Optimal Trajectory Planning for a Space Robot Docking with a Moving Target via Homotopy Algorithms

Y. Chen and L. T. Watson

TR 93-25

Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061

August 16, 1993

Optimal Trajectory Planning for a Space Robot Docking with a Moving Target via Homotopy Algorithms

Y. Chen and L. T. Watson*

The mathematical formulation of optimal trajectory planning for a space robot docking with a moving target is derived. The calculus of variations is applied to the problem so that the optimal robot trajectory can be obtained directly from the target information without first planning the trajectory of the end-effector. The nonlinear two-point boundary value problem resulting from the problem formulation is solved numerically by a globally convergent homotopy algorithm. The algorithm guarantees convergence to a solution for an arbitrarily chosen initial guess. Numerical simulation for three examples demonstrates the approach.

1. Introduction

Space robots will play an important role in future space exploration. Their tasks may include docking payload to another spacecraft, picking up space debris and rescue/repair of stranded satellites in orbit, etc. Some tasks require that a space robot be able to dock to a moving target accurately and smoothly. Unlike industrial robots, space robots are mobile and usually redundant. The redundancy can be utilized to help achieve optimality, such as the criteria of minimum energy or minimum time, which are highly desirable in a space environment.

The topic of trajectory planning for robots with redundancy has been studied exten-

* Professor, Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061

sively in the recent robotics literature. Often the end-effector's trajectory is planned (or prescribed) first based on the target information and/or task requirements. Then either global or local optimization methods can be used to obtain an optimal robot configuration trajectory under a certain criterion[1-3] for a specific end-effector's trajectory.

As far as the space robot itself is concerned, a lot of research has been conducted on free-flying space robots [4-6], where the robot is subject to no external forces and torques so that the linear and angular momentum of the robot system are constant. This problem possesses some special characteristics with respect to robot kinematics. The relation between robot end-effector variables and robot configuration variables is not merely kinematical. This leads to more complexity in the forward and inverse kinematics calculations, as well as in trajectory planning.

In this article, we assume that the space robot is capable of travelling in space and maneuvering to dock with a moving target. It is not assumed free-flying because robot base control may be on during its mission. The target motion in an orbit is given. We define the target trajectory vector as

$$\mathbf{r}_t = [\,\mathbf{R}_t^T \quad \Theta_t^T\,]^T\,, \tag{1}$$

where $\mathbf{R}_t$ is the target position vector in inertial space and Θ_t is the target orientation vector. The problem considered here is to find a robot configuration trajectory $\mathbf{q}^*(t)$ that can meet the requirement of docking with the target and also minimize the performance measure defined by

$$L = \int_{t_0}^{t_f} \phi(\mathbf{q}, \dot{\mathbf{q}})dt. \tag{2}$$

It is well known that the kinematical relation between the robot end-effector variable vector $\mathbf{r}(t)$ and its configuration variable vector $\mathbf{q}(t)$ can be expressed as

$$\mathbf{r}(t) = \mathbf{f}(\mathbf{q}(t)), \tag{3}$$

where $\mathbf{r}$ is an m-vector and $\mathbf{q}$ is an n-vector. Let J denote the Jacobian matrix of $\mathbf{f}$. We consider two problems:

Problem 1: Find $\mathbf{q}^*$ so that

i) the end-effector reaches the target at free final time t_f, i.e., $\mathbf{r}_t(t_f) = \mathbf{f}(\mathbf{q}(t_f))$;

ii) L is minimized.

Problem 2: Find $\mathbf{q}^*$ so that

i) the same as i) in Problem 1;

ii) the velocity of the end-effector matches the velocity of the target at free final time t_f, i.e., $\dot{\mathbf{r}}_t(t_f) = J(\mathbf{q}(t_f))\dot{\mathbf{q}}(t_f)$ (this is the condition referred to as "smooth/soft docking"[7]);

iii) the same as ii) in Problem 1.

Here we simply apply the calculus of variations to our problems[8] instead of using global or local optimization methods for resolving robot redundancy[1]. The advantage of this approach is that it enables us to obtain the optimal robot configuration trajectory directly from the target information without first dealing with the end-effector's trajectory. A set of nonlinear two-point boundary value equations is derived by this approach. An algorithm based on homotopy theory[9] is used which guarantees global convergence to a solution.

The mathematical formulation of the problems is presented in the next section, followed by an introduction to homotopy algorithms for solving nonlinear equations. Finally, a numerical example of a planar two-arm space manipulator is used to demonstrate the approach.

