Time-Optimal Control of Robotic Manipulators Along Specified Paths

J.E. Bobrow, S. Dubowsky and J.S. Gibson

The International Journal of Robotics Research 1985; 4; 3
DOI: 10.1177/027836498500400301

Time-Optimal Control of Robotic Manipulators Along Specified Paths

J. E. Bobrow
Department of Mechanical Engineering
University of California, Irvine
Irvine, California 92717

S. Dubowsky
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

J. S. Gibson
Mechanical, Aerospace, and Nuclear Engineering
University of California, Los Angeles
Los Angeles, California 90024

Abstract

The minimum-time manipulator control problem is solved for the case when the path is specified and the actuator torque limitations are known. The optimal open-loop torques are found, and a method is given for implementing these torques with a conventional linear feedback control system. The algorithm allows bounds on the torques that may be arbitrary functions of the joint angles and angular velocities. This method is valid for any path and orientation of the end-effector that is specified. The algorithm can be used for any manipulator that has rigid links, known dynamic equations of motion, and joint angles that can be determined at a given position on the path.

1. Introduction

For many industrial applications, present robotic manipulators are too slow to justify their use economically. Their speed and hence their productivity are limited by the capability of their actuators. Increasing actuator size and power is not the best solution; it is largely self-defeating because of the increased inertia of the actuators themselves and because of the increased cost and power consumption of the larger actuators. A more successful approach is to minimize the time needed to perform a given task, subject to the constraints imposed by the actuators. The subject of this paper is the minimum-time control problem for applications where the path of the manipulator is specified.

Work on minimum-time control problems for manipulators began as early as the late 1960s (Kahn 1970; Kahn and Roth 1971). The limits on the actuator torques were assumed to be constant, and the path was not constrained (only the endpoints were specified). Although this approach is suitable for some applications, it is often necessary to specify the manipulator trajectory in order to avoid obstacles. This additional collision-avoidance constraint may be added to the unconstrained path minimum-time problem. The result is a highly nonlinear, difficult-to-solve optimal control problem.

Niv and Auslander (1984) show some progress toward solving this problem using a parameter optimization scheme on the joint actuator switching times. During the motion, each actuator exerts maximum control torque (bang-bang) while enabling the manipulator to avoid all obstacles and reach its final destination. This method involves considerable computation and may be difficult to implement for general manipulators. Another approach (Dubowsky and Shiller 1984) is to minimize the time along any known path using the algorithm described in this paper and to vary the path to find the one that avoids all obstacles and gives the shortest time.

The research reported here was supported by the National Science Foundation under grants ECS 78-04753 and CME 80-08926.

The International Journal of Robotics Research,
Vol. 4, No. 3, Fall 1985,

Several other researchers have addressed the time-optimal control problem. Lynch (1981) developed a specialized minimum-time algorithm for the Stanford-type manipulator, assuming that the actuator torque bounds were constant. Another method (Hollerbach 1984) has been developed that scales any known path velocity profile to make full use of the actuators. While this method produces shorter traveling times than conventional techniques, it does not produce the minimum-time solution. A technique also was developed (Luh and Walker 1977; Luh and Lin 1981) to minimize the time required to move along a specified path consisting of straight lines and circular arcs. In this work, piecewise constant acceleration and maximum velocity constraints were assumed. Although these assumptions are common in manipulator control, the maximum achievable accelerations and velocities actually can vary substantially with manipulator configuration and angular velocities, both because of the nonlinear manipulator dynamics and because the maximum torques that electric motors and hydraulic actuators can produce depend on the angular velocities of the joints (Herrick 1982; Kollmorgen Corporation 1983).

This paper presents in detail the solution derived by Bobrow (1982) and presented in Bobrow, Dubowsky, and Gibson (1983) of the minimum-time control problem with specified manipulator path and state-dependent constraints on the actuators. The actuator torque and force constraints can be arbitrary functions of the joint positions and velocities. Rigid links are assumed, and the full manipulator dynamics are modeled. The solution given here is valid for any smooth path along which the joint angles, including end-effector orientation, can be determined uniquely at each point.

In the solution presented here, the distance and velocity of the end-effector along the specified path are taken as the state vector, and the nonlinear manipulator dynamics and actuator constraints are transformed into state-dependent constraints on the acceleration along the path. The problem thus becomes a time-optimal problem for a second-order linear system with nonlinear state-dependent constraints on the control, which is the acceleration along the path. The basic idea of the solution is to select the acceleration profile that produces the largest velocity profile such that, at each point on the path, the velocity is no greater than the maximum velocity at which the actuators can hold the manipulator on the path.

This solution is given in terms of a switching curve in the phase plane for the tip motion along the path. When the velocity lies below the switching curve, maximum acceleration is optimal; when the velocity lies on the switching curve, either maximum acceleration or maximum deceleration is optimal, depending on the location on the switching curve. Because the actuator efforts can be determined from the position, velocity, and acceleration along the path, the switching curve provides a graphical representation of the feedback law for the time-optimal control. The switching curve can be computed efficiently by iteration on one switching point at a time while solving a first-order nonlinear differential equation.

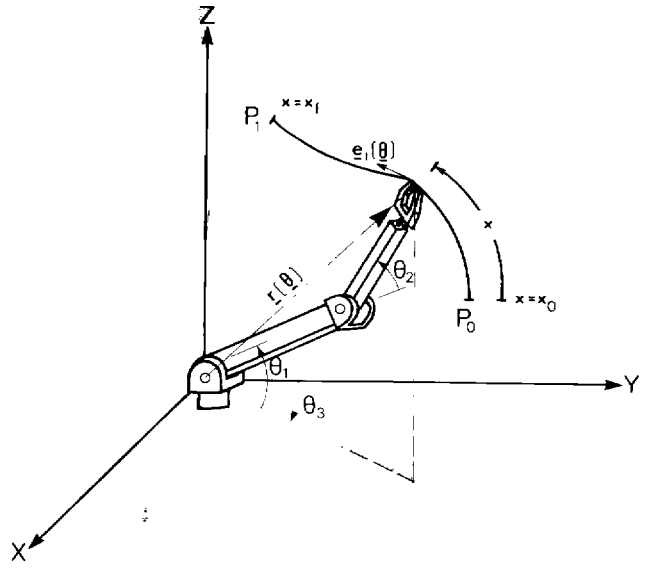
We should note that Shin and McKay (1984) have subsequently derived a similar algorithm. With the assumptions that the bound on each actuator torque is a quadratic function of the joint velocity and that each joint position is written as a polynomial in the path parameter, they have obtained added insight into the case of high joint friction.

Because the mathematical optimal control problem involves a second-order system with constraints on the control, we first tried standard optimal control methods — in particular, Pontryagin's maximum principle — for the solution. However, in even the simplest cases, the numerical algorithms did not converge to a solution. The solution finally came from the nonstandard, though conceptually straightforward, approach in this paper, and the resulting numerical algorithm has performed well on numerous examples.

The control problem is formulated first in Section 2 for a three-degree-of-freedom elbow-type manipulator. After the mathematical optimal control problem is derived, the solution is given in the form of an algorithm for constructing the switching curve. While the motivation for the algorithm is discussed in Section 2, the complete mathematical proof that the algorithm indeed gives the optimal control is deferred until the Appendix. At the end of Section 2 an example with a curved three-dimensional path is given.

Section 3 shows how the dynamics and orientation of the end-effector can be included in the problem. In fact, any number of degrees of freedom can be han-

Fig. 1. The manipulator model used in this study.



dled as long as all of the joint angles can be determined by an inverse arm solution at each point on the path. In this case, the algorithm in Section 2 still applies. Also, it is pointed out in Section 3 that, while it is easiest initially to think of the state variable in the mathematical optimal control problem as the distance along the path, the results of the paper apply for almost any parameterization of the path.

The development presented in Section 2 uses Lagrange's equations with the joint positions as the generalized coordinates. Section 4 extends this development to cases where more than the minimum required number of coordinates are used to describe the system. Lagrange's equations are used along with Lagrange multipliers to formulate the equations that must be solved. A detailed example is presented that demonstrates that this approach often reduces the complexity of the equations.

