Proceedings of Design Engineering Technical Conference &Computers and Information 2008: Mechanisms and Robotics Conference August 3-6, 2008, New York City, NY

DETC2008-49268

OPTIMAL TRAJECTORY PLANNING FOR REDUNDANT MANIPULATORS BASED ON MINIMUM JERK

Jingzhou Yang
Center for Computer Aided Design
The University of Iowa

Joo Kim
Center for Computer Aided Design
The University of Iowa

Esteban Pena Pitarch
Universitat Polit`ecnica de Catalunya (UPC)
Spain

Karim Abdel-Malek
Center for Computer Aided Design
The University of Iowa

ABSTRACT

This paper presents an optimization-based method to solve the smooth trajectory planning problem where the user knows only the start and end points of the end-effector or the via point plus the start and end target points. For the start and end target points, we use an optimization approach to determine the manipulator configurations. Having obtained the desired minimum jerk path in the Cartesian space using the minimum jerk theory and having represented each joint motion by the third-degree B-spline curve with unknown parameters (i.e., control points), an optimization approach, rather than the pseudoinverse technique for inverse kinematics, is used to calculate the control points of each joint spline curve. The objective function includes several parts: (a) dynamic effort; (b) the inconsistency function, which is the joint rate change (first derivative) and predicted overall trend from the initial point to the end point; and (c) the nonsmoothness function of the trajectory, which is the second derivative of the joint trajectory. This method can be used for robotic manipulators with any number of degrees of freedom. Minimum jerk trajectories are desirable for their similarity to human joint movements, for their amenability to limit robot vibrations, and for their control (i.e., enhancement of control performance). Illustrative examples are presented to demonstrate the method.

1. INTRODUCTION

Trajectory planning for serial manipulators has been an important research topic for many years. The traditional trajectory planning problem for robotic manipulators is defined as follows: given a geometric path in Cartesian space, find the joint angles in joint space. The geometric path and the kinematic and dynamic constraints are the inputs of the trajectory planning. The trajectory joints, expressed as a time sequence of position, velocity, and acceleration values, are the output.

However, if one is given only the start and end points or a via point on the path in Cartesian space, the path not only has to be determined, but the joint angles also need to be generated. Park and Bobrow (2004) presented a minimum time motion planning method to determine the path and joint angles. Yamane et al (2004) proposed a path planner and a data-driven inverse kinematics. To date, nobody else has addressed this type of trajectory planning problem as a whole, and it is a practical problem in both the robotics and human modeling fields. Piazzi and Visioli (2000) investigated a global minimum-jerk framework using cubic splines. However, it did not take into account the dynamics of the manipulators. This paper reviews a 3D minimum jerk model based on the 2D human motion planning model in the appendix. We propose an optimization-based method to solve the trajectory planning problem based on the 3D minimum jerk path. The minimum jerk model within human motion planning is similar to robot manipulator trajectory planning, and it is desirable to consider using this minimum jerk model for manipulator planning to reduce robot vibrations and improve control performance.

Researchers have developed various trajectory planning methods for robotic systems considering different kinematic

 Copyright © 2008 by ASME

and dynamic criteria such as obstacle avoidance, singularity avoidance, time minimization, torque optimization, energy optimization, and other objective functions (Shin and McKay, 1986; Bobrow, 1988; Shiller and Dubowsky, 1989; Zhou and Nguyen, 1997; Antonelli and Chiaverini, 1998; Hirakawa and Kawamura, 1997; Furuno et al., 2002; Valero et al., 2006, Kim et al., 2007). Yun and Xi (1996) used genetic algorithms for optimum motion planning in joint space for robots. Constantinescu and Croft (2000) put forth a smooth and time-optimal trajectory planning that minimizes time under path constraints, torque limits, and torque rate limits. Saramago et al. (1998; 2000; 2001; 2002) have studied robot path planning by considering a dynamic system with payload constraints, and in the presence of moving obstacles. Pugazhenthi et al. (2002) studied the optimal trajectory planning for Stewart-platform-based machine tools. Li and Ceglarek (2002) presented an optimal trajectory planning application for material handling of compliant sheet metal parts in which they considered part permanent deformation, trajectory smoothness, and static obstacle avoidance. Zha (2002) presented optimal pose trajectory planning using genetic algorithm. Valero et al. (2006) studied the trajectory planning in workspaces with obstacles. Recently, Gasparetto and Zanotto (2007) surveyed all the different approaches and also proposed a new method where the integral of the squared jerk and total execution time is used for the objective function.

In the human motion simulation field, lots of attention has been directed at finding the best method for solving the redundant problem because the human is a highly redundant system. Flash and Hogan (1985) presented a mathematical model that was shown to predict both the qualitative features and the quantitative details observed experimentally in planar (2D), multi-joint arm movements. Uno et al. (1989) proposed a mathematical model that is formulated by defining the square of the rate of change of torque integrated over the entire movement as an objective function. Kawato et al. (1988) studied the problems of coordinate transformation from the desired trajectory to the body coordinates and motor command generation. A randomized planner introduced by Barraquand and Latombe (1991) was able to solve complex path-planning problems for many-DOF robots by alternating "down motions" to track the negated gradient of a potential field and "random motions" to escape local minima. The unique trajectory that yields the best performance is determined using dynamic optimization theory. The objective function is the square of the magnitude of jerk (rate of change of acceleration) of the hand integrated over the entire movement. This is equivalent to assuming that a major goal of motor coordination is the production of the smoothest possible movement of the hand. Alexander (1997) hypothesized that trajectories are chosen to minimize metabolic energy costs.

This paper presents an optimization-based approach to considering robot dynamic effort, joint angle consistency, and smoothness. In this work, we consider robot traveling time as a user-assigned constant rather than a part of the objective function.

This paper is organized as follows. In Section 2, a recursive formulation is used for determining robot kinematics and dynamics. Section 3 reviews the B-spline curve definition. Section 4 illustrates the procedure for determining the configurations for the initial and final target points. Section 5 gives the details of the optimization formulation. Section 6 provides examples to demonstrate the proposed method, and concluding remarks are in Section 7.

2. KINEMATICS AND DYNAMICS

The Denavit-Hartenberg method (Denavit and Hartenberg, 1955) was created to systematically represent the relation between two coordinate systems. This method is based upon characterizing the configuration of link *i* with respect to link (*i*-1) by a (4×4) homogeneous transformation matrix. For an *n*-degrees-of-freedom model, the position vector of the end-effector of a serial robotic manipulator can be written as

$$\mathbf{x} = \mathbf{x}(\mathbf{q}), \tag{1}$$

where $\mathbf{q} \in \mathbf{R}^n$ is the vector of *n*-generalized coordinates, and $\mathbf{x}(\mathbf{q})$ can be obtained from the multiplication of the homogeneous transformation matrices defined by the DH method as

$$\begin{bmatrix} \mathbf{x}(\mathbf{q}) \\ 1 \end{bmatrix} = {}^0\mathbf{T}_n \begin{bmatrix} \mathbf{x}^n \\ 1 \end{bmatrix} = \mathbf{T}_1\mathbf{T}_2 \ldots \mathbf{T}_n \begin{bmatrix} \mathbf{x}^n \\ 1 \end{bmatrix}, \tag{2}$$

where $\mathbf{T}_i$ is the transformation matrix relating coordinate frames *i* and *i-1*. $\mathbf{x}^n$ denotes the position vector of the end-effector with respect to local frame *n*. The transformation matrix can be defined as

$${}^{i-1}\mathbf{T}_i = \left[\begin{array}{ccc|c} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\alpha_i\cos\theta_i & -\sin\alpha_i\cos\theta_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ \hline 0 & 0 & 0 & 1 \end{array}\right], \tag{3}$$

where $\theta_i, d_i, \alpha_i, a_i$ are four DH parameters.