2. Formulation

Formulation for Problem 1

Given the measure $L = \int_{t_0}^{t_f} \phi(\mathbf{q}, \dot{\mathbf{q}})dt$ and the boundary conditions of Problem 1:

$$\mathbf{q}(t_0) = \mathbf{C}_0, \tag{4a}$$

$$\mathbf{f}(\mathbf{q}(t_f)) = \mathbf{r}_t(t_f), \tag{4b}$$

the calculus of variations (note that t_f is free) yields the necessary conditions:

$$\frac{\partial \phi}{\partial \mathbf{q}} - \frac{d}{dt}\left(\frac{\partial \phi}{\partial \dot{\mathbf{q}}}\right) = 0, \qquad t_0 < t < t_f, \tag{5a}$$

$$\mathbf{q}(t_0) = \mathbf{C}_0, \tag{5b}$$

$$\left.\frac{\partial \phi}{\partial \dot{\mathbf{q}}}\right|_{t_f} \delta\mathbf{q}\Big|_{t_f} + \phi(\mathbf{q}, \dot{\mathbf{q}})\Big|_{t_f} \delta t_f = 0. \tag{5c}$$

Based on the relation

$$\delta\mathbf{q}_f = \delta\mathbf{q}\Big|_{t_f} + \dot{\mathbf{q}}(t_f)\delta t_f \tag{6}$$

shown in Figure 1, Eq. (5c) can be expressed as

$$\mathbf{g}\left\{\begin{array}{c} \delta\mathbf{q}_f \\ \delta t_f \end{array}\right\} = 0, \tag{7}$$

where $\mathbf{g} = \left[\left(\frac{\partial \phi}{\partial \dot{\mathbf{q}}}\right)^T \quad \left(\phi(\mathbf{q}, \dot{\mathbf{q}}) - \frac{\partial \phi}{\partial \dot{\mathbf{q}}}\dot{\mathbf{q}}\right)\right]_{t_f}$ is an $(n+1)$-dimensional row vector. Equation (7) is the transversality condition at $t = t_f$. It is required that $m \leq n+1$ for the existence of solutions. In the case $m = n+1$, the problem becomes trivial in the sense that the final state $\mathbf{q}^*(t_f)$ and the final time t_f can be determined from Eq. (4b) itself and a standard two-point boundary value problem is formulated as

$$\begin{aligned} &\frac{\partial \phi}{\partial \mathbf{q}} - \frac{d}{dt}\left(\frac{\partial \phi}{\partial \dot{\mathbf{q}}}\right) = 0, \qquad t_0 < t < t_f, \\ &\mathbf{q}(t_0) = \mathbf{C}_0, \\ &\mathbf{q}(t_f) = \mathbf{q}^*. \end{aligned} \tag{8}$$

If $m < n+1$, we denote the hypersurfaces of Eq. (4b) by

$$\mathbf{F}(\mathbf{q}(t_f), t_f) = \mathbf{0}. \tag{9}$$

By variation and writing $\mathbf{q}_f$ for $\mathbf{q}(t_f)$, we have

$$\left[\left(\frac{\partial F_j}{\partial \mathbf{q}_f}\right)^T \quad \frac{\partial F_j}{\partial t_f}\right]\left\{\begin{array}{c} \delta\mathbf{q}_f \\ \delta t_f \end{array}\right\} = 0, \qquad j = 1, \ldots, m. \tag{10}$$

Defining the $(n+1)$-vector

$$\mathbf{u}_j = \left[\left(\frac{\partial F_j}{\partial \mathbf{q}_f} \right)^T \quad \frac{\partial F_j}{\partial t_f} \right]^T \tag{11}$$

as the gradient vector of hypersurface j, $(j = 1, \ldots, m)$, we conclude that the necessary condition of Eq. (7) is that $\mathbf{g}$ is a linear combination of m vectors $\mathbf{u}_j$, i.e.,

$$\mathbf{g} = d_1\mathbf{u}_1 + d_2\mathbf{u}_2 + \cdots + d_m\mathbf{u}_m. \tag{12}$$

Now we write Eqs. (5a), (4a), (9) and (12) together as:

$$\frac{\partial \phi}{\partial \mathbf{q}} - \frac{d}{dt}\left(\frac{\partial \phi}{\partial \dot{\mathbf{q}}}\right) = 0, \qquad t_0 < t < t_f, \tag{13a}$$

$$\mathbf{q}(t_0) = \mathbf{C}_0, \tag{13b}$$

$$\mathbf{F}(\mathbf{q}(t_f), t_f) = \mathbf{0}, \tag{13c}$$

$$\mathbf{g}(\mathbf{q}_f, \dot{\mathbf{q}}_f) = d_1\mathbf{u}_1(\mathbf{q}_f, t_f) + d_2\mathbf{u}_2(\mathbf{q}_f, t_f) + \cdots + d_m\mathbf{u}_m(\mathbf{q}_f, t_f). \tag{13d}$$

Equations (13) represent a nonlinear two-point boundary value problem. The $(2n+m+1)$ boundary conditions of Eqs. (13b)–(13d) determine $2n$ constants of integration, m variables $d_1, \ldots, d_m$, and one variable t_f, altogether $(2n+m+1)$ unknowns.

