplify the system analysis.

## 2.4 Computational Algorithm

This section deals with the control algorithm for the manipulator, and an impulse-based control strategy is proposed. The basic idea is as following. First, the impulse at each instant of calculation of torque can be set as in the following equation:

$$\mathbf{impulse} = \mathbf{I} \times \Delta\dot{\Theta} \qquad (20)$$

where I is the moment of inertia of rotor, and $\Delta\dot{\Theta}$ is the joint angular velocity error between the desired joint velocity and actual joint velocity. This impulse is used as an error correction tool. Generally the way to build a high-performance control system is to use of feedback

from the joint sensors. Typically this feedback is used to compute the servo error by finding the difference between the desired position and the actual position, and likewise between desired and actual velocities.

$$\mathbf{E} = \Theta_d - \Theta_a \tag{21}$$

$$\dot{\mathbf{E}} = \dot{\Theta}_d - \dot{\Theta}_a \tag{22}$$

Figure 1 shows the control flow of our scheme. The design of controllers for the gross motion of the manipulators may be considered in two stages: (1) to design a primary controller that under ideal condition makes the end-effector track the desired trajectory. (2) based on impulse concept to provide feedback compensation. From the trajectory generation procedure, position, velocity and acceleration of the end-effector can be calculated, and then from a geometric approach to find a manipulator's solution, the joint angles $\Theta$ can be solved.[17] The joint angular velocity then can be obtained by introducing the jacobian which relates the joint velocity to the Cartesian velocity of the tip of the arm. The joint angular acceleration can be obtained by differentiation the joint angular velocity with respect to time. For example:

$$\mathbf{V} = \mathbf{J}(\Theta)\dot{\Theta} \tag{23}$$

where $\Theta$ is the vector of joint angles of the manipulator, and V is a vector of Cartesian velocities.

Generally, the inverse of the jacobian matrix can be found by changing equation (23) into an inverse configuration. Consequently, the joint angular velocities can be expressed as:

$$\dot{\Theta} = \mathbf{J}^{-1}(\Theta)\mathbf{V} \tag{24}$$

To obtain the joint angular accelerations, equation (24) is differentiated with respect to time. As a result of differentiation, we obtain the following equation.

$$\ddot{\Theta} = (\mathbf{J}^{-1}(\Theta))'\mathbf{V} + \mathbf{J}^{-1}(\Theta)\mathbf{V}' \tag{25}$$

With equations (23), (24) and (25), the desired joint angle, desired joint angular velocity, and desired joint angular acceleration are finally obtained. Combining these equations leads to the desired actuator torque of the manipulator. By integrating the dynamic equations of the manipulator, the actual joint angle and joint angular velocity of the manipulator are obtained. The velocity error signal is then utilized as a feedback compensation tool. The control system can compute how much impulse to send to the actuators as a function of the servo rate error. These procedures repeat to solve the equation (19) with a newly computed angular joint angle, a newly computed joint angular velocity, and a newly computed torque during every step.

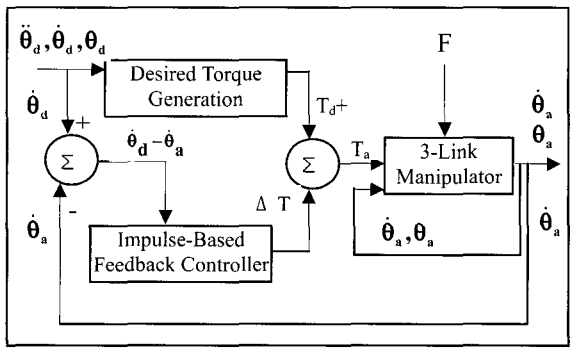


Fig. 1 Block diagram of control strategy

## 3. Simulation

As a simulation model, a 3-DOF planar rigid manipulator was used to test this control algorithm. Simulation parameters (masses, inertias, lengths and dimensions) used in the simulation model are given in table 1. The end-effector moved from point (x = 1.0, y = -0.35, $\Theta$ = 0°) to point (x = 0.3, y = 0.6, $\Theta$ = 90°). The total simulation time was 2 seconds. The simulation scheme was tested by using three sampling times (0.005 sec, 0.01 sec, 0.02 sec). As an additional device, a damper was attached at each joint to improve the tracking performance.

