Robotica (2006) volume 24, pp. 683–696. © 2006 Cambridge University Press
doi:10.1017/S0263574706002852 Printed in the United Kingdom

Optimization-based trajectory planning of the human upper body

Karim Abdel-Malek†*, Zan Mi†, Jingzhou Yang†* and Kyle Nebel‡

†*Virtual Soldier Research Program, Center for Computer-Aided Design, The University of Iowa, 111 Engineering Research Facility, Iowa City, IA 52242-1000 (USA)*
‡*U.S. Army TACOM/RDECOM, AMSRD-TAR-NAC/157, 6501 East 11 Mile Rd., Warren, MI 48397-5000 (USA)*

(Received in Final Form: April 3, 2006. First published online: July 3, 2006)

SUMMARY

This paper presents studies of the coordination of human upper body voluntary movement. A minimum-jerk 3D model is used to obtain the desired path in Cartesian space, which is widely used in the prediction of human reach movement. Instead of inverse kinematics, a direct optimization approach is used to predict each joint's profile (a spline curve). This optimization problem has four cost function terms: (1) *Joint displacement function* that evaluates displacement of each joint away from its neutral position; (2) *Inconsistency function*, which is the joint rate change (first derivative) and predicted overall trend from the initial target point to the final target point; (3) The *non-smoothness function* of the trajectory, which is the second derivative of the joint trajectory; (4) The *non-continuity function*, which consists of the amplitudes of joint angle rates at the initial and final target points, in order to emphasize smooth starting and ending conditions. This direct optimization technique can be used for potentially any number of degrees of freedom (DOF) system and it reduces the cost associated with certain inverse kinematics approaches for resolving joint profiles. This paper presents a high redundant upper-body modeling with 15 DOFs. Illustrative examples are presented and an interface is set up to visualize the results.

KEYWORDS: Planning trajectory; Biomechanics; Minimum jerk; B-splines; Cartesian space; Joint space; Direct optimization-based method.

1. INTRODUCTION

Trajectory planning of human upper-body movement is one of the most challenging problems in digital human simulation. Many tasks require the arm to move from its initial position to a specified target position without any constraints, or via a point for a curved path in case of obstacle avoidance.

Researchers have developed various trajectory planning methods for robotic systems considering different kinematic and dynamic criteria such as obstacle avoidance, singularity avoidance, time minimization, torque optimization, energy optimization, and other objective functions.[1–7] For a kinematically redundant system, the mapping between task–space trajectory and the joint–space trajectory is not unique. It admits an infinite number of joint–space solutions for a given task–space trajectory.

* Corresponding authors. E-mail: amalek@engineering.uiowa.edu, jyang@engineering.uiowa.edu

Flash and Hogan[8] presented a mathematical model that was shown to predict both the qualitative features and the quantitative details observed experimentally in planar, multi-joint arm movements. Uno *et al.*[9] proposed a mathematical model, which is formulated by defining an objective function, square of the rate of change of torque integrated over the entire movement. Kawato *et al.*[10] studied the problems of coordinate transformation from the desired trajectory to the body coordinates and motor command generation. This approach appears very attractive, but it lacks generalizability. Bobrow[2] presented a path-planning technique, which makes use of approximations of an initial feasible trajectory in conjunction with an iterative, nonlinear parameter optimization algorithm to produce time-optimal motions for a manipulator with three degrees of freedom in a workspace containing obstacles. A randomized planner was introduced by,[11] which 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 minimum jerk model[8] is a mathematical model that is shown to predict both the qualitative features and the quantitative details observed experimentally in planar, multi-joint arm movements. Coordination is modeled mathematically by defining an objective function, a measure of performance for any possible movement. 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. This model is a well-known one that is widely used in the prediction of human reach movement.[12–18]

Yun and Xi[19] used genetic algorithms for optimum motion planning in joint space for robots. Similarly, Constantinescu and Croft[20] put forth a smooth and time-optimal trajectory planning, which minimizes time under path constraints, torque limits, and torque rate limits.

Saramago *et al.*[21–23] have studied robot path planning by taking into consideration dynamic system, with payload constraints, and in the presence of moving obstacles. Pugazhenthi *et al.*[24] studied the optimal trajectory planning for Stewart platform-based machine tools. Li and Ceglarek[25] presented an optimal trajectory planning application for material handling of compliant sheet metal parts in which

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

they considered part permanent deformation, trajectory smoothness, and static obstacle avoidance.

Alexander[26] hypothesized that trajectories are chosen to minimize metabolic energy costs. Ohta*et al.*[27] presented a criterion minimizing the hand contact force change and muscle force change of a movement over time.