Formulation for Problem 2

The difference between Problem 2 and Problem 1 is that the former includes the additional boundary conditions

$$J(\mathbf{q}(t_f))\dot{\mathbf{q}}(t_f) = \dot{\mathbf{r}}_t(t_f), \tag{14}$$

where J is the Jacobian matrix of $\mathbf{f}$. Therefore the constraint equations at final time t_f are

$$\mathbf{F}(\mathbf{q}(t_f), t_f) = \mathbf{f}(\mathbf{q}(t_f)) - \mathbf{r}_t(t_f) = 0, \tag{15a}$$

$$\mathbf{G}(\mathbf{q}(t_f), \dot{\mathbf{q}}(t_f), t_f) = J(\mathbf{q}(t_f))\dot{\mathbf{q}}(t_f) - \dot{\mathbf{r}}_t(t_f) = 0. \tag{15b}$$

Based on the differential equation (13a) and its initial condition for $\mathbf{q}(t_0)$, we can regard the final state $\mathbf{q}(t_f)$ and $\dot{\mathbf{q}}(t_f)$ as functions of $\dot{\mathbf{q}}(t_0)$ and t_f, i.e.,

$$\mathbf{q}(t_f) = \mathbf{p}_1(\dot{\mathbf{q}}(t_0), t_f), \tag{16a}$$

$$\dot{\mathbf{q}}(t_f) = \mathbf{p}_2(\dot{\mathbf{q}}(t_0), t_f). \tag{16b}$$

From Eqs. (16), we have in general a relation between $\dot{\mathbf{q}}(t_f)$ and $\mathbf{q}(t_f)$ expressed as

$$\dot{\mathbf{q}}(t_f) = \mathbf{p}(\mathbf{q}(t_f), t_f). \tag{17}$$

Substituting Eq. (17) into Eq. (15b) gives

$$\mathbf{H}(\mathbf{q}(t_f), t_f) = J(\mathbf{q}(t_f))\mathbf{p}(\mathbf{q}(t_f), t_f) - \dot{\mathbf{r}}_t(t_f) = 0. \tag{18}$$

It is required that $2m \leq n+1$ for the existence of solutions. In the case $2m = n+1$, the problem becomes trivial in the sense that the final state $\mathbf{q}^*(t_f)$ and the final time t_f can be determined from Eqs. (15a) and (18) alone and a standard two-point boundary value problem is formulated the same as Eqs. (8).

If $2m < n+1$, we follow the same procedure as for Problem 1 concerning the hypersurfaces of Eqs. (15a) and (18). By defining the gradient vectors $\mathbf{u}_j$ from Eq. (11) and $\mathbf{v}_j$ from

$$\mathbf{v}_j = \left[\left(\frac{\partial H_j}{\partial \mathbf{q}_f} \right)^T \quad \frac{\partial H_j}{\partial t_f} \right]^T, \tag{19}$$

the transversality condition of Eq. (7) is satisfied if and only if $\mathbf{g}$ is a linear combination of $2m$ vectors $\mathbf{u}_j$ and $\mathbf{v}_j$ $(j = 1, \cdots, m)$, i.e.,

$$\mathbf{g} = \sum_{i=1}^{m} d_i \mathbf{u}_i + e_i \mathbf{v}_i. \tag{20}$$

We can then formulate Problem 2 as

$$\frac{\partial \phi}{\partial \mathbf{q}} - \frac{d}{dt}\left(\frac{\partial \phi}{\partial \dot{\mathbf{q}}} \right) = 0, \qquad t_0 < t < t_f, \tag{21a}$$

$$\mathbf{q}(t_0) = \mathbf{C}_0, \tag{21b}$$

$$\mathbf{F}(\mathbf{q}(t_f), t_f) = \mathbf{0}, \tag{21c}$$

$$\mathbf{G}(\mathbf{q}(t_f), \dot{\mathbf{q}}(t_f), t_f) = \mathbf{0}, \tag{21d}$$

$$\mathbf{g}(\mathbf{q}_f, \dot{\mathbf{q}}_f) = \sum_{i=1}^{m} d_i \mathbf{u}_i(\mathbf{q}_f, t_f) + e_i \mathbf{v}_i(\mathbf{q}_f, t_f). \tag{21e}$$

Equations (21) represent a nonlinear two-point boundary value problem. The $(2n+2m+1)$ boundary conditions of Eqs. (21b)–(21e) determine $2n$ constants of integration, m variables $d_1, \ldots, d_m$, m variables $e_1, \ldots, e_m$, and one variable t_f, altogether $(2n+2m+1)$ unknowns.