A. *Forward Recursive Kinematics*

Based on Eq. (2), the position, velocity, and acceleration for the j^{th} joint are obtained from a recursive formulation (Toogood, 1989). Given the link transformation matrix (T_j) and the kinematic state variables for each joint (q_j, $\dot{q}_j$, and $\ddot{q}_j$), for *j* =1 to *n*, the 4×4 transformation matrices $\mathbf{A}_j$, $\mathbf{B}_j$, and $\mathbf{C}_j$ are obtained by

$$\mathbf{A}_j = \mathbf{T}_1\ \mathbf{T}_2\ \mathbf{T}_3 \cdots \mathbf{T}_j = \mathbf{A}_{j\text{-}1}\mathbf{T}_j \tag{4}$$

$$\mathbf{B}_j = \dot{\mathbf{A}}_j = \mathbf{B}_{j\text{-}1}\mathbf{T}_j + \mathbf{A}_{j\text{-}1}\frac{\partial \mathbf{T}_j}{\partial q_j}\dot{q}_j \tag{5}$$

Copyright © 2008 by ASME

$$\mathbf{C}_j = \dot{\mathbf{B}}_j = \ddot{\mathbf{A}}_j = \mathbf{C}_{j-1}\mathbf{T}_j + 2\mathbf{B}_{j-1}\frac{\partial \mathbf{T}_j}{\partial q_j}\dot{q}_j + \mathbf{A}_{j-1}\frac{\partial^2 \mathbf{T}_j}{\partial q_j^2}\dot{q}_j^2 + \mathbf{A}_{j-1}\frac{\partial \mathbf{T}_j}{\partial q_j}\ddot{q}_j \tag{6}$$

$\mathbf{A}_0 = \mathbf{I}_{4\times4}$ and $\mathbf{B}_0 = \mathbf{C}_0 = \mathbf{0}_{4\times4}$.

After obtaining all the transformation matrices, $\mathbf{A}_j$, $\mathbf{B}_j$, and $\mathbf{C}_j$, the global position, velocity, and acceleration of a point in the Cartesian coordinate system can be calculated using the following formulas:

$$\begin{bmatrix}\mathbf{x}(\mathbf{q})\\ 1\end{bmatrix} = \mathbf{A}_n\begin{bmatrix}\mathbf{x}^n\\ 1\end{bmatrix};\quad \begin{bmatrix}\dot{\mathbf{x}}(\mathbf{q})\\ 1\end{bmatrix} = \mathbf{B}_n\begin{bmatrix}\mathbf{x}^n\\ 1\end{bmatrix};\quad \begin{bmatrix}\ddot{\mathbf{x}}(\mathbf{q})\\ 1\end{bmatrix} = \mathbf{C}_n\begin{bmatrix}\mathbf{x}^n\\ 1\end{bmatrix} \tag{7}$$

B. Backward Recursive Dynamics

Based on forward recursive kinematics, the backward recursive dynamic analysis is accomplished by defining a 4×4 transformation matrix $\mathbf{D}_i$ and 4×1 vectors $\mathbf{E}_i$, $\mathbf{F}_i$, and $\mathbf{G}_i$, as follows.

Given the mass and inertia properties of each link, the external force $\mathbf{f}_k^{\mathrm{T}} = \begin{bmatrix} f_x^k & f_y^k & f_z^k & 0\end{bmatrix}$, and the moment $\mathbf{h}_k^{\mathrm{T}} = \begin{bmatrix} h_x^k & h_y^k & h_z^k & 0\end{bmatrix}$ at link k ($1 \le k \le n$) defined in the global coordinate system, the joint actuation torques τ_i for $i = n$ to 1 are determined as follows (Toogood, 1989):

$$\tau_i = tr[\frac{\partial \mathbf{A}_i}{\partial q_i}\mathbf{D}_i] + \mathbf{g}^{\mathrm{T}}\frac{\partial \mathbf{A}_i}{\partial q_i}\mathbf{E}_i + \sum_{k=i}^{n}\mathbf{f}_k^{\mathrm{T}}\frac{\partial \mathbf{A}_i}{\partial q_i}\mathbf{F}_i^k + \sum_{k=i}^{n}(\mathbf{G}_i^k)^{\mathrm{T}}\mathbf{A}_{i-1}\mathbf{z}_0 \tag{8}$$

$$\mathbf{D}_i = \mathbf{I}_i\mathbf{C}_i^{\mathrm{T}} + \mathbf{T}_{i+1}\mathbf{D}_{i+1},\ \mathbf{E}_i = m_i{}^i\mathbf{r}_i + \mathbf{T}_{i+1}\mathbf{E}_{i+1}$$

$$\mathbf{F}_i^k = {}^k\mathbf{r}_f\delta_{ik} + \mathbf{T}_{i+1}\mathbf{F}_{i+1}^k,\ \mathbf{G}_i^k = \mathbf{h}_k\delta_{ik} + \mathbf{G}_{i+1}$$

where $\mathbf{D}_{n+1} = \mathbf{0}_{4\times4}$, $\mathbf{D}_{n+1} = \mathbf{E}_{n+1} = \mathbf{F}_{n+1}^k = \mathbf{G}_{n+1}^k = \mathbf{0}_{4\times1}$; $\mathbf{I}_i$ is the inertia matrix for link i; m_i is the mass of link i; $\mathbf{g}$ is the gravity vector; ${}^i\mathbf{r}_i$ is the location augmented vector of the center of mass of link i in the local frame i; ${}^k\mathbf{r}_f$ is the position of the external force in the local frame k; $\mathbf{z}_0 = \begin{bmatrix}0 & 0 & 1 & 0\end{bmatrix}^{\mathrm{T}}$; and $\delta_{ik} = \begin{cases}1, & \text{if } i = k\\ 0, & \text{if } i \ne k\end{cases}$. The first term in Eq. (8) is the inertia and Coriolis torque, the second term is the torque due to gravity, the third term is the torque due to external forces, and the fourth term represents the torque due to the external moments.

3. JOINT PROFILES USING B-SPLINES CURVES

There are many ways to define the B-spline basis functions, where the most useful for computer implementation is the recursive formula (Pigel, 1997). Let $\mathbf{U} = \{t_0, ..., t_{n_m}\}$ be a non-decreasing sequence of real numbers, i.e., $t_i \le t_{i+1}$, $i = 0, ..., n_m - 1$. The t_i are called knots, and $\mathbf{U}$ is the knot vector. The ith B-spline basis function of p-degree (order p+1), denoted by $N_{i,p}(t)$, is defined as

$$N_{i,0}(t) = \begin{cases}1 & \text{if } t_i \le t < t_{i+1}\\ 0 & \text{otherwise}\end{cases}$$
$$N_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i}N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}}N_{i+1,p-1}(t) \tag{9}$$

A pth-degree B-spline curve is defined by

$$C(t) = \sum_{i=0}^{m}N_{i,p}(t)P_i \quad a \le t \le b, \tag{10}$$

where the $\{P_i\}$ are the control points, and the $\{N_{i,p}(t)\}$ are the pth-degree B-spline basis functions defined on the non-periodic knot vector

$$U = \{\underbrace{a, ..., a}_{p+1}, t_{p+1}, ..., t_{n_n - p - 1}, \underbrace{b, ..., b}_{p+1}\}, \tag{11}$$

The polygon formed by the $\{P_i\}$ is called the control polygon, and its calculation is the objective of this work. Three steps are required to compute a point on a B-spline curve at a fixed t value: (1) find the knot span in which t lies; (2) compute the nonzero basis functions; (3) multiply the values of the nonzero basis functions with the corresponding control points. A third-degree B-Spline with seven control points is shown in Fig. 1.

