SUBOPTIMAL CONTROL OF INDUSTRIAL MANIPULATORS WITH A WEIGHTED MINIMUM TIME-FUEL CRITERION[1]

Byung Kook Kim and Kang G. Shin

Department of Electrical and Computer Engineering
The University of Michigan
Ann Arbor, Michigan 48109

ABSTRACT

Even if a manipulator does not have to follow a prespecified path (i.e. a geometric path and a velocity schedule), due to the complexity and nonlinearity of the manipulator dynamics, control of manipulators has been conventionally divided into two subproblems, namely path planning and path tracking, which are then separately and independently solved. This may result in mathematically tractable solutions but can not offer a solution that utilizes manipulators' maximum capabilities (e.g. operating them at their maximum speed).

To combat this problem, we have developed a suboptimal method for controlling manipulators that provides improved performance in both their operating speed and use of energy. The nonlinearity and the joint couplings in the manipulator dynamics--a major hurdle in the design of robot control--are handled by a new concept of averaging the dynamics at each sampling interval. With the averaged dynamics, we have derived a feedback controller which (i) has a simple structure allowing for on-line implementation with inexpensive microprocessors, and (ii) offers a near minimum time-fuel(NMTF) solution, thus enabling manipulators to perform nearly up to their maximum capacity and efficiency.

As a demonstrative example, we have applied the method to control the Unimation PUMA 600 series manipulator and simulated its performance on a DEC VAX-11/780. The simulation results agree with the expected high performance nature of the control method.

1. INTRODUCTION

Recently, robotics has emerged as a hot field in engineering mainly because of its potential for improving both manufacturing productivity and working environment. Industrial manipulators are computer-controlled mechanical devices and are the primary component in contemporary automation systems. It is therefore essential to design optimal manipulator systems with a suitable performance criterion which is consistent with the foregoing goal.

The performance of manipulators can be bettered by improving their mechanical construction and/or by using more effective controllers. In this paper we are only concerned with the latter. Although manipulator control problems can in general be classified into four different types depending upon if (i) they have to follow a prespecified path and/or if (ii) they operate in a collision-free workspace (see Luh[10] for detail), there are many applications which do not require robotic manipulators to strictly follow a prescribed path and, also, collision with obstacles can be avoided by specifying a few appropriate intermediate points in the workspace for the manipulator to pass through[11]. Consequently, in such a case the manipulator control problem can be converted to a more general form in which the manipulator is given freedom to move along any path between two given intermediate or end points.

[1] *The work reported here is supported in part by the Postdoctoral Program of the Korea Science and Engineering Foundation, Republic of Korea, the US AFOSR Contract No. F49620-82-C-0089 and Center for Robotics and Integrated Manufacturing, The University of Michigan, Ann Arbor, Michigan. All correspondence should be addressed to Prof. Kang G. Shin.*

In view of the preceding fact, for most cases manipulators are desired to move from one point to another as fast as possible. Consequently, it is important to design an efficient controller which requires less time and energy, thus pushing the manipulators to be operated at near maximum capacity. This consideration naturally leads to an optimal control problem of robotic manipulators with a minimum time-fuel criterion. However, it is in general very difficult to obtain an exact closed-form optimal solution to the problem since the dynamics of manipulators are highly nonlinear, coupled functions of their positions and velocities, and also of their payloads. There are two alternative approaches conceivable for this problem:

(1) off-line minimum time path planning followed by on-line path tracking, and
(2) derivation of a suboptimal controller with realistic approximations of the manipulator dynamics.

In the first approach, it is assumed that there are known optimal path planners which minimize the total traveling time for a given sequence of specified positions (describing the desired path) in joint coordinates [2], or in Cartesian coordinates [3]. And then, one can use one of many well-known, on-line path tracking algorithms [4], [5], [9] which force the manipulator to follow the path with the prescheduled velocities. There are path tracking algorithms which take the manipulator dynamics into consideration, but there are no known methods for path planning which include the manipulator dynamics. This implies that path planning has to be made with a global least upper bound for manipulators' capabilities. Note that the capability of a manipulator changes with its position, payload, etc. For example, an optimal path for a manipulator has to be generated on the basis of the maximum speed allowed under the worst (global) condition, since it may otherwise not be able to follow the prespecified path with the prescheduled velocities. It means that the full capacity of the manipulator cannot be utilized if this approach is used.

The second approach can be adopted to nearly fully utilize the capacity of individual manipulators. Only a few attempts have been made in this context due mainly to the difficulty in obtaining amenable solutions. Kahn and Roth[1] linearized the manipulator dynamics at the final target point, and used the decoupled dynamic model to derive the minimum-time controller. This method suffers from the fact that the linearized dynamic equations would not be valid if the manipulator is not located in the vicinity of the final target. Hence this method is acceptable only when manipulator motion is confined to a small region in the neighborhood of the target point. Lynch[6] developed a minimum-time controller for sequential axis operation. Only one axis was moved at one time, which simplifies the dynamic equations of the manipulator (e.g. linear time-invariant equations for a 2-axis cylindrical manipulator) and hence the controller. However, the controller requires much more time (approximately n times more for an n jointed manipulator) than the minimum-time controller allowing simultaneous operation of all axes.