Existing approaches are applied to trajectory planning of manipulators, which normally have only two to three degrees of freedom or up to six at most. On the other hand, for realistic motion generation, human models normally have more than 10 DOFs. Moreover, the criteria used for motion planning will be quite different. For example, optimum time is always selected for the manipulator trajectory planning in application. But for human motion, this is not always important; instead, humans tend to adopt the motion with least discomfort and effort, and greatest smoothness. This paper presents a methodology for predicting and simulating the path generated by humans in a natural motion of the torso and upper extremity, which entails two parts: Minimum jerk model to define the path of human hand in Cartesian space and direct optimization method to find the motion in joint space that matches the desired Cartesian path.

2. HUMAN MODELING

To establish a systematic method for the biomechanical human modeling, researchers have implemented conventions for representing segmental links and joints. Perhaps the most important element of a joint is its function, which may vary according to the joint's location and physiology. The physiology becomes important when we discuss the loading conditions of a joint. As for kinematics, we shall address the function in terms of the number of DOFs associated with its overall movement. Muscle action, ligaments, and tendons at a joint are also important and contribute to the function.

For example, the elbow joint is considered a hinge or a one-DOF rotational joint (e.g., the hinge of a door) because it allows for flexibility and extension in the sagittal plane (Fig. 1) as the radius and ulna rotate about the humerus. We shall represent this joint by a cylinder that rotates about one axis and has no other motions (i.e., has one DOF). Therefore, we can now say that the elbow is characterized by one DOF and is represented as a cylindrical rotational joint also shown in Fig. 1.

Our shoulder model takes into consideration the final gross movement of the joint[28,29] as abduction/adduction (about the anteroposterior axis of the shoulder joint), flexion/extension, and transverse flexion/extension (about the mediolateral axis of the shoulder joint). Note that these motions provide for three rotational DOFs, with their axes intersecting at one point. This effectively gives rise to a spherical joint typically associated with the shoulder joint (Fig. 2). In addition, the upward/downward rotation of the scapula gives rise to two substantial translational (i.e., sliding) DOFs for a total of five DOFs in the shoulder complex. This model allows for consideration of the coupling between some of the joints, as is the case in the shoulder where muscles extend over more than one segment. When muscles are used to lift the arm in a rotational motion, a translational motion of the shoulder occurs inadvertently. In Fig. 2, the global coordinate system $\mathbf{x}_0\mathbf{y}_0\mathbf{z}_0$ is located on the hip joint with three orthogonal anterior–posterior, medial–lateral, and vertical axes. Two translational DOFs are along $\mathbf{x}_0$ and $\mathbf{z}_0$, respectively. The rest three DOFs are revolute joints with the axes perpendicular to each other.

The human upper body (torso, shoulder, and arm) is modeled using a total of 15 DOFs (Fig. 3) as described later. In Fig. 3, the global coordinate system is the same as Fig. 2; the torso includes six DOFs, the clavicle has two DOFs, and the arm entails seven DOFs. The two or three arrows such as $\mathbf{z}_{10}$, $\mathbf{x}_8$, and $\mathbf{x}_9$ denote that these vectors have the same direction. The thumb tip is the point of interest (end-effector).

Using Denavit–Hartenberg (D–H) method[30] to define the transformation matrix of one coordinate system with respect to another coordinate system, the position and orientation of each axis determine the four parameters $\theta_i, d_i, a_i, \alpha_i$ in Fig. 4, and thus determine the resulting (4×4) transformation matrix. To establish this matrix, one can observe that a vector ${}^{i}\mathbf{v}$ resolved in the ith coordinate system may be expressed in the $(i-1)th$ coordinate system (${}^{i-1}\mathbf{v}$) by performing four successive transformations as follows:

(a) A rotation about the $\mathbf{z}_{i-1}$ axis by an angle θ_i to align the $\mathbf{x}_{i-1}$ axis with the $\mathbf{x}_i$ axis (as shown in Fig. 4, $\mathbf{x}_{i-1}//\mathbf{x}_i$ and pointing in the same direction).
(b) A translation $\mathbf{x}_{i-1}$ along $\mathbf{z}_{i-1}$ by a distance of d_i units to make $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$ aligned.
(c) A translation along the $\mathbf{x}_i$ axis by a distance of a_i units to make the two origins of the i and $i-1$ systems coincide ($\mathbf{x}_i$ and the $\mathbf{x}_{i-1}$ will also be aligned).
(d) A rotation about the $\mathbf{x}_i$ axis by an angle α_i to coincide the two coordinate systems.