2. Formulation and Solution of the Optimal Control Problem

2.1. Problem Formulation

To illustrate the minimum-time control algorithm, we first consider the relatively simple case where the tip of the three-degree-of-freedom manipulator shown in Fig. 1 is required to move along a specified path $\mathbf{P}_0$ to $\mathbf{P}_1$, starting and finishing at rest. This case does not include the motion of the end-effector relative to the second link. In this case, the dynamics of the end-effector are assumed to have negligible influence on the dynamics of the manipulator, as in many industrial robotic systems. The method can handle such motion if the orientation of the end-effector is specified at each point on the path, as illustrated by the third example in this paper, and in Dubowsky and Shiller (1984).

The equations of motion for this system can be derived using Lagrange's equations (see Bobrow 1982), which have the form

$$\mathbf{M}(\theta)\ddot{\theta} + \mathbf{h}(\theta, \dot{\theta}) = \mathbf{T}. \tag{1}$$

The vectors $\theta = (\theta_1, \theta_2, \theta_3)^T$ and $\mathbf{T} = (T_1, T_2, T_3)^T$ are the joint angles and the applied actuator torques. The torque T_i acts about the θ_i-axis, and T_2 acts between links 1 and 2. The detailed definitions of the mass matrix $\mathbf{M}$ and vector $\mathbf{h}$ are given in Bobrow (1982).

The optimal control problem is as follows: Given the manipulator equations of motion (Eq. 1), a path through which the tip must move (Fig. 1), and actuator torque constraints of the form

$$T_{i_{\min}}(\theta, \dot{\theta}) \leq T_i \leq T_{i_{\max}}(\theta, \dot{\theta}), \tag{2}$$

find the torques $\mathbf{T}(t)$ that will drive the manipulator from the initial position at $\mathbf{P}_0$ to the final position at $\mathbf{P}_1$ in minimum time.

In our solution, the time-optimal control problem is transformed into an equivalent mathematical optimal control problem in which the single control variable is $\ddot{x}$, the tip acceleration along the path; that is, $\ddot{x} = d^2x/dt^2$ *where x is measured along the path from* $\mathbf{P}_0$. For the transformed problem, we must determine at each position and velocity on the path the constraints on the linear acceleration $\ddot{x}$ corresponding to the actuator torques. This requires that the joint angles be computed as functions of x and that the angular velocities and accelerations be computed as functions of x, $\dot{x}$, and $\ddot{x}$; that is, we must have

$$\theta = \theta(x), \tag{3A}$$

$$\dot{\theta} = \dot{\theta}(x, \dot{x}), \tag{3B}$$

$$\ddot{\theta} = \ddot{\theta}(x, \dot{x}, \ddot{x}). \tag{3C}$$

With the path of the tip specified, the function $\theta(x)$ for Eq. (3A) is determined at least implicitly. We emphasize that θ need not be written as an explicit function of x; the ability to compute θ numerically for each value of x is sufficient. Similarly, instead of explicit expressions for Eqs. (3B) and (3C), numerical evaluation of $\dot{\theta}$ and $\ddot{\theta}$ according to the following kinematic development is sufficient.

To obtain Eqs. (3B) and (3C), we note that the position vector $\mathbf{r}$ of the tip can be thought of as either a function of the joint angles or a function of the distance along the path. Hence we write

$$\mathbf{r} = \mathbf{r}(\theta) = \tilde{\mathbf{r}}(x). \tag{4}$$

Differentiating Eq. (4) with respect to time yields

$$[r_\theta]\dot{\theta} = \tilde{\mathbf{r}}_x \dot{x}, \tag{5}$$

where $[r_\theta]$ is the Jacobian matrix of partial derivatives of the position vector components with respect to the joint angles and $\tilde{\mathbf{r}}_x$ is the unit vector tangent to the path. Then, when the manipulator is not at a singular point, the Jacobian $[r_\theta]$ is invertible and we can solve Eq. (5) for the expression in Eq. (3B):

$$\dot{\theta}(x, \dot{x}) = [r_\theta]^{-1}\tilde{\mathbf{r}}_x \dot{x}. \tag{6}$$

We should note that, in practice, manipulator trajectories with singularities are avoided.

For Eq. (3C), differentiating Eq. (5) with respect to time yields

$$[r_\theta]\ddot{\theta} + [\dot{r}_\theta]\dot{\theta} = \tilde{\mathbf{r}}_x \ddot{x} + \tilde{\mathbf{r}}_{xx}\dot{x}^2, \tag{7}$$

where $[\dot{r}_\theta]$ is the time derivative of the Jacobian matrix and $\tilde{\mathbf{r}}_{xx}$ is the second derivative of $\tilde{\mathbf{r}}$ with respect to x. (The expression $[\dot{r}_\theta]\dot{\theta}$ contains the term $(\partial^2 r_i/\partial\theta_j\partial\theta_k)\dot{\theta}_j\dot{\theta}_k$, familiar in rigid-body dynamics; see Bobrow 1982.)

Note that the first term on the right-hand side of Eq. (7) is the tangential acceleration of the tip along the path and the second term is the normal acceleration. Still assuming that the manipulator is not in a singular configuration, we can solve Eq. (7) for Eq. (3C):

$$\ddot{\theta} = [r_\theta]^{-1}(\tilde{\mathbf{r}}_x \ddot{x} + \tilde{\mathbf{r}}_{xx}\dot{x}^2 - [\dot{r}_\theta]\dot{\theta}). \tag{8}$$

Now we can derive expressions for the maximum acceleration and deceleration the actuators can produce, at any distance x along the path and velocity $\dot{x}$. Substituting Eq. (8) into Eq. (1) yields

$$\ddot{x}\mathbf{c}_1(x) + \mathbf{c}_2(x, \dot{x}) = \mathbf{T}, \tag{9}$$

where

$$\mathbf{c}_1(x) = \mathbf{M}(\theta)[r_\theta]^{-1}\tilde{\mathbf{r}}_x \tag{10}$$

and

$$\mathbf{c}_2(x, \dot{x}) = \mathbf{M}(\theta)[r_\theta]^{-1}(\tilde{\mathbf{r}}_{xx}\dot{x}^2 - [\dot{r}_\theta]\dot{\theta}) + \mathbf{h}(\theta, \dot{\theta}). \tag{11}$$

Given the distance x along the path, the tip velocity $\dot{x}$, and the tangential acceleration $\ddot{x}$, Eq. (9) shows the unique values of the three actuator torques to be

$$T_i = c_{1i}(x)\ddot{x} + c_{2i}(x, \dot{x}), \qquad i = 1, 2, 3. \tag{12}$$

With Eq. (12), the torque constraints in Eq. (2) yield the following constraints on acceleration:

$$T_{i\min} - c_{2i}(x, \dot{x}) \leqslant c_{1i}(x)\ddot{x} \leqslant T_{i\max} - c_{2i}(x, \dot{x}), \qquad i = 1, 2, 3. \tag{13}$$

Since, for any x and $\dot{x}$, $\theta(x)$ and $\dot{\theta}(x, \dot{x})$ can be computed from the inverse arm solution and Eq. (6), $T_{i\min}$ and $T_{i\max}$ can be written as functions of x and $\dot{x}$. Then, if $c_{1i}(x) \neq 0$, Eq. (13) can be written as

$$f_i(x, \dot{x}) \leqslant \ddot{x} \leqslant g_i(x, \dot{x}), \tag{14}$$

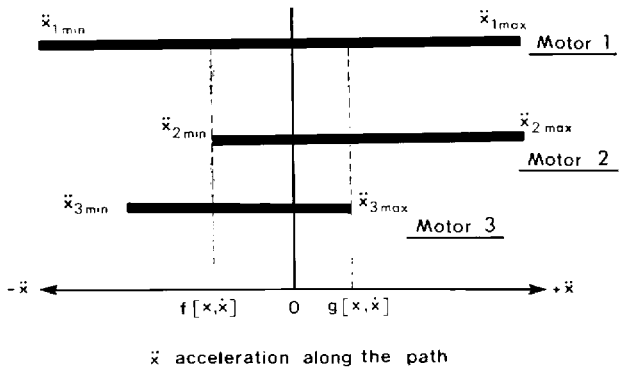
where

$$f_i(x, \dot{x}) = \begin{cases} (T_{i\min} - c_{2i})/c_{1i}, & c_{1i} > 0, \\ (T_{i\max} - c_{2i})/c_{1i}, & c_{1i} < 0, \end{cases} \tag{15}$$

and

$$g_i(x, \dot{x}) = \begin{cases} (T_{i\max} - c_{2i})/c_{1i}, & c_{1i} > 0, \\ (T_{i\min} - c_{2i})/c_{1i}, & c_{1i} < 0. \end{cases} \tag{16}$$

Fig. 2. Ranges of acceleration possible for each actuator at some instant.