Also some results were derived on the basis of minimum energy control with a fixed terminal time [7], [8], in which they used over-simplified dynamic models without considering manipulator's operating speed.

The second approach with a suitable suboptimal controller, if possible, is highly desirable due to its capability of almost full utilization of the manipulator capacity. In this paper, we have adopted the second approach and developed a suboptimal *feedback controller* for industrial manipulators with the weighted minimum time-fuel(MTF) criterion. The choice of this criterion is

0191-2216/83/0000-1199 $1.00 © 1983 IEEE

justified by its direct link to the goal of improving productivity and saving energy. The near minimum time-fuel(NMTF) controller is derived in a *feedback form* with a judicious approximation in the calculation of the manipulator dynamics. Although the optimal controller is developed for continuous time domain, the manipulator dynamics are updated at each of discrete sampling instants. The approximation is based on the fact that any manipulator control algorithm has to be implemented on a digital computer in discrete form. Namely, to cope with the nonlinearity and joint couplings in the manipulator dynamics, the model parameters of the manipulator are updated continually at each sampling interval on the basis of the feedback information of positions and velocities. Then, an *averaged dynamics* concept -- which utilizes all available dynamic information of the current and the final states to update the dynamics continually -- is newly introduced to design the proposed NMTF controller. Since the approximation error at a sampling instant is compensated at the next iteration through feedback, this approximation can effectively handle the nonlinearity and joint couplings in the manipulator dynamics. Note that this approximation is extremely simple and thus suitable for real-time implementation with inexpensive microprocessors as we shall see.

The main contribution of this paper lies in that (a) the complexity of the manipulator dynamics--which has been a major hurdle in the design of robot control--is handled effectively by continuous update of the dynamics with the averaging method, (b) its structural simplicity allows for on-line implementation with microprocessors, (c) it has high potential for improving the manipulator performance (e.g. operating speed), and (d) it results in a closed-loop feedback controller, whereas most existing ones for manipulators are open-loop MTF controllers.

This paper is organized as follows: In Section 2 the algorithm for suboptimal control of manipulators with the MTF criterion is derived, and the method of updating the manipulator dynamics with the averaging concept is presented. Also considered is the synchronization of each joint controller for simultaneous convergence of all manipulator joints to a target point. Section 3 applies the NMTF controller to the Unimation PUMA 600 series manipulator and simulates its performance on a DEC VAX-11/780, and then conclusions follow in Section 4.

2. The Near Minimum Time-Fuel Controller

2.1. Derivation of the Weighted Near Minimum Time-Fuel Controller

Using the Lagrangian mechanics, one can derive explicit manipulator dynamic equations which represent generalized forces/torques in terms of the joint positions, velocities, and accelerations:

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \mathbf{u} \tag{1}$$

where $\mathbf{u}$ is an nx1 generalized force/torque vector, $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$ are vectors of generalized coordinates, velocities, and accelerations, respectively. $\mathbf{D}(\mathbf{q})$ is an nxn inertia matrix, $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})$ is an nx1 Coriolis and centrifugal force vector, $\mathbf{g}(\mathbf{q})$ is an nx1 gravitational loading vector, and n is the number of joints in the manipulator. The inertia, the gravity loading, and the Coriolis and centrifugal terms depend on the position of each joint as well as on the mass, first moment, and inertia of each link. These terms are also functions of manipulator's payload (i.e. tool and parts). The dynamic equations (1) can be converted into a state-variable representation with a 2n-dimensional state vector $\mathbf{y} = [\mathbf{y}_p^T\ \ \mathbf{y}_v^T]^T = [\mathbf{q}^T\ \dot{\mathbf{q}}^T]^T$ (T denotes here 'transpose') as follows:

$$\dot{\mathbf{y}} = \mathbf{a}(\mathbf{y}) + \mathbf{B}(\mathbf{y})\mathbf{u} \tag{2}$$

where

$$\mathbf{a}(\mathbf{y}) = \begin{bmatrix} \mathbf{y}_p \\ -\mathbf{D}^{-1}(\mathbf{y}_p)[\mathbf{g}(\mathbf{y}_p) + \mathbf{h}(\mathbf{y}_p, \mathbf{y}_v)] \end{bmatrix}$$

$$\mathbf{B}(\mathbf{y}) = \begin{bmatrix} 0 \\ \mathbf{D}^{-1}(\mathbf{y}_p) \end{bmatrix}$$

Each joint of the manipulator is separately driven by an electric motor or by a hydraulic motor, and naturally there exist certain limits in the magnitudes of driving forces or torques. Hence constraints on the input vector $\mathbf{u}$ can be represented in general:

$$\mathbf{u}^- \le \mathbf{u} \le \mathbf{u}^+ \tag{3}$$

where $\mathbf{u}^-$, $\mathbf{u}^+$ are nx1 vectors representing the minimum and the maximum values of input force/torque, respectively.