Each step given above is expressed by a *basic homogeneous rotation* or *translation matrix*. For example, $\mathbf{T}_{z,d}$ denotes a translation matrix along $\mathbf{z}$ with d units. The product of these four matrices results in a composite

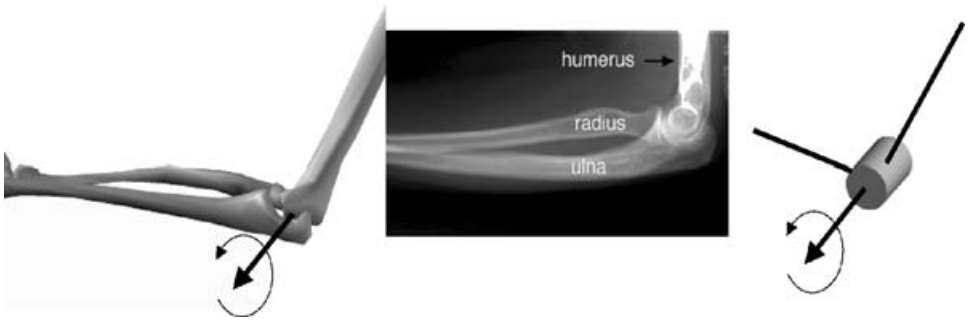

Fig. 1. One-DOF elbow.

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

homogeneous transformation matrix

$$
{}^{i-1}\mathbf{T}_i = \mathbf{T}_{z,d}\mathbf{T}_{z,\theta}\mathbf{T}_{x,a}\mathbf{T}_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{i-1}\mathbf{T}_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{1}
$$

where $d_7 = q_7$, $d_8 = q_8$, and $\theta_i = q_i$, $i = 1, \ldots, 6, 9, \ldots, 15$.

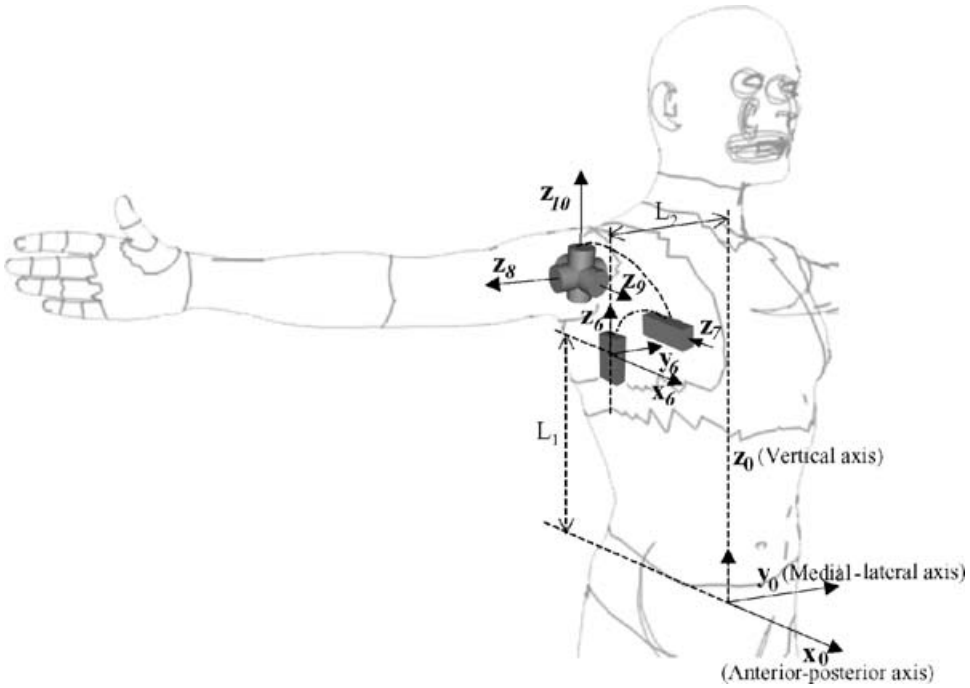

Fig. 2. Modeling of the shoulder complex as three revolute and two sliding DOFs.