3. Homotopy Algorithm

Equations (13) or (21) constitute a nonlinear two-point boundary value problem. The solution of such problems is generally very difficult to obtain, particularly for systems whose dimension is not small. The shooting method reduces the problem to the solution of a system of nonlinear equations

$$\mathbf{F}(\mathbf{w}) = \mathbf{0}, \qquad (\mathbf{F}, \mathbf{w} \in R^n) \tag{22}$$

which is usually solved by Newton's method. The approach suffers from sensitivity to initial guesses and the Newton iteration can diverge. As an alternative, one can consider the continuation method with Eq. (22) replaced by

$$\Lambda(\alpha, \mathbf{w}) = \alpha \mathbf{F}(\mathbf{w}) + (1 - \alpha)\mathbf{S}(\mathbf{w}) = \mathbf{0}, \tag{23}$$

where $\alpha \in [0, 1]$ is a parameter and $\mathbf{S}(\mathbf{w})$ is a function such that the equation $\mathbf{S}(\mathbf{w}) = \mathbf{0}$ is relatively easy to solve. However, if Newton's method is applied to Eq. (23), failure may still occur, because continuation postulates that $\mathbf{w} = \mathbf{w}(\alpha)$ in Eq. (23), and the zero set of Eq. (23) does not necessarily increase monotonically in α.

Quite recently, a new version of the continuation method, known as probability-one homotopy algorithms, has been developed[9]. In the algorithm, a homotopy map $\mathbf{\Gamma_a} : [0,1) \times E^m \rightarrow E^n$ is defined by

$$\mathbf{\Gamma_a}(\alpha, \mathbf{w}) = \mathbf{\Gamma}(\mathrm{a}, \alpha, \mathbf{w}) = \alpha\mathbf{F}(\mathbf{w}) + (1-\alpha)(\mathbf{w} - \mathrm{a}). \tag{24}$$

According to the probability-one homotopy theory, for almost all $\mathbf{a} \in E^m$, there is a zero curve γ of $\mathbf{\Gamma_a}$ emanating from $(0, \mathrm{a})$, along which the Jacobian matrix $D\mathbf{\Gamma_a}$ has full rank. The curve γ is continuously differentiable, does not intersect itself, and is disjoint from all the other zeros of $\mathbf{\Gamma_a}$. Furthermore, γ must either reach a point $(1, \bar{\mathbf{w}})$, in which case $\bar{\mathbf{w}}$ solves Eq. (22), or go off to infinity. Since γ is smooth, it can be parameterized by arc length $s : \ \alpha = \alpha(s)$, $\mathbf{w} = \mathbf{w}(s)$. Then the zero curve γ of Eq. (24) satisfies

$$\mathbf{\Gamma_a}(\alpha(s), \mathbf{w}(s)) = \mathbf{0} \tag{25}$$

identically in s. Thus

$$\frac{d}{ds}\mathbf{\Gamma}_{\mathrm{a}}(\alpha(s), \mathbf{w}(s)) = \mathbf{0}. \tag{26}$$

Equation (26) can be written as

$$\begin{bmatrix} \dfrac{\partial \mathbf{\Gamma_a}}{\partial \alpha} & \dfrac{\partial \mathbf{\Gamma_a}}{\partial \mathbf{w}} \end{bmatrix} \begin{pmatrix} \dfrac{d\alpha}{ds} \\ \dfrac{d\mathbf{w}}{ds} \end{pmatrix} = \mathbf{0}, \tag{27a}$$

and since the derivative is a unit tangent vector,

$$\left\| \begin{pmatrix} \dfrac{d\alpha}{ds} & \dfrac{d\mathbf{w}}{ds} \end{pmatrix} \right\|_2 = 1, \tag{27b}$$

where $\| \ \|_2$ denotes the Euclidean norm. Equations (27) are subject to the initial conditions

$$\begin{aligned} \alpha(0) &= 0, \\ \mathbf{w}(0) &= \mathbf{a}. \end{aligned} \tag{28}$$

At the point $s = s^*$ for which $\alpha(s^*) = 1$, the solution $\bar{\mathbf{w}} = \mathbf{w}(s^*)$ is obtained.

In solving Eqs. (27), α is allowed to increase or decrease arbitrarily for $s \in [0, s^*]$, as long as the zero curve γ is followed. This is a distinct advantage over a Newton-based continuation method. Under fairly general assumptions on $\mathbf{F}(\mathbf{w})$, the homotopy theory says that for almost all $\mathbf{a}$ the Jacobian matrix $\left[\frac{\partial \Gamma_{\mathbf{a}}}{\partial \alpha} \quad \frac{\partial \Gamma_{\mathbf{a}}}{\partial \mathbf{w}}\right]$ has full rank along the zero curve γ and γ is bounded. This implies the global convergence (with probability one) to a solution of the nonlinear problem.