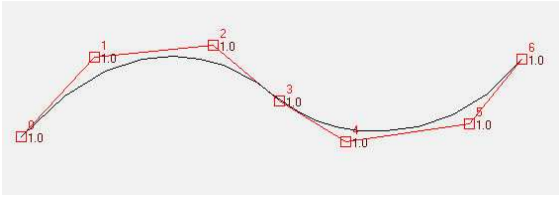


Fig. 1 B-Spline curve

In this work, we define the joint angle for joint j as

$$q_j(t) = \sum_{i=0}^{m}N_{i,3}(t)P_i^j \tag{12}$$

where $j = 1, 2, ..., n$ with degree 3, P_i^j represents control points for joint j, and the total number of control points for joint j is $m+1$. The relationship between the degree p, the number of control points $m+1$, and the number of knots n_m+1 is defined by $n_m = m + p + 1$.

4. OPTIMAL TRAJECTORY PLANNING

In this section, we illustrate the general method for optimal trajectory planning. The problem is defined as: given the start and end position of the end-effector, a smooth path for the end-effector is determined by the minimum jerk model, and then we determine the joint angles for the optimal trajectory planning. The overall procedure is shown in Fig. 2. The inputs to the algorithm are the start and end target points of the motion, the via point position for a curved path in the case of obstacle avoidance, the DH parameters for the manipulator, and the time

Copyright © 2008 by ASME

to travel along the path. Note that the absolute time is not very important; it is the relative time at that instance that determines the shape of the velocity. The first part of this section is to demonstrate the optimization-based procedure for determining the configurations for the start and end target points. The second part is to illustrate the trajectory planning algorithm.

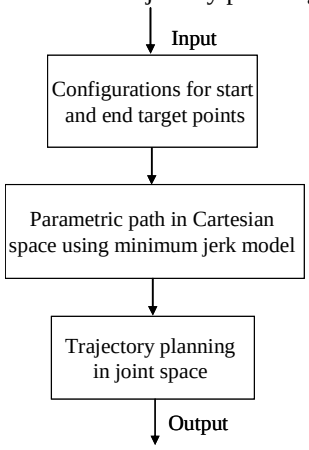


Fig. 2 Flow chart of trajectory planning procedure

4.1. Configurations for Start and End Target Points

Given the start and end target points, the first step is to determine the start configuration $\mathbf{q}^0$ and end configuration $\mathbf{q}^f$, or $q_j(0)$ and $q_j(t_f)$ in Eq. (12), corresponding to $t=0$ and $t=t_f$. This will transfer to the problem to determine P_0^j and P_m^j because the first and last control points of each joint angle are also on the B-slpine curve. We formulate this problem as follows:

Find: P_0^j or P_m^j

to minimize: $\sum_{i=1}^{n}\sum_{k=1}^{n_t}\tau_i^2(t_k)$

subject to:

$$\| \mathbf{x}(\mathbf{q}(0)) - \mathbf{p}(0) \| \le \varepsilon \quad \text{or} \quad \| \mathbf{x}(\mathbf{q}(t_f)) - \mathbf{p}(t_f) \| \le \varepsilon \tag{13}$$

$$q_j^L \le P_0^j \le q_j^U \ \text{ or } \ q_j^L \le P_m^j \le q_j^U$$

where τ_i is the joint torques from Eq. (8), $\mathbf{p}(t)$ is the path obtained from the minimum jerk model in Cartesian space in the Appendix with the detail, n_t is the total number of discretized time points on the path, and $\mathbf{x}(\mathbf{q}(t))$ is the position vector of the end-effector at time t. In the above formulation, we transfer the joint limit constraints from $q_j^L \le q_0^j \le q_j^U$ to $q_j^L \le P_0^j \le q_j^U$ because B-Spline curves have strong convex hull property.

4.2. Planning Formulation

After the geometric path and configurations for the start and end target points are determined, we can formulate the optimal trajectory planning problem as follows:

Find: Control points P_i^j, $i=1,...,m-1$, $j=1,...,n$

to minimize

$$\text{cost} = w_1\sum_{i=1}^{n}\sum_{k=1}^{n_t}\tau_i^2(t_k) + w_2 f_{inconsistency} + w_3 f_{nonsmoothness} \tag{14}$$

subject to:

$$\| \mathbf{x}(\mathbf{q}(t)) - \mathbf{p}(t) \| \le \varepsilon,\ q_j^L \le P_i^j \le q_j^U$$

where $0 \le t \le t_f$; ε is a small positive number as the tolerance; and w_1, w_2, and w_3 are the weights added to each performance index. The total number of design variables is $m-1$ for one joint since the beginning and end control points are the same as the joint angles determined by the initial and final configuration prediction of the robotic manipulator determined above. Discrete time moments t_i on the path are chosen as the components for the knot vector.

(1) *The joint torque* is obtained from the recursive formulation in Eq. (8).

(2) *The inconsistency function*: By comparing the two configurations (the initial configuration $\mathbf{q}^0$ and the end configuration $\mathbf{q}^f$), an overall changing trend of each joint (increasing or decreasing) can be predicted to avoid the abrupt change of the joint velocity. As a result, the consistency between the joint rate change (first derivative) and the predicted overall trend is evaluated and will be added to the cost function. The detailed formulation of this consistency is as follows:

$$\left.\begin{array}{l}\mathbf{x}_0 \to \mathbf{q}^0 \\ \mathbf{x}_f \to \mathbf{q}^f\end{array}\right\} \to trend_j = \begin{cases} 1 & \text{if } (q_j^f - q_j^0) \ge 0 \\ -1 & \text{if } (q_j^f - q_j^0) < 0 \end{cases} \tag{15}$$

and

$$f_{inconsistency} = \sum_{j=1}^{n}\left(\left|sign(\dot{q}_j(t)) - trend_j\right| + 1\right)\left|\dot{q}_j(t)\right| \tag{16}$$

where $$sign(\dot{q}_j(t)) = \begin{cases} 1 & \text{if } \dot{q}_j(t) \ge 0 \\ -1 & \text{if } \dot{q}_j(t) < 0 \end{cases}. \tag{17}$$

The (+1) in Eq. (16) is to make the amplitude of the joint rate change still have an effect towards optimizing a smooth joint trajectory when the first term within the parenthesis is evaluated to be zero. The multiplication with the amplitude of this joint change rate is to enforce the underlying assumption that the smaller the joint angle change rate, the smoother the joint trajectory. It also has a significant effect on the optimization process, not only by qualifying the consistency, but also by quantifying it to avoid the zero gradient of this objective that is characteristic of an ill-stated optimization problem statement.

(3) *The nonsmoothness function*: The second derivative of the joint trajectory is considered in a nonsmoothness function as

$$f_{nonsmoothness} = \sum_{j=1}^{n}(\ddot{q}_j(t))^2 \tag{18}$$

Copyright © 2008 by ASME

Once the control points of the joint curves are selected by the iterative optimization algorithm, the cost function of Eq. (14) can be integrated to obtain the total cost at any point along the path. The same principle applies to the distance, where the total deviation along the path can be obtained by the integration of the distance between the calculated and desired paths from the start to the end points. Since each joint's profile has $m+1$ control points, the total number of the design variables will be $n(m+1)$ initially. In our calculation, the joint values at the start and end have been obtained directly using the configuration prediction algorithm, where we need only to calculate the remaining $m-1$ control points for each joint, i.e., the design variables for the optimization are reduced to $n(m-1)$.