When $c_{1i}(x) \neq 0$, Eqs. (14)–(16) give the range of tip acceleration $\ddot{x}$ for which the actuators can hold the tip on the path without violating the ith constraint in Eq. (2). For the tip to remain on the path, $\ddot{x}$ must lie in the intersection of the intervals $[f_i(x, \dot{x}), g_i(x, \dot{x})]$, with the intersection taken over the values of i for which $c_{1i}(x) \neq 0$. It can happen, usually when the velocity is too great, that the intervals $[f_i(x, \dot{x}), g_i(x, \dot{x})]$ do not intersect, in which case the manipulator tip will leave the path immediately. On the other hand, if $c_{1i}(x) \neq 0$ for all i and the three intervals $[f_i(x, \dot{x}), g_i(x, \dot{x})]$ have nonempty intersection, then the necessary and sufficient condition for the tip to stay on the path is that Eq. (14) hold for all i.

Although the constraints in Eq. (13) must hold for all i, if $c_{1i}(x) = 0$ for some i, then the selection of $\ddot{x}$ cannot affect whether Eq. (13) holds for that i. In this case, $\ddot{x}$ must be chosen to satisfy the two remaining constraints. As long as the manipulator is not in a singular configuration, there will be at least one nonzero $c_{1i}(x)$. This follows from the right-hand side of Eq. (10) since the mass matrix $\mathbf{M}(\theta)$ is always positive definite and $\tilde{\mathbf{r}}_x$ is a unit vector.

For given x and $\dot{x}$, an *admissible acceleration* is any tangential acceleration $\ddot{x}$ at which the actuators can hold the tip on the prescribed path without violating the constraints. The foregoing discussion shows that, if any admissible acceleration exists, then the range of admissible accelerations is given by

$$f(x, \dot{x}) \leq \ddot{x} \leq g(x, \dot{x}), \tag{17}$$

where

$$f(x, \dot{x}) = \max_i f_i(x, \dot{x}) \tag{18}$$

and

$$g(x, \dot{x}) = \min_i g_i(x, \dot{x}), \tag{19}$$

with the maximum and minimum taken over those i for which $c_{1i}(x) \neq 0$. See Fig. 2. Note that if two of the intervals $[f_i(x, \dot{x}), g_i(x, \dot{x})]$ do not intersect, then $f(x, \dot{x}) > g(x, \dot{x})$. Maintaining $f(x, \dot{x}) \leq g(x, \dot{x})$, so that an admissible acceleration exists, is a key idea in solving the time-optimal problem. We now have the mathematical time-optimal control problem:

> Given $x(0)$ and $\dot{x}(0)$, choose $\ddot{x}(t)$ to minimize the final time t_f for which $x(t_f) = x_f$ and $\dot{x}(t_f) = \dot{x}_f$, subject to Eq. (17) at each t.

Although we have transformed the manipulator control problem so that the tangential acceleration becomes the control variable for the mathematical time-optimal problem, we have not lost sight of the torques, which are the physical controls. Recall that Eq. (12) gives the actuator torques/forces in terms of x, $\dot{x}$, and $\ddot{x}$.

2.2. Solution of the Time-Optimal Problem

The basic idea of the time-optimal solution is to choose the acceleration $\ddot{x}$ to make the velocity $\dot{x}$ as large as possible at every point without violating the condition $f(x, \dot{x}) \leq g(x, \dot{x})$. This is suggested by the identity

$$t_f = \int_{x_0}^{x_f} (dx)/\dot{x} \tag{20}$$

It is proved in the Appendix that to minimize t_f, $\ddot{x}$ always takes either its largest or its smallest possible value; that is, either $\ddot{x} = g(x, \dot{x})$ or $\ddot{x} = f(x, \dot{x})$. Therefore, finding the optimal control law amounts to finding the times, or positions, at which $\ddot{x}$ switches between maximum acceleration and maximum deceleration.

Fig. 3. A typical minimum-time trajectory with one switching time.

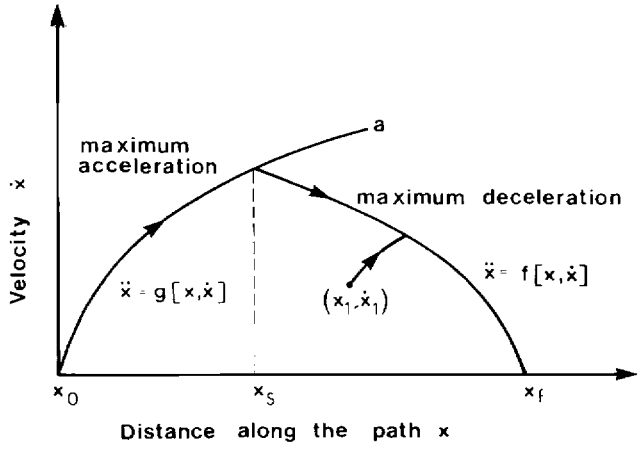
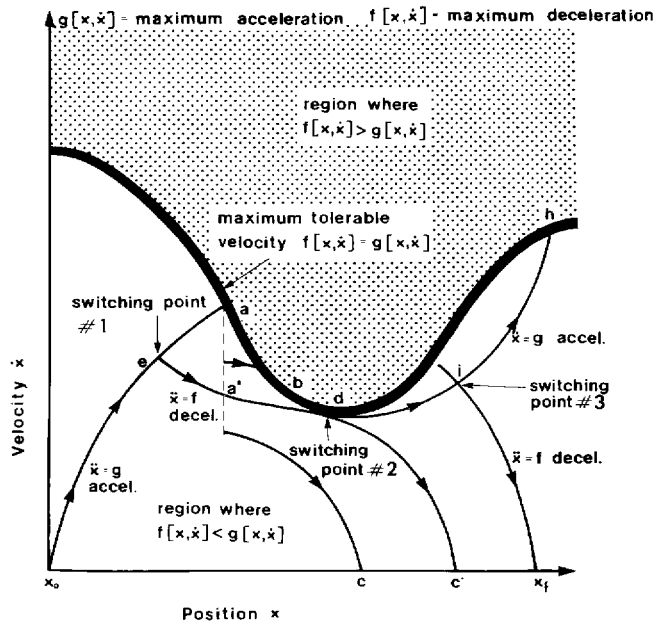


Fig. 4. An example showing the minimum-time trajectory construction method.



2.2.1. *Problems with One Switching*

The best approach for finding the switching position is to construct the switching curves in the $x - \dot{x}$ phase plane. We will give an algorithm for the construction for problems with multiple switchings, but first it should help to consider the case of one switching. A typical minimum-time trajectory with one switching is shown in Fig. 3. The manipulator tip starts accelerating from the initial position x_0 with $\ddot{x} = g(x, \dot{x})$. At the switching position x_s, the acceleration switches to $\ddot{x} = f(x, \dot{x})$ and continues this deceleration until coming to rest at the final position x_f.

To find x_s, we solve $\ddot{x} = g(x, \dot{x})$ forward in time from the point $x = x_0$, $\dot{x} = 0$ to some point a as shown in Fig. 3. Then we solve $\ddot{x} = f(x, \dot{x})$ backward in time from $x = x_f$, $\dot{x} = 0$ until the two trajectories intersect at x_s. The phase plane trajectory that results from solving $\ddot{x} = f(x, \dot{x})$ backward from $x = x_f$, $\dot{x} = 0$ is the switching curve for this case. If the manipulator starts at some position x_1 with tip velocity $\dot{x}_1$, as shown in the figure, the optimal control policy is to use the maximum acceleration $\ddot{x} = g(x, \dot{x})$ until the phase plane trajectory intersects the switching curve and then switch to maximum deceleration $\ddot{x} = f(x, \dot{x})$.

2.2.2. *Multiple Switchings*

The minimum-time problem becomes considerably more difficult when the maximum acceleration curve $\ddot{x} = g(x, \dot{x})$ proceeding from x_0 and the maximum deceleration curve $\ddot{x} = f(x, \dot{x})$ proceeding backward from x_f do not intersect before the velocity becomes too large and the condition $f(x, \dot{x}) \leq g(x, \dot{x})$ is violated. In this case, before the final switch to the deceleration curve that brings the tip to rest at x_f, the optimal control policy requires earlier switches between acceleration and deceleration to avoid building up velocities, and hence large inertial forces, at which the actuators can no longer hold the tip on the specified path.