4. Numerical Examples

A planar satellite-based manipulator is used as an example system to demonstrate the approach presented here. As shown in Figure 2, the space robot consists of a satellite base, two manipulator arms and an end-effector. The arm lengths are $L_1 = L_2 = 10.0\,\mathrm{m}$, $L_3 = 2.0\,\mathrm{m}$. We define $\mathbf{r} = [\,x \quad y \quad \psi\,]^T$ as the manipulation vector of the end-effector and $\mathbf{q} = [\,x_0 \quad y_0 \quad \theta_0 \quad \theta_1 \quad \theta_2 \quad \theta_3\,]^T$ as the robot configuration vector. We also assume that the base orientation θ_0 of the robot is controlled to be fixed in orbit (at $\theta_0 = 0$), so that the kinematical relation in Eq. (3) is simply

$$\begin{aligned} x &= x_0 + L_1 \cos\theta_1 + L_2 \cos\theta_2 + L_3 \cos\theta_3, \\ y &= y_0 + L_1 \sin\theta_1 + L_2 \sin\theta_2 + L_3 \sin\theta_3, \\ \psi &= \theta_3. \end{aligned} \tag{29}$$

The performance measure is given by

$$L = \int_{t_0}^{t_f} \dot{\mathbf{q}}^T \dot{\mathbf{q}} dt \tag{30}$$

and the initial configuration of the robot is $\mathbf{q}(t_0) = \mathbf{C}_0 = \mathbf{0}$.

Numerical computations have been done for three examples.

Example 1:

The target trajectory in space is specified as

$$\begin{aligned} x_t &= 10t, \\ y_t &= 10 + 10t, \\ \psi_t &= \beta \quad \text{(const)}. \end{aligned} \tag{31}$$

It is required that the position and orientation of the robot end-effector match the position and orientation of the target at some final time t_f and meanwhile minimize the performance measure defined in Eq. (30). Therefore Eqs. (13) associated with Problem 1 are used. Here we have $n = 5$ (θ_0 is dropped from $\mathbf{q}$ as the robot base orientation θ_0 is assumed fixed) and $m = 3$. The solution exists and is nontrivial since $m < n + 1$. The dimension of the problem is $(2n + m + 1) = 14$.

The details of the derivation of Eqs. (13) and their conversion to a nonlinear system Eq. (22) are easy but rather lengthy, and $\mathbf{F}(\mathbf{w})$ in (22) is only defined implicitly. Here shooting is used to define $\mathbf{F}(\mathbf{w})$, whose components are thus given in terms of the solution to an initial value problem, evaluated at t_f. Hence $\mathbf{F}(\mathbf{w})$ can only be expressed implicitly. Note that Eqs. (13) can also be converted to a nonlinear system $\mathbf{F}(\mathbf{w}) = \mathbf{0}$ by finite differences, collocation, or finite elements. Each of these methods has certain advantages over shooting, but shooting was sufficient for the examples here. Several different cases were solved by a homotopy based nonlinear equation solver, using the HOMPACK software described in ref. 9. The optimal robot configuration trajectories with respect to different target orientations β are shown in Figures 3–4.

Example 2:

The target trajectory in space is specified as

$$\begin{aligned} x_t &= 10t, \\ y_t &= 10 + 10t, \\ \psi_t &= t. \end{aligned} \tag{32}$$

It is required that the position and orientation of the robot end-effector, *as well as their velocities*, match those of the target at some final time and meanwhile minimize the performance measure defined in Eq. (30). This is an example of Problem 2, with $n = 5$ and $m = 3$ so that $n + 1 = 2m$. Therefore, as described in Section 2, the problem becomes trivial. The final state $\mathbf{q}(t_f)$ and the final time t_f can be obtained by solving directly the constraint equations. Then, solving a standard (simple explicit boundary conditions) nonlinear two-point boundary value problem, the optimal robot configuration trajectory can be obtained as shown in Figure 5.

Example 3:

With the same target trajectory defined in Eq. (32), we now drop the requirement on orientation in the constraint equations. It is only required that the end-effector's position and velocity match those of the target and meanwhile minimize the performance measure of Eq. (30). Now $n = 5$ and $m = 2$ so that $2m < n+1$. Therefore Eqs. (21) associated with a nontrivial solution of Problem 2 are used. The dimension of this problem is $2n+2m+1 = 15$.

The optimal robot configuration trajectory is shown in Figure 6. Figure 7 is the homotopy zero curve for this problem, clearly not monotone in α, and showing why continuation in α would fail. The many sharp turns also show why sophisticated mathematical software (such as HOMPACK[10]) is required to track the homotopy zero curve.

5. Conclusions

In this article, a mathematical formulation of optimal trajectory planning for a space robot docking with a moving target was derived. The calculus of variations was applied to the problem so that the optimal robot trajectory can be obtained directly from the target information, without first planning the trajectory of the end-effector.

Two problems were considered: the first for the position (and/or orientation) match of two docking objects, the second for the position (and/or orientation) and velocity match of two docking objects. The degrees of freedom n of the space robot, must satisfy an inequality for the existence of a solution. For Problem 1, the inequality is $n \geq (m-1)$; for Problem 2, $n \geq (2m-1)$. The difference between the two problems' formulations seems only their dimensions. However, the formulation of Problem 2 required the relation between $\dot{\mathrm{q}}(t_f)$ and $\mathrm{q}(t_f)$, as expressed in Eq. (17). In most cases, unlike Example 3 presented here, the relation will be implicit. Some special treatment would be required and the dimension of the equations $\mathbf{F}(\mathbf{w}) = \mathbf{0}$ might increase further. Thus, from a numerical point of view, Problem 2 is much more difficult than Problem 1.

