

**DISTANCE FUNCTIONS AND THEIR APPLICATION  
TO ROBOT PATH PLANNING IN THE  
PRESENCE OF OBSTACLES**

**Elmer G. Gilbert**

**Daniel W. Johnson**

**Department of Electrical Engineering and Computer Science**

**The University of Michigan**

**Ann Arbor, Michigan 48109-1109**

**September 1984**

**CENTER FOR ROBOTICS AND INTEGRATED MANUFACTURING**

**Robot Systems Division**

**COLLEGE OF ENGINEERING**

**THE UNIVERSITY OF MICHIGAN**

**ANN ARBOR, MICHIGAN 48109-1109**

**THE UNIVERSITY OF MICHIGAN  
ENGINEERING LIBRARY**

Enqm  
UMK  
1575

## TABLE OF CONTENTS

1. Introduction .....	3
2. Path Planning as a Problem of Optimal Control .....	6
3. Properties of Distance Functions .....	12
4. Numerical Solution of the Optimal Control Problem .....	19
5. Numerical Solution of the Example .....	25
6. Conclusion .....	32
7. Appendix .....	32
8. References .....	34



**DISTANCE FUNCTIONS AND THEIR APPLICATION TO ROBOT  
PATH PLANNING IN THE PRESENCE OF OBSTACLES<sup>1</sup>**

**ABSTRACT**

An approach to robotic path planning, which allows optimization of useful performance indices in the presence of obstacles, is given. The main idea is to express obstacle avoidance in terms of the distances between potentially colliding parts. Mathematical properties of the distance functions are studied and it is seen that various types of derivatives of the distance functions are easily characterized. The results lead to a general formulation of path planning problems and suggest numerical procedures for their solution. A simple numerical example involving a 3-degree of freedom cartesian manipulator is described.

---

<sup>1</sup>This work was supported by the Air Force Office of Scientific Research, AF Systems Command, USAF under grant F 49620-82-C-0089.

## **1. Introduction**

The generation of robotic motion to accomplish a given task is often broken down into two steps: path planning - the off-line determination of time functions which specify a suitable time dependence of mechanism configuration variables such as manipulator joint angles, path tracking - accurate on-line implementation of the motion through feedback control. Other types of motion control may occur too. Examples include the use of a vision-directed control system for part acquisition at the beginning of a motion, and force control in an assembly operation at the end of a motion. See [1, 2] for more details and references.

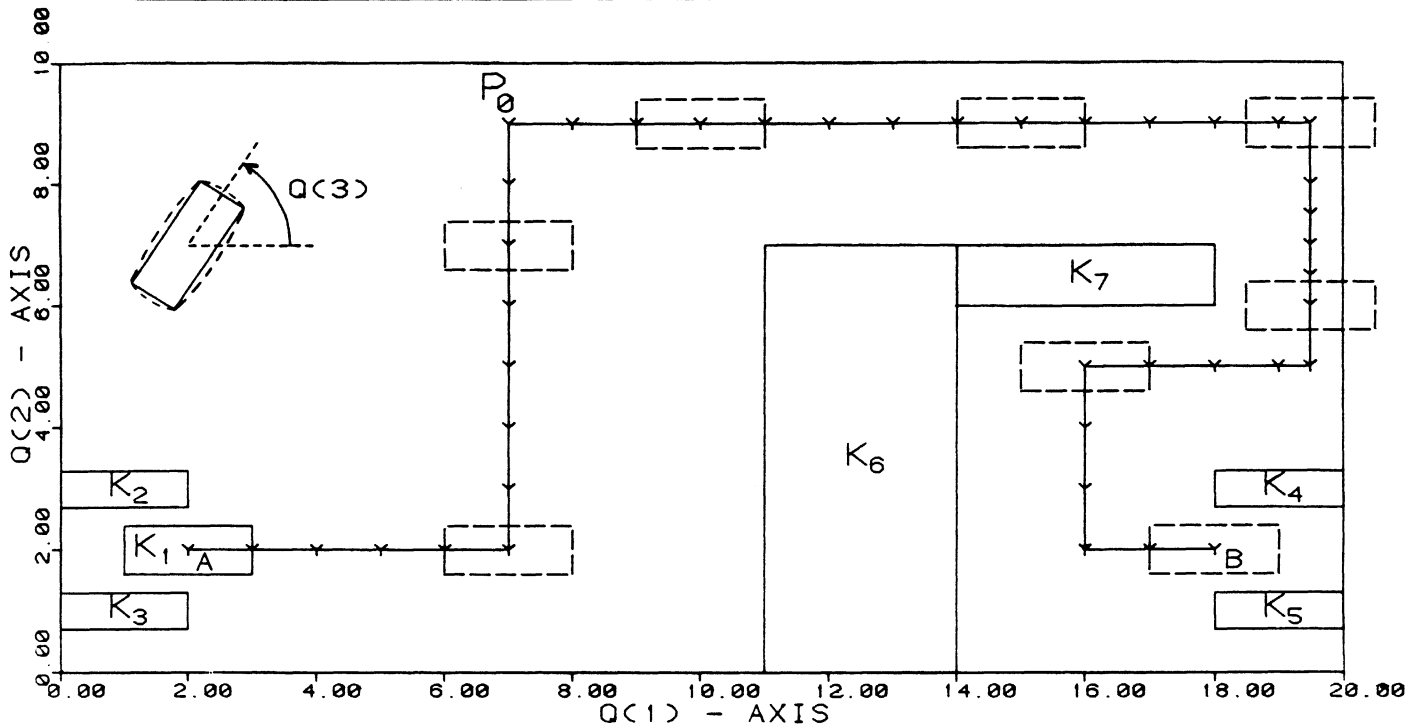
This paper is concerned with the path planning problem. The standard approach is to specify a geometric path (lacking time evolution) as a series of segments in end-effector space. Then smooth, time-dependent functions are fitted to these segments in the space of configuration variables. A crucial aspect of such path planning methods is the implementation of physical constraints. The most important of these are limits on the configuration variables and actuator inputs, and obstacle avoidance. There is a rather large literature on the computation of shortest geometric paths to avoid obstacles (see, e.g., [1, 3, 4, 5] and the literature discussed in these references). But the research to date has limitations in treating multiple-member mechanisms and objects in three dimensional space. The determination of motions to meet actuator constraints is difficult

because of the complexity and nonlinearity of the equations of motion. The result has been a rather conservative approach to the fitting of time functions to geometric paths [1, 2]. Some attempts have been made to optimize motion, but they are in one way or another limited. See, for example [1, 6, 7, 8].

A few examples may give a better understanding of these issues. Figure 1 shows the first, a crane or cartesian manipulator. The payload  $K_1$  is translated and rotated in the plane. There are limits on the translations  $q_1(t)$  and  $q_2(t)$  and  $K_2, \dots, K_7$  are the obstacles. There is no limit on the rotation  $q_3(t)$ . The task is to move the payload from position A to position B and minimize the energy consumed in the move. Determining an admissible motion as a sequence of translations and rotations such as  $P_o$  is not difficult. But determining a motion which meets the control constraints and minimizes the energy is.

The second, more complex, example is the interaction of two manipulators in a common workspace. Again, there are objects  $K_1, K_2, \dots$ : the moving parts (links) of the manipulators and the fixed obstacles in the workspace. During the motion of two manipulators, interference of the objects, including collisions between the moving parts of the two manipulators, must be avoided. Minimization of a suitable performance index, say the time of travel to transfer a part from the gripper of one manipulator to the gripper of the other, would tend to bring the manipulators together. Thus collision avoidance is likely to be an active constraint. The constraint is especially complex because there may be

many pairs of objects which can collide.



**Figure 1. The Cartesian Manipulator with Payload  $K_1$ , and  
Obstacles  $K_2$ - $K_7$ .**

The distances between potentially colliding parts, such as  $K_1$  and  $K_2$  in Figure 1, can be used to express the obstacle constraints. They must all be greater than zero if collision is to be avoided. This observation provides the central focus for this paper. We examine carefully the properties of the distances expressed as functions of the configuration variables. Under certain con-



ditions we find they are differentiable functions and have derivatives which can be expressed by reasonably simple formulas. These results are useful in path planning problems. They lead to an optimal-control formulation and suggest procedures for obtaining numerical solutions of quite general problems.

The plan of the paper is as follows. In Section 2 the optimal control problem is given a precise statement and its connections to the examples are discussed. Section 3 contains the main mathematical results on the continuity and differentiability of distance functions. The results are expressed in a fairly general form and may have other applications such as on-line collision avoidance control schemes [9]. One approach to the numerical solution of the optimal control problem is outlined in Section 4. The problem of numerical solutions has many diverse and difficult aspects and our study of it has just begun. Thus the presentation in Section 4 should not be considered as final or completely authoritative. It does raise critical issues and gives some useful ideas for addressing them. Section 5 shows a few numerical results for the example of Figure 1. Section 6 contains some concluding remarks.

This report is an amplification of results which first appeared in [10].

## **2. Path Planning as a Problem of Optimal Control**

In this section we establish a general framework for path planning problems. It is an optimal control problem with special state constraints expressed

by distance functions.

The following notations and definitions are used:  $x \in R^n$  is a column with components  $x^i$ ,  $A \in R^{m \times n}$  is an  $m$  by  $n$  matrix with elements  $A^{ij}$ ,  $I_n \in R^{n \times n}$  is the identity matrix, the  $'$  denotes transpose,  $\langle x, x \rangle = x'x$  is the inner product on  $R^n$ ,  $|x| = \langle x, x \rangle^{\frac{1}{2}}$  is the norm on  $R^n$ ,  $x \leq 0$  means  $x^i \leq 0$  for  $i=1, \dots, n$ . For  $S_1, S_2 \subset R^n$ ,  $x \in R^n$ ,  $A \in R^{n \times n}$  let:  $S_1 \pm S_2 = \{z_1 \pm z_2 : z_1 \in S_1, z_2 \in S_2\}$ ,  $AS_1 = \{Az_1 : z_1 \in S_1\}$ , and  $S_1 = \{x\}$  denote a singleton. Functions of several variables which have continuous partials of order up to order  $k$  are said to be  $C^k$ .