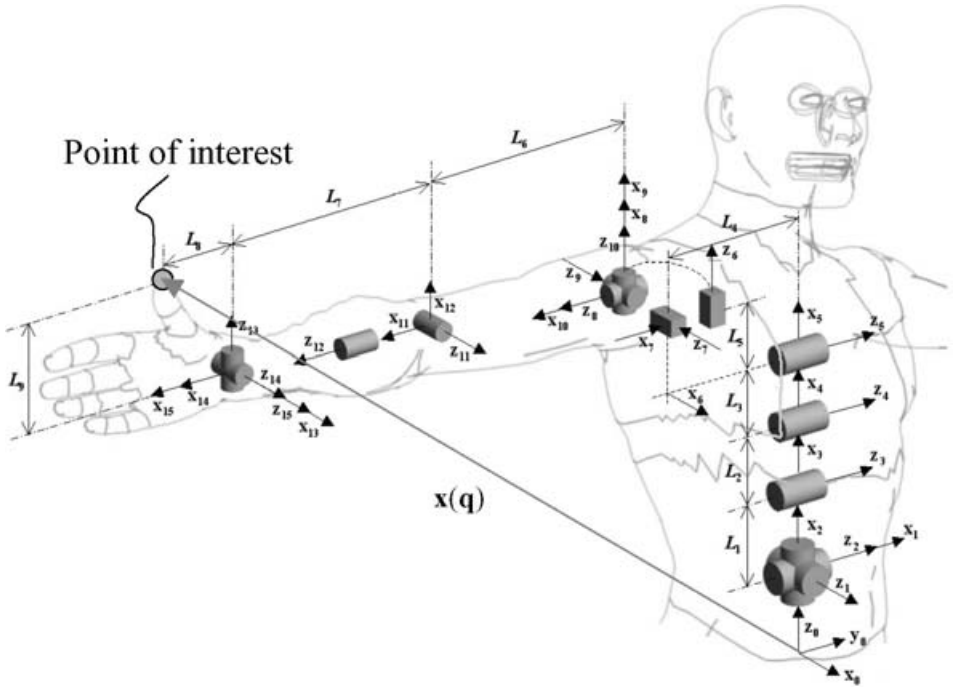

Fig. 3. Modeling of the torso–shoulder–arm.

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

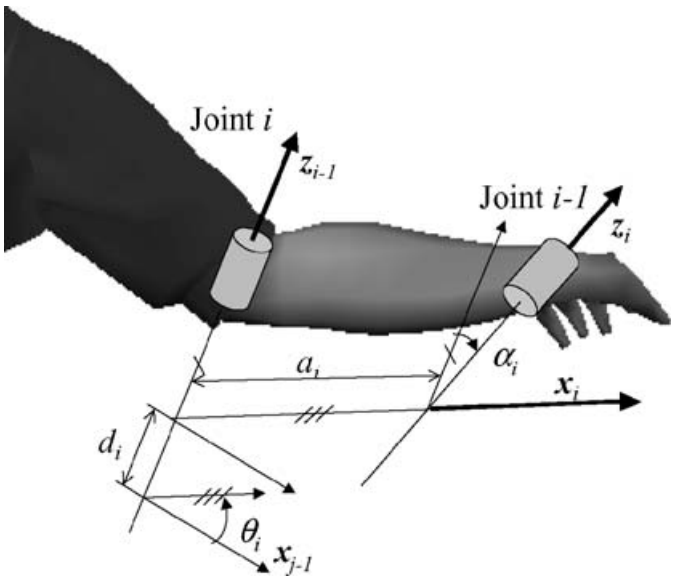

Fig. 4. Establishing coordinate systems and the four D–H parameters.

For any sequence of segmental links and joints, two reference frames are represented by the multiplication of the transformations between them. To relate the coordinate frames $m-1$ and $m+n$, the multiplication of the transformations is carried out as

$$^{m-1}\mathbf{T}_m\,{}^{m}\mathbf{T}_{m+1}\ldots{}^{m+n-1}\mathbf{T}_{m+n} = {}^{m-1}\mathbf{T}_{m+n}. \tag{2}$$

Equally, to relate the base frame usually denoted by zero, and perhaps embedded in the shoulder, to the nth frame, we multiply all transformation matrices such as

$${}^{0}\mathbf{T}_n(\mathbf{q}) = {}^{0}\,\mathbf{T}_1(q_1)\,{}^{1}\mathbf{T}_2(q_2)\ldots{}^{n-1}\,\mathbf{T}_n(q_n), \tag{3}$$

where $\mathbf{q} = [q_1, \ldots, q_n]^T$ are joint angles. Subsequently, a vector described in the hand coordinate system ${}^{n}\mathbf{x}$ (local coordinate system) can be resolved in the base coordinate system (e.g., embedded in the shoulder) by pre-multiplying the overall transformation matrix as

$$\begin{bmatrix}\mathbf{x}(\mathbf{q})\\ 1\end{bmatrix} = {}^{0}\mathbf{T}_n(q_1, \ldots, q_n)\begin{bmatrix}{}^{n}\mathbf{x}\\ 1\end{bmatrix}, \tag{4}$$