The weighted time-fuel optimal control problem can be stated as follows:

Problem I. Find control $\mathbf{u}^*(t)$, $t_0 \le t \le t_f$, such that the system given by Eq. (2) is steered to a given target point

$$\mathbf{y}(t_f) = \mathbf{y}_f \tag{4}$$

from a given initial point

$$\mathbf{y}(t_0) = \mathbf{y}_0 \tag{5}$$

while minimizing the performance index

$$J(\mathbf{u}) = \int_{t_0}^{t_f} [\lambda + \sum_{i=1}^{n} |u_i|]dt \tag{6}$$

subject to the input constraint (3).

Note that this is an open terminal-time problem, and the parameter λ is introduced to set a relative weight between time and fuel. When the value of λ approaches zero, this becomes the fuel-optimal control problem, and when the value of λ becomes infinity, this approaches the time-optimal control problem.

Using Pontryagin's maximum principle[12], we can derive necessary conditions for the optimal solution, which results in a 4n-dimensional differential equation with the optimal state and costate vectors and the boundary conditions described by Eqs. (4) and (5), which result in a two-point boundary value problem; this is in general very difficult, if not impossible, to solve. Only numerical solutions for all but extremely simple cases may be obtained due to the nonlinearity and inertial couplings in the manipulator dynamics.

In order to overcome this difficulty, we consider first optimal control of an approximated dynamic system for each individual axis and then update the concerned dynamics with feedback information to compensate for the errors induced by the approximation as well as for the nonlinearity in the dynamics. The system dynamic model, i.e. Eq.(1), can be rewritten:

$$\ddot{q}_i = \sum_{j=1}^{n} d'_{ij}(\mathbf{q})u_j - r_i(\mathbf{q}, \dot{\mathbf{q}}), \quad i=1, \cdots, n \tag{7}$$

where q_i is the i-th element of $\mathbf{q}$, $d'_{ij}(\mathbf{q})$ is the ij-th element of $\mathbf{D}^{-1}(\mathbf{q})$, $r_i(\mathbf{q}, \dot{\mathbf{q}})$ is the i-th element of $\mathbf{D}^{-1}(\mathbf{q})[\mathbf{g}(\mathbf{q}) + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})]$

The control inputs are coupled in Eq. (7), direct use of which complicates the control system design. Instead, we rearrange Eq. (7) into:

$$\ddot{q}_i = \alpha_i(\mathbf{q})u_i + \beta_i(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}), \quad i=1, \cdots, n \tag{8}$$

where $\alpha_i(\mathbf{q}) = d'_{ii}(\mathbf{q})$ and $\beta_i(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u}) = \sum_{j=1, j\ne i}^{n} d'_{ij}(\mathbf{q})u_j - r_i(\mathbf{q}, \dot{\mathbf{q}})$.

The second term of Eq. (8) represents the coupling effects from other joints on the i-th joint as well as Coriolis, centrifugal, and gravitational forces which are a major hindrance in obtaining amenable optimal solutions to the manipulator control problem. However, Eq. (8) could be regarded as an uncoupled subsystem model for the i-th joint of the manipulator if the value of $\beta_i(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u})$ is calculated or approximated by some judicious means. In such a case Problem I can be solved and will therefore yield a computationally simple solution that can be implemented with microprocessors. To this end, we will in this paper pursue a method for calculating approximate values of both $\alpha_i(\mathbf{q})$, and $\beta_i(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u})$, which are nonlinear functions of the manipulator position, velocity, and the control input.

The optimal control problem for each joint system of the manipulator is then stated as follows.

Problem II. Find the control $u_i^*(t)$, $0 \le t \le t_f$, such that the i-th joint system with the state vector $(x_{ip}, x_{iv})^T = (q_i, \dot{q}_i)^T$

$$\dot{x}_{ip} = x_{iv} \tag{9}$$

$$\dot{x}_{iv} = \alpha_i(\mathbf{q})u_i + \beta_i(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u})$$

is steered to a given target point $\mathbf{x}_{if} = (q_{if}, \dot{q}_{if})^T$ from a given initial point $\mathbf{x}_{io} = (q_{io}, \dot{q}_{io})^T$, while minimizing the performance index $J_i(u_i) = \int_{t_0}^{t_f}[\lambda_i + |u_i|]dt$ subject to the input constraint $u_i^- \le u_i \le u_i^+$.

Since the values of $\alpha_i(\mathbf{q})$, and $\beta_i(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u})$ are nonlinear functions of the manipulator position, velocity, and control input, it is still impossible to obtain a closed form solution to Problem II. However, it would be possible to obtain a closed-form optimal solution if the values of α_i and β_i are time-invariant. Also, in this case, we can synthesize the solution in a feedback form which is essential in robotic applications to effectively handle the manipulator dynamics that vary widely with its position and payload. Consequently we (i) assume both α_i and β_i to be constant over a sampling interval, and (ii) update the both at each sampling time to include the preceding nonlinear dependence on $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\mathbf{u}$. The latter is made on the basis of position and velocity feedbacks and more on this will be discussed in Section 2.2. Hence Problem II will be solved for a constant, continuous-time system which is updated at each of discrete sampling intervals.