The optimal control problem involves the minimization of

$$J = L_o(x(0), x(\tau)) + \int_0^\tau L(x(t), u(t)) dt \quad (2.1)$$

subject to the constraints

$$\dot{x}(t) = f(x(t), u(t)), \text{ a. a. } t \in [0, a], \quad (2.2)$$

$$h(x(0), x(\tau)) = 0, \quad (2.3)$$

$$g(x(t), u(t)) \leq 0, \quad t \in [0, \tau]. \quad (2.4)$$

Additional requirements are:  $0 < \tau < a$  where  $\tau$  is either fixed or free and  $a$  is fixed;  $x(\cdot) \in X$ , where  $X$  denotes the set of absolutely continuous functions from  $[0, a]$  into  $R^n$ ;  $u(\cdot) \in U$ , where  $U$  denotes the set of measurable, essentially

bounded functions from  $[0, a]$  into  $R^p$ . It is possible that the components of  $g$  are either functions of  $x$  only or  $u$  only. Thus (2.4) can implement both state and control constraints. The problem data satisfy

Assumption A.1: Let  $n, p, N_h, N_g$  be positive integers. The functions  $L_o: R^n \times R^n \rightarrow R$ ,  $L: R^n \times R^p \rightarrow R$ ,  $f: R^n \times R^p \rightarrow R^n$ ,  $h: R^n \times R^n \rightarrow R^{N_h}$ ,  $g: R^n \times R^p \rightarrow R^{N_g}$  are  $C^1$ .

To this rather general optimal control problem we add constraints corresponding to obstacle avoidance. These constraints require

$$K_i(x(t)) \cap K_j(x(t)) = \phi, \quad t \in [0, \tau], (i, j) \in \mathbf{I}, \quad (2.5)$$

where the sets  $K_i(x) \subset R^m$  describe the space occupied by parts of the robotic mechanism and its environment,  $\mathbf{I}$  is the set of index pairs corresponding to the parts which must not collide and  $\phi$  is the empty set. The sets have the form

$$K_i(x) = T_i(x)C_i + \{p_i(x)\}. \quad (2.6)$$

In applications  $m = 2$  or  $3$ . The sets  $C_i$  characterize the shape of the (rigid) parts; the matrix  $T_i$  and the vector  $p_i$  specify the rotation and translation of the parts due to robotic motion. In applications, the matrices  $T_i$  are orthogonal and, therefore, nonsingular.

It is convenient to describe the requirement (2.5) by the distance between the sets:

$$d_{ij}(x) = \min\{|z_i - z_j| : z_i \in K_i(x), z_j \in K_j(x)\}. \quad (2.7)$$

To make sure (2.5) holds with some margin for error, we impose the conditions

$$d_{ij}^o - d_{ij}(x(t)) \leq 0, \quad t \in [0, \tau], \quad (i, j) \in \bar{I}, \quad (2.8)$$

where the  $d_{ij}^o > 0$  are measures of the margin. The constraint data satisfy

Assumption A.2: Let  $m$  and  $N_o$  be positive integers and  $\bar{I} \subset \{1, \dots, N_o\}^2$ .

The sets  $C_i \subset R^m$ ,  $i = 1, \dots, N_o$ , are compact. The functions  $p_i : R^n \rightarrow R^m$ ,  $T_i : R^n \rightarrow R^{m \times m}$ ,  $i = 1, \dots, N_o$ , are  $C^1$ . For each  $x \in R^n$ ,  $T_i(x)$  is orthogonal,  $i = 1, \dots, N_o$ .

Thus the complete optimal control problem consists of minimizing  $J$  subject to (2.1) - (2.4) and (2.6) - (2.8). The problem data satisfy assumptions A.1 and A.2.

As will be seen from the results of the next section, the properties of  $d_{ij}$  depend on the properties of  $C_i$  and  $C_j$ . In particular, A.2 implies  $d_{ij}$  is Lipschitz continuous but not necessarily differentiable. The lack of differentiability does not upset most results on the existence of optimal solutions (see, e.g. [11]), but it does complicate the application of necessary conditions and numerical optimization techniques. As discussed in Section 3, if either  $C_i$  or  $C_j$  is strictly convex  $d_{ij}$  is continuously differentiable on the domain of interest and the complications are avoided. In specific applications it is often possible to achieve the strict convexity without introducing significant modelling errors.

For instance,  $K_1$  in Figure 1 can be approximated by the slightly bulged set shown by the dashed lines. Moreover, most nonconvex objects, such as  $K_6 \cup K_7$ , can be represented satisfactorily by the union of convex sets.

In this paper we restrict our attention to robotic mechanisms which satisfy dynamic equations of the form

$$M(q(t))\ddot{q}(t) + F(q(t), \dot{q}(t)) = H(q(t))u(t). \quad (2.9)$$

Here:  $q(t) \in R^p$  describes the configuration of the mechanism,  $M(q) \in R^{p \times p}$  is a nonsingular inertia matrix,  $F(q, \dot{q})$  represents a variety of force terms including actuator damping,  $H(q) \in R^{p \times p}$  is nonsingular,  $u(t) \in R^p$  is the actuator input. By choosing  $n = 2p$  and  $x = (q, \dot{q})$  it is clear (2.9) can be written in the form (2.2).

Consider the formulation of the optimal control problem for the example of Figure 1. A fairly accurate dynamic model is

$$m^i \ddot{q}^i(t) + c^i \dot{q}^i(t) = h^i u^i(t), \quad i = 1, 2, 3, \quad (2.10)$$

where the  $m^i$  are the inertias of the (essentially) uncoupled axes, the  $c^i, h^i$  are parameters of DC motor drives, and the  $u^i$  are motor control voltages. Limit constraints on  $q^1(t), q^2(t)$  and  $q^3(t)$ , and on  $u^1(t), u^2(t), u^3(t)$  are imposed by the choice of  $g$ . The beginning and ending points  $A, B$  give  $q(0) = q_A, q(\tau) = q_B$  and  $\dot{q}(0) = \dot{q}(\tau) = 0$ ; these conditions determine  $h$ . The energy required to transfer the payload in a fixed time  $\tau$  is

$$J = \int_0^{\tau} \sum_{i=1}^3 (u^i(t)^2 - b^i u^i(t) \dot{q}^i(t)) dt, \quad (2.11)$$

where the  $b^i > 0$  are motor parameters. Clearly,  $N_o = 7, \bar{I} = \{(1,2), \dots, (1,7)\}$ , and  $m = 2$ . Only  $K_1$  is dependent on  $x; K_i = C_i$  for  $i = 2, \dots, 7$ .

Similar remarks apply to the second example of Section 1. The components of  $q$  are variables such as joint angles which describe the configuration of the two manipulators. Since the dynamics of the two manipulators are not coupled, the matrix and vector functions  $M(q), H(q), F(q, \dot{q})$  have a special, partly uncoupled structure. Of course, the system (2.9) is nonlinear and much more complex than (2.10). In addition, there are constraints of the form (2.3) and (2.4) to implement the desired starting and ending positions and limits on the configuration and control variables (joint torques or forces). For minimum time,  $\tau$  is free and  $L_o \equiv 0, L \equiv 1$ . The index set  $\bar{I}$  is likely to have many elements, particularly if the various links are to be represented by the union of several convex sets.

Our numerical approach produces a smooth approximation to the optimal  $q(t)$  and  $u(t)$ . This is an advantage because they may be used to provide nominal inputs for the manipulator control system. Similarly,  $M(q(t))$  and  $F(q(t), \dot{q}(t))$  are available and may be useful in implementing the feedback control law.

Provided motions satisfying the constraints exist, both of the example problems have optimal solutions. This follows from standard existence theorems.

See [11], Chapter 9.

### 3. Properties of Distance Functions

We now consider the distance between two sets in  $R^m$  which are the images of fixed compact sets under affine mappings. It will be shown that the dependence of the distance on the parameters of the mappings is Lipschitzian, but not necessarily differentiable in the Frechet sense. Clarke [12, 13] has developed a theory of derivatives for Lipschitz continuous functions and we apply it to obtain important properties of the dependence. Our results have obvious connections to the problem of the preceding section.

To simplify the presentation, we abstract the notation of the previous section. For  $i = 1, 2$  let:  $C_i \subset R^m$  be compact,  $p_i \in R^m$ ,  $T_i \in R^{m \times m}$ ,

$$\begin{aligned} K_i(T_i, p_i) &= T_i C_i + \{p_i\} \\ &= \{z: z = T_i w + p_i, w \in C_i\}. \end{aligned} \quad (3.1)$$

The quadruple  $(T_1, p_1, T_2, p_2)$  is written as

$$P = (T_1, p_1, T_2, p_2) \in \mathbb{P} = (R^{m \times m} \times R^m)^2. \quad (3.2)$$

The distance between  $K_1$  and  $K_2$  is

$$d(P) = \min\{|z_1 - z_2|: z_1 \in K_1(T_1, p_1), z_2 \in K_2(T_2, p_2)\}. \quad (3.3)$$

Clearly, the minimum exists and thus  $d: \mathbb{P} \rightarrow R$  is defined by (3.3).

It is worth noting that the compactness of  $C_1$  and  $C_2$  seems to be essential. The example,  $C_1 = \{z: z_2 \geq e^{-z_1}\} \subset R^2, C_2 = \{z: z_2 \leq -1\} \subset R^2, P = (I_2, 0, I_2, 0)$ , shows that the minimum may not exist if  $C_1$  and  $C_2$  are merely closed. Replacing the "min" by "inf" does not solve the problem but leads to another difficulty:  $d$  may be discontinuous. This can be seen by perturbing  $P$  in the example (for instance rotate  $C_2$  about the origin).