The nonlinear two-point boundary value problem resulting from the problem formulation can be solved numerically by a globally convergent homotopy method. The examples verified the global convergence of the algorithm to a solution for an arbitrarily chosen initial guess. The algorithm has been used successfully in other fields where highly nonlinear, large dimensional systems of equations need to be solved. Here was an example of the use of homotopy algorithms in robotics.

References

1. Nakamura, Y., *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, Reading, Massachusetts, 1991.
2. Koivo, A. J., and Arnautovic, S. H., "Dynamic Optimum Control of Redundant Manipulators," *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, Sacramento, CA, 1991, pp. 466–471.
3. Kazerounian, K., and Wang, Z., "Global versus Local Optimization in Redundancy Resolution of Robotic Manipulators," *International Journal of Robotics Research*, **7**(5), 3–12 (1988).
4. Longman, R. W., "The Kinetics and Workspace of a Satellite-Mounted Robot," *Journal of the Astronautical Sciences*, **38**(4), 423–439 (1990).
5. Vafa, Z., and Dubowsky, S., "On the Dynamics of Space Manipulators Using the Virtual Manipulator, with Applications to Path Planning," *Journal of the Astronautical Sciences*, **38**(4), 441–472 (1990).
6. Nenchev, D., Umetani, Y., and Yoshida, K., "Analysis of a Redundant Free-Flying Spacecraft/Manipu- lator System," *IEEE Transactions on Robotics and Automation*, **8**(1), 1–6 (1992).
7. Wang, P. K. C., "Automatic Assembly of Space Station," *Proceedings of the Workshop*

on Identification and Control of Flexible Space Structures, San Diego, CA, 1985, Vol. 1, pp. 67–101.

8. Leitmann, G., *An Introduction to Optimal Control*, McGraw-Hill, New York, 1966.

9. Watson, L. T., "Globally Convergent Homotopy Algorithms for Nonlinear Systems of Equations," *Nonlinear Dynamics*, **1**, 143–191 (1990).

10. Watson, L. T., Billups, S. C., and Morgan, A. P., "HOMPACK: A Suite of Codes for Globally Convergent Homotopy Algorithms," *ACM Transactions on Mathematical software*, **13**(3), 281–310 (1987).

List of Illustrations

Figure 1. An extremal and a neighboring comparison curve.

Figure 2. Space robot and orbiting target.

Figure 3. Optimal robot trajectory: β=45 (degrees), final time=2.183(s).

Figure 4. Optimal robot trajectory: β=-135 (degrees), final time=0.977(s).

Figure 5. Optimal robot trajectory (trivial case), final time=0.996(s).

Figure 6. Optimal robot trajectory (nontrivial case), final time=1.717(s).

Figure 7. A homotopy zero curve for a nontrivial case of Problem 2.

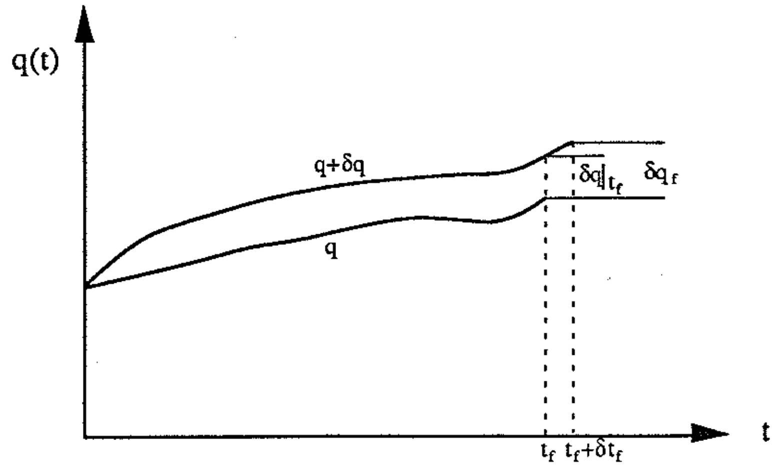

Figure 1. An extremal and a neighboring comparison curve.