5. ILLUSTRATIVE EXAMPLES

In this section, we present two examples to demonstrate the proposed method. The first is a 6-DOF robotic manipulator planning example, and the second is a 15-DOF human upper body motion prediction example.

Fig. 3 shows the 6-DOF manipulator. The D-H table is shown in Table 1. Table 2 denotes the link masses and center of masses, and Table 3 represents the moment of inertia. The ranges of motions are $-2.793 \le q_1 \le 2.793$, $-3.927 \le q_2 \le 0.785$, $-0.785 \le q_3 \le 3.927$, $-1.920 \le q_4 \le 2.967$, $-1.745 \le q_5 \le 1.745$, and $-4.643 \le q_6 \le 4.643$.

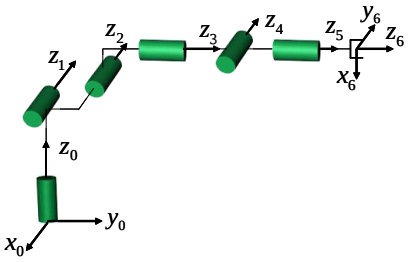


Fig. 3 A 6-DOF robot

Table 1 D-H Table

Joint	θ_i	d_i (mm)	α_i	a_i (mm)
1	$q_1+\pi/2$	0	$-\pi/2$	0
2	q_2	149.09	0	431.80
3	$q_3+\pi/2$	0	$\pi/2$	-20.32
4	q_4	433.07	$-\pi/2$	0
5	q_5	0	$\pi/2$	0
6	q_6	56.25	0	0

Table 2 Link Masses and Center of Gravity

Link	Mass (kg)	$\bar{x}$ (m)	$\bar{y}$ (m)	$\bar{z}$ (m)
1	2.27	0	0	0.073
2	15.91	-0.4318	0	0
3	6.82	0	0	0.1
4	3.18	0	0.1	0
5	0.91	0	0	0.01
6	2.75	0	0	0.08018

Table 3 Moment of Inertia

Link	I_{xx} ($\mathrm{kg\cdot cm^2}$)	I_{yy} ($\mathrm{kg\cdot cm^2}$)	I_{zz} ($\mathrm{kg\cdot cm^2}$)
1	84.58	170.9	117.6
2	62.85	2323	2369
3	132.6	416.7	329.2
4	106.3	3.145	103.1
5	33.97	33.97	4.396
6	92.4	92.4	6.109

Based on simulation experiments, a set of weights (10, 1000, 100) have been selected for w_1, w_2, and w_3 with the examples, and all optimization problems are solved using the software SNOPT (Gill et al., 2002), which uses sequential quadratic programming.

5.1. Point-to-point example

The end-effector moves from the target point $\mathbf{p}_1 = [-0.589 \quad 0.723 \quad 0.02]^T$, in meters, to the end point $\mathbf{p}_2 = [0.200 \quad -0.288 \quad 0.764]^T$, in meters; the external load is $\mathbf{F}^6 = [-5 \quad 0 \quad -10]^T$ in N, and the moment is $\mathbf{M}^6 = [0 \quad 0 \quad 0]^T$ in Nm. For the start point $\mathbf{p}_1$, the predicted configuration is $q_1 = -0.0315$, $q_2 = -0.4307$, $q_3 = 0.1473$, $q_4 = 1.5340$, $q_5 = 0.4902$, $q_6 = -0.9824$. For the end target point $\mathbf{p}_2$, the predicted configuration is $q_1 = 1.0057$, $q_2 = -0.3790$, $q_3 = 0.4138$, $q_4 = 1.4042$, $q_5 = 0.7089$, $q_6 = -2.1458$. In Fig. 4, the small spheres on the path are the constraints enforced on the end-effector when predicting the joint B-Splines. From the time stamps of the shown snapshots, it is easy to observe that the end-effector moves more slowly at the start and end than in the middle. This is the so-called bell-shape velocity profile, a characteristic of a smooth and natural movement (Flash and Hogan, 1985), and the predictability of this profile is actually the strength of the minimum jerk model. The predicted joint profiles for the 6 joints are shown in Fig. 5. From the joint profiles, we can see that each joint moves smoothly towards the final position.

5.2. Curved and obstacle avoidance example

For curved and obstacle avoidance movements, it is assumed that, in the motion between the end points, the end-effector is

Copyright © 2008 by ASME

required to pass through a third specified point (for example, an artificial intelligence engine can provide a via point to pass so as to go around the obstacle by examining the diameter of the obstacle). Fig. 6 shows the path with a via-point $\mathbf{p}_v = [0.15 \quad 0.5 \quad 0.8]^T$. The big red spheres are the start and end points. The teal sphere is the via point. The blue spheres are the discretized points on the path. The curve is the Cartesian path predicted using the minimum jerk model. The joint profiles shown in Fig. 7 also indicate the smooth movement of each joint.

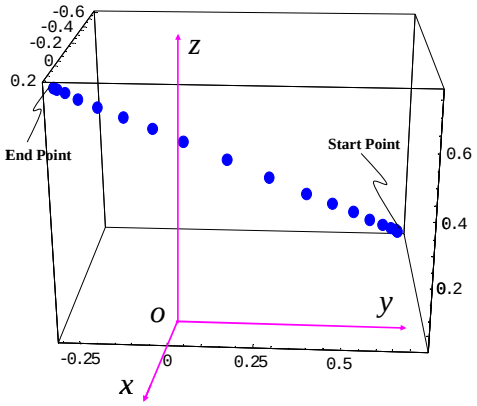


Fig. 4 Points on the path in Cartesian space

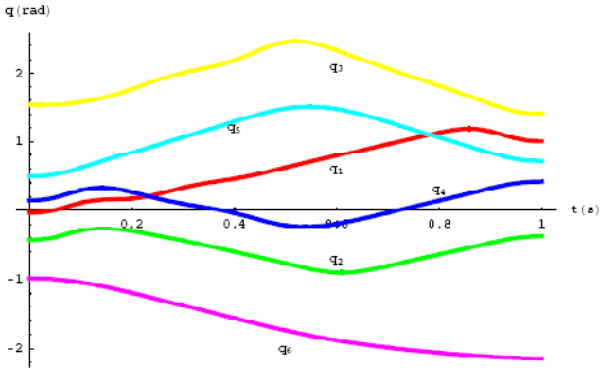


Fig. 5 Joint profiles for point-to-point planning

Consider another example that the left hand moves from the starting point $[32.95 \quad 42.40 \quad 37.30]^T$ to the end point $[50.51 \quad 7.04 \quad 49.13]^T$ holding one 2kg object. The human upper body model has total 15 degrees of freedom (Abdel-Malek, et al. 2006). Snapshots of the predicted results are shown in Fig. 8. The predicted joint profiles for the 15 DOFs are shown in Figs. 9-11.

Consider left hand curved motion prediction holding a 2kg object. The start point is $[15.35 \quad 75.77 \quad 43.53]^T$, the via point is $[43.61 \quad 36.20 \quad 38.74]^T$, and the end point is $[35.95 \quad -6.412 \quad 52.05]^T$. Snapshots for the motion are shown in Figs. 12. The predicted joint profiles are shown in Figs. 13-15.

As shown in the two examples above, the proposed method and algorithm can predict smooth and ideal movements for robotic or human models with any number of degrees of freedom.