Inner products and norms on  $R^{m \times m}$  and  $P$  are introduced in the obvious way. For  $T, \bar{T} \in R^{m \times m}$  and  $P, \bar{P} \in P$ :

$$\begin{aligned} \langle T, \bar{T} \rangle &= \sum_{i,j=1}^m T^{ij} \bar{T}^{ij}, \\ \langle P, \bar{P} \rangle &= \sum_{k=1,2} (\langle T_k, \bar{T}_k \rangle + \langle p_k, \bar{p}_k \rangle). \end{aligned} \tag{3.4}$$

The inner product on  $R^{m \times m}$  has the property that for  $z, w \in R^m$

$$\langle zw', T \rangle = z' Tw. \tag{3.5}$$

The norms are:  $|T| = \langle T, T \rangle^{\frac{1}{2}}, |P| = \langle P, P \rangle^{\frac{1}{2}}$ .

Theorem 3.1 There exists  $M > 0$  such that for all  $P, \bar{P} \in P$

$$|d(P) - d(\bar{P})| \leq M|P - \bar{P}|. \tag{3.6}$$

Proof. Clearly,



$$\begin{aligned}
d(\bar{P}) &= \min\{|\bar{T}_1 w_1 + \bar{p}_1 - \bar{T}_2 w_2 - \bar{p}_2| : w_1 \in C_1, w_2 \in C_2\} \\
&\leq |T_1 \hat{w}_1 + p_1 - T_2 \hat{w}_2 - p_2 + (\bar{T}_1 - T_1) \hat{w}_1 \\
&\quad + (\bar{p}_1 - p_1) - (\bar{T}_2 - T_2) \hat{w}_2 - (\bar{p}_2 - p_2)|
\end{aligned} \tag{3.7}$$

for all  $\hat{w}_1 \in C_1$ ,  $\hat{w}_2 \in C_2$ . Pick  $\hat{w}_1, \hat{w}_2$  to minimize  $|T_1 \hat{w}_1 + p_1 - T_2 \hat{w}_2 - p_2|$ .

Then it follows that

$$\begin{aligned}
d(\bar{P}) &\leq d(P) + \|\bar{T}_1 - T_1\| M_1 + |\bar{p}_1 - p_1| \\
&\quad + \|\bar{T}_2 - T_2\| M_2 + |\bar{p}_2 - p_2|,
\end{aligned} \tag{3.8}$$

where  $M_i = \max\{|w| : w \in C_i\}$ ,  $i = 1, 2$ , and  $\|T\|$  is the spectral norm of  $T$ . Noting  $\|T\| \leq |T|$ ,  $|T_i| \leq |P|$ ,  $|p_i| \leq |P|$  and setting  $M = M_1 + M_2 + 2$  gives  $d(\bar{P}) - d(P) \leq M|\bar{P} - P|$ . Interchanging the roles of  $P$  and  $\bar{P}$  gives  $d(P) - d(\bar{P}) \leq M|P - \bar{P}|$  and thus (3.6) follows.

While  $d$  is uniformly Lipschitz it does not necessarily have a Frechet derivative (gradient)  $\nabla d$ . This can be seen from the sets  $K_1$  and  $K_2$  in Figure 1. If  $K_1$  is rotated clockwise from its indicated position about its center, the distance between  $K_1$  and  $K_2$  changes at a different rate than if it is rotated counterclockwise. The theory of Clarke [12, 13] does apply. Specifically, it will show that  $d$  has a generalized gradient  $\partial d$  and a directional derivative  $D^+ d$ . See the Appendix for definitions of the various derivatives and a summary of results appropriate to what follows.

Let

$$P^+ = \{P \in P : d(P) > 0\}. \quad (3.9)$$

By the continuity of  $d$  it follows that  $P^+$  is an open set. Consider  $\eta: P^+ \times C_1 \times C_2 \rightarrow R$  defined by

$$\eta(P, w_1, w_2) = |T_1 w_1 + p_1 - T_2 w_2 - p_2|. \quad (3.10)$$

Then

$$d(P) = \min\{\eta(P, w_1, w_2) : w_1 \in C_1, w_2 \in C_2\}. \quad (3.11)$$

Note  $\eta \neq 0$  on its domain. Thus,  $\eta$  is differentiable with respect to  $P$ . Specifically, letting

$$\gamma(P, w_1, w_2) = \eta(P, w_1, w_2)^{-1}(T_1 w_1 + p_1 - T_2 w_2 - p_2) \quad (3.12)$$

gives

$$\frac{\partial \eta}{\partial T_j^i}(P, w_1, w_2) = \gamma^i(P, w_1, w_2) w_1^j, \quad (3.13)$$

$$\frac{\partial \eta}{\partial p_1^i}(P, w_1, w_2) = \gamma^i(P, w_1, w_2). \quad (3.14)$$

By collecting together all the partial derivatives of  $\eta$  with respect to the elements of  $P = (T_1, p_1, T_2, p_2)$  we can view  $\nabla_P \eta(P, w_1, w_2)$  as an element of  $P$ .

In particular, it is the quadruple

$$\begin{aligned} \nabla_P \eta(P, w_1, w_2) &= (\hat{\gamma} w_1', \hat{\gamma}, -\hat{\gamma} w_2', -\hat{\gamma}), \\ \hat{\gamma} &= \gamma(P, w_1, w_2). \end{aligned} \quad (3.15)$$

With these notations and those of the Appendix it is now possible to characterize the derivatives of  $d$ .

Theorem 3.2 For all  $P \in \mathbf{P}^+$ ,  $\partial d(P)$  and  $D^+d(P; \cdot)$  exist and are given by

$$\partial d(P) = \text{co}\{(\hat{\gamma}w_1', \hat{\gamma}, -\hat{\gamma}w_2', -\hat{\gamma}) : \hat{\gamma} = \gamma(P, w_1, w_2), (w_1, w_2) \in W(P)\}, \quad (3.16)$$

$$\begin{aligned} D^+d(P; \tilde{P}) &= \min\{\langle \nabla_P \eta(P, w_1, w_2), \tilde{P} \rangle : (w_1, w_2) \in W(P)\} \\ &= \min\{\gamma(P, w_1, w_2)' (\tilde{T}_1 w_1 + \tilde{p}_1 - \tilde{T}_2 w_2 - \tilde{p}_2) : (w_1, w_2) \in W(P)\} \end{aligned} \quad (3.17)$$

where

$$W(P) = \{(w_1, w_2) \in C_1 \times C_2 : \eta(P, w_1, w_2) = d(P)\}. \quad (3.18)$$

Proof. It is only necessary to note (3.11) and use Results 3 and 4 of the Appendix with the following substitutions:  $P \rightarrow x, (w_1, w_2) \rightarrow u, \mathbf{P}^+ \rightarrow \mathbf{O}, C_1 \times C_2 \rightarrow \mathbf{U}, \eta \rightarrow -g, d \rightarrow -f, \tilde{P} \rightarrow v$ . The required conditions on  $g$  and  $\nabla_x g$  are verified immediately from (3.10), (3.12), and (3.15). The second formula for  $D^+d$  follows from (3.5) and (3.15).

Theorem 3.3 Suppose  $P \in \mathbf{P}^+$  and

$$W(P) = \{(\psi_1(P), \psi_2(P))\} = \text{singleton}. \quad (3.19)$$

Then  $d$  has a gradient at  $P$  which is given by

$$\nabla d(P) = \nabla_P \eta(P, \psi_1(P), \psi_2(P)). \quad (3.20)$$

Moreover, for  $\tilde{P} \in \mathbf{P}$

$$\begin{aligned} \langle \nabla d(P), \tilde{P} \rangle = & \gamma(P, \psi_1(P), \psi_2(P))' (\tilde{T}_1 \psi_1(P) \\ & + \tilde{p}_1 - \tilde{T}_2 \psi_2(P) - \tilde{p}_2). \end{aligned} \quad (3.21)$$

Proof. Use Result 1 of the Appendix with the same notation as used in the proof of Theorem 3.2.

Some simplifications in the above results occur if convexity assumptions are added.

Theorem 3.4 Suppose  $C_1$  and  $C_2$  are convex. Then for all  $P \in \mathbb{P}$ : (C1)  $W(P)$  is convex, (C2)  $\gamma(P, w_1, w_2)$  is independent of  $(w_1, w_2)$  for  $(w_1, w_2) \in W(P)$ .

Proof. Result C1 follows because  $W(P)$  is the set of minimizers for a convex function  $(\eta(P, \cdot, \cdot))$  on a convex set  $(C_1 \times C_2)$ . It is clear that for  $(w_1, w_2) \in W(P)$ ,  $\gamma(P, w_1, w_2) = |z|^{-1}z$  where  $z$  is a minimizer of  $|z|$  on  $K_1(T_1, p_1) - K_2(T_2, p_2)$ . Since there is only one element of minimum norm on a closed convex set, result C2 follows.

Because of C2 it is possible to write

$$\gamma(P, w_1, w_2) = \gamma_c(P), (w_1, w_2) \in W(P). \quad (3.22)$$

From (3.15) with  $\hat{\gamma} = \gamma_c(P)$  this shows that  $\nabla_P \eta(P, \cdot, \cdot)$  is affine on  $C_1 \times C_2$ .

Hence (3.16) and (3.17) are simplified:

$$\partial d(P) = \{(\hat{\gamma} w_1', \hat{\gamma}, -\hat{\gamma} w_2', -\hat{\gamma}) : \hat{\gamma} = \gamma_c(P), (w_1, w_2) \in W(P)\} \quad (3.23)$$

$$D^+d(P;\tilde{P}) = \min\{\gamma_c(P)' (\tilde{T}_1 w_1 + \tilde{p}_1 - \tilde{T}_2 w_2 - \tilde{p}_2) : (w_1, w_2) \in W(P)\}. \quad (3.24)$$

Note  $D^+d(P;\tilde{P})$  is obtained by minimizing an affine function on the convex set  $W(P)$ .