The optimal solution to Problem II when $\alpha_i(\mathbf{q})$ and $\beta_i(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{u})$ are constant, i.e., $\ddot{q}_i = \alpha_i u_i + \beta_i$, can be synthesized in a feedback form as follows. (Note that this system is controllable and the solution has no singularity region. Subscript i is omitted for convenience):

A. When $\lambda > |\beta|$;

(i). When $z_v \ge 0$;

$$u^*(t) = \begin{cases} u^+ & \text{if} \quad z_p \le \delta^- z_v^2/2\gamma^- \\ 0 & \text{if} \quad z_v^2/2\gamma^- > z_p > \delta^- z_v^2/2\gamma^- \\ u^- & \text{if} \quad z_p \ge z_v^2/2\gamma^- \end{cases} \tag{10}$$

(ii). When $z_v < 0$;

$$u^*(t) = \begin{cases} u^+ & \text{if} \quad z_p \le z_v^2/2\gamma^+ \\ 0 & \text{if} \quad z_v^2/2\gamma^+ < z_p < \delta^+ z_v^2/2\gamma^+ \\ u^- & \text{if} \quad z_p \ge \delta^+ z_v^2/2\gamma^+ \end{cases} \tag{11}$$

where

$$z_p = x_p - x_{fp}$$

$$z_v = x_v - x_{fv}$$

$$\gamma^+ = \alpha u^+ + \beta$$

$$\gamma^- = \alpha u^- + \beta$$

$$\delta^+ = 1 + \frac{4\lambda\alpha u^+}{(\lambda + \beta)^2}$$

$$\delta^- = 1 - \frac{4\lambda\alpha u^-}{(\lambda + \beta)^2}$$

These equations indicate that $u^*(t)$ can be determined in a feedback form with the switching curves as illustrated in Fig. 1.

B. When $\beta > \lambda > 0$;

(i). When $z_v \ge 0$;

$$u^*(t) = \begin{cases} 0 & \text{if} \quad z_p < z_v^2/2\gamma^- \\ u^- & \text{if} \quad z_p \ge z_v^2/2\gamma^- \end{cases} \tag{12}$$

(ii). When $z_v < 0$;

$$u^*(t) = \begin{cases} u^+ & \text{if} \quad \delta^- z_v^2 \le z_p < z_v^2/2\beta \\ 0 & \text{if} \quad z_p/2\gamma^- < \delta^- z_v^2/2\gamma^- \ \text{or} \ z_p = z_v^2/2\beta \\ u^- & \text{if} \quad z_p > z_v^2/2\beta \end{cases} \tag{13}$$

These equations are illustrated in Fig. 2.

C. When $-\beta > \lambda > 0$;

(i). When $z_v \ge 0$;

$$u^*(t) = \begin{cases} u^+ & \text{if} \quad z_p < z_v^2/2\beta \\ 0 & \text{if} \quad z_p = z_v^2/2\beta \ \text{or} \ z_p > \delta^+ z_v^2/2\gamma^+ \\ u^- & \text{if} \ z_v^2/2\beta < z_p \le \delta^+ z_v^2/2\gamma^+ \end{cases} \tag{14}$$

(ii). When $z_v < 0$;

$$u^*(t) = \begin{cases} u^+ & \text{if} \quad z^p \le z_v^2/2\gamma^+ \\ 0 & \text{if} \quad z^p > z_v^2/2\gamma^+ \end{cases} \tag{15}$$

These equations are illustrated in Fig. 3.

2.2. The Averaged Dynamics

In order to utilize the above solution when α_i and β_i are constant, we have developed a method for obtaining their average values at time t, $\bar{\mathbf{p}}_i(t) = [\bar{\alpha}_i(t), \bar{\beta}_i(t)]^T$, and continuously updating those values to cope with the variation in the manipulator dynamics due to their nonlinear dependence on $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\mathbf{u}$. We call this the *averaged dynamics* method, which is described below. At time t_1, using the current position $\mathbf{y}_p(t_1)$ and the current velocity $\mathbf{y}_v(t_1)$, and the previous input $\mathbf{u}(t_1 - \Delta T)$, we calculate an approximate value of $\mathbf{p}_i(t_1) = [\alpha_i(t_1), \beta_i(t_1)]^T$ where ΔT denotes the sampling interval. In other words, the inertial couplings at time t_1 among different joints are approximated with the control input at time $t_1 - \Delta T$, and the nonlinear dependence of the manipulator dynamics on $\mathbf{q}, \dot{\mathbf{q}}$ is taken into consideration by using manipulator's actual behavior (i.e. its position and velocity). The latter implies that the approximation and update are made in a *feedback form;* so is the resulting controller. Any discrepancy at a sampling time between the approximated and the real dynamic parameters will be compensated at the next iteration through feedback. Observe that $\mathbf{p}_i(t_f) = [\alpha_i(t_f), \beta_i(t_f)]^T$ can be determined *a priori* by using the final target position and velocity. Then, the arithmetic average[2] of these values at time t_1,

$$\bar{\mathbf{p}}_i(t_1) = \frac{\mathbf{p}_i(t_1) + \mathbf{p}_i(t_f)}{2} \tag{16}$$

is used to determine the optimal control input, u^*_i, using the switching curve shown in Figures 1 thru 3.