Evaluation of the robot arm's mechanical performance has largely been made on specifications such as payload capacity, accuracy, repeatability, and geometrical configuration. A more effective means of evaluating a robot's performance, from the accuracy view point, is to compute the joint torque needed for error compensation.

In this study the performance of the manipulator, which is the end-effector's error, was optimized by using

a parametric search. A parametric search was undertaken to evaluate the effect of different design and operating variables on the system performance and stability. To demonstrate the effects of different design variables, various damping coefficients were used. Different sampling time was considered as an operating variable. As a result, a useful guidance for the design of manipulators which minimize the torque necessary was provided for controlling the accuracy of the end-effector movement. As a tool of reducing the rippling of the end effector's tip and improving the performance, a damper was attached to the each joint of the manipulator.

Table 1 Manipulator parameters used for simulation

| Link | Length (m) | Centroid location (m) | Mass(kg) | Inertia (kg· $m^2$) |
|---|---|---|---|---|
| 1 | 0.6 | 0.25 | 4.5 | 1.0 |
| 2 | 0.5 | 0.23 | 3.8 | 0.6 |
| 3 | 0.15 | 0.1 | 3.2 | 0.1 |

"Contours of performance" allow synthesis for upper bound error zone for different condition. They can be extended to predict the possible optimum conditions of the manipulators. To visualize the optimum condition, the contours of impulse summation value at each joint were plotted. The impulses at each joint were accumulated during the simulation with various conditions, such as damping coefficient value, $\mathbf{K}_i$ (where, $\mathbf{K}_i = \mathbf{C}_i / \mathbf{I}_i$, $\mathbf{C}_i$ : damping value, $\mathbf{I}_i$ : moments of inertia of link) and sampling time to plot the sampling time versus $\mathbf{K}_i$ for the purpose of finding optimum condition of this total impulse. The evaluation of a robot's performance using contour plot can provide the groundwork for the design of better system.

### 3.1 Effects of Damping

In this study, the effect of damping of manipulator was investigated by considering different values of damping in the simulation. In cases 1, each link had a different joint damping coefficient value $\mathbf{K}_i$ while, in cases 2, each link had the same damping value $\mathbf{C}_i$.

In every case, the actual trajectory of 1st and 2nd joints showed a good accuracy, but the 3rd joint had a severe rippling that deteriorated the system's performance with inappropriate parameters.

For the case of different damping coefficient value $\mathbf{K}_i$ (where, $\mathbf{K}_i = \mathbf{C}_i / \mathbf{I}_i$) for each joint, the rippling of the 3rd joint angular position is occurred (see, fig. 2). If the damping value of the 3rd joint's is then increased to 3.9, no ripplings occur over the entire trajectory as a result of the increased damping coefficient (see, fig. 3).

For the case of the same damping value $\mathbf{C}_i$ for the entire system, the effects of damper were revealed (see, fig. 4, fig. 5) by comparing the trajectory of x-direction of the end-effector. Due to the damper, position of the end-effector showed less rippling than one had no damping. As a result, the better tracking performance, less error of the end effector, and less rippling at the end of the execution were observed. The increase of the coefficient of the damping suppressed the error oscillation of the system.