where the extended vector is necessary to allow for the (4×4) matrix multiplication. The vector function $\mathbf{x}(\mathbf{q})$ characterizes the xyz coordinates of the hand with respect to the shoulder (Fig. 3). The joint limits for the 15 DOF human upper body are obtained by experiments on human subjects, which are $-\pi/6 \le q_1 \le \pi/6$, $-\pi/12 \le q_2 \le \pi/12$, $-\pi/18 \le q_3 \le \pi/16$, $-\pi/18 \le q_4 \le \pi/6$, $-\pi/18 \le q_5 \le \pi/6$, $-\pi/18 \le q_6 \le \pi/6$, $-3.81 \le q_7 \le 3.81$, $-3.81 \le q_8 \le 3.81$, $-\pi/2 \le q_9 \le \pi/2$, $-2\pi/3 \le q_{10} \le 11\pi/18$, $-\pi/3 \le q_{11} \le 2\pi/3$, $-5\pi/6 \le q_{12} \le 0$, $-\pi \le q_{13} \le 0$, $-\pi/3 \le q_{14} \le \pi/3$, and $-\pi/9 \le q_{15} \le \pi/9$. q_7 and q_8 are translational joints and they are in centimeters; other joints are in radians.

3. PATH IN CARTESIAN SPACE (MINIMUM-JERK 3D MODEL)

First, we will extend the concept of 2D minimum jerk[8] to 3D in Cartesian space. This minimum-jerk 3D model is developed and used as the path of human upper-body reach movement. This model entails unconstrained point-to-point movement and curved point-to-point (via point) movement.

3.1 Unconstrained point-to-point movement

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

$$\mathbf{x}(t) = [x(t)\ \ y(t)\ \ z(t)]^T \tag{5}$$

the first derivative is the velocity and the second derivative is the acceleration. The third derivative, introduced by Flash and Hogan,[8] is the jerk along a path and is best measured by an integration over the motion time along the path, such that

$$C = \frac{1}{2}\int_0^{t_f}\left(\left(\frac{\mathrm{d}^3x}{\mathrm{d}t^3}\right)^2 + \left(\frac{\mathrm{d}^3y}{\mathrm{d}t^3}\right)^2 + \left(\frac{\mathrm{d}^3z}{\mathrm{d}t^3}\right)^2\right)\mathrm{d}t. \tag{6}$$

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 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 $L[t, x, \dot{x}, \ddot{x}, \ldots, \mathrm{d}^n x/\mathrm{d}t^n]$, which is integrable over the same interval, the unconstrained cost function

$$C(x(t)) = \int_0^{t_f} L\left[t, x, \dot{x}, \ddot{x}, \ldots, \frac{\mathrm{d}^n x}{\mathrm{d}t^n}\right]\mathrm{d}t \tag{7}$$

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

$$\frac{\partial L}{\partial x} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial x}\right)\cdots + (-1)^n\frac{\mathrm{d}^n}{\mathrm{d}t^n}\frac{\partial L}{\partial(x^n)} = 0. \tag{8}$$

In our case,

$$L = \tfrac{1}{2}((\dddot{x})^2 + (\dddot{y})^2 + (\dddot{z})^2) \tag{9}$$

and the Euler–Poisson equation

$$\frac{\mathrm{d}^3}{\mathrm{d}t^3}\left(\frac{\partial\dddot{x}^2}{\partial\dddot{x}}\right) = \frac{\mathrm{d}^3}{\mathrm{d}t^3}\left(\frac{\partial\dddot{y}^2}{\partial\dddot{y}}\right) = \frac{\mathrm{d}^3}{\mathrm{d}t^3}\left(\frac{\partial\dddot{z}^2}{\partial\dddot{z}}\right) = 0 \tag{10}$$

we can get

$$\frac{\mathrm{d}^6x}{\mathrm{d}t^6} = 0; \qquad \frac{\mathrm{d}^6y}{\mathrm{d}t^6} = 0; \qquad \frac{\mathrm{d}^6z}{\mathrm{d}t^6} = 0. \tag{11}$$

If we assume the movement to start and end with zero velocity and acceleration, then we have

$$\begin{aligned} x(t) &= x_0 + (x_0 - x_f)(15\tau^4 - 6\tau^5 - 10\tau^3),\\ y(t) &= y_0 + (y_0 - y_f)(15\tau^4 - 6\tau^5 - 10\tau^3),\\ z(t) &= z_0 + (z_0 - z_f)(15\tau^4 - 6\tau^5 - 10\tau^3), \end{aligned} \tag{12}$$

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

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

3.2 Curved point-to-point movement

Consider motion along a curve, where the hand 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 points, by examining the largest diameter of the obstacle, an artificial intelligence engine can determine and introduce a via point to pass through for going around the obstacle. The objective is to generate the smoothest motion to bring the hand from the initial position to the final position at a given time, while the hand must move to the final position through a via point at an unspecified time. The requirement that the hand should move through a specified via point defines equality constraints on the hand-position coordinates $\mathbf{x}(t) = [x(t), y(t), z(t)]^T$; i.e., if the location of the via point with respect to a Cartesian coordinate system is given by $\mathbf{x}_1 = [x_1, y_1, z_1]^T$, the equality constraints are

$$\mathbf{x}(t_1) = \mathbf{x}_1, \tag{13}$$

where the time t_1 at which the hand has to pass through the via point is not specified *a priori*, but rather is derived from 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.[31]