The averaged dynamics method is illustrated in Fig. 4. At time $t_1 \in [0, t_f]$, the optimal control input should be determined based on the information of parameter function $\mathbf{p}_i(t)$, for $t_1 \le t \le t_f$. However, the parameter values are known for both the current state, $\mathbf{p}_i(t_1)$, and the final state, $\mathbf{p}_i(t_f)$, but unknown for $t_1 < t < t_f$. Since the switching times in the optimal control are determined on the basis of the dynamic equations in $[t_1, t_f]$, it is necessary to find an approximate value of the parameter which represents the behavior of the manipulator during the entire remaining time period, i.e. $[t_1, t_f]$. In order to obtain such an approximate value of the parameter, $\bar{\mathbf{p}}_i(t_1)$, we used the two known values of the parameter, i.e. $\mathbf{p}_i(t_1)$ and $\mathbf{p}_i(t_f)$, and made a zero-th order approximation of the parameter function, $\mathbf{p}_i(t)$, for $t_1 \le t \le t_f$. The same procedure is repetitively applied for $t = t_2, t_3, \cdots$. It is apparent from Figure 4 that $\bar{\mathbf{p}}_i(t) \to \mathbf{p}_i(t_f)$ as $t \to t_f$.

The averaged dynamics method takes advantage of the fact that the optimal control is dependent on the accumulation of nonlinear dynamics from the current state to the final state, and we know the coefficients of the manipulator dynamics for the current state and the final state. The modeling error in the averaged dynamics is also implicitly compensated at the time of the next update through position and velocity feedbacks which are used in the averaging process.

[2] *Actually, this does not have to be strictly an arithmetic average. A more general form would be* $\bar{\mathbf{p}}_i(t_1) = \eta\mathbf{p}_i(t_1) + (1 - \eta)\mathbf{p}_i(t_f)$ *where* $0 \le \eta \le 1$.

2.3. Algorithm of the Near Minimum Time-Fuel Controller

Using (i) n local near minimum-time-fuel(NMTF) controllers for an n-jointed manipulator, (ii) the synchronization method discussed above, and (iii) the continual updating of the parameters with the average dynamics method, we can derive a global near-minimum-time-fuel feedback controller as in Fig. 5, which is described below.

1. Given an initial condition and a final condition, compute parameters α_{if}, β_{if}, and let k = 0.
2. Calculate coefficients of the manipulator dynamics using the current state information.
3. Compute $\alpha_i(k)$, $\beta_i(k)$ using Eq. (20).
4. Determine $u_i(k)$ using the switching curves in Equations (14) thru (19).
5. Repeat Steps 3 and 4 for each joint, i.e. for i =1,...,n.
6. Let k = k + 1, and repeat Steps 2 thru 5 until the manipulator reaches the target point.

The NMTF control proposed in this paper deviates from the exact optimal solution due to the following reasons:

(i). The NMTF control is based on the averaged dynamic equations, and thus may not be an optimal control for the overall nonlinear system.

(ii). Using the value of inputs at sampling time k-1 for calculating β_i's, the inertial coupling terms are approximated and used for determining optimal control at sampling time k; this may introduce an error in the calculation of the switching times.

3. SIMULATION RESULTS

We have performed numerical simulation on DEC VAX11/780 for the PUMA 600 series manipulator. The manipulator is manufactured by Unimation, Inc. and consists of six rotational joints, each of which is driven by a DC servomotor.

We employed the Lagrangian formulation to derive the PUMA manipulator dynamics which is then used to simulate the behavior of the first three joints of the manipulator. The remaining three joints are not considered here, since they are used only for orienting the end-effector and, therefore, do not play an important role in the area of our present concern. Note that this simulator for the forward dynamics of the manipulator is not a part of the controller and will be replaced by the manipulator in case of actual implementation.

The controller computes first the parameters associated with the final position when both the initial and final positions are given. At each sampling time, the controller computes the current values of the manipulator dynamic parameters, and then determines the parameters for the next iteration with the averaged dynamics. Control input is then determined using the switching curves as described in Eqs. (14) thru (19) which are constructed on the basis of the parameters updated.

Numerical values used in this simulation are as follows.

(i). The mass, center of mass, and inertia of each joint of the PUMA manipulator are given in Table 1.

(ii). The bounds on the control input torque are assumed to be $|u_1| \le 300$ Nm, $|u_2| \le 400$ Nm, and $|u_3| \le 200$ Nm.

Observe that the choice of these parameters is made almost arbitrary for the sake of numerical demonstration and any of such choices does not change the basic performance of our manipulator control method.

In order to examine the performance of the NMTF controller for different ranges of motion, the first three joints of the manipulator are commanded to move from various initial points to final points. The corresponding simulation results are given in Table 2 and show that the NMTF controller performs well for wide range of motions. Particularly, the results indicate the relative insensitivity to the range of motion and are therefore in sharp contrast with those reported in [1] which showed 20% or more overshoots for a small range of motion.