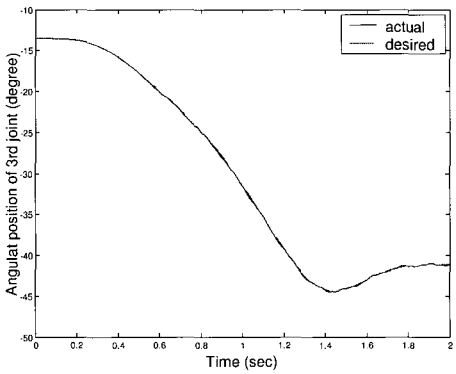


Fig. 2 3rd joint angular position with payload and no damping $K_i=(C_i/I_i)=0$, Sampling time: 0.005 (sec), Payload: 20 (N)

### 3.2 Effects of Sampling time

An important design consideration is to keep the robot manipulator's position accuracy within the error limit during the operation. Accuracy can be influenced by design and kinematics of the linkages, as well as by its payload, and types of drives. If the end point is oscillating at the termination of motion, it will seriously damage the entire performance of the system.

In this simulation, the response of the proposed control schemes to the payload and the effects of sampling time on the tracking performance of manipulator were of interest.

To check the influence of the sampling time of the simulation on the tracking performance, three kinds of

the models which have different sampling time were demonstrated. For achieving the convergence of the control algorithm, the computation must be repeated very frequently. In most robot manipulator the frequency ranges above 60 Hz[18] since the resonant frequency of most of the mechanical manipulators is around 10 Hz. Consequently, computations below 60 Hz can cause an instability state.

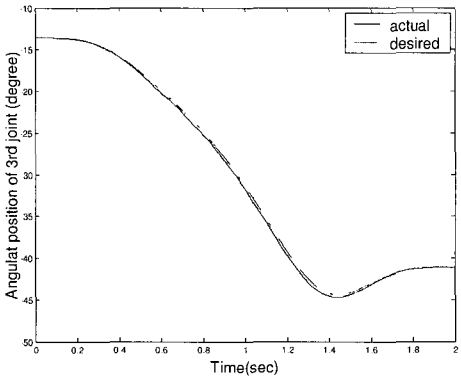


Fig. 3 3rd joint angular position with payload and damping $K_i=(C_i/I_i)=3.9$, Sampling time: 0.005 (sec), Payload: 20 (N)

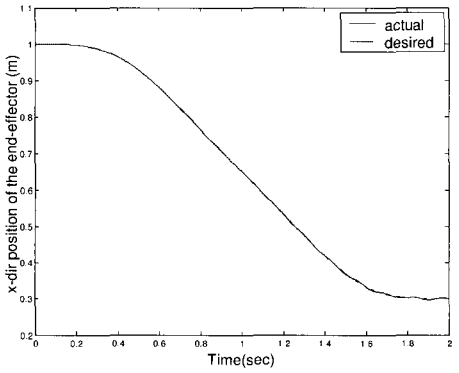


Fig. 4 X-dir position of the end-effector with payload and no damping $C_i=0$, Sampling time: 0.005 (sec), Payload: 20 (N)

In our simulation, following sampling times were used: 0.02 seconds (=50 Hz), 0.01 seconds (=100 Hz), and 0.005 seconds (=200 Hz). A step disturbance was applied after 0.5 seconds. The response of the manipulator was checked when this disturbance was applied.

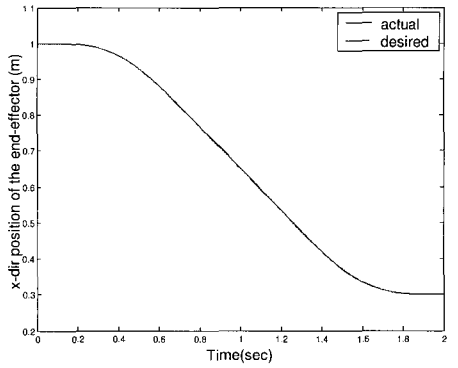


Fig. 5 X-dir position of the end-effector with payload and no damping $C_i=1.9$, Sampling time: 0.005 (sec), Payload: 20 (N)

After the step disturbance was applied to the manipulator at 0.5 seconds, profiles of the x direction error, y direction error, and orientation error for the payload state and the no payload state were obtained (Figures 6-8). The error fluctuated in spite of the damping. It is natural for the no payload case to make the stable system. Even though the errors in payload case fluctuated, our control scheme showed the tendency of reducing errors.