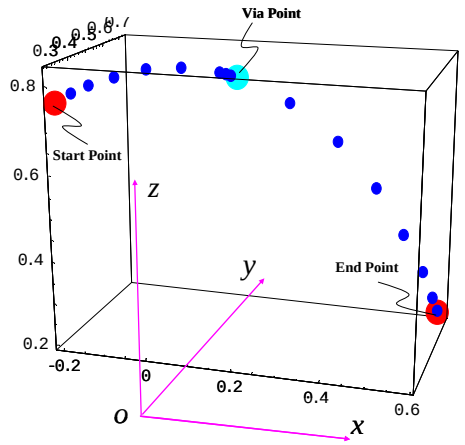


Fig. 6 Path in Cartesian space

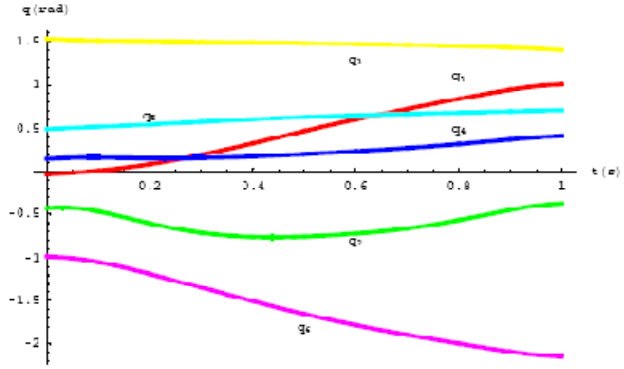
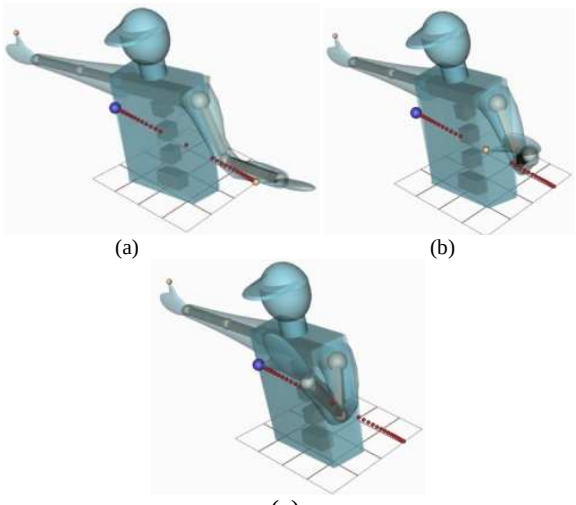


Fig. 7 Joint angles for via point planning



(a) (b)

(c)

Fig. 8 Motion at time (a) 0 s, (b) $0.5t_f$, and (c) t_f

Copyright © 2008 by ASME

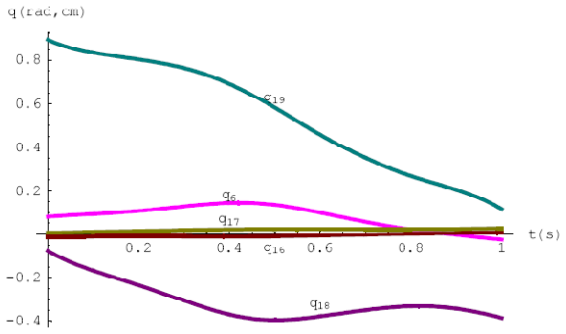


Fig. 9 Joint profiles for $q_1 \sim q_5$

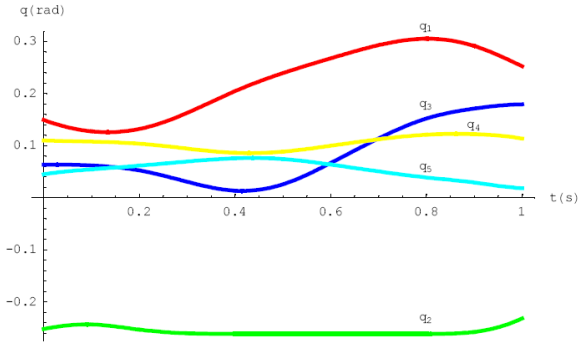


Fig. 10 Joint profiles for q_6, $q_{16} \sim q_{19}$

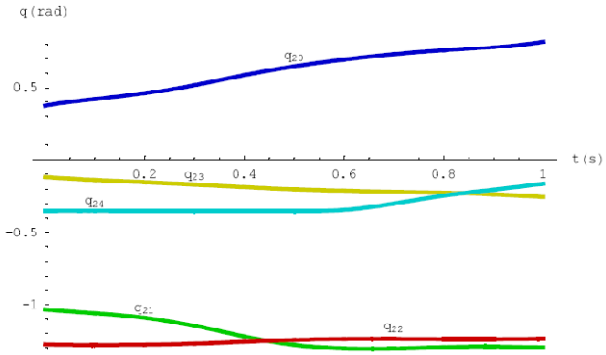
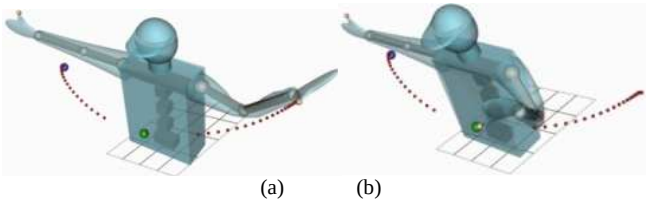
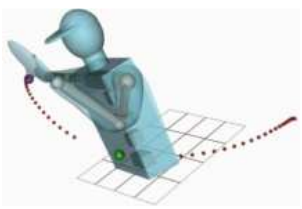


Fig. 11 Joint profiles for $q_{20} \sim q_{24}$



(a) (b)



(c)

Fig. 12 At time (a) 0 s, (b) $0.5t_f$, and (c) t_f

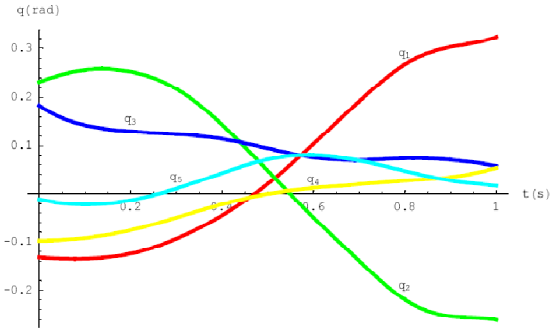


Fig. 13 Joint profiles for $q_1 \sim q_5$

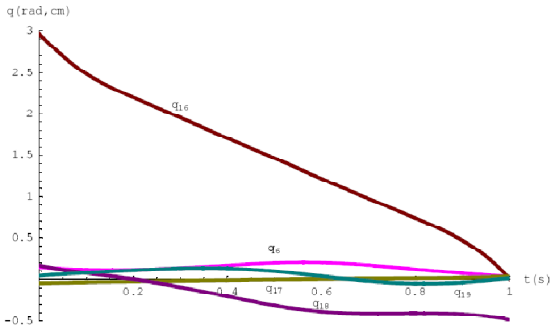


Fig. 14 Joint profiles for q_6, $q_{16} \sim q_{19}$

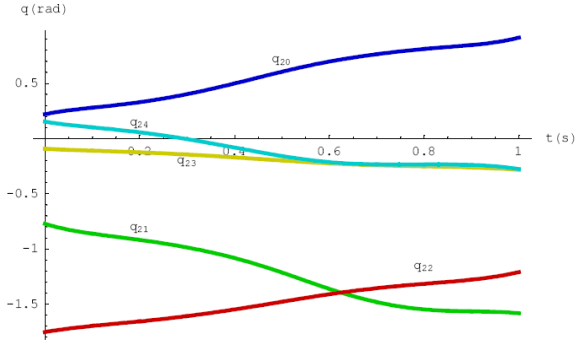


Fig. 15 Joint profiles for $q_{20} \sim q_{24}$

6. CONCLUSION

This paper proposed a general method for predicting joint profiles that is broadly applicable to both linear (straight) and nonlinear (curved) path trajectories of robotic manipulators. Nonlinear paths are applicable to obstacle-avoidance problems, where trajectories deviate from the typical linear point-to-point motion with minimum jerk. A mathematical formulation that is applicable to any number of degrees of freedom and that predicts joint profiles as functions of time was demonstrated. The objective is to minimize a dynamic effort function and maximize smoothness and consistency functions. This work introduces a framework of trajectory planning for any serial mechanisms with any number of degrees of freedom. Other challenges include how to consider other factors within this optimization-based method, such as moving obstacles or efficiency (minimum time planning). Ultimately, the general

Copyright © 2008 by ASME

trajectory planning system will include either the given Cartesian path or the minimum jerk path considering other constraints within this framework.