With still stronger conditions  $d$  is  $C^1$ . Let

$$P_N^\dagger = \{P \in P^+ : T_1, T_2 \text{ are nonsingular}\}. \quad (3.25)$$

Since  $\det T_i, i = 1, 2$ , is continuous on  $R^{m \times m}$  it follows that  $P_N^\dagger$  is an open set.

Theorem 3.5 Suppose the sets  $C_1, C_2$  are both convex and at least one is strictly convex. Then: (SC1)  $W(P)$  is a singleton for all  $P \in P_N^\dagger$ , (SC2)  $d$  restricted to  $P_N^\dagger$  is  $C^1$ .

Proof. Conclusion SC2 is a consequence of SC1 and Result 2 of the Appendix. Now consider the proof of SC1. For  $P \in P^+$  it is easily shown that

$$\nabla_w \eta(P, w_1, w_2) = -(-1)^i T_i' \gamma(P, w_1, w_2). \quad (3.26)$$

From this  $\nabla_w \eta(P, w_1, w_2) \neq 0, i = 1, 2$ , for  $P \in P_N^\dagger$  and  $(w_1, w_2) \in C_1 \times C_2$ . Since  $(w_1, w_2) \in W(P)$  solves (3.11),  $(w_1, w_2) \in W(P)$  implies  $w_i \in \text{boundary } C_i, i=1, 2$ .

Assume, contrary to SC1, that  $(\bar{w}_1, \bar{w}_2) \neq (\tilde{w}_1, \tilde{w}_2)$  are both in  $W(P)$ . It follows from  $T_1$  and  $T_2$  nonsingular and C2 that  $\bar{w}_1 \neq \tilde{w}_1$  and  $\bar{w}_2 \neq \tilde{w}_2$ . From C1,  $\lambda \bar{w}_i + (1 - \lambda) \tilde{w}_i \in \text{boundary } C_i, i = 1, 2$  and  $0 \leq \lambda \leq 1$ . This contradicts the

assumption that either  $C_1$  or  $C_2$  is strictly convex.

Extensions of the above results are possible. First, Clarke's theory may be applied to the characterization of  $d$  on all of  $P$ . This is because  $\eta(P, w_1, w_2)$  is Lipschitz at  $\eta = 0$ , even though it doesn't have a Frechet derivative at  $\eta = 0$ . However, the details are more complex and of no value in the collision avoidance problem where we require  $d(P) > 0$ . Second, if  $T_2$  and  $p_2$  are fixed, Theorems 3.1-3.5 still apply to dependence of  $d$  on  $T_1, p_1$ , even though  $C_2$  is only closed. A more complex method of proof must be used and, again, the result has limited practical value. In the real world fixed obstacles are of finite extent.

#### 4. Numerical Solution of the Optimal Control Problem

Numerical solution of the optimal control problem (2.1)-(2.4), (2.6)-(2.8) is a complex task which raises a number of questions. These include: (1) approximation of the infinite dimensional system by a finite dimensional system, (2) imposition of the equality constraints (2.2) and (2.3), (3) treatment of the inequality constraints (2.4) and (2.8), (4) choice of the descent algorithm, (5) efficient calculation of the various functions and their derivatives, (6) determination of a good initial approximate solution which is feasible, (7) achievement of reasonable computational cost. It is not clear what algorithmic approach addresses these questions in the best way. In this section we present one approach which appears to have a number of advantages. It assumes the system dynamics are

described by (2.9) and the function  $h$  is affine. This permits the solution of a wide class of path planning problems, including the examples discussed in Sections 1 and 2. To keep the notation reasonably simple and to emphasize the conceptual issues, the approach is described in general terms. Actual implementation of the approach involves attention to important, complex details.

Questions (1) and (2) are resolved together in the following way. The state  $x(t)$  is approximated by restricting it to a linear subspace of  $X$ . Specifically, the approximation is

$$q(t, \alpha) = \sum_{i=1}^K \alpha^i \phi_i(t) \quad (4.1)$$

where  $\alpha \in R^K$  and the  $\phi_i$  are basis functions for the subspace. It is assumed that the  $\phi_i$  are at least twice differentiable. The most natural choice for the  $\phi_i$  are piecewise polynomial splines of limited support [14]. The equality constraint (2.9), which is another way of writing (2.2), is satisfied by choosing  $u(t)$  equal to

$$u(t, \alpha) = H^{-1}(q(t, \alpha))(M(q(t, \alpha))\ddot{q}(t, \alpha) + F(q(t, \alpha), \dot{q}(t, \alpha))). \quad (4.2)$$

The functions  $\dot{q}(t, \alpha)$  and  $\ddot{q}(t, \alpha)$  have obvious expressions in terms of  $\dot{\phi}_i$  and  $\ddot{\phi}_i$ , which are easily computed if the  $\phi_i$  are splines. Moreover, for mechanical manipulators there are efficient recursive procedures [15, 16] for evaluating the left side of (4.2). Specifying the finite dimensional approximations of  $q(t), \dot{q}(t)$  and  $u(t)$  in the above way provides an exact solution of (2.9), avoiding numerical integration of (2.9). By defining  $x(t, \alpha) = (q(t, \alpha), \dot{q}(t, \alpha))$  it is clear  $x(t, \alpha), u(t, \alpha)$

solves (2.2). The integration of (2.1) still requires a numerical scheme, but a simple quadrature formula suffices. Similar ideas have appeared previously in the literature. See, e.g. [17].

Now consider the constraint (2.3). Since  $h$  is affine,

$$h(x(0), x(\tau)) = H_{11}q(0) + H_{12}\dot{q}(0) + H_{21}q(\tau) + H_{22}\dot{q}(\tau) + h_o = 0, \quad (4.3)$$

where  $h_o \in R^{N_k}$ ,  $H_{ij} \in R^{N_k \times p}$ . Setting  $q(t) = q(t, \alpha)$  in (4.3) yields an affine constraint on  $\alpha$ . Provided the  $\phi_i$  and  $H_{ij}$  satisfy reasonable assumptions the constraint on  $\alpha$  has a solution which can be expressed by

$$\alpha = \Pi\bar{\alpha} + \pi, \quad (4.4)$$

where  $\bar{\alpha} \in R^{K-N_k}$ ,  $\pi \in R^K$ ,  $\Pi \in R^{K \times (K-N_k)}$ . Thus (2.3) is satisfied by expressing  $\alpha$  in terms of  $\bar{\alpha}$  by (4.4).

Having obtained exact solutions of the (more troublesome) equality constraints, we accept approximate implementation of the inequality constraints. The approach is to use interior penalty functions [18, 19]. This causes the constraints (2.4), (2.8) to be satisfied, but increases  $J$  somewhat because the penalty functions force the solution away from the constraint boundaries. The penalty function approach requires the constraint set to have an interior. This is a physically reasonable condition for our problem.

The penalized cost has the form



$$\bar{J}(\alpha, \mu) = L_o(x(0, \alpha), x(\tau, \alpha)) + \int_0^{\tau} \bar{L}(x(t, \alpha), u(t, \alpha), \mu) dt \quad (4.5)$$

where

$$\begin{aligned} \bar{L}(x, u, \mu) = & L(x, u) + \mu \sum_{i=1}^{N_f} \beta_i (-g_i(x, u)) \\ & + \mu \sum_{(i,j) \in I} \beta_{ij} (d_{ij}(x) - d_{ij}^o). \end{aligned} \quad (4.6)$$

The functions  $\beta_i$  and  $\beta_{ij}$  have the following properties:  $\beta: (0, \infty) \rightarrow R$  is  $C^1$ ,  $\beta(e) > 0$  and  $\frac{\partial \beta}{\partial e}(e) \leq 0$  for all  $e$ ,  $\beta(e) \rightarrow \infty$  as  $e \rightarrow 0$ . An example of such a function is

$$\beta(e) = \bar{\beta} (e\bar{e}^{-1} + e^{-1}\bar{e} - 2), \quad 0 < e < \bar{e}, \quad \beta(e) = 0, \quad e \geq \bar{e}, \quad (4.7)$$

where the numbers  $\bar{e}, \bar{\beta} > 0$  measure respectively the point where the penalty becomes active and the strength of the penalty. The parameter  $\mu > 0$  is selected in the usual way to establish a trade-off between bad conditioning of  $\bar{J}$  and acceptably small degradation of  $J$ .

Finally,  $\bar{J}$  must be integrated numerically. This gives  $\bar{J}(\alpha, \mu) \approx \tilde{J}(\alpha, \mu)$ , where

$$\tilde{J}(\alpha, \mu) = L_o(x(0, \alpha), x(\tau, \alpha)) + \sum_{i=0}^{\nu} \sigma_i \bar{L}(x(t_i, \alpha), u(t_i, \alpha), \mu) \quad (4.8)$$

Here, the  $t_i, \sigma_i$  determine the "time grid" and the method of numerical

integration. It is important that the  $t_i$  be close together, particularly where the penalty terms are changing rapidly. Otherwise, the penalty terms in  $\bar{L}$  may fail to enforce the inequalities (2.4), (2.8). On the other hand, the computational cost grows almost linearly with  $\nu$ . At a given  $t = t_i$  only a few of the constraints (2.8) are likely to be active. Therefore, many calculations can be eliminated by having a simple scheme which omits evaluation of  $d_{ij}(x(t, \alpha))$  when the sets  $K_i$  and  $K_j$  are widely separated.