Finding the multiple switching points is the most difficult part of the minimum-time problem. Several approaches were tried, including a conjugate gradient optimization algorithm that treated the switching times as parameters and the numerical solution of the two-point boundary value problem that results from the maximum principle with state-dependent constraints on the control (Leitmann 1966). In neither of these approaches did the numerical algorithm converge to a solution in even the simplest cases. We finally developed a nonstandard approach that is actually more straightforward and that has been successful on numerous examples. This method yields a simple numerical algorithm for constructing the switching curve.

The method is motivated by noting that, for most points on the path, there is a tip velocity above which no combination of admissible torques will hold the

Fig. 5. A three-dimensional path. Dimensions are in feet.

manipulator on the path. For a given path, then, we have a maximum velocity curve in the phase plane, as shown in Fig. 4. Below the curve, we have $f(x, \dot{x}) \leqslant g(x, \dot{x})$, so that $\ddot{x}$ can be chosen to satisfy Eq. (17). Above the curve, we have $f(x, \dot{x}) > g(x, \dot{x})$, so that no admissible $\ddot{x}$ exists. The curve satisfies $f(x, \dot{x}) = g(x, \dot{x})$ and is the key to the algorithm.

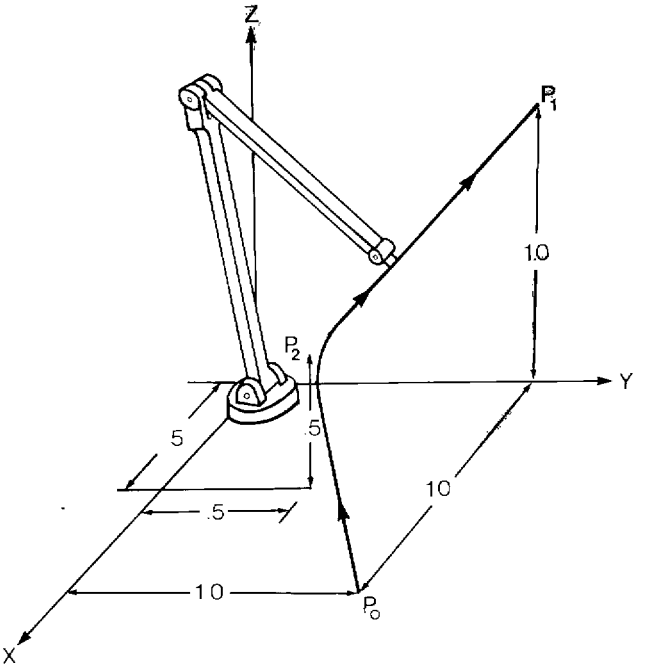
We state here our most understandable version of the algorithm for constructing the switching curve. In the Appendix we prove that the control policy determined by this switching curve is optimal. The idea is that the higher the phase plane trajectory, the shorter the traveling time, as indicated by Eq. (20). To minimize the traveling time, $\ddot{x}$ is always chosen as either the maximum possible acceleration $g(x, \dot{x})$ or the maximum possible deceleration $f(x, \dot{x})$. The switchings between $g(x, \dot{x})$ and $f(x, \dot{x})$ are chosen so that the phase plane trajectory just touches the maximum velocity curve. The switching curve is constructed by the following six steps. Refer to Fig. 4.

Step 1. Integrate the equation $\ddot{x} = g(x, \dot{x})$ from the initial state $(x_0, \dot{x}_0)$ until the maximum velocity curve is reached at some point a.

Step 2. From point a, drop to some lower velocity on the dotted vertical line and then integrate the equation $\ddot{x} = f(x, \dot{x})$ forward in time. One of two things will happen: Either the trajectory will intersect the maximum velocity curve again at some point b or the trajectory will intersect the x-axis at some point c. The object is to find, by iteration, the point a' such that the deceleration $(\ddot{x} = f)$ trajectory emanating from a' just touches the maximum velocity curve at a single point d and then continues downward, intersecting the x-axis at c'. If $c' \geqslant x_f$, then there is only one switching point.

Step 3. From a', integrate $\ddot{x} = f(x, \dot{x})$ backward in time until the acceleration trajectory from x_0 to a is intersected at some point e.

Step 4. Integrate the equation $\ddot{x} = g(x, \dot{x})$ forward in time from point d until either x_f is passed or the trajectory again intersects the maximum velocity curve, as at point h. It is proved in the Appendix that it is possible to resume maximum acceleration $(\ddot{x} = g)$ at d without immediately violating $f(x, \dot{x}) \leqslant g(x, \dot{x})$.



Step 5. Finally, integrate the equation $\ddot{x} = f(x, \dot{x})$ backward in time from the final position until the trajectory from d to h is crossed at point i.

Step 6. If the deceleration trajectory proceeding backward from x_f does not intersect the acceleration trajectory from d to h, then there are more than three switchings. In this case the switching point between d and h is determined as point e is determined in steps 2 and 3, and the algorithm is continued until x_f is reached.

The curve $x_0 edix_f$ is the switching curve. For any initial conditions $(x(0), \dot{x}(0))$ that lie beneath this switching curve, the optimal control policy is to use maximum acceleration until the switching curve is reached and then switch to deceleration and follow the switching curve to x_f.

Some computation can be saved by solving the first-order equations $d\dot{x}/dx = g/\dot{x}$ and $d\dot{x}/dx = f/\dot{x}$ instead of $\ddot{x} = g$ and $\ddot{x} = f$. Also, these first-order equations are more convenient for the more rigorous statement of the algorithm in the Appendix.

Finally, note that the initial and final velocities actually need not be zero. By simply letting the x-axis in Fig. 4 intersect the $\dot{x}$-axis at $\dot{x} = \dot{x}_f$, the desired final

Fig. 6. The actuator torque constraints.

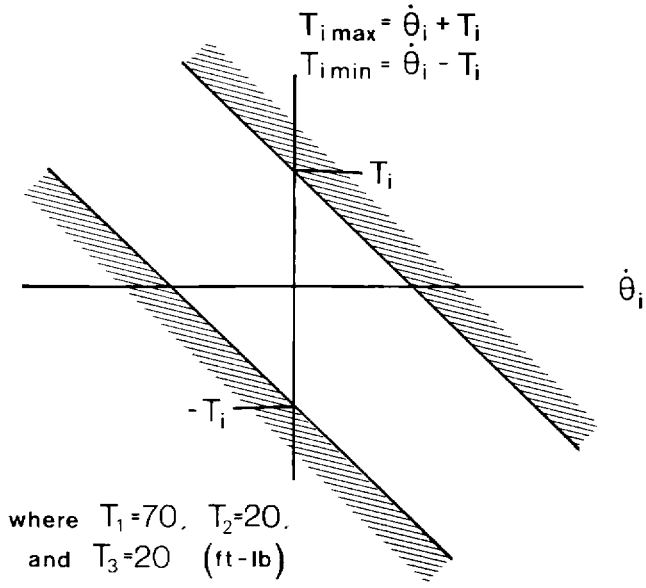
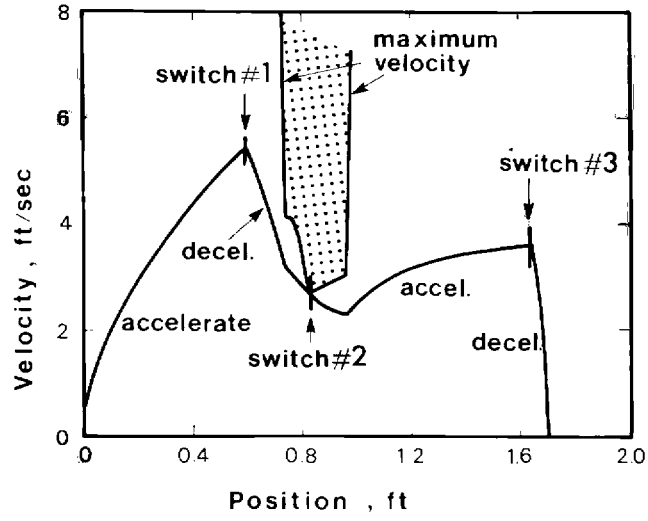


Fig. 7. The time-optimal trajectory for the path in Fig. 5.



velocity along the path, we can construct the switching curve just as above. Therefore, the method can be applied directly, for example, to a problem where an object is to be moved from one moving conveyor to another in minimum time.