7. ACKNOWLEDGMENTS

This work was supported in part by the U.S. Army TACOM Project.

REFERENCES

[1] Abdel-Malek, K., Mi, Z., Yang, J., and Nebel, K. (2006) "Optimization-based trajectory planning of the human upper body," *Robotica*, Vol. 24, 683-697.

[2] Alexander, R. M. (1997) "A minimum energy cost hypothesis for human arm trajectories," *Biological Cybernetics*, Vol. 76, No. 2, 1997.

[3] Antonelli, G. and Chiaverini, S. (1998) "Adaptive tracking control of underwater vehicle-manipulator systems," *IEEE Conference on Control Applications - Proceedings*, Vol. 2, pp. 1089-1093.

[4] Barraquand, J. and Latombe, J.C. (1991) "Robot motion planning: a distributed representation approach," *International Journal of Robotics Research*, Vol. 10, No. 6.

[5] Bobrow, J.E. (1988) "Optimal robot path planning using the minimum-time criterion," *IEEE Journal of Robotics and Automation*, Vol. 4, No. 4, pp. 443-450.

[6] Bryson, A.E. and Ho, Y.C. (1975) *Applied Optimal Control*, Hempshire Publ. Co.

[7] Constantinescu, D. and Croft, E.A. (2000) "Smooth and time-optimal trajectory planning for industrial manipulators along specified paths," *Journal of Robotic Systems*, Vol. 17, No. 5, pp. 233-249.

[8] Denavit, J. and Hartenberg, R.S. (1955) A kinematic notation for lower-pair mechanisms based on matrices. *Journal of Applied Mechanics*, Vol. 77, pp. 215-221.

[9] Flash, T. and Hogan, N. (1985) "The coordination of arm movements: an experimentally confirmed mathematical model," *The Journal of Neuroscience*, Vol. 5, No. 7, pp. 1688-1703.

[10] Furuno, S., Yamamoto, M., and Mohri, A. (2003) "Trajectory planning of mobile manipulator with stability considerations," *Transactions of the Japan Society of Mechanical Engineers, Part C*, Vol. 69, No. 5, pp. 1330-1335.

[11] Gasparetto, A. and Zanotto, V. (2007) "A new method for smooth trajectory planning of robot manipulators," *Mechanism and Machine Theory*, Vol. 42, Issue 4, pp. 455-471.

[12] Gill, P., Murray, W., and Saunders, A. (2002) "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Journal of Optimization*, Vol. 12, No. 4, pp. 979-1006.

[13] Hirakawa, R. and Kawamura, A. (1997) "Trajectory planning of redundant manipulators with consumed energy minimization by variational approach," *Transactions of the Institute of Electrical Engineers of Japan, Part D*, v 117-D, n 6, June 1997, p 724-32.

[14] Kawato, M., Isobe, M., Maeda, Y. and Suzuki, R. (1988) "Coordinates transformation and learning control for visually-guided voluntary movement with iteration: a Newton-like method in a function space," *Biological Cybernetics*, Vol. 59, pp. 161-177.

[15] Kim, J., Yang, J., and Abdel-Malek, K. (2007) "Planning load-effective dynamic motions of redundant manipulators," *ASME International Design Engineering Technical Conferences*, September 4-7, 2007, Las Vegas, NV.

[16] Li, H. and Ceglarek, D., (2002) "Optimal trajectory planning for material handling of compliant sheet metal parts," *Journal of Mechanical Design, Transactions of the ASME*, Vol. 124, No. 2, pp. 213-222.

[17] Park, J. and Bobrow, J. (2004) "Reliable computation of minimum-time motions for manipulators moving in obstacle fields using a successive search for minimum-overload trajectories," *Journal of Robotic Systems*, Vol. 22, Issue 1, pp. 1-14.

[18] Piazzi, A. and Visioli, A. (2000) "Global minimum-jerk trajectory planning of robot manipulators," *IEEE Transactions on Industrial Electronics*, Vol. 47, No. 1, pp. 140-149.

[19] Piegl, L. and Tiller, W. (1997) *The NURBS Book*, Springer Verlag.

[20] Pontryagin, L.S., Boltyanskii, V., Gamkrelidze, R., and Mishchenko, E. (1962) *The Mathematical Theory of Optimal Processes*, Interscience Publishers Inc., New York.

[21] Pugazhenthi, S., Nagarajan, T., and Singaperumal, M. (2002) "Optimal trajectory planning for a hexapod machine tool during contour machining," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 216, No. 12, pp. 1247-1256.

[22] Saramago, S.F.P. and Steffen, V. (1998) "Optimization of the trajectory planning of robot manipulators taking into account the dynamics of the system," *Mechanism & Machine Theory*, Vol. 33, No. 7, pp. 883-894.

[23] Saramago, S.F.P. and Steffen, V. (2000) "Optimal trajectory planning of robot manipulators in the presence of moving obstacles," *Mechanism and Machine Theory*, Vol. 35, No. 8, pp. 1079-1094.

[24] Saramago, S.F.P. and Steffen, V. (2001) "Trajectory modeling of robot manipulators in the presence of obstacles," *Journal of Optimization Theory and Applications*, Vol. 110, No. 1, pp. 17-34.

[25] Saramago, S.F.P. and Ceccarelli, M. (2002) "An optimum robot path planning with payload constraints," *Robotica*, Vol. 20, Pt. 4, pp. 395-404.

[26] Shiller, Z. and Dubowsky, S. (1989) "Robot path planning with obstacles, actuator, gripper, and payload constraints," *International Journal of Robotics Research*, Vol. 8, No. 6, pp. 3-18.

[27] Shin, K.G. and McKay, N.D. (1986) "A dynamic programming approach to trajectory planning of robotic manipulators," *IEEE Transactions on Automatic Control* Vol. 31, No. 6, pp. 491-500.

[28] Toogood, R.W. (1989) "Efficient robot inverse and direct dynamics algorithms using micro-computer based symbolic generation," *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 3, pp. 1827-1832.

[29] Uno Y., Kawato, M., and Suzuki, R. (1989) "Formation and control of optimal trajectory in human multijoint arm movement," *Biological Cybernetics*, Vol. 61, pp. 89-101.

[30] Valero, F., Mata. V., and Besa, A. (2006) "Trajectory planning in workspaces with obstacles taking into account the dynamic robot behaviour," *Mechanism and Machine Theory*, Vol. 41, Issue 5, pp. 525-536.

[31] Yamane K., Kuffner J., Hodgins J., "Synthesizing Animations of Human Manipulation Tasks" SIGGRAPH2004.