For the case of no load, Fig. 6 shows the response of each joint of the manipulator to the NMTF controller command.

The first three joints of the manipulator are now ordered to move from initial joint angles (45°, −30°, 150°) to final joint angles (0°, −60°, 210°). When the manipulator picks up a maximum load (i.e. 5 pounds $\approx$ 2.25 Kg), the simulation reslut shows the capability that, even with a maximum payload, the controller can drive the manipulator to its capacity. This result is not unexpected since the precise parameters and the load characteristics for the manipulator are assumed to be known to the NMTF controller. If these are unknown, one has to appeal to an adaptive control method similar to those in Dubowsky and DesForges[13], Koivo and Guo[14], and Kim and Shin[15]. Table 3 summarizes the simulated effects on overshoot in each joint when the weighting factor, λ, is varied. It indicates that as the weighting factor increases, the NMTF controller approaches a near time-optimal controller, and tends to have a slight increase in overshoot.

The effects of the variation of the sampling interval on overshoot are simulated and presented in Table 4. Since the dynamic models used for the NMTF controller is updated at every sampling interval, these effects can be employed to measure implicitly the modeling errors induced by the averaging process. The sampling interval is varied from 0.001 second to 0.02 second and the result shows that as the sampling interval increases, the overshoot increases accordingly. This result is expected; the NMTF controller is derived for continuous-time domain with the averaged dynamic equations, and, therefore, it would become more accurate when the sampling interval gets smaller. Hence, it is desirable to select the sampling interval which (i) provides acceptable/required accuracy, and (ii) does not require excessive computation.

Our simulation results have indicated that the manipulator can be driven at high speed with reasonable accuracy regardless of the load it is carrying. One can observe that since our NMTF controller is developed on the basis of the averaged dynamics method, it is not restricted to any range of motion (unlike the one in [1]). In addition to high operation speed, this controller requires very simple computations making the implementation easier and less expensive, when compared to the conventional two stage algorithms with separate path planning and path tracking. As a whole this simulation exhibits that the proposed control method has great potential for high performance with the capability of coping with the nonlinear, coupled dynamics of the manipulator.

4. CONCLUSION

We have presented the design of a manipulator feedback control system on the basis of the minimum time-fuel criterion, which has a direct link to the goal of improving productivity and saving energy. The design has focussed on the following three important features. First, unlike most other existing methods, this control method is not based on a conventional, separate path planning algorithm which does not take the manipulator dynamics into account and therefore results in low utilization of the manipulator capabilities. The design of the NMTF controller has placed emphasis, particularly on the improvement of the manipulator performance by nearly fully utilizing the capacity of manipulators. This was made possible with a judicious approximation with the averaging of the manipulator dynamics. Secondly, this controller provides a feedback control algorithm with a very simple structure so that it can be implemented on-line with microprocessors. Primarily, this controller is intended to provide fast operation speed with less energy as well as reasonable settling accuracy.

The averaged dynamics method is a good engineering solution to one of the most difficult problems in robot control, namely the nonlinearity and joint couplings in the manipulator dynamics. This is a simple yet novel departure from conventional robot control techniques toward the goal of efficient control of the manipulator.

Considering all the preceding facts, the manipulator control method proposed here has great potential use-- due to its high performance capability and simplicity in structure-- in the design of intelligent controllers for the growing number of sophisticated industrial manipulators.

REFERENCES

[1] M. E. Kahn, and B. E. Roth, "The near-minimum-time control of open-loop articulated kinematic chains", *ASME J. DSMC*, vol. 93, no. 3, pp. 164 - 172, Sep. 1971.

[2] C. S. Lin, P. R. Chang, and J. Y. S. Luh, "Formulation and

optimization of cubic polynomial joint trajectories for mechanical manipulators", *IEEE Conf. on Decision and Control*, Orlando, Florida, Dec. 1982.

[3] J. Y. S. Luh, and C. S. Lin, "Optimum path planning for mechanical manipulators", *ASME J. DSMC*, vol. 102, pp. 142 - 151, June 1981.

[4] J. Y. S. Luh, and R. P. C. Paul, "Resolved-acceleration control of mechanical manipulators", *IEEE Trans. Automatic Control*, vol. AC-25, no. 3, pp. 468-474, June 1980.

[5] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "On-line computational scheme for mechanical manipulators", *ASME J. DSMC*, vol. 102, pp. 69-76, June 1980.

[6] P. M. Lynch, "Minimum-time, sequential axis operation of a cylindrical, two-axis manipulator", *Joint Automatic Control Conference*, vol. 1, WP-2A, 1981.

[7] W. E. Snyder, W. A. Gruver, "Microprocessor implementation of optimal control for a robotic manipulator system", *IEEE Conference on Decision and Control*, pp. 839 - 841, 1979.

[8] T. L. Turner, and W. A. Gruver, "A viable suboptimal controller for robotic manipulators", *IEEE Conference on Decision and Control*, pp. 83 - 87, Dec. 1980.

[9] D. E. Whitney, "Resolved motion rate control of manipulators and human prosthesis", *IEEE Trans. Man-Machine Systems*, vol. MMS-10, No. 2, pp. 47-53, June, 1969.