2.3. Example 1

For each link of the manipular in Fig. 1, we take the length to be 1 ft and the weight to be 32.2 lb. We require the tip to move along the three-dimensional path in Fig. 5, which consists of two straight lines connected by a circular arc of radius .2 ft. The arc smooths out the corner formed by the intersecting lines P_0P_2 and P_1P_2, where $P_0 = (1', 0, 1')$, $P_1 = (1', 1', 0)$ and $P_2 = (.5', .5', .5')$. The torque constraints are the linear functions of the motor angular velocities shown in Fig. 6.

The minimum-time algorithm given in steps 1–6 yields the maximum velocity and switching curves shown in Fig. 7. Note that the segment of the switching curve just past the second switch indicates that the maximum acceleration $g(x, \dot{x})$ is actually negative for a short time. Figure 8 shows the optimal actuator torques as functions of time.

3. Modeling the Dynamics and Orientation of the End-Effector

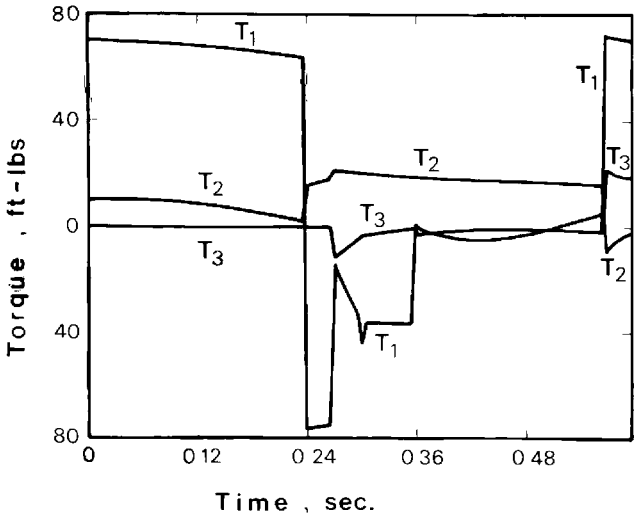
Most industrial manipulators have two or three degrees of freedom for the end-effector in addition to the three degrees of freedom for the links considered so far. In many applications, the dynamics and orientation of the end-effector are significant and must be modeled. The method of Section 2 can include such applications if the orientation of the end-effector is prescribed for each point on the trajectory, as usually is the case with programmed trajectories. We now show how the method of Section 2 applies to this case.

We assume that there are k(1, 2, or 3) additional degrees of freedom for the end-effector, each represented by a joint variable θ_i, and that there is an additional actuator for each degree of freedom, producing a joint torque/force T_i about the θ_i-axis. The θ and T vectors then are expanded to $\theta = (\theta_1, \theta_2, \ldots, \theta_{3+k})^T$ and $\mathrm{T} = (T_1, T_2, \ldots, T_{3+k})^T$.

We assume that on the end-effector there is a point P whose path is specified as in Fig. 1. The 3-vector $\mathbf{r}(\theta)$ contains the coordinates of P as before. The orientation of the end-effector is represented by a k-vector $\mathbf{s}(\theta)$. Since the position of P can be written as a function of θ and of x, we have

$$\mathbf{r}(\theta) = \tilde{\mathbf{r}}(x). \tag{21}$$

Fig. 8. Optimal actuator torque versus time.



Also, the orientation of the end-effector will be prescribed along the path by k scalar equations having the vector form

$$\mathbf{s}(\theta) = \tilde{\mathbf{s}}(x). \tag{22}$$

To generalize the method of Section 2 to include the present case, we expand the vector $\mathbf{r}(\theta)$ by writing

$$\mathbf{R}(\theta) = \begin{pmatrix} \mathbf{p}(\theta) \\ \mathbf{s}(\theta) \end{pmatrix} = \tilde{\mathbf{R}}(x). \tag{23}$$

Then Eq. (23) replaces Eq. (4) and, from here, the development in Section 2 can be retraced easily to see that, after Eq. (4), all the equations hold with the more general $(3 + k)$-vectors θ, $\mathbf{R}(\theta)$, and $\mathbf{T}$.

Recall that when the dynamics and orientation of the end-effector were neglected, the vector $\tilde{\mathbf{r}}_x$ was the unit vector tangent to the path. We now have the more general

$$\tilde{\mathbf{R}}_x = \frac{d\tilde{\mathbf{R}}}{dx} = \begin{pmatrix} \tilde{\mathbf{r}}_x \\ \tilde{\mathbf{s}}_x \end{pmatrix}, \tag{24}$$

where $\tilde{\mathbf{r}}_x$ is the unit vector tangent to the path of P and $\tilde{\mathbf{s}}_x = d\tilde{\mathbf{s}}/dx$. The $\tilde{\mathbf{R}}_x$ of Eq. (24) is now used in Eqs. (5)–(11). The $(3 + k) \times (3 + k)$ Jacobian $[r_\theta]$ will be singular at most at isolated points in θ-space, which, again, are avoided in practice.

Another easy generalization can be useful in applications. The variable x need not be the actual distance along the path; rather, x can represent any parameterization of the path that is continuously differentiable with respect to distance. We need only require that $\tilde{\mathbf{r}}_x = d\tilde{\mathbf{r}}/dx \neq 0$ at each point on the path, so the argument for at least one component of the vector $\mathbf{c}_1$ being nonzero still holds. This is not restrictive for sensible path parameterizations, although generally $\tilde{\mathbf{r}}_x$ will not be a unit vector.

For the generalizations discussed in this section, the time-optimal control algorithm in Section 2.2 and the proof of its optimality in the Appendix are unchanged.

4. Lagrange Multipliers

So far we have assumed that the equations of motion of the manipulator have the form of Eq. (1), where the joint angles θ_i are independent generalized coordinates. In applying our optimal control algorithm, we have found it convenient to use additional coordinates along with Lagrange multipliers. With Lagrange multipliers, the coordinates of the center of mass of the end-effector can be included among the generalized coordinates to make the chore of writing Lagrange's equations more tractable.

4.1. Including the Extra Degrees of Freedom and Constraints

If $\mathbf{z}$ is the vector containing the Cartesian coordinates of the center of mass of the end-effector, we have the holonomic constraints

$$\mathbf{z} = \mathbf{z}(\theta), \tag{25}$$

which result from writing the position vector $\mathbf{z}$ in terms of the joint angles. In general, $\mathbf{z}$ is a 3-vector; however, if (as in Example 3 below) the end-effector moves only in the plane, $\mathbf{z}$ can be a 2-vector. Differentiating Eq. (25) with respect to time, we obtain the nonholonomic form

$$[z_\theta]\dot{\theta} - \dot{\mathbf{z}} = 0, \tag{26}$$

where the matrix $[z_\theta]$ contains the partial derivatives $\partial z_i/\partial\theta_j$. Then, selecting the generalized coordinate vector

$$\mathbf{q} = \begin{pmatrix} \theta \\ \mathbf{z} \end{pmatrix}, \tag{27}$$

we obtain the Lagrange equations

$$\begin{aligned} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + h(\mathbf{q}, \dot{\mathbf{q}}) &= \begin{bmatrix} \mathbf{B}_1 & [\mathbf{z}_\theta]^T \\ 0 & -\mathbf{I} \end{bmatrix} \begin{pmatrix} \mathbf{T} \\ \lambda \end{pmatrix} \\ &= B(\mathbf{q}) \begin{pmatrix} \mathbf{T} \\ \lambda \end{pmatrix}, \end{aligned} \tag{28}$$

where $\mathbf{M}(\mathbf{q})$ is the mass matrix; $h(\mathbf{q}, \dot{\mathbf{q}})$ is a vector that arises from the rotation of the links and from gravity; $\mathbf{T}$ is the vector of actuator torques; and λ is the Lagrange multiplier vector. The term $B_1\mathbf{T}$ is the generalized force corresponding to the torques and forces on the joints.