Next, the sampling time effects on the manipulator were investigated by changing sampling time. An illustration of investigating the effect of the sampling time with low damping coefficient ( $\mathbf{K}_i = 0.1$) on the tracking accuracy is shown in Figs. 9, and 10. As the sampling frequency increased the tracking performance improved considerably.

Fig. 11 and Fig. 12 describe the x-direction position of the end-effector with high damping ( $\mathbf{K}_i = 1.9$). As discussed in section 2, the trajectory of the cases with high damping showed improved performance with no ripplings over the path.

As discussed at the beginning of this section, usually, low sampling frequency deteriorates the system's performance. With increased sampling frequency, the trajectory errors of the end-effector diminish.

### 3.3 Optimization

The final objective of this paper is to minimize trajectory error with minimum total joint impulse.

Fig. 13 shows the contour plot of the absolute

impulse summation of the first joint which was used in the control implementation. Fig. 14 and Fig. 15 depict the impulse summation need to compensate for the error in the 2nd joint of the manipulator and for the error in the 3rd joint of the manipulator respectively. The total impulses for 60 different cases were compared. Simulations were performed with selected 20 different $\mathbf{K}_i$ values and repeated with three different sampling times (0.02, 0.01, and 0.005 sec). Table 2 summarizes results of the 4 cases with the highest sampling time.

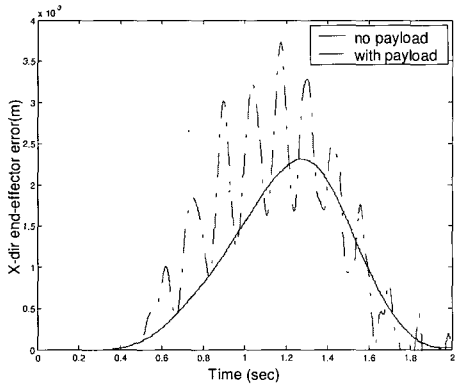


Fig. 6 X-dir end-effector error compensation between payload and payload $K_i=(C_i/I_i)=1.9$, Sampling time: 0.005 (sec)

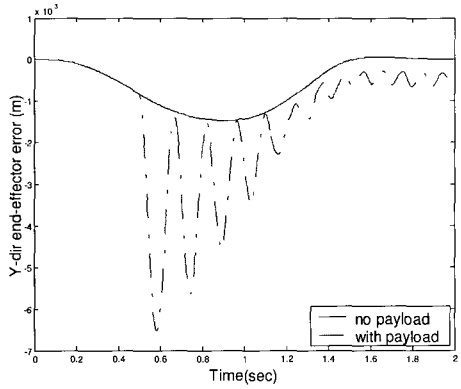


Fig. 7 Y-dir end-effector error compensation between payload and payload $K_i=(C_i/I_i)=1.9$, Sampling time: 0.005 (sec)

It can be seen that cases 3, 4 give the good tracking performance with the minimum total impulse. Using this impulse controller, the best accuracy can be obtained with the least impulse summation. This fact gives the desirable circumstances to the control system. With the aid of this procedure the optimum damping and sampling time can be determined for the manipulator used in the considered illustrative example as can be seen in table 2. The total absolute impulse was found to be equal to 1844.942( $\mathbf{kg \cdot m^2/sec}$ ), and the ratios of the control impulse to the total impulse are also listed in table 2.