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

$$\dot{\mathbf{s}} = \mathbf{f}[\mathbf{s}(t), \mathbf{u}(t), t], \tag{14}$$

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

$$C(t) = \int_0^{t_f} L[\mathbf{s}(t), \mathbf{u}(t), t]\,\mathrm{d}t, \tag{15}$$

where $L[\mathbf{s}(t), \mathbf{u}(t), t]$ is the performance index. This problem can be solved using the Pontryagin method.[32] One defines an n component co-state (Lagrange multipliers) vector $\boldsymbol{\lambda}(t)$ and a scalar Hamiltonian

$$H[\mathbf{s}(t), \mathbf{u}(t), t)] = L[\mathbf{s}(t), \mathbf{u}(t), t] + \boldsymbol{\lambda}^T(t)\mathbf{f}[\mathbf{s}(t), \mathbf{u}(t), t]. \tag{16}$$

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

$$\dot{\mathbf{s}} = \mathbf{f}[\mathbf{s}(t), \mathbf{u}(t), t] \tag{17a}$$

$$\dot{\boldsymbol{\lambda}}(t) = -\frac{\partial H}{\partial \mathbf{s}} \tag{17b}$$

$$\frac{\partial H}{\partial \mathbf{u}} = 0. \tag{17c}$$

For optimal control problems with interior point equality constraints, there is a set of constraints at some time t_1

$$\boldsymbol{\Psi}(\mathbf{s}(t_1), t_1) = 0, \tag{18}$$

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

$$C = \boldsymbol{\kappa}^T\boldsymbol{\Psi} + \int_0^{t_f} L[\mathbf{s}(t), \mathbf{u}(t), t]\,\mathrm{d}t. \tag{19}$$

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 Lagrange coefficients $\boldsymbol{\lambda}^+(t)$ and Hamiltonian $H^+(t)$ for $t \geq t_1$ and a vector $\boldsymbol{\lambda}^-(t)$ and Hamiltonian $H^-(t)$ for $t \leq t_1$. At time t_1, these variables satisfy the equations

$$\boldsymbol{\lambda}^-(t_1) = \boldsymbol{\lambda}^+(t_1) + \boldsymbol{\kappa}^T\frac{\partial \boldsymbol{\Psi}}{\partial \mathbf{s}(t_1)} \tag{20}$$

and

$$H^-(t_1) = H^+(t_1) - \boldsymbol{\kappa}^T\frac{\partial \boldsymbol{\Psi}}{\partial t_1}. \tag{21}$$

The ρ components of $\boldsymbol{\kappa}$ are determined by the constraint Eq. (18), while time t_1 is fully determined by Eq. (21).

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 $\mathbf{u}(t) = [\delta, \gamma, \eta]^T$, and the components of these vectors are defined by the system equations

$$\begin{aligned}
\dot{x} &= u \\
\dot{y} &= v \\
\dot{z} &= w \\
\dot{u} &= \ddot{x} = a \\
\dot{v} &= \ddot{y} = b \\
\dot{w} &= \ddot{z} = c \\
\dot{a} &= \dddot{x} = jerk_x = \delta \\
\dot{b} &= \dddot{y} = jerk_y = \gamma \\
\dot{c} &= \dddot{z} = jerk_z = \eta
\end{aligned} \tag{22}$$

and the Hamiltonian is

$$\begin{aligned}
H = {} & \lambda_x u + \lambda_y v + \lambda_z w + \lambda_u a + \lambda_v b + \lambda_w c + \lambda_a \delta \\
& + \lambda_b \gamma + \lambda_c \eta + \tfrac{1}{2}(\delta^2 + \gamma^2 + \eta^2).
\end{aligned} \tag{23}$$

The necessary conditions for a minimum to exist are

$$\left.\begin{array}{lll}
-\dfrac{\mathrm{d}\lambda_x}{\mathrm{d}t} = 0 & -\dfrac{\mathrm{d}\lambda_u}{\mathrm{d}t} = \lambda_x & -\dfrac{\mathrm{d}\lambda_a}{\mathrm{d}t} = \lambda_u \\
-\dfrac{\mathrm{d}\lambda_y}{\mathrm{d}t} = 0 & -\dfrac{\mathrm{d}\lambda_v}{\mathrm{d}t} = \lambda_y & -\dfrac{\mathrm{d}\lambda_b}{\mathrm{d}t} = \lambda_v \\
-\dfrac{\mathrm{d}\lambda_z}{\mathrm{d}t} = 0 & -\dfrac{\mathrm{d}\lambda_w}{\mathrm{d}t} = \lambda_z & -\dfrac{\mathrm{d}\lambda_c}{\mathrm{d}t} = \lambda_w.
\end{array}\right\} \tag{24}$$