Since we are assuming an independent actuator for each degree of freedom (i.e., for each θ_i), the matrix B_1 is invertible, and therefore

$$B(\mathbf{q})^{-1} = \begin{bmatrix} B_1^{-1} & B_1^{-1}[z_\theta]^T \\ 0 & -I \end{bmatrix}. \tag{29}$$

This allows us to solve Eq. (28) for $\mathbf{T}$ as a function of q, $\dot{q}$, and $\ddot{q}$. Now, just as before, after using the inverse arm solution to obtain $\theta(x)$, we can use Eqs. (6) and (8) to obtain $\dot{\theta}(x, \dot{x})$ and $\ddot{\theta}(x, \dot{x}, \ddot{x})$. Then Eqs. (25)–(27) show how to compute

$$\mathbf{q} = \mathbf{q}(x), \tag{30A}$$

$$\dot{\mathbf{q}} = \dot{\mathbf{q}}(x, \dot{x}), \tag{30B}$$

and

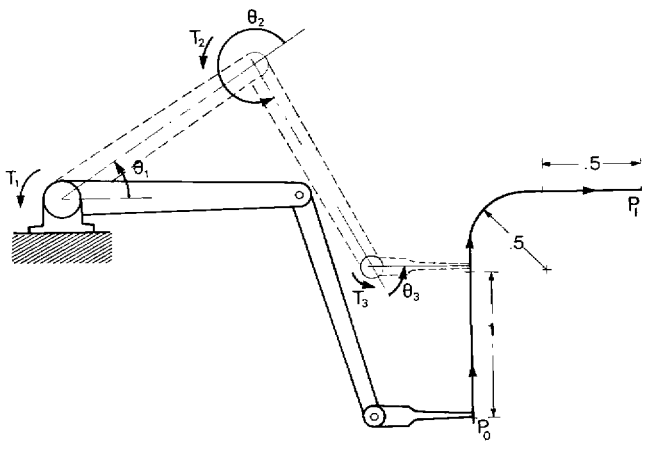
$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}(x, \dot{x}, \ddot{x}). \tag{30C}$$

In particular, $\ddot{\mathbf{q}}$ has the form

$$\ddot{\mathbf{q}} = v(x, \dot{x}) + \ddot{x}\mathbf{w}(x). \tag{31}$$

With these expressions, Eq. (28) can be solved for $\mathbf{T}$ and λ to obtain

Fig. 9. A three-degree-of-freedom planar manipulator.



$$\mathbf{T} = \ddot{x}\mathbf{c}_1(x) + \mathbf{c}_2(x, \dot{x}), \tag{32}$$

which is the same as Eq. (9) except that

$$c_1(x) = [B_1^{-1}B_1^{-1}[z_\theta]^T]\mathbf{M}(q(x))w(x) \tag{33}$$

and

$$\begin{aligned} c_2(x, \dot{x}) = [B_1^{-1}B_1^{-1}[z_\theta]^T](&\mathbf{M}(\mathbf{q}(x))\mathbf{v}(x, \dot{x}) \\ &+ \mathbf{h}(\mathbf{q}(x), \dot{\mathbf{q}}(x, \dot{x}))). \end{aligned} \tag{34}$$

Recall also that $[z_\theta] = [z_\theta(\theta(x))]$.

With Eqs. (32)–(34), the functions $f_i(x, \dot{x})$ and $g_i(x, \dot{x})$ for the time-optimal algorithm are defined as in Eqs. (15) and (16), and the algorithm proceeds just as in Section 2.

4.2. Example 2

This example is a planar problem in which we have added a third link to represent the end-effector. The third link must traverse the path P_0P_1 shown in Fig. 9 while remaining horizontal, perhaps to avoid spilling a payload.

Introducing the Cartesian coordinates of the center of mass of the third link reduces the effort required to write the kinetic energy and derive Lagrange's equations. We will denote these extra coordinates by z_1 and z_2. The generalized coordinate vector then is $q = (\theta_1,$

$\theta_2, \theta_3, z_1, z_2)^T$. In terms of these coordinates, the kinetic energy is

$$\begin{aligned} KE = \tfrac{1}{2}[&J_1\dot{\theta}_1^2 + m_1 I_1'^2\dot{\theta}_1^2 + J_2(\dot{\theta}_1 + \dot{\theta}_2)^2 + m_2 I_1^2\dot{\theta}_1^2 \\ &+ m_2 I_2'^2(\dot{\theta}_1 + \dot{\theta}_2)^2 \\ &+ 2m_2 I_1 I_2'\dot{\theta}_1(\dot{\theta}_1 + \dot{\theta}_2)\cos\theta_2 + m_3(\dot{x}_3^2 + \dot{y}_3^2) \\ &+ J_3(\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3)^2], \end{aligned} \tag{35}$$

and the potential energy is

$$\begin{aligned} PE = m_1 g I_1' \sin\theta_1 + m_2 g(I_2' \sin(\theta_1 + \theta_2) \\ + I_1 \sin\theta_1) + m_3 g y_3. \end{aligned} \tag{36}$$

The constraints in Eq. (25) are

$$\begin{aligned} z_1 &= I_1\cos\theta_1 + I_2\cos(\theta_1 + \theta_2) \\ &\quad + I_3\cos(\theta_1 + \theta_2 + \theta_3), \\ z_2 &= I_1\sin\theta_1 + I_2\sin(\theta_1 + \theta_2) \\ &\quad + I_3\sin(\theta_1 + \theta_2 + \theta_3). \end{aligned} \tag{37}$$

By applying Lagrange's equations to the kinetic and potential energies, we obtain the following nonzero components of the mass matrix $\mathbf{M}(q)$:

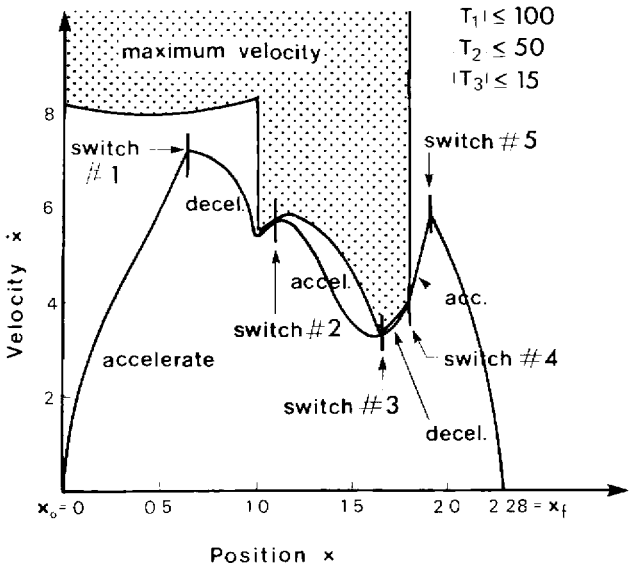
$$\begin{aligned} m_{11} &= J_1 + J_2 + J_3 \\ &\quad + m_2(I_1^2 + I_2'^2 + 2I_1 I_2\cos\theta_2) + m_1 I_1'^2, \\ m_{12} &= m_{21} = J_2 + J_3 + m_2(I_2'^2 + I_1 I_2'\cos\theta_2), \\ m_{13} &= m_{31} = J_3, \\ m_{22} &= J_2 + J_3 + m_2 I_2'^2, \\ m_{23} &= m_{32} = J_3, \\ m_{33} &= J_3, \\ m_{44} &= m_3, \\ m_{55} &= m_3. \end{aligned} \tag{38}$$

The vector $\mathbf{h}(q, \dot{q})$ in Eq. (28) has components

$$\begin{aligned} h_1 &= -m_2 I_2 I_2'\dot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2)\sin\theta_2 \\ &\quad + m_2 g(I_2'\cos(\theta_1 + \theta_2) + I_1\cos\theta_1), \\ h_2 &= m_2 I_1 I_2'\dot{\theta}_1^2\sin\theta_2 + m_2 g I_2'\cos(\theta_1 + \theta_2), \\ h_3 &= h_4 = 0, \\ h_5 &= m_3 g. \end{aligned} \tag{39}$$

Finally, the 5×5 $\mathbf{B}$ matrix is

Fig. 10. The minimum-time trajectory for the path in Fig. 9.



$$\mathbf{B} = \left[\begin{array}{c|cc} & b_{14} & b_{15} \\ I & b_{24} & b_{25} \\ & b_{34} & b_{35} \\ \hline (3\times 3) & & \\ \hline O & -I & \\ (2\times 3) & (2\times 2) & \end{array}\right], \tag{40}$$

where

$$\begin{aligned} b_{14} &= -I_1\sin\theta_1 - I_2\sin(\theta_1 + \theta_2) \\ &\quad - I_3\sin(\theta_1 + \theta_2 + \theta_3), \\ b_{15} &= I_1\cos\theta_1 + I_2\cos(\theta_1 + \theta_2) \\ &\quad + I_3\cos(\theta_1 + \theta_2 + \theta_3), \\ b_{24} &= -I_2\sin(\theta_1 + \theta_2) \\ &\quad - I_3\sin(\theta_1 + \theta_2 + \theta_3), \\ b_{25} &= I_2\cos(\theta_1 + \theta_2) + I_3\cos(\theta_1 + \theta_2 + \theta_3), \\ b_{34} &= -I_3\sin(\theta_1 + \theta_2 + \theta_3), \\ b_{35} &= I_3\cos(\theta_1 + \theta_2 + \theta_3). \end{aligned} \tag{41}$$

For this case, we have the very nice property that $B^{-1} = B$ (this can be seen by computing BB to obtain the identity matrix).