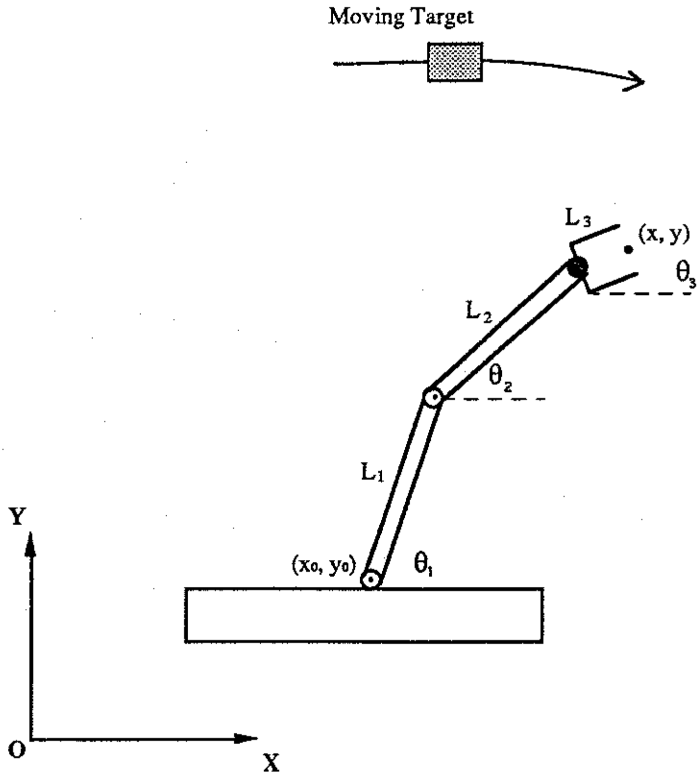

Figure 2. Space robot and orbiting target.

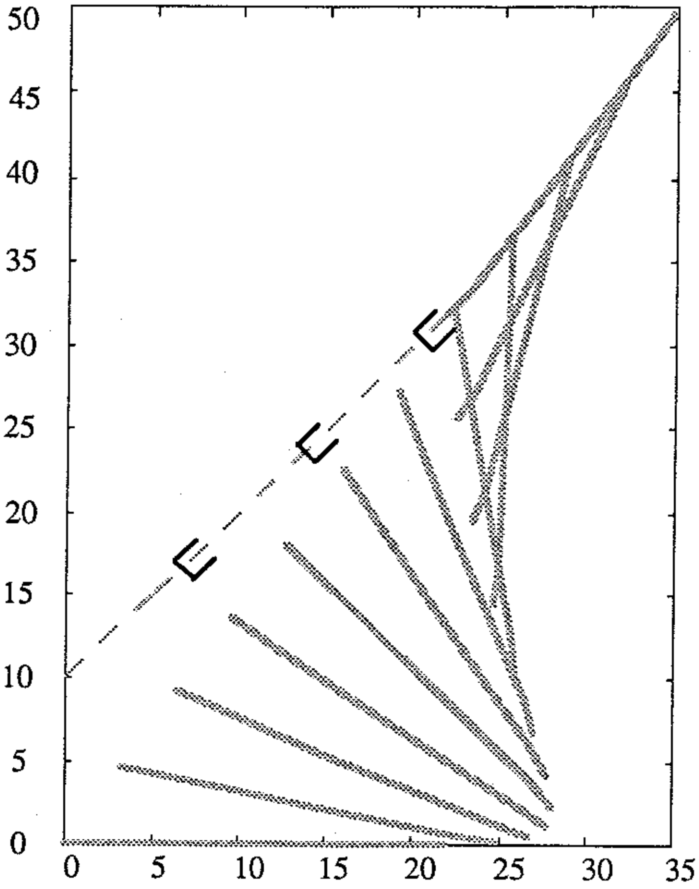

Figure 3. Optimal robot trajectory: β=45 (degrees), final time=2.183 (s).

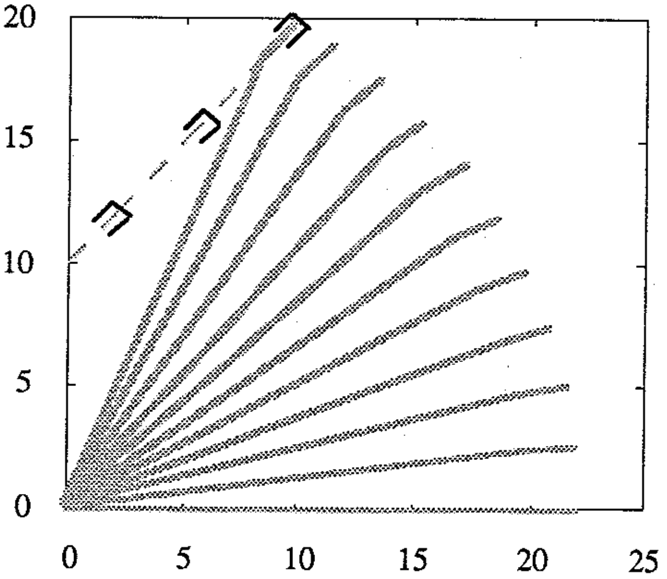

Figure 4. Optimal robot trajectory: β=-135 (degrees), final time=0.977(s).