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

The necessary conditions on the control variables are

$$\begin{aligned}\frac{\partial H}{\partial \delta} &= \delta + \lambda_a = 0\\ \frac{\partial H}{\partial \gamma} &= \gamma + \lambda_b = 0\\ \frac{\partial H}{\partial \eta} &= \eta + \lambda_c = 0.\end{aligned} \tag{25}$$

For our specific problem, the constraints are at the hand position at time t_1

$$\begin{aligned}x(t_1) &= x_1\\ y(t_1) &= y_1\\ z(t_1) &= z_1.\end{aligned} \tag{26}$$

The Hamiltonian H^- for all times $t \le t_1$ is

$$\begin{aligned}H^- = {}& \lambda_x^- u^- + \lambda_y^- v^- + \lambda_z^- w^- + \lambda_u^- a^- + \lambda_v^- b^- + \lambda_w^- c^-\\ & + \lambda_a^- \delta^- + \lambda_b^- \gamma^- + \lambda_c^- \eta^- + \tfrac{1}{2}((\delta^-)^2 + (\gamma^-)^2 + (\eta^-)^2)\end{aligned} \tag{27}$$

and the Hamiltonian H^+ for all times $t \ge t_1$ is

$$\begin{aligned}H^+ = {}& \lambda_x^+ u^+ + \lambda_y^+ v^+ + \lambda_z^+ w^+ + \lambda_u^+ a^+ + \lambda_v^+ b^+ + \lambda_w^+ c^+\\ & + \lambda_a^+ \delta^+ + \lambda_b^+ \gamma^+ + \lambda_c^+ \eta^+ + \tfrac{1}{2}((\delta^+)^2 + (\gamma^+)^2 + (\eta^+)^2).\end{aligned} \tag{28}$$

Since the constraint equations only relate to position, the only discontinuities are in λ_x, λ_y, and λ_z; therefore, according to Eq. (20), we get

$$\begin{aligned}\lambda_x^- &= \lambda_x^+ + \kappa_1\\ \lambda_y^- &= \lambda_y^+ + \kappa_2\\ \lambda_z^- &= \lambda_z^+ + \kappa_3.\end{aligned} \tag{29}$$

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

$$\begin{aligned}\lambda_u^- &= \lambda_u^+\\ \lambda_v^- &= \lambda_v^+\\ \lambda_w^- &= \lambda_w^+\\ \lambda_a^- &= \lambda_a^+\\ \lambda_b^- &= \lambda_b^+\\ \lambda_c^- &= \lambda_c^+.\end{aligned} \tag{30}$$

Since time t_1 is not explicitly specified, the Hamiltonian must be continuous at t_1 according to Eq. (21)

$$H^+(t_1) = H^-(t_1). \tag{31}$$

Now we can derive the necessary conditions for the existence of a minimum separately for $t \le t_1$ and $t \ge t_1$ as shown in Eqs. (24) and (25). In addition, we require continuity of velocities and accelerations at t_1, so that

$$\begin{aligned}u^+(t_1) &= u^-(t_1)\\ v^+(t_1) &= v^-(t_1)\\ w^+(t_1) &= w^-(t_1)\\ a^+(t_1) &= a^-(t_1)\\ b^+(t_1) &= b^-(t_1)\\ c^+(t_1) &= c^-(t_1).\end{aligned} \tag{32}$$

Applying the following boundary conditions

$$\begin{aligned}x(0) &= x_0 \quad & x(t_\mathrm{f}) &= x_\mathrm{f}\\ y(0) &= y_0 \quad & y(t_\mathrm{f}) &= y_\mathrm{f}\\ z(0) &= z_0 \quad & z(t_\mathrm{f}) &= z_\mathrm{f}\\ u(0) &= 0 \quad & u(t_\mathrm{f}) &= 0\\ v(0) &= 0 \quad & v(t_\mathrm{f}) &= 0\\ w(0) &= 0 \quad & w(t_\mathrm{f}) &= 0\\ a(0) &= 0 \quad & a(t_\mathrm{f}) &= 0\\ b(0) &= 0 \quad & b(t_\mathrm{f}) &= 0\\ c(0) &= 0 \quad & c(t_\mathrm{f}) &= 0.\end{aligned} \tag{33}$$