[32] "Synthesizing Animations of Human Manipulation Tasks"

[33] SIGGRAPH2004

[34] Yang, J., Abdel-Malek, K., and Nebel, K. (2005) "The reach envelope of a 9 degree of freedom model of the upper extremity," *International Journal of Robotics and Automation.* Vol. 20, No. 4, pp. 240-259.

[35] Yun, W.M. and Xi, Y.G. (1996) "Optimum motion planning in joint space for robots using genetic algorithms," *Robotics and Autonomous Systems*, Vol. 18, pp. 373-393.

[36] Zha, X. (2002) "Optimal pose trajectory planning for robot manipulators," *Mechanism and Machine Theory*, Vol. 37, Issue 10, pp. 1063-1086.

[37] Zhou, Z. and Nguyen, C. (1997) "Globally optimal trajectory planning for redundant manipulators using state space augmentation method," *Journal of Intelligent and Robotic Systems: Theory & Applications*, Vol. 19, No. 1, May 1997, pp. 105-117.

APPENDIX: MINIMUM JERK MODEL

The 3D minimum jerk model is used as the geometric path for robotic manipulators based on the 2D minimum jerk (Flash and Hogan, 1985). This model entails unconstrained point-to-point movement and curved point-to-point (via point) movement.

(1) Point-to-Point Model

Given a path trajectory as a parametric curve in space, such as

 Copyright © 2008 by ASME

$$\mathbf{x}(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix}^T, \quad \text{(A1)}$$

the first derivative is the velocity and the second derivative is the acceleration. The third derivative is the jerk along a path and is best measured by an integration over the motion time along the path, such that

$$C = \frac{1}{2}\int_0^{t_f} \left(\left(\frac{d^3x}{dt^3}\right)^2 + \left(\frac{d^3y}{dt^3}\right)^2 + \left(\frac{d^3z}{dt^3}\right)^2\right)dt . \quad \text{(A2)}$$

In order to include the concept of minimum jerk as a driving function in the design (or prediction) of a path trajectory, we will adapt some mathematics to allow for the calculation of minima and maxima. Generally, for any function $x(t)$, which is sufficiently differentiable in the interval $0 \le t \le t_f$, and for any performance index $L[t, x, \dot{x}, \ddot{x}, ..., d^n x / dt^n]$, which is integrable over the same interval, the unconstrained cost function

$$C(x(t)) = \int_0^{t_f} L\left[t, x, \dot{x}, \ddot{x}, ..., \frac{d^n x}{dt^n}\right] dt \quad \text{(A3)}$$

assumes an extremum when $x(t)$ is the solution of Euler-Poisson equation

$$\frac{\partial L}{\partial x} - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) ... + (-1)^n \frac{d^n}{dt^n}\frac{\partial L}{\partial (x^n)} = 0 . \quad \text{(A4)}$$

In our case,

$$L = \tfrac{1}{2}((\ddot{x})^2 + (\ddot{y})^2 + (\ddot{z})^2), \quad \text{(A5)}$$

and the Euler-Poisson equation

$$\frac{d^3}{dt^3}\left(\frac{\partial \ddot{x}^2}{\partial \ddot{x}}\right) = \frac{d^3}{dt^3}\left(\frac{\partial \ddot{y}^2}{\partial \ddot{y}}\right) = \frac{d^3}{dt^3}\left(\frac{\partial \ddot{z}^2}{\partial \ddot{z}}\right) = 0 . \quad \text{(A6)}$$

We can get

$$\frac{d^6x}{dt^6} = 0; \quad \frac{d^6y}{dt^6} = 0; \quad \frac{d^6z}{dt^6} = 0 . \quad \text{(A7)}$$

If we assume that the movement starts and ends with zero velocity and acceleration, then we have

$$\begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + \begin{bmatrix} x_0 - x_f \\ y_0 - y_f \\ z_0 - z_f \end{bmatrix} (15\eta^4 - 6\eta^5 - 10\eta^3) \quad \text{(A8)}$$

where $\eta = t/t_f$, x_0, y_0, z_0 are the initial end-effector position coordinates at $t = 0$, and x_f, y_f, z_f are the final end-effector coordinates at $t = t_f$.

(2) Curved Point-to-Point Model

Consider motion along a curve, where the end-effector has to traverse a specified point (called a via point) during its motion. Study of such movements will provide a way to model obstacle-avoidance motions. For example, if there is an obstacle in the path between two end target points, by examining the largest diameter of the obstacle, an artificial intelligence engine can determine and introduce a via point through which to pass in order to go around the obstacle. The objective becomes to generate the smoothest motion that brings the end-effector from the initial position to the final position at a given time while moving through a via point at an unspecified time. The requirement that the end-effector move through a specified via point defines the equality constraints on the end-effector position coordinate $\mathbf{x}(t) = [x(t), y(t), z(t)]^T$; i.e., if the location of the via point with respect to a Cartesian coordinate system is given by $\mathbf{x}_v = [x_v, y_v, z_v]^T$, the equality constraints are

$$\mathbf{x}(t_v) = \mathbf{x}_v , \quad \text{(A9)}$$

where the time t_v at which the end-effector has to pass through the via point is not specified a priori, but rather is derived form the optimization procedure to minimize the jerk function. Problems of this kind are known as dynamic optimization problems with interior point equality constraints, and techniques have been established for their solution (Bryson and Ho, 1975).

We will now introduce the dynamic optimization method. Generally, optimization problems similar to the problem solved here involve a system that can be described by a set of nonlinear differential equations:

$$\dot{\mathbf{s}} = \mathbf{f}[\mathbf{s}(t), \mathbf{u}(t), t], \quad \text{(A10)}$$

where $\mathbf{s}(t)$ is an n vector function of state variables and $\mathbf{u}(t)$ is an m vector control function. The problem is to find the control $\mathbf{u}(t)$, such that carrying the system from an initial state $\mathbf{s}(0)$ to a final state $\mathbf{s}(t_f)$ optimizes the cost function $C(t)$. $C(t)$ is defined as

$$C(t) = \int_0^{t_f} L[\mathbf{s}(t), \mathbf{u}(t), t]dt , \quad \text{(A11)}$$

where $L[\mathbf{s}(t), \mathbf{u}(t), t]$ is the performance index. This problem can be solved using the Pontryagin method (Pontryagin et al., 1962). One defines an n component co-state (Lagrange multipliers) vector $\boldsymbol{\lambda}(t)$ and a scalar Hamiltonian

$$H[\mathbf{s}(t), \mathbf{u}(t), t)] = L[\mathbf{s}(t), \mathbf{u}(t), t] + \boldsymbol{\lambda}^T(t)\mathbf{f}[\mathbf{s}(t), \mathbf{u}(t), t] . \quad \text{(A12)}$$

The following differential equations define the necessary conditions for a minimum to exist:

$$\dot{\mathbf{s}} = \mathbf{f}[\mathbf{s}(t), \mathbf{u}(t), t] \quad \text{(A13)}$$

$$\dot{\boldsymbol{\lambda}}(t) = -\frac{\partial H}{\partial \mathbf{s}} \quad \text{(A14)}$$

$$\frac{\partial H}{\partial \mathbf{u}} = 0 \quad \text{(A15)}$$

For optimal control problems with interior point equality constraints, there is a set of constraints at some time t_v:

$$\boldsymbol{\Psi}(\mathbf{s}(t_v), t_v) = \mathbf{0} , \quad \text{(A16)}$$

where $\boldsymbol{\Psi}$ is a ρ-component vector function. These interior point constraints can augment the cost function by a Lagrange multiplier vector $\boldsymbol{\kappa}$ so that the new cost function is

$$C = \boldsymbol{\kappa}^T\boldsymbol{\Psi} + \int_0^{t_f} L[\mathbf{s}(t), \mathbf{u}(t), t]dt . \quad \text{(A17)}$$