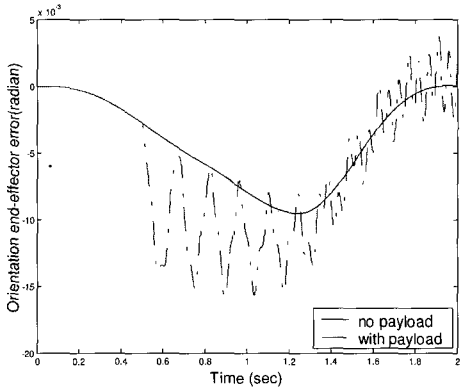


Fig. 8 Orientation end-effector error compensation between payload and payload $K_i=(C_i/I_i)=1.9$, Sampling time: 0.005 (sec)

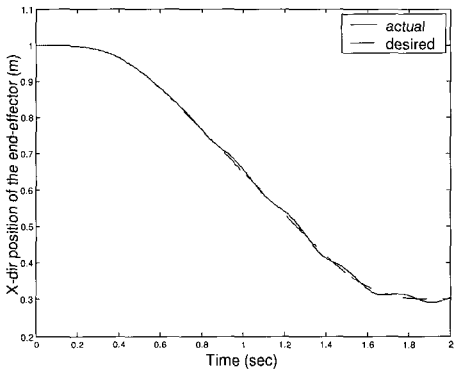


Fig. 9 X-dir position of the end-effector with damping and payload $K_i=(C_i/I_i)=0.1$, Sampling time: 0.02 (sec), Payload: 20(N)

## 4. Conclusions

The study reported in this paper investigated a control strategy using impulse as the error minimization tool. To show the effect of system variables such as damping and sampling rate, illustrative cases are considered.

From the result of the simulation, the concept of the

impulse correction can be successfully applied to the serial rigid link manipulators as a controller. The sampling time of the system can significantly influence the system's tracking performance.

Appropriate selection of the sampling time is important. The fastest sampling time in the simulation (0.005 seconds, 200 Hz) showed the best tracking performance. Introducing damping in the system affected the system's rippling response and stability. As would be expected, increasing the damping value reduced the rippling effect in the system response. Contour plot of the total impulse and end-effector errors can be used to determine the optimum condition of the manipulator's response. They can determine damping and sampling rate necessary to minimize the tracking error with minimum total joint impulse.

In order to implement this approach in practice, the motor voltage is to be modified as a control input rather than the actuator torque. This requires incorporating the dynamic model of the drive systems in the total system model. Although the proposed control strategy was applied to the rigid body motion, this concept should extend to the flexible joint problem which is generally the case in practical manipulators.

Table 2 Absolute final peak error in x, y, orientation direction, total correction impulse and ratio to the total impulse

| | X-dir (m) | Y-dir (m) | Angle (radian) | Total correction impulse | Ratio to total impulse |
|---|---|---|---|---|---|
| 1* | 0.00525 | 0.00030 | 0.01297 | 133 | 0.072 |
| 2* | 0.00468 | 0.00029 | 0.01173 | 128 | 0.069 |
| 3* | 0.00192 | 0.00027 | 0.00633 | 95 | 0.052 |
| 4* | 0.00088 | 0.00025 | 0.00508 | 85 | 0.046 |

1* (case1) : $\mathbf{K}_i = 0$ with 0.005 sec sampling time.
2* (case2) : $\mathbf{K}_i = 0.1$ with 0.005 sec sampling time.
3* (case3) : $\mathbf{K}_i = 0.9$ with 0.005 sec sampling time.
4* (case4) : $\mathbf{K}_i = 1.9$ with 0.005 sec sampling time.