We now have all the terms in Eqs. (32)–(34), so we can apply the time-optimal control algorithm of Section 2.2 to obtain the switching curve.

For this problem, we used constant torque constraints $|T_i| \leq T_{i\max}$, as indicated in Fig. 10, with torque in lb · ft. Figure 10 shows the maximum velocity and switching curves. Note that the maximum velocity curve is discontinuous where the tip passes into and out of the circular arc. This results from the normal acceleration on the arc suddenly appearing and then disappearing. Also note that there are five switchings between maximum acceleration and maximum deceleration in this example. The second switch occurs tangent to the maximum velocity curve after the first discontinuity. It would probably be impossible to find these switching points using maximum principle–based algorithms.

5. Discussion

The preceding results show that the actuator torques required for a typical optimal trajectory are discontinuous functions of time. In practice, the dynamic properties of the actuators make it impossible to produce these torques exactly. For typical direct-current motors used in manipulators, the time constants relating the input voltage to output torques range from 0.0001 sec to 0.025 sec (Kollmorgen Corporation 1983), which is short enough that for most robots these dynamic effects can be neglected (Bobrow 1982). The open-loop input voltage that produces the optimal open-loop torque found with our algorithm can then be computed algebraically. For cases when the actuator time constant is not small, the actuator input voltage can be found by solving an optimal linear quadratic tracking problem (Kirk 1970). The actuator input voltage thus calculated will produce an output torque that is as close as possible to the desired open-loop torque.

Once the open-loop actuator input voltage has been determined, it should drive the manipulator along the desired path if the dynamic model is accurate. However, in most cases this is not a realistic assumption; there is always some uncertainty about the values of the system parameters and unmodeled disturbances. For this reason it is necessary to apply the open-loop voltage in conjunction with feedback. Bobrow (1982) presents a straightforward method for accomplishing this. The method was tested with a complete dynamic simulation of the system, including the motor dynamics. The results showed that in spite of significant errors in the manipulator model, the feedback control did a remarkable job of keeping the manipulator on the required trajectory.

6. Conclusion

We have presented an algorithm for computing the actuator torques that will move the manipulator along a specified path in minimum time, subject to constraints on the torques.

For a specific manipulator, the algorithm requires that

1. the path of the end-effector, including its orientation, is specified;
2. the joint angles can be calculated in terms of the position on the path;
3. the dynamic equations of motion for the manipulator are known;
4. the maximum and minimum torques that can be produced by each actuator are known as functions of the joint angles and angular velocities.

The algorithm is computationally efficient since it involves the numerical integration of only a second-order differential equation, and it requires iteration on only one variable to find the switching curve in the phase plane. While the method can be used to generate the optimal open-loop torques, it also gives the optimal feedback control law for motions along the path in terms of a switching curve in the phase plane.

Appendix

Rules for Constructing a Switching Curve

Before proving rigorously that the algorithm presented in Section 2 does in fact produce the minimum-time control, we must state more precisely the rules for constructing the switching curve. First note that we can write the second-order differential equation $\ddot{x} = g$ in first-order form by the relation $d\dot{x}/dt = (d\dot{x}/dx)(dx/dt) = g$. Thus, the equivalent equations for maximum acceleration and deceleration are

Fig. A1. An optimal trajectory with sets of points labeled.

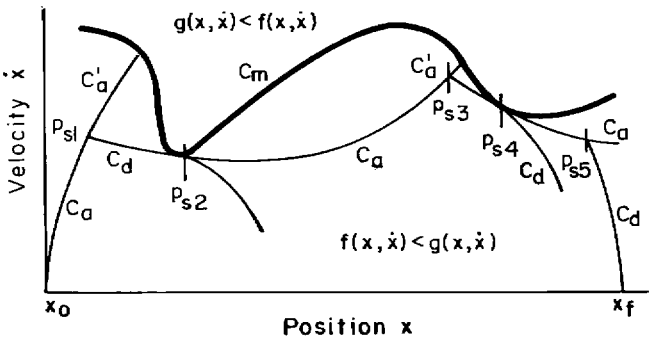
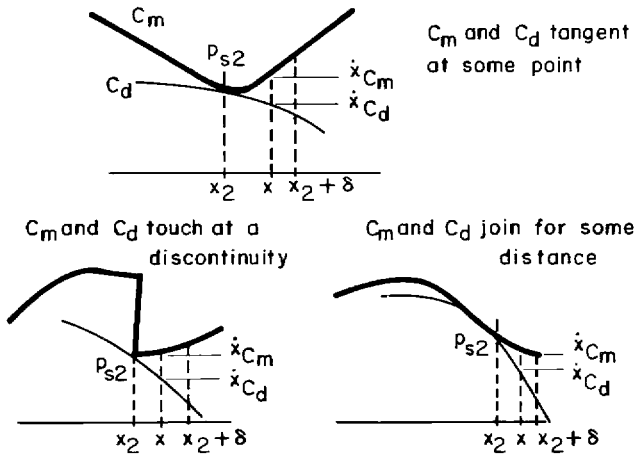


Fig. A2. Three possible occurrences of the second switch.



$$d\dot{x}/dx = g(x, \dot{x})/\dot{x} \qquad \text{acceleration,} \qquad \text{(A1)}$$

$$d\dot{x}/dx = f(x, \dot{x})/\dot{x} \qquad \text{deceleration} \qquad \text{(A2)}$$

When $\dot{x} = 0$, we can obtain the solution to these differential equations by solving $\ddot{x} = g$ and $\ddot{x} = f$ in place of Eqs. (A1) and (A2), respectively. Also, as stated earlier, points on the maximum velocity curve satisfy the relation $f(x, \dot{x}) = g(x, \dot{x})$. Let C_m be the set of these points p:

$$C_m = \{p = (x, \dot{x}): f(x, \dot{x}) = g(x, \dot{x})\}.$$

To construct the optimal trajectory, we will make the following assumptions:

Hypothesis 1: The time required for the manipulator to move from initial condition x_0 to final condition x_f is finite.

Hypothesis 2: There are a finite number of switches between maximum acceleration and maximum deceleration.

We then use the following rules for the construction:

Step A1. Generate the solution to the acceleration equation (A1) from the initial condition $(x_0, \dot{x}_0)$ until the constraint C_m is violated. Let C_a be the set of points on this trajectory (see Fig. A1); i.e.,

$$C_a = \{p = (x, \dot{x}): (x, \dot{x}) \text{ are the solution to Eq. A1) starting from } (x_0, x_0)\}.$$

Step A2. Let

$$C'_a = \{p \in C_a: \text{the solution of Eq. (A2) emanating from } p \text{ intersects } C_m\}.$$

Let $P_{s1} = \min C'_a$ that is, the point in C'_a that has the smallest value for x. [*Note:* The method for choosing P_{s1} requires that the solution C_a touches but does not violate the constraint C_m.]

Step A3. Generate the solution to the deceleration equation A2 starting at P_{s1} and ending some distance beyond the intersection with C_m. Call the set of points p on this trajectory C_d:

$$C_d = \{p = (x, \dot{x}): (x, \dot{x}) \text{ lies on the solution to Eq. (A2) starting from } P_{s1}\}.$$

Step A4. The second switch occurs at P_{s2}, the last (usually only) point that C_m has in common with C_a. That is, p_{s2} satisfies the following two conditions:

(a) $p_{s2} \in C_d \cap C_m$,

(b) for any $\delta > 0$, if $x_2 < x < x_2 + \delta$, then $\dot{x}_{C_m} > \dot{x}_{C_d}$,

where x_2 is the position of p_{s2} and $\dot{x}_{C_m}$ and $\dot{x}_{C_d}$ are the velocities in C_m and C_d corresponding to position x (see Fig. A2).