[10] J. Y. S. Luh, "An anatomy of industrial robots and their controls", *IEEE Trans. Automatic Control*, vol. AC-28, No. 2, pp. 133-153, February 1983.

[11] K. G. Shin and S. M. Malin, "Dynamic adaptation of robot kinematic control to its actual behavior", *Proc. the 1981 IEEE Int'l Conf. on Cybernetics and Society*, Atlanta, Georgia, October 1981.

[12] A. P. Sage, *Optimum systems control*, Prentice Hall, 1968.

[13] S. Dubowsky and D. T. DesForges, "The application of model-referenced adaptive control to robot manipulators", *ASME J. DSMC*, vol. 101, pp. 193-200, Sep. 1979.

[14] A. J. Koivo and T. H. Guo, "Adaptive linear controller for robotic manipulators", *IEEE Trans. Automatic Control*, vol. AC-28, no. 2, pp. 162-171, Feb. 1983.

[15] B. K. Kim and K. G. Shin, "Adaptive control of robotic manipulators to unpredictable changes in their dynamics", to appear in *IEEE Trans. Aerospace and Electronic Systems*.

Case		I	II	III
Initial	$q_1(0)$	60.0	45.0	-45.0
condition	$q_2(0)$	-80.0	-30.0	45.0
(deg)	$q_3(0)$	80.0	150.0	60.0
Final	$q_1(t_f)$	40.0	0.0	0.0
condition	$q_2(t_f)$	-90.0	-60.0	-45.0
(deg)	$q_3(t_f)$	90.0	210.0	150.0
Settling time (sec)	t	0.11	0.19	0.31
Overshoot	Joint 1	0.7	5.0	5.2
(%)	Joint 2	6.5	0.0	5.9
	Joint 3	3.7	1.2	6.1

Table 2. Simulation results with various initial and final conditions.

λ	Joint 1	Joint 2	Joint 3
10	2.2	0.0	0.8
100	5.0	0.0	1.3
1000	6.9	3.3	2.0
100,000	12.2	3.3	16.0

Table 3. Effects of the weighting factor on overshoot(%).

Link	Mass	Center of Mass			Inertia		
	M (Kg)	$\bar{x}$ (m)	$\bar{y}$ (m)	$\bar{z}$ (m)	I_x (Kg m^2)	I_y (Kg m^2)	I_z (Kg m^2)
1	2.27	0.0	0.0	0.075	0.00376	0.00376	0.0169
2	15.91	-0.216	0.0	0.0	0.9897	0.1237	0.1237
3	11.36	0.0	0.0	0.216	0.0074	0.0074	0.7067

Table 1. Mass, first moments, and inertias of the first three joints for the PUMA 600 manipulator.

ΔT	Joint 1	Joint 2	Joint 3
0.001	5.0	0.0	1.3
0.002	5.0	0.0	3.9
0.005	12.6	0.0	2.2
0.01	22.1	12.0	27.7
0.02	25.2	22.0	29.7

Table 4. Effects of sampling interval (ΔT) on overshoot(%).

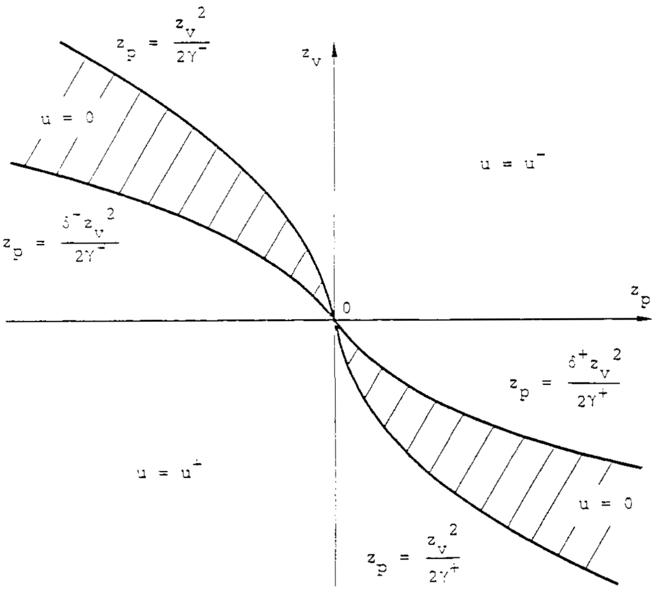

Fig. 1. Switching curves when $\lambda > |\beta|$.

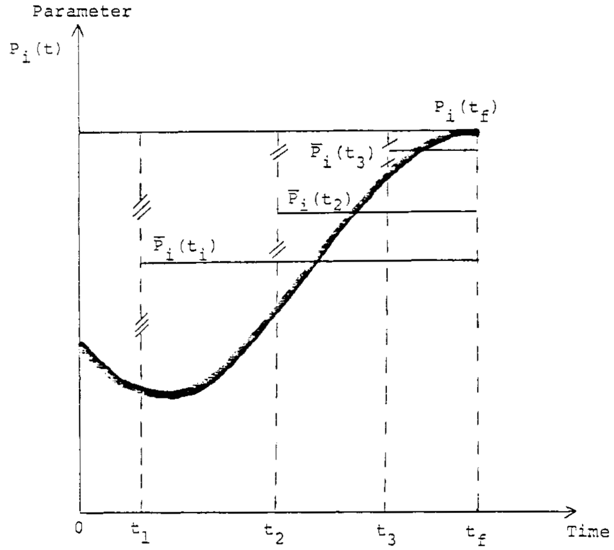

Fig. 4. The averaged dynamics method.