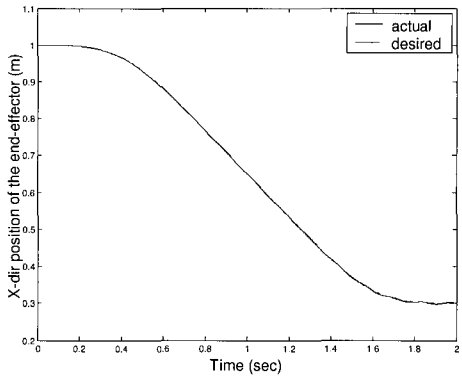


Fig. 10 X-dir position of the end-effector with damping and payload $K_i=(C_i/I_i)=0.1$, Sampling time: 0.005 (sec), Payload: 20(N)

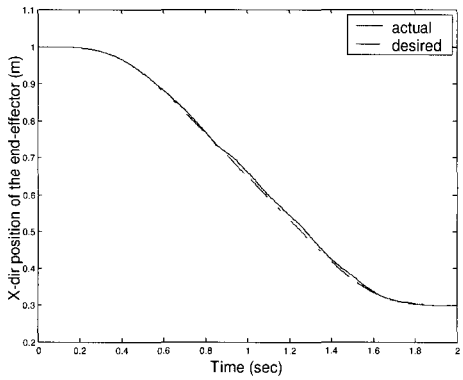


Fig. 11 X-dir position of the end-effector with damping and payload $K_i=(C_i/I_i)=1.9$, Sampling time: 0.02 (sec), Payload: 20(N)

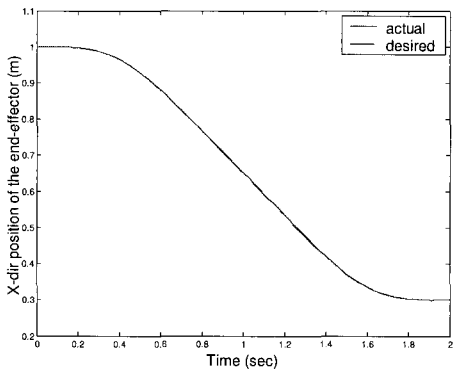


Fig. 12 X-dir position of the end-effector with damping and payload $K_i=(C_i/I_i)=1.9$, Sampling time: 0.005 (sec), Payload: 20(N)

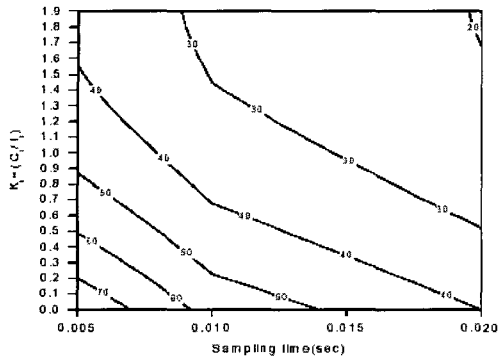


Fig. 13 Contour plot of the absolute sum at the 1st joint

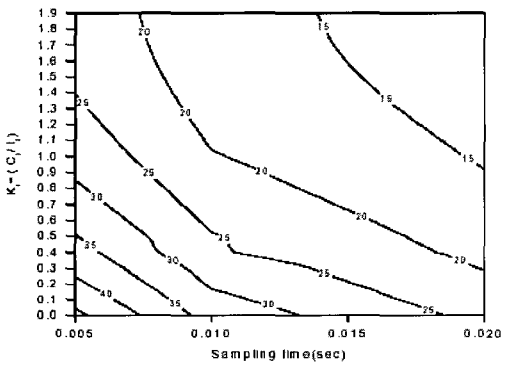


Fig. 14 Contour plot of the absolute sum at the 2nd joint

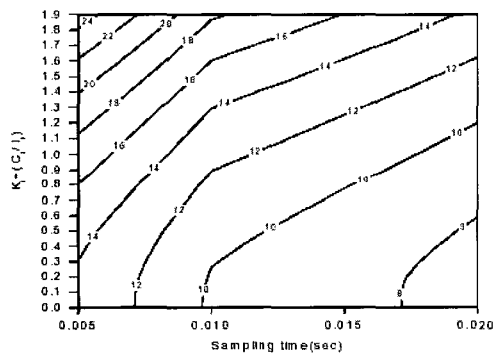


Fig. 15 Contour plot of the absolute sum at the 3rd joint

## Acknowledgement

In memory of Dr. A. Seireg, a true scholar. I deeply appreciate his passion for research and teaching at UF.