Step A5. Generate the solution to the acceleration equation (A1) as in step A1, but start from p_{s2}. Redefine C_a as the set of these points:

$$C_a = \{p = (x, \dot{x}): (x, \dot{x}) \text{ are the solution to Eq. (A1) starting from } p_{s2}\}.$$

Step A6. Repeat the entire process to find more switch points until the solution goes beyond x_f (see Fig. A1). This will ensure that when (A2) is solved backward from the final condition, the solution will cross the trajectory constructed.

Proof That the Construction Gives the Minimum-Time Trajectory

To prove that it is possible to switch from deceleration to acceleration at the p_{s2} found in step A4 without immediately violating the constraint, we need the following standard result (Birkhoff and Rota 1969, pp. 24–26).

Theorem 1 Comparison Theorem. Let y and z be solutions of the differential equations

$$y' = F(x, y), \qquad z' = G(x, z),$$

respectively, where $F(x, y) \leq G(x, z)$ in the strip $a \leq x \leq b$ and F or G satisfies a Lipschitz condition. Also let $y(a) = z(a)$. Then $y(x) \leq z(x)$ for all $x \in [a, b]$.

For our problem, the theorem says that for any position x, the velocity corresponding to the solution of the deceleration equation (A2) is less than or equal to the velocity corresponding to the solution of the acceleration equation (A1) for the same initial condition. The theorem is valid in our case because under the maximum velocity curve, $f/\dot{x} < g/\dot{x}$. Also, f and g are piecewise continuously differentiable functions of x, so they satisfy a Lipschitz condition (Birkhoff and Rota 1969). This can be seen by noting how f and g were obtained (see Eqs. 15–19) and observing that it is possible to differentiate these functions except when the Jacobian $[r_\theta]$ is singular (these positions are normally avoided).

We are now able to prove that it is possible to switch from deceleration to acceleration at p_{s2} (Fig. A2).

Theorem 2. The solution to $d\dot{x}/dx = g/\dot{x}$ starting at p_{s2} remains below C_m for some $\delta > 0$.

Proof. First observe that C_d was constructed so that it continues below C_m. Also, below C_m, $f/\dot{x} < g/\dot{x}$, and above C_m, $f/\dot{x} > g/\dot{x}$. If the solution to Eq. (A1) starting from p_{s2} went immediately above C_m, then by the comparison theorem the solution to Eq. (A2) would be above that, since $f/\dot{x} > g/\dot{x}$ above C_m. This cannot happen since C_d (the solution to Eq. A2) is below C_m. Hence the solution to the acceleration equation (A1) must be below C_m for some $\delta > 0$.

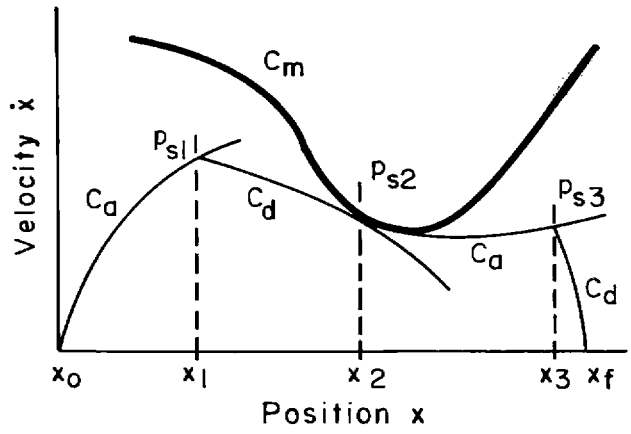


Fig. A3. Switch positions for the minimum-time proof.

Finally, we will prove that the trajectory constructed is the minimum-time solution.

Theorem 3. The solution that moves the manipulator from initial condition x_0, $\dot{x}_0$ to final condition x_f, $\dot{x}_f$ in minimum time is constructed by steps A1–A6.

Proof. Assume first that there are three switches between acceleration and deceleration, at the positions x_1, x_2, and x_3 (see Fig. A3). In the arguments given below, we have implicitly used the result that the solutions to Eqs. (A1) and (A2) are unique. This is true because the functions $f(x, \dot{x})$ and $g(x, \dot{x})$ satisfy a Lipschitz condition.

The time for the maneuver is given by $t = \int_{x_0}^{x_f} dx/\dot{x}$. This integral exists under our hypothesis that it is possible to move from x_0 to x_f in finite time. Assume there is another trajectory that gives a shorter time $t = \int_{x_0}^{x_f} dx/\hat{\dot{x}}$. Then $\hat{\dot{x}} > \dot{x}$ for some $x < x_f$. In the first segment, $x_0 < x < x_1$. This cannot happen because $d\dot{x}/dx < g/\dot{x}$, and, since g is as large as possible, the comparison theorem guarantees no higher solution. We can make the same argument in the last segment by making the change of variable $\hat{x} = x_f - x$ and integrating

backward from the final condition x_f, $\dot{x}_f$. If $\dot{\hat{x}} > \dot{x}$ for $x_1 < x < x_2$ in the second segment C_d, then no solution starting at $\dot{x}$, $\dot{\hat{x}}$ could go beneath C_d before violating the constraint C_m, by the rule for obtaining C_d. Finally, if $\dot{\hat{x}} > \dot{x}$ for some $x_2 < x < x_3$ in the third segment C_a, then this solution starting at the position corresponding to p_{s2} must have a higher velocity since there is no acceleration curve that can cross C_a. This is not possible because the constraint C_m would then be violated. This completes the proof for three switches. When there are more than three, the proof that the intermediate segments are optimal is the same.

References

Birkhoff, G., and Rota, G. 1969. *Ordinary differential equations.* New York: Blaisdell.

Bobrow, J. E. 1982. Optimal control of robotic manipulators. Ph.D. dissertation, University of California, Los Angeles.

Bobrow, J. E., Dubowsky, S., and Gibson, J. S. 1983 (San Francisco). On the optimal control of robotic manipulators with actuator constraints. *Proc. American Control Conference* 2:782–787.

Dubowsky, S., and Desforges, D. T. 1979. The application of model-referenced adaptive control to robotic manipulators. *ASME J. Dyn. Sys., Meas., Contr.* 101:193–200.

Dubowsky, S., and Shiller, Z. 1984 (Udine, Italy). Optimal dynamic trajectories for robotic manipulators. *Proc. V CISM-IFToMM Symp. Theory and Practice of Robots and Manipulators.*

Herrick, S. L. 1982. Model referenced adaptive control of an industrial robot. Master's thesis, University of California, Los Angeles.

Hollerbach, J. M., 1984. Dynamic scaling of manipulator trajectories. *ASME J. Dyn. Sys., Meas., Contr.* 106:102–106.

Kahn, M. E. 1970. The near-minimum time control of open-loop articulated kinematic chains. Ph.D. dissertation, Stanford University.

Kahn, M. E., and Roth, B. 1971. The near-minimum-time-control of open-loop articulated kinematic chains. *ASME J. Dyn. Sys., Meas., Contr.,* Sept., pp. 164–171.

Kirk, D. E. 1970. *Optimal control theory.* Englewood Cliffs, N.J.: Prentice-Hall, pp. 219–227.

Kollmorgen Corporation, Inland Motor Specialty Products Division, 1983. *Direct drive dc motors.* Radford, Va.

Leitmann, G. 1966. *An introduction to optimal control.* New York: McGraw-Hill, pp. 88–97.

Luh, J. Y. S., and Lin C. S. 1981. Optimal path planning for mechanical manipulators. *ASME J. Dyn. Sys., Meas., Contr.* 12:142–151.

Luh, J. Y. S., and Walker, W. M. 1977 (New Orleans). Minimum-time along the path for a mechanical arm. *Proc. 1977 IEEE Conf. Dec. Contr.* 2:755–759.

Lynch, P. M. 1981 (Charlottesville, Va.). Minimum-time, sequential axis operation of a cylindrical, two axis manipulator. *Proc. Joint Automatic Contr. Conf.*

Niv, M., and Auslander, D. M. 1984 (San Diego). Optimal control of a robot with obstacles. *Proc. American Contr. Conf.* 1:280–287.

Ogata, K. 1970. *Modern control engineering.* Englewood Cliffs, N.J.: Prentice-Hall, pp. 98–101.

Paul, R. P. 1981. *Robotic manipulators.* Cambridge: MIT Press, pp. 59–62.

Shin, K. G., and McKay, N. D. 1984 (San Diego). Open-loop minimum-time control of mechanical manipulators and its application. *Proc. American Contr. Conf.,* pp. 296–303.