The solution is obtained by allowing discontinuities in the co-state variables (Lagrange coefficients) $\boldsymbol{\lambda}(t)$'s and in the Hamiltonian $H[t, \boldsymbol{\lambda}(t), \mathbf{s}(t)]$. One can define a vector of

Copyright © 2008 by ASME

Lagrange coefficients $\boldsymbol{\lambda}^+(t)$ and Hamiltonian $H^+(t)$ for $t \geq t_v$, and a vector $\boldsymbol{\lambda}(t)$ and Hamiltonian $H(t)$ for $t \leq t_v$. At time t_v, these variables satisfy the equations

$$\boldsymbol{\lambda}^-(t_v) = \boldsymbol{\lambda}^+(t_v) + \boldsymbol{\kappa}^T \frac{\partial \boldsymbol{\Psi}}{\partial \mathbf{s}(t_v)} \tag{A18}$$

and

$$H^-(t_v) = H^+(t_v) - \boldsymbol{\kappa}^T \frac{\partial \boldsymbol{\Psi}}{\partial t_v}. \tag{A19}$$

The ρ components of $\boldsymbol{\kappa}$ are determined by the constraint Eq. (A16) while time t_v is fully determined by Eq. (A19).

For our problem, we define a state vector $\mathbf{s}(t) = [x, y, z, u, v, w, a, b, c]^T$ and a control vector $\mathbf{u}(t) = [\delta, \gamma, \eta]^T$. The components of these vectors are defined by the system equations

$$\begin{aligned}
&\dot{x} = u \\
&\dot{y} = v \\
&\dot{z} = w \\
&\dot{u} = \ddot{x} = a \\
&\dot{v} = \ddot{y} = b \\
&\dot{w} = \ddot{z} = c \\
&\dot{a} = \dddot{x} = jerk_x = \delta \\
&\dot{b} = \dddot{y} = jerk_y = \gamma \\
&\dot{c} = \dddot{z} = jerk_z = \eta
\end{aligned} \tag{A20}$$

and the Hamiltonian is

$$\begin{aligned}
H = {} & \lambda_x u + \lambda_y v + \lambda_z w + \lambda_u a + \lambda_v b + \lambda_w c \\
& + \lambda_a \delta + \lambda_b \gamma + \lambda_c \eta + \frac{1}{2}(\delta^2 + \gamma^2 + \eta^2)
\end{aligned}. \tag{A21}$$

The conditions necessary for a minimum to exist are

$$\begin{aligned}
&-\frac{d\lambda_x}{dt} = 0 \quad -\frac{d\lambda_u}{dt} = \lambda_x \quad -\frac{d\lambda_a}{dt} = \lambda_u \\
&-\frac{d\lambda_y}{dt} = 0, \quad -\frac{d\lambda_v}{dt} = \lambda_y, \quad -\frac{d\lambda_b}{dt} = \lambda_v \\
&-\frac{d\lambda_z}{dt} = 0 \quad -\frac{d\lambda_w}{dt} = \lambda_z \quad -\frac{d\lambda_c}{dt} = \lambda_w
\end{aligned} \tag{A22}$$

The necessary conditions on the control variables are

$$\begin{aligned}
&\frac{\partial H}{\partial \delta} = \delta + \lambda_a = 0 \\
&\frac{\partial H}{\partial \gamma} = \gamma + \lambda_b = 0 \\
&\frac{\partial H}{\partial \eta} = \eta + \lambda_c = 0
\end{aligned} \tag{A23}$$

For our specific problem, the constraints are at the end-effector at time t_v:

$$\begin{aligned}
&x(t_v) = x_v \\
&y(t_v) = y_v \\
&z(t_v) = z_v
\end{aligned} \tag{A24}$$

The Hamiltonian H^- for all times $t \leq t_v$ is

$$\begin{aligned}
H^- = {} & \lambda_x^- u^- + \lambda_y^- v^- + \lambda_z^- w^- + \lambda_u^- a^- + \lambda_v^- b^- + \lambda_w^- c^- + \lambda_a^- \delta^- + \lambda_b^- \gamma^- + \lambda_c^- \eta^- \\
& + \frac{1}{2}((\delta^-)^2 + (\gamma^-)^2 + (\eta^-)^2)
\end{aligned} \tag{A25}$$

and the Hamiltonian H^+ for all times $t \geq t_v$ is

$$\begin{aligned}
H^+ = {} & \lambda_x^+ u^+ + \lambda_y^+ v^+ + \lambda_z^+ w^+ + \lambda_u^+ a^+ + \lambda_v^+ b^+ + \lambda_w^+ c^+ \\
& + \lambda_a^+ \delta^+ + \lambda_b^+ \gamma^+ + \lambda_c^+ \eta^+ + \frac{1}{2}((\delta^+)^2 + (\gamma^+)^2 + (\eta^+)^2)
\end{aligned} \tag{A26}$$

Since the constraint equations relate only to position, the only discontinuities are in λ_x, λ_y, and λ_z; therefore, according to Eq. (A18), we get

$$\begin{aligned}
&\lambda_x^- = \lambda_x^+ + \kappa_1 \\
&\lambda_y^- = \lambda_y^+ + \kappa_2 \\
&\lambda_z^- = \lambda_z^+ + \kappa_3
\end{aligned} \tag{A27}$$

While all the other Lagrange coefficients are continuous at $t = t_1$,

$$\begin{aligned}
&\lambda_u^- = \lambda_u^+ \\
&\lambda_v^- = \lambda_v^+ \\
&\lambda_w^- = \lambda_w^+ \\
&\lambda_a^- = \lambda_a^+ \\
&\lambda_b^- = \lambda_b^+ \\
&\lambda_c^- = \lambda_c^+
\end{aligned} \tag{A28}$$

Since time t_v is not explicitly specified, the Hamiltonian must be continuous at t_v according to Eq. (A19):

$$H^+(t_v) = H^-(t_v) \tag{A29}$$

$\mathbf{x}(t)$ for $t \leq t_v$ is obtained as follows:

$$\begin{aligned}
\mathbf{x}^-(\eta) = {} & \frac{t_f^5}{720}(\boldsymbol{\kappa}(\eta_v^4(15\eta^4 - 30\eta^3) + \eta_v^3(80\eta^3 - 30\eta^4) \\
& -60\eta^3\eta_v^2 + 30\eta^4\eta_v - 6\eta^5) + \mathbf{c}(15\eta^4 - 10\eta^3 - 6\eta^5)) + \mathbf{x}_0
\end{aligned} \tag{A30}$$

For times $t \geq t_v$, the expression is

$$\begin{aligned}
\mathbf{x}^+(\eta) = {} & \frac{t_f^5}{720}(\boldsymbol{\kappa}(\eta_v^4(15\eta^4 - 30\eta^3 + 30\eta - 15) \\
& + \eta_v^3(-30\eta^4 + 80\eta^3 - 60\eta^2 + 10)) \\
& + \mathbf{c}(-6\eta^5 + 15\eta^4 - 10\eta^3 + 1)) + \mathbf{x}_f \\
= {} & \mathbf{x}^-(\eta) + \boldsymbol{\kappa}\frac{t_f^5(\eta - \eta_v)^5}{120}
\end{aligned} \tag{A31}$$

Copyright © 2008 by ASME