The last step in the overall solution process is the numerical minimization of  $\hat{J}(\bar{\alpha}) = \tilde{J}(\Pi\bar{\alpha} + \pi, \mu)$ . From the preceding section it follows that  $\hat{J}$ , which depends on  $d_{ij}(x)$ , is Lipschitz but may not be  $C^1$ . The numerical minimization of such functions presents both theoretical and practical difficulties [20, 21, 22]. The resulting algorithms tend to be complicated, since they involve a smoothing of the generalized gradient in order to determine a direction of descent. For our problem it seems much simpler to carry out the "smoothing" directly by using the suggestion of Section 2: Assume the  $C_i, i = 1, \dots, N_o$ , are convex and at least one of the sets  $C_i, C_j$  is strictly convex for each  $(i, j) \in \bar{I}$ . By Theorem 3.5 this causes the  $d_{ij}(x)$  and hence  $\hat{J}$  to be  $C^1$ . Then standard minimization algorithms, which have excellent convergence properties, may be used.

To begin the minimization an initial choice of  $\bar{\alpha}$  must be made. Because of the interior penalty function, the corresponding initial path must satisfy strictly

the inequalities (2.4) and (2.8). In many problems, an admissible path is evident from the geometry; alternatively, one may be generated algorithmically by the methods of [3, 4]. Once an admissible path is known,  $\bar{\alpha}$  can be determined by a spline fit of the path (this another advantage of doing the computations in terms of  $q(t)$  rather than  $u(t)$ ). For example, in Figure 1 there is an obvious sequence of translations and rotations which achieve a transfer from A to B.

Efficient minimization algorithms also require the evaluation of  $\nabla \hat{J}$ . Formulas for  $\nabla \hat{J}$  are obtained by applying the chain rule of differentiation to the series of expressions which define  $\hat{J}$ . We omit the details, but comment on the terms which are computationally most expensive: the derivatives of  $d_{ij}(x)$  and  $u(t, \alpha)$ .

Application of (3.21) gives

$$\begin{aligned} \frac{\partial d_{ij}}{\partial x^k}(\bar{x}) &= |\bar{z}_i^* - \bar{z}_j^*|^{-1} (\bar{z}_i^* - \bar{z}_j^*) \left( \frac{\partial T_i}{\partial x^k}(\bar{x}) \bar{w}_i^* \right. \\ &\quad \left. + \frac{\partial p_i}{\partial x^k}(\bar{x}) - \frac{\partial T_j}{\partial x^k}(\bar{x}) \bar{w}_j^* - \frac{\partial p_j}{\partial x^k}(\bar{x}) \right). \end{aligned} \quad (4.9)$$

Here,  $\bar{z}_i^*, \bar{z}_j^*$  are the (unique) minimizers from (2.7) at  $x = \bar{x}$  and  $\bar{w}_l^* = T_l^{-1}(\bar{x})(\bar{z}_l^* - p_l(\bar{x})), l = i, j$ . All the terms which show explicit dependence on  $\bar{x}$  are rather simple functions of  $q$  and  $T_l^{-1}(\bar{x}) = T_l'(\bar{x}), l = i, j$ . Thus, the greatest potential for computational expense is the solution of (2.7). There are a variety of approaches ranging from the use of quadratic program-

ming algorithms to the derivation of explicit solution formulas. Several of the approaches are being investigated for numerical efficiency, along with procedures for approximating convex sets by strictly convex sets.

The most difficult computation in the differentiation of (4.2) with respect to  $\alpha$  is the differentiation of  $M(q)$  and  $F(q, \dot{q})$  with respect to the components of  $q$  and  $\dot{q}$ . For some systems, such as the one in Figure 1, the computation is easy. General manipulator dynamics offer a greater challenge. Some interesting methods for treating them have been developed and will be reported in a later paper.

It is worth noting that the computations described in the preceding two paragraphs have exactly the same form at each  $t_i$ . Moreover, the distance minimizing problem for each  $(i,j) \in \bar{I}$  are independent of each other. Thus it would be possible to greatly speed the computations by using a computer with parallel processors.

## 5. Numerical Solution of the Example

To test the ideas of Section 4 the system in Figure 1 was solved numerically for several cases. The following paragraphs give the details.

The dynamics and cost are given by (2.10) and (2.11) with  $c^i = b^i = h^i = 1, i = 1,2,3$ , and  $m^1 = m^2 = 1$ . There are six cases as summarized in Table 1. The constraints (2.3) and (2.4) correspond to:

$$q(0)=(2,2,0), q(\tau)=(18,2,0), \dot{q}(0) = \dot{q}(\tau) = 0, 0 \leq q^1(t) \leq 20, 0 \leq q^2(t) \leq 10.$$

The payload  $K_1$  is .8 units wide and 2 units long. It is made strictly convex by bulging its sides slightly into circular arcs of radius 100. The distance minimizing problems are solved easily so that their solutions can be expressed explicitly by relatively simple formulas.

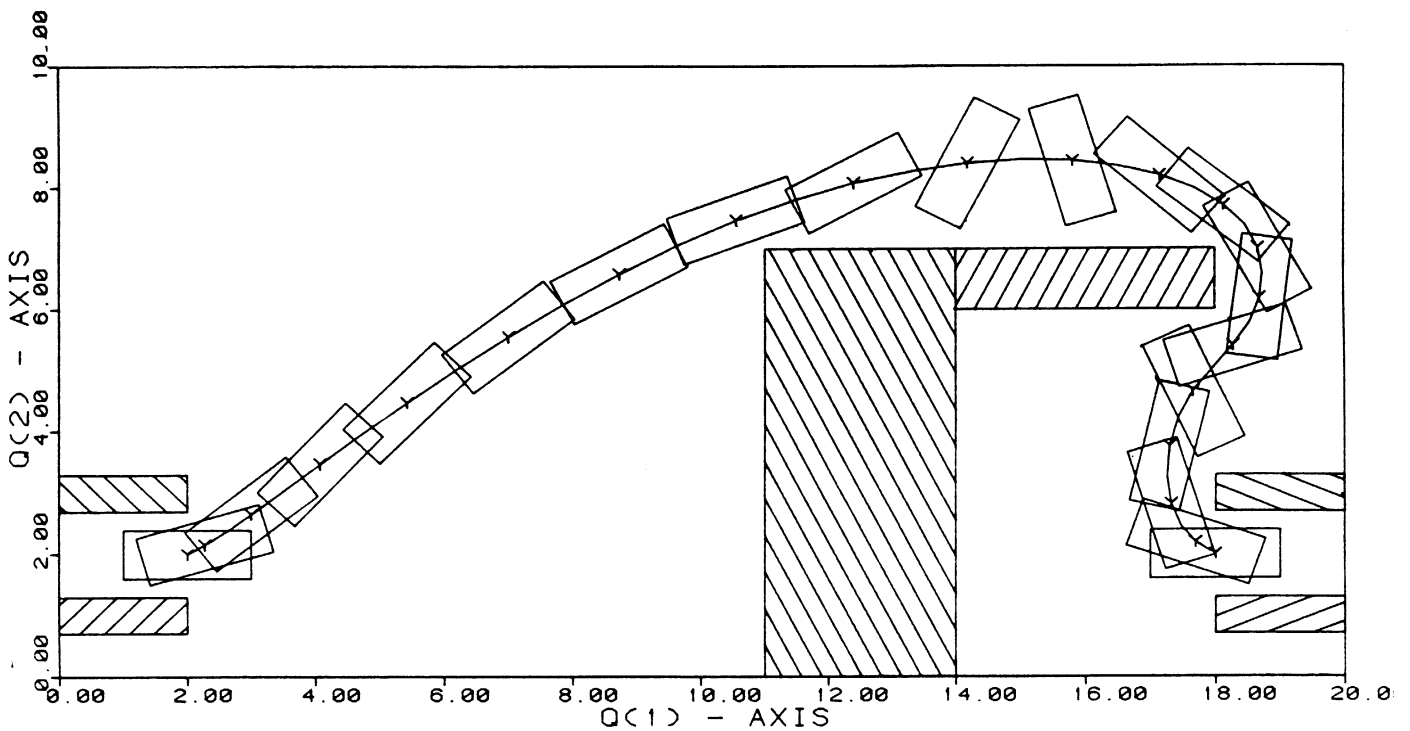
The functions  $\phi_i(t)$  represent a cubic spline basis for  $q^1(t), q^2(t), q^3(t)$ ; specifically, the components of  $\phi_i(t)$  are normalized B-splines of order 4 [14] with the knots producing 40 equal intervals in  $[0, \tau]$ . Thus  $K = 3 \times 43$  and  $\dim \bar{\alpha} = 3 \times 39 = 117$ . The  $\beta_i$  and  $\beta_{ij}$  in (4.6) have the form (4.7) with  $\bar{\epsilon} = .2$  and  $\bar{\beta} = 1$ . The remaining parameters in (4.6) are  $d_{ij}^g = .1$  and  $\mu = .1$  for all cases. The formula (4.8) was obtained from the composite Simpson's rule where the time grid partitions each spline knot interval into  $\Gamma$  equal sections, where  $\Gamma = 8$ . Points along the piecewise rectilinear path  $P_o$  in Figure 1 were used to generate an initial spline for the optimization algorithm. A BFGS - type algorithm [23] was used to carry out the minimization. Typically,  $\nabla \hat{J} \approx 0$  was obtained in about 300-600 evaluations of  $\hat{J}$  and  $\nabla \hat{J}$ . Smaller values of  $\mu$  ( $\mu = 10^{-3}, 10^{-5}$ ) were also tried in several cases. The result was a somewhat improved cost (approximately 10%) and a closer approach to the obstacles. The poorer conditioning of the penalized cost did not affect the number of iterations significantly, although convergence did require the use of a better initial path than  $P_o$ .

The resulting paths for cases 1 through 6 are indicated in Figures 2 through 7. The energies of the initial and final paths as obtained in the computational process are given in Table 1. The optimization produces appreciable reductions in cost. This is no doubt due to the rather abrupt starting and stopping along the initial path. As expected, the optimal cost decreases as either  $\tau$  increases or  $m^3$  decreases.