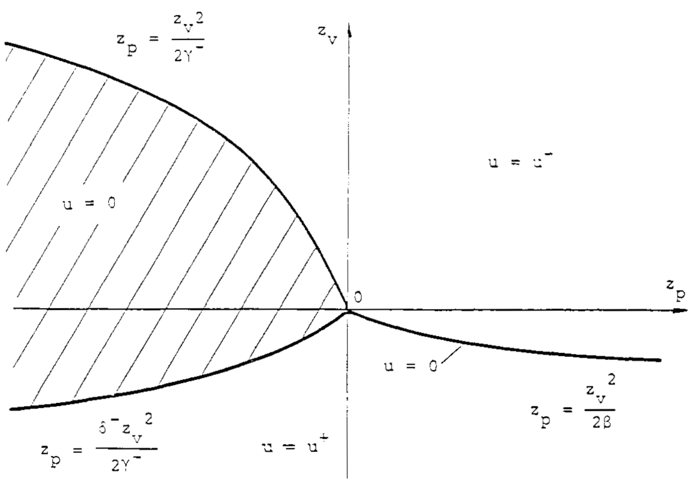

Fig. 2. Switching curves when $\beta > \lambda$.

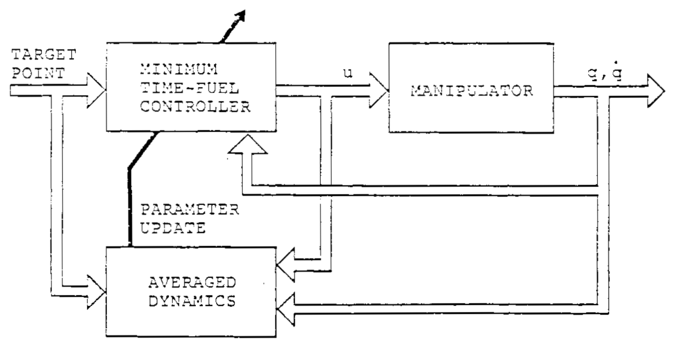

Fig. 5. Block diagram of the near-minimum time-fuel controller.

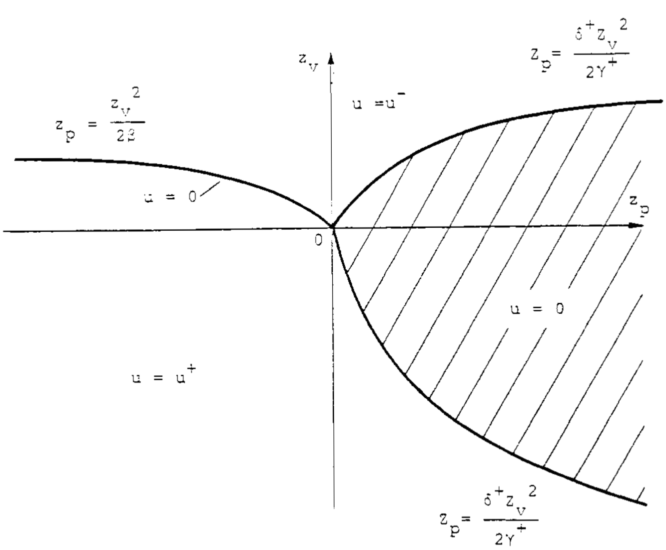

Fig. 3. Switching curves when $-\beta > \lambda$.

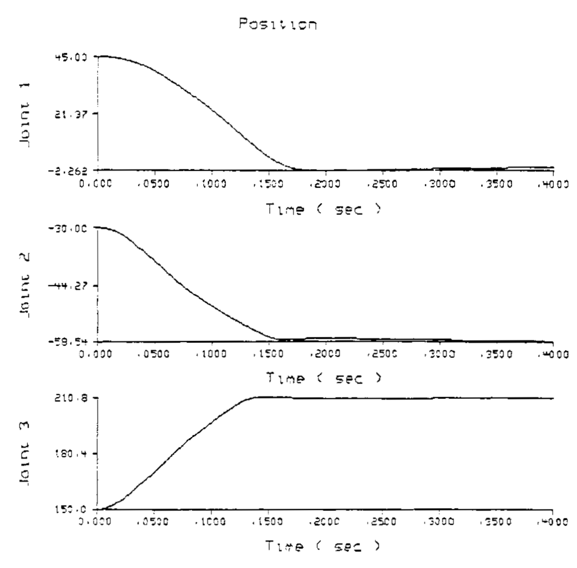

Fig. 6. Manipulator responses with no load.