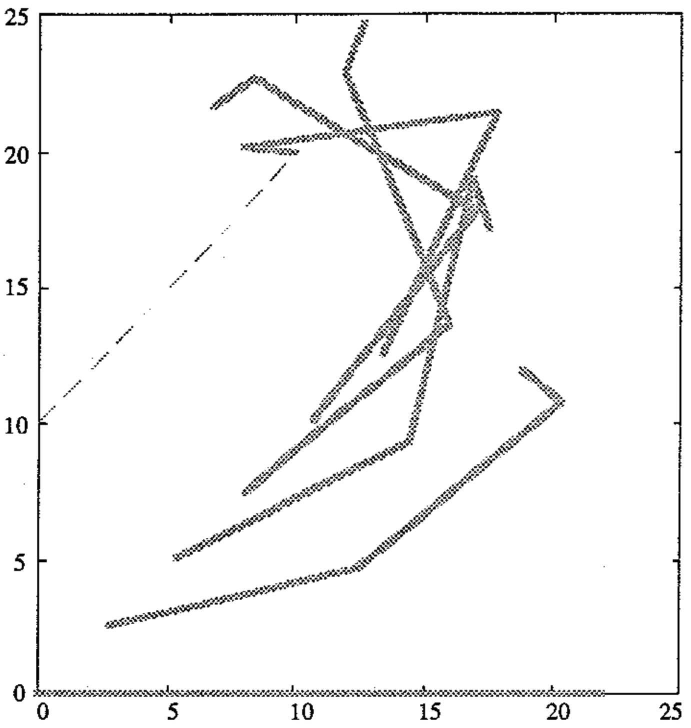

Figure 5. Optimal robot trajectory (trivial case), final time=0.996 (s).

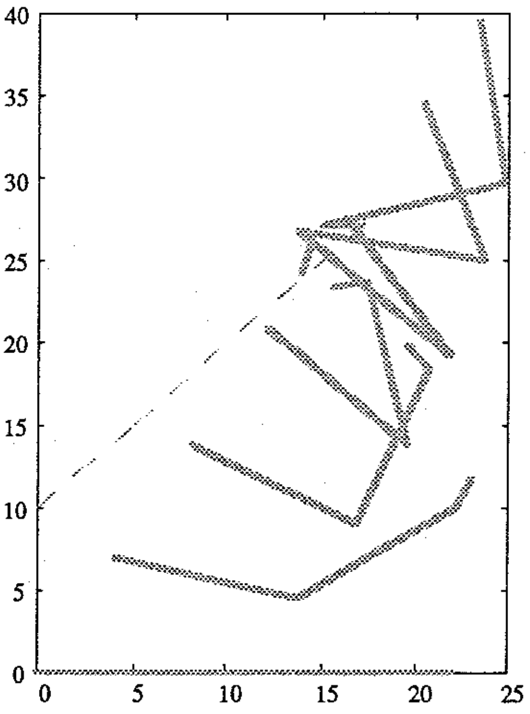

Figure 6. Optimal robot trajectory (nontrivial case), final time=1.717 (s).

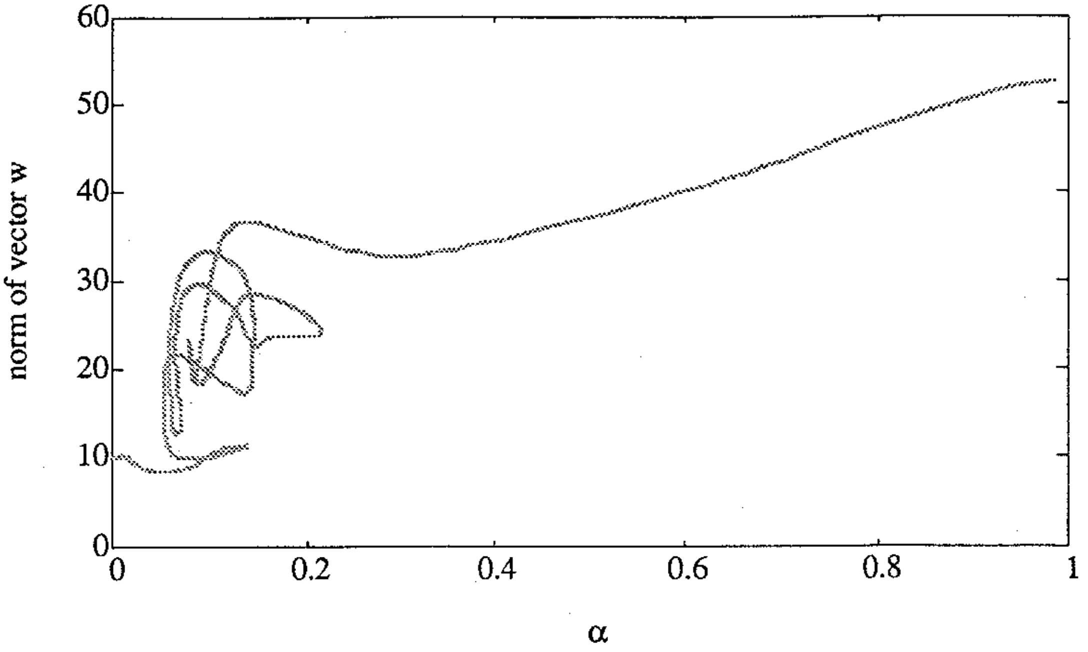

Figure 7. A homotopy zero curve for a nontrivial case of Problem 2.