The use of Simpson's rule ensures that the actual cost, being a piecewise cubic polynomial in this example, is integrated exactly for all the above cases. However, the penalized portion of the cost is not, and the distance constraints (2.8) may be violated at values of  $t$  between grid points. This was observed to happen when  $\Gamma = 1$ . The paths obtained in [10] had this difficulty, although it was not detected at the time [10] was written. A plot of distance versus time for each object pair in the index set  $\mathbf{I}$  is given in Figure 8 for case 1.

**TABLE 1. NUMERICAL RESULTS**

CASE	$\tau$	$m^3$	INITIAL COST	FINAL COST
1	20.0	0.25	214.75	3.8374
2	20.0	1.0	214.75	4.9214
3	20.0	4.0	214.75	5.6072
4	20.0	16.0	214.75	6.0235
5	40.0	0.25	26.84	0.5030
6	40.0	1.0	26.84	0.6397

**Figure 2. Final Path for Case 1.**

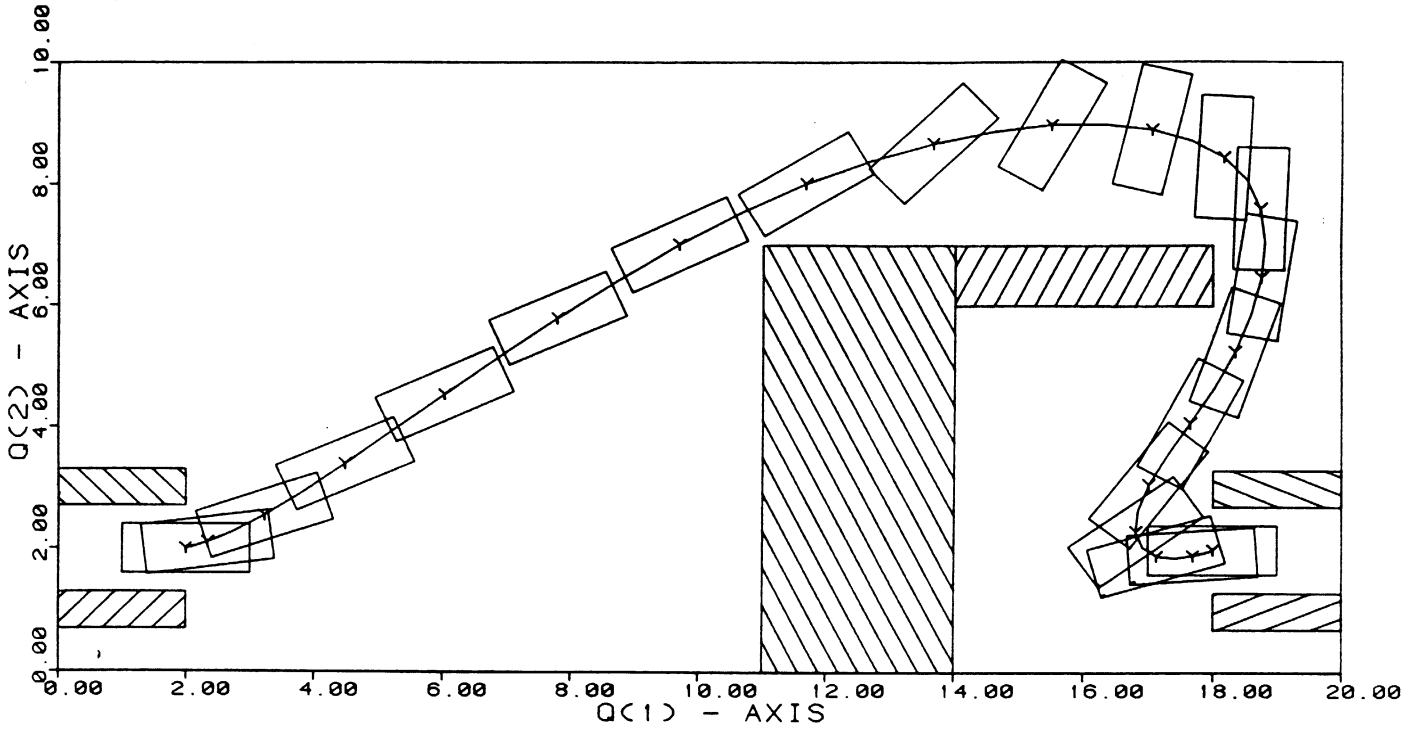


Figure 3. Final Path for Case 2.

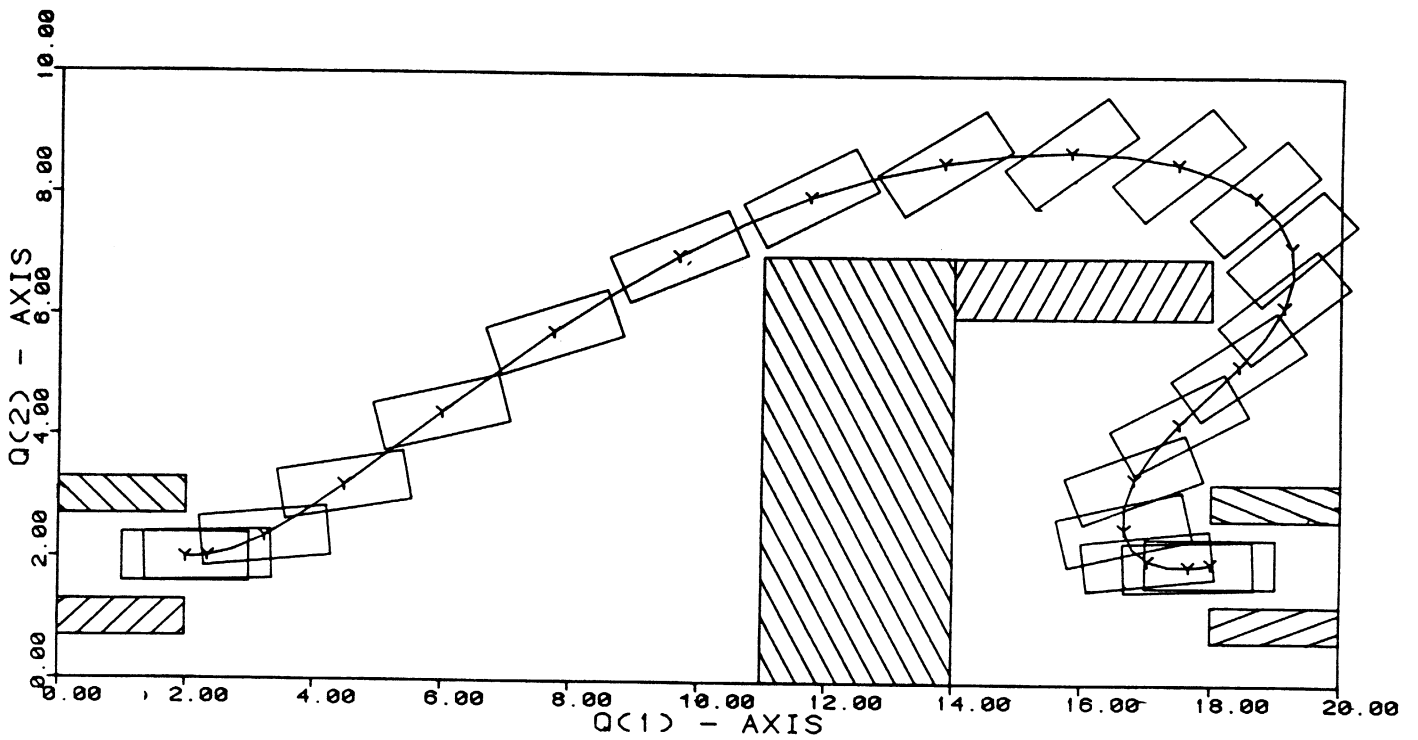


Figure 4. Final Path for Case 3.



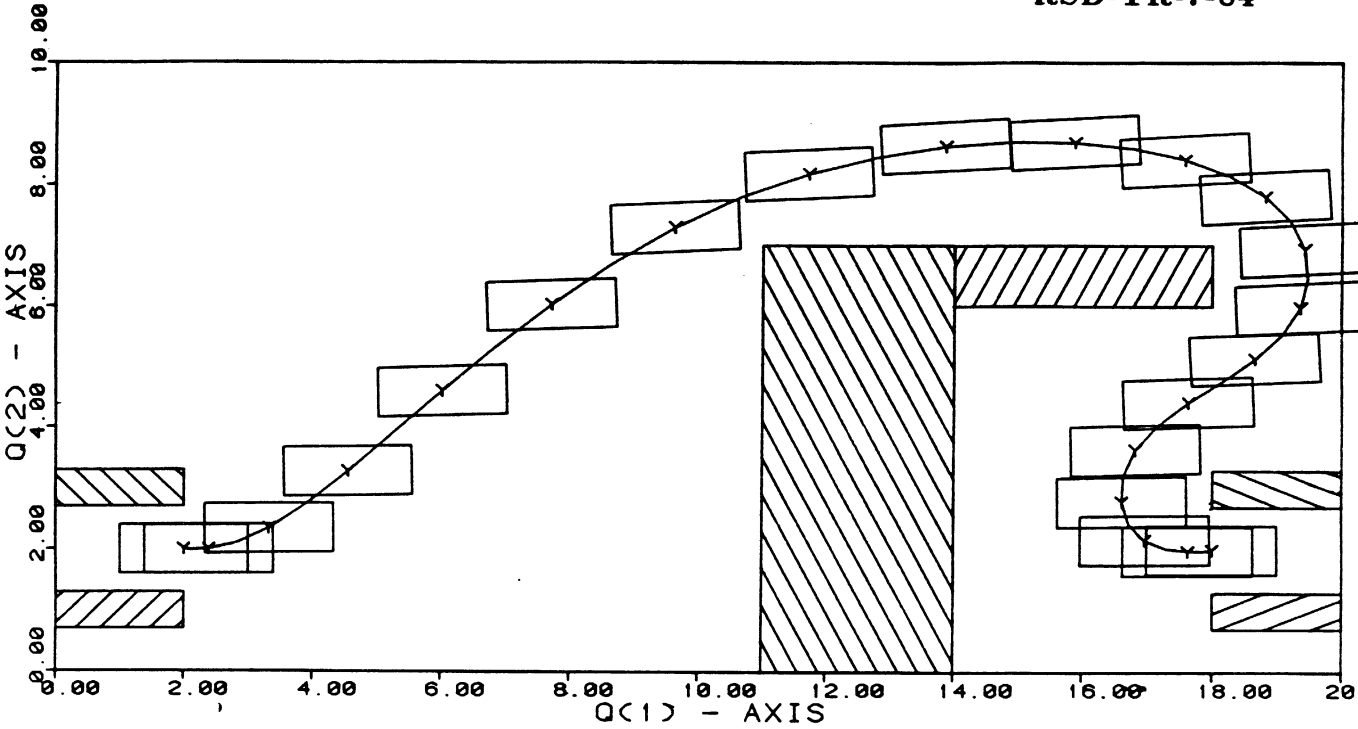


Figure 5. Final Path for Case 4.