## References

1. Markiewicz, B. R., "Analysis of the Computer Torque Drive Method and Comparison with Conventional Position Servo for a Computer Controlled Manipulator," Tech. Memorandum 33-601, Jet Propulsion Lab., Pasadena, CA. 1973.
2. Bejczy, A. K., "Robot Arm Dynamics and Control," Tech. Memorandum, Jet Propulsion Lab., Pasadena, CA. 1974.
3. Egeland, O., "On the Robustness of Computed Torque Method in Manipulator Control," Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1203-1208, 1986.
4. Slotine, J-J. and Li, W., "On the Adaptive Control of Robot Manipulators," International Journal of Robotics Research, Vol. 6, No. 3, pp. 49-59, 1987.
5. Middleton, R. and Goodwin, G., "Adaptive Computed Torque Control for Rigid Link Manipulators," Systems and Control Letters, Vol. 10, No. 1, pp. 9-16, 1988.
6. Koivo, A. and Guo, T., "Adaptive Linear Controller for Robotic Manipulators," IEEE Trans. on Automatic Control, Vol. 28, No. 2, pp.162-171, 1983.
7. Slotine, J. J. and Sastry, S. S., "Tracking Control of Non-Linear Systems Using Sliding Surfaces, with Application to Robot Manipulators," Int. J. Control, Vol. 38, No. 2, pp.465-492, 1983.
8. Chen, Y. F., Mita, T. and Wakui, S., "A New and Simple Algorithm for Sliding Mode Trajectory Control of the Robot arm," IEEE Trans. on Automatic Control, Vol. 33, No.7, July 1990.
9. Singer, N. C. and Seering, W. P., "Preshaping Command Inputs to Reduce System Vibration," ASME J. of Dynamic Systems, Measurement, and Control, pp. 76-82, Mar. 1990.
10. Luh, J. Y. S., Walker, M. W. and Paul, R. P. C., "Resolved Acceleration Control of Mechanical Manipulators," IEEE Trans. on Automatic Control,"pp. 468-474, July, 1980.
11. Gilbert, E. G.. and Ha, I. J., "An Approach to Nonlinear Feedback Control with Application to Robotics," Proc. 22nd IEEE Decision and Control Conference, San Antonio, TX., pp. 14-16, Dec. 1983.
12. Uebel, M. A., Minis, I. and Cleary, K., "Improved

Computed Torque Control for Industrial Robots," Proc. IEEE Int. Conf. on Robotics and Automation, Nice, France, pp. 528-533, May, 1992.

13. Furuta, K., Mayeda H., "Robust Control of a Robot Manipulator with Nonlinearity," Robotica, Vol. 2, pp. 75-81, 1984.
14. Seraji, H., "An approach to Multivariable Control of Manipulators," Trans. ASME, Vol. 109, pp. 146-154, 1987.
15. Thomas, M. and Tesar, D., "Dynamic Modeling of Serial Manipulator Arms," ASME J. of Dynamic Systems, Measurement, and Control, Vol. 104, pp. 218-228, Sep. 1982.
16. Benedict, C. E. and Tesar, D., "Dynamic Response Analysis of Quasi-Rigid Mechanical Systems Using Kinematic Influence Coefficients," Journal of Mechanisms, Vol. 6, pp. 383, 1971.
17. Craig, J. J., "Introduction to Robotics Mechanics and Control," Addison-Wesley, Reading, Massachusetts, 1989.
18. Luh, J. Y. S., Walker, M. W. and Paul, R. P. C., "On-Line Computational Scheme for Mechanical Manipulators," ASME J. of Dynamic Systems, Measurement, and Control, Vol. 102, pp.69-76, June, 1980.