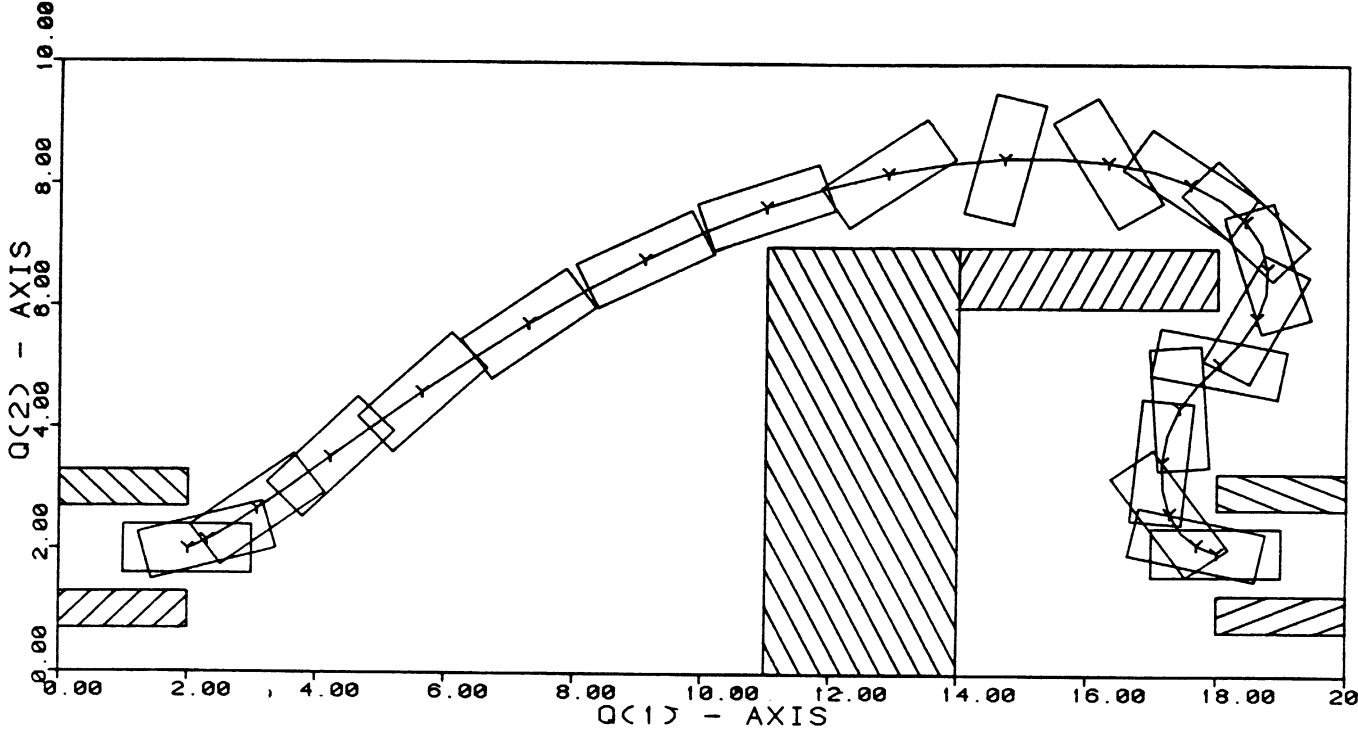


Figure 6. Final Path for Case 5.

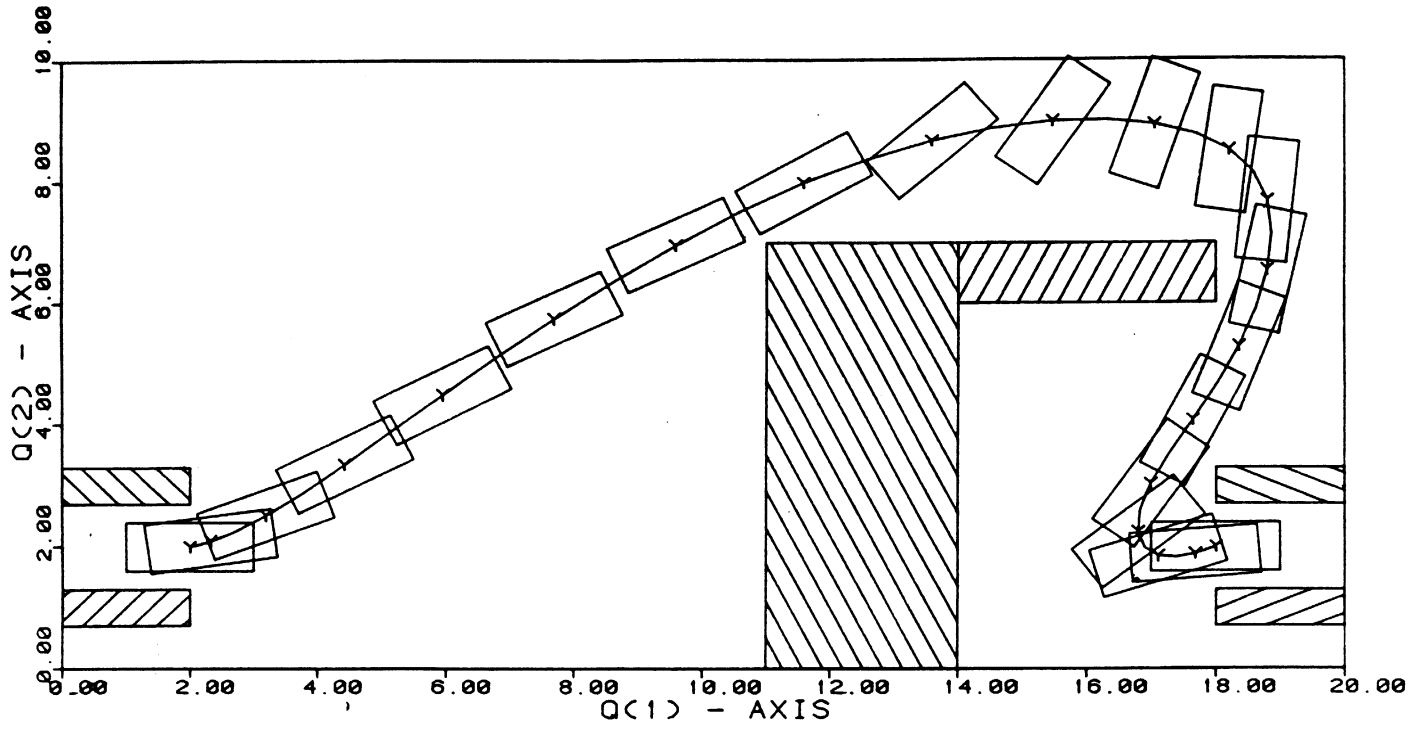


Figure 7. Final Path for Case 6.

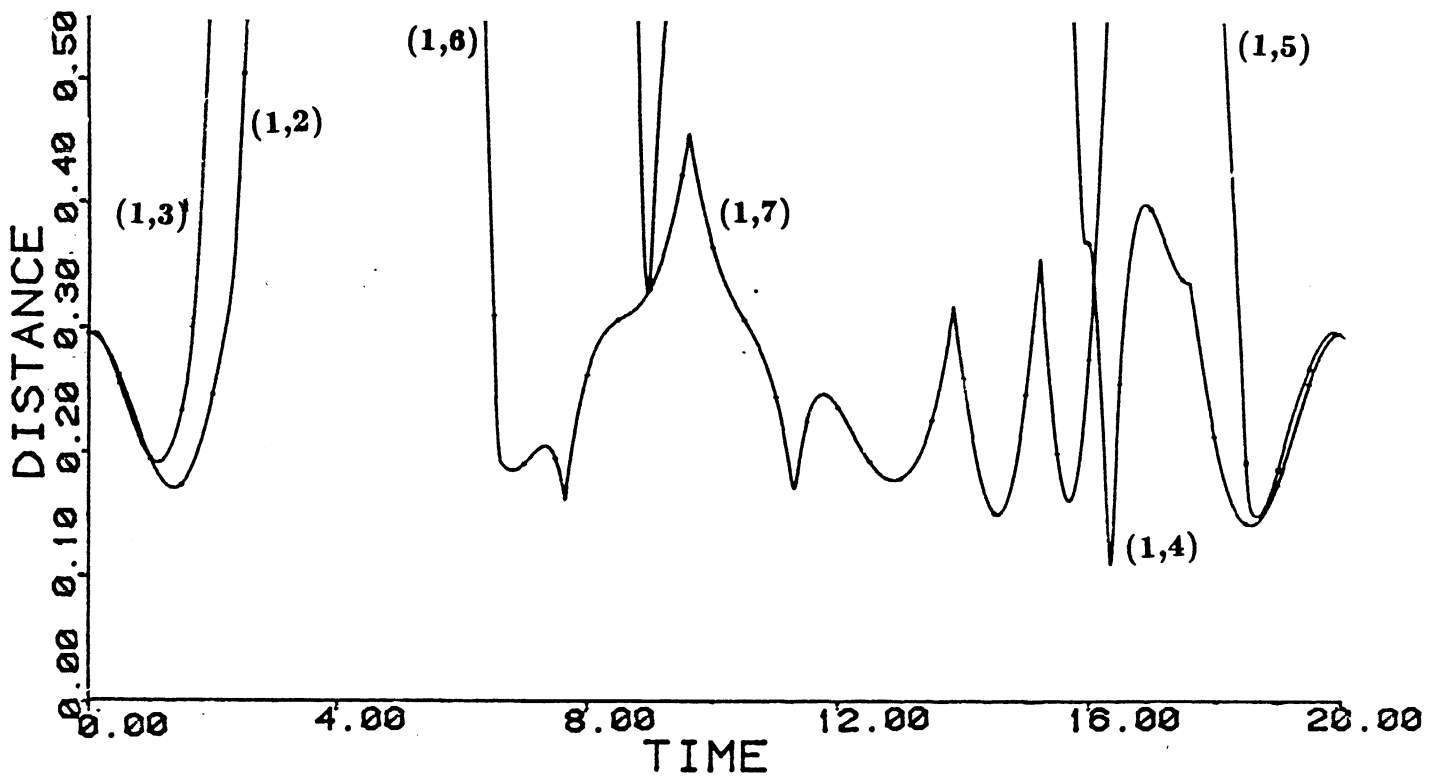


Figure 8. Distances of Approach for Case 1.

## 6. Conclusion

In this paper we have formulated an optimal control problem which includes a great variety of path planning problems. Obstacle constraints are expressed in terms of the distances between potentially colliding parts. We have studied the properties of these functions and found that they are continuously differentiable and have simple gradient formulas provided "strictly convex" modelling of part shapes is used. An algorithmic approach to the optimal control problem has been proposed and has proven successful for a cartesian manipulator in 2-space.

Manipulator problems in 3-space are much more complex than our example, but we feel that the ideas we have set forth will prove fruitful in their solution. The distance function approach may also be useful in finding admissible geometric paths; in this case the "dynamics" are very simple since they are integrators whose function is to produce smooth connecting paths. Unlike many of the presently known geometric approaches [1, 3, 4, 5], there is no conceptual problem in treating multi-member mechanisms in 3-space with this approach.

## 7. Appendix

Here we summarize some results of Clarke [12, 13] on the differentiability of Lipschitz functions. In all of what follows  $O$  is an open subset of  $R^n$  and  $f:O \rightarrow R$  is locally Lipschitz.

It is known that  $f$  has a Frechet derivative (gradient) at almost all points of  $\mathbf{O}$ , i.e., for such points  $x$  there exists  $\nabla f(x) \in R^n$  with

$$|\delta x|^{-1}(f(x + \delta x) - f(x) - \langle \nabla f(x), \delta x \rangle) \rightarrow 0, \delta x \rightarrow 0. \quad (\text{A.1})$$

The usual one-sided directional derivative at  $x$  is denoted by  $D^+f(x;v)$  and, provided the following limit exists for all  $v \in R^n$ , is defined and characterized by

$$\alpha^{-1}(f(x + \alpha v) - f(x) - \alpha D^+f(x;v)) \rightarrow 0, \alpha \downarrow 0. \quad (\text{A.2})$$

The generalized gradient of  $f$  at  $x$  is

$$\partial f(x) = \text{co}\{w: \nabla f(x_i) \rightarrow w, \nabla f(x_i) \text{ exists}, x_i \rightarrow x\}, \quad (\text{A.3})$$

where  $\text{co}$  denotes the convex hull. For all  $x \in \mathbf{O}$  the set  $\partial f(x) \in R^n$  is nonempty, compact, and convex.

We now state the results which are needed in Section 3.

Result 1 ([12], Prop. 1.13). The following statements are equivalent:  
 $\partial f(x) = \{\xi\}$ , a singleton;  $\nabla f(x)$  exists;  $\nabla f(x) = \xi$ .

Result 2 ([12], Cor. of Prop. 1.13). The following statements are equivalent:  
 $f$  is  $C^1$ ;  $\partial f(x)$  is a singleton for each  $x \in \mathbf{O}$ .

The remaining results concern the function

$$f(x) = \max\{g(x,u): u \in \mathbf{U}\}, \quad (\text{A.4})$$

where  $g: \mathbf{O} \times \mathbf{U} \rightarrow R$  and  $\mathbf{U}$  is compact. It is assumed that  $g$  satisfies the following conditions (somewhat stronger than those in [12]): (1)  $g$  is continuous;

(2)  $g$  is locally Lipschitz in  $x$ , uniformly for  $u \in \mathbf{U}$ ; (3) the Frechet derivative of  $g$  with respect to  $x, \nabla_x g(x, u)$ , exists and is continuous on  $\mathbf{O} \times \mathbf{U}$ .

Result 3.  $D^+f(x;v)$  exists for all  $x \in \mathbf{O}$ . Moreover,

$$D^+f(x;v) = \max\{\langle \nabla_x g(x, u), v \rangle : u \in M(x)\}, \quad (\text{A.5})$$

where

$$M(x) = \{u \in \mathbf{U} : g(x, u) = f(x)\}. \quad (\text{A.6})$$

Result 4. The generalized gradient exists and is given by

$$\partial f(x) = \text{co}\{\nabla_x g(x, u) : u \in M(x)\}. \quad (\text{A.7})$$

These results follow from Theorem 2.1 of [12], with the trivial modification in the proof of the Theorem that  $\mathbf{O}$  replaces  $R^n$ .

## 8. References

- [1] J.Y.S. Luh, "An anatomy of industrial robots and their controls," *IEEE Trans. Automat. Contr.*, vol. AC-28, pp. 133-153, Feb. 1983.
- [2] R. P. Paul, *Robot Manipulators: Mathematics, Programming and Control*. Cambridge, MA: M.I.T. Press, 1981.
- [3] T. Lozano-Perez, "Spatial planning: a configuration space approach," *IEEE Trans. Comput.*, vol. C-32, Feb. 1983.

- [4] T. Lozano-Perez, "Automatic planning of manipulator transfer movements," *IEEE Trans. Sys., Man, Cybern.* , vol. SMC-11, pp. 681-698, Oct. 1981.
- [5] J. Y. S. Luh, C. E. Campbell, "Minimum distance collision-free path-planning for industrial robots with a prismatic joint," *IEEE Trans. Automat. Contr.*, vol. AC-29, pp. 675-680, Aug. 1984.
- [6] M. Vukobratovic and M. Kircanski, "A method for optimal synthesis of manipulation robot trajectories," *Trans. of the ASME* , vol. 104, pp. 188-193, June 1982.
- [7] K. G. Shin and N. D. McKay, "Minimum-time control of robotic manipulators with geometric path constraints," to appear in *IEEE Trans. Automat. Contr.*
- [8] J. E. Bobrow, S. Dubowsky and J. S. Gibson, "On the optimal control of robotic manipulators with actuator constraints," in *Proc. Automat. Contr. Conf.* , pp. 782-787, June 1983.
- [9] H. P. Moravec, "The Stanford cart and the CMU rover", *Proc. of the IEEE* , vol. 71, July 1983.
- [10] E. G. Gilbert and D. W. Johnson, "Distance functions and their application to robot path planning in the presence of obstacles," in *Proc. Eighteenth Annual Conf. on Information Sciences and Systems*, Prince-

ton University, March 1984.

- [11] L. Cesari, *Optimization Theory and Applications*. New York: Springer-Verlag, 1983.
- [12] F. H. Clarke, "Generalized gradients and applications," *Trans. American Mathematical Society*, vol. 205, pp. 247-262, 1975.
- [13] F. H. Clarke, *Optimization and Nonsmooth Analysis*. New York: John Wiley & Sons, 1983.
- [14] L. L. Schumaker, *Spline Functions: Basic Theory*. New York: John Wiley & Sons, 1981.
- [15] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "On-line computational scheme for mechanical manipulators", *Trans. ASME, J. Dynamic Syst. Meas. Contr.* , vol. 102, pp. 69-76, June 1980.
- [16] M. W. Walker and D. E. Orin, "Efficient dynamic computer simulation of robotic mechanisms," *Proc. Joint Automat. Contr. Conf.* 1981.
- [17] R. K. Mehra and R. E. Davis, "A generalized gradient method for optimal control problems with inequality constraints and singular arcs," *IEEE Trans. Automat. Contr. Conf.* , vol. AC-17, pp. 69-79, Feb. 1972.
- [18] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques* New York: John Wiley & Sons, 1968.

- [19] E. Polak, *Computational Methods in Optimization, A Unified Approach*. New York: Academic Press, 1971.
- [20] P. Wolfe, "A method of conjugate subgradients for minimizing nondifferentiable functions," *Mathematical Programming Study 3* , pp. 145-173, 1975.
- [21] E. Polak, D. Q. Mayne and Y. Wardi, "On the extension of constrained optimization algorithms from differentiable to nondifferentiable problems," *SIAM J. Contr. and Optimization* , vol. 21, pp. 179-203, Mar. 1983.
- [22] E. Polak, D. Q. Mayne, "Algorithm models for nondifferentiable optimization", *Proc. IEEE Conf. on Decision and Contr.*, pp. 934-939, Dec. 1983.
- [23] R. Fletcher, *Practical Methods of Optimization, Unconstrained Optimization* , vol I. New York, John Wiley & Sons, 1980.