Solving Eqs. (24) and (25), and applying boundary conditions (33) we can obtain an expression for $\mathbf{x}(t)$ for times $t \le t_1$

$$\begin{aligned}\mathbf{x}^-(\tau) = \frac{t_\mathrm{f}^5}{720}\big(&\boldsymbol{\kappa}\big(\tau_1^4(15\tau^4 - 30\tau^3) + \tau_1^3(80\tau^3 - 30\tau^4)\\ &- 60\tau^3\tau_1^2 + 30\tau^4\tau_1 - 6\tau^5\big)\\ &+ \mathbf{c}(15\tau^4 - 10\tau^3 - 6\tau^5)\big) + \mathbf{x}_0\end{aligned} \tag{34}$$

and for times $t \ge t_1$ the expression is

$$\begin{aligned}\mathbf{x}^+(\tau) = {}& \frac{t_\mathrm{f}^5}{720}\big(\boldsymbol{\kappa}\big(\tau_1^4(15\tau^4 - 30\tau^3 + 30\tau - 15)\\ &+ \tau_1^3(-30\tau^4 + 80\tau^3 - 60\tau^2 + 10)\big)\\ &+ \mathbf{c}(-6\tau^5 + 15\tau^4 - 10\tau^3 + 1)\big) + \mathbf{x}_\mathrm{f}\\ = {}& \mathbf{x}^-(\tau) + \boldsymbol{\kappa}\frac{t_\mathrm{f}^5(\tau - \tau_1)^5}{120},\end{aligned} \tag{35}$$

where $\mathbf{x}(t) = [x(t), y(t), z(t)]^T$, $\boldsymbol{\kappa} = [\kappa_1, \kappa_2, \kappa_3]^T$, and $\mathbf{c} = [c_1, c_2, c_3]^T$ are vectors of constants, $\tau = t/t_\mathrm{f}$ and $\tau_1 = t_1/t_\mathrm{f}$. From Eq. (26), we have

$$\mathbf{x}^+(t_1) = \mathbf{x}^-(t_1) = \mathbf{x}_1, \tag{36}$$

where $\mathbf{x}_1 = [x_1, y_1, z_1]^T$. Substituting Eqs. (34) and (35) into (36), we obtain the following

$$\begin{aligned}\boldsymbol{\kappa} = \frac{1}{t_\mathrm{f}^5\tau_1^5(1-\tau_1)^5}\big(&(\mathbf{x}_\mathrm{f} - \mathbf{x}_0)\big(120\tau_1^5 - 300\tau_1^4 + 200\tau_1^3\big)\\ &- 20(\mathbf{x}_1 - \mathbf{x}_0)\big)\end{aligned} \tag{37}$$

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

$$\mathbf{c} = \frac{1}{t_f^5 \tau_1^2 (1-\tau_1)^5}\big((\mathbf{x}_f - \mathbf{x}_0)\big(300\tau_1^5 - 1200\tau_1^4 + 1600\tau_1^3\big) + \tau_1^2(-720\mathbf{x}_f + 120\mathbf{x}_1 + 600\mathbf{x}_0) + (\mathbf{x}_0 - \mathbf{x}_1)(300\tau_1 - 200)\big), \quad (38)$$

where $\mathbf{x}_0 = [x_0, y_0, z_0]^T$ and $\mathbf{x}_f = [x_f, y_f, z_f]^T$.

Next we substitute Eqs. (37) and (38) into (31), which reduces to

$$\kappa_1 u(t_1) + \kappa_2 v(t_1) + \kappa_3 w(t_1) = 0 \quad (39)$$

and we can obtain a polynomial equation in $\tau_1 = t_1/t_f$

$$(\boldsymbol{\kappa}^* \cdot \boldsymbol{\kappa}^*)\left(2\tau_1^3 - 7\tau_1^2 + 8\tau_1 - 3\right) + (\boldsymbol{\kappa}^* \cdot \mathbf{c}^*)\left(-\tau_1^3 + 2\tau_1^2 - \tau_1\right) = 0, \quad (40)$$

where

$$\boldsymbol{\kappa}^* = (\mathbf{x}_f - \mathbf{x}_0)\left(6\tau_1^5 - 15\tau_1^4 + 10\tau_1^3\right) - (\mathbf{x}_1 - \mathbf{x}_0) \quad (41)$$

$$\mathbf{c}^* = (\mathbf{x}_f - \mathbf{x}_0)\big(15\tau_1^5 - 60\tau_1^4 + 80\tau_1^3\big) + \tau_1^2(-36\mathbf{x}_f + 6\mathbf{x}_1 + 30\mathbf{x}_0) + (\mathbf{x}_0 - \mathbf{x}_1)(15\tau_1 - 10). \quad (42)$$

We then find the real roots of this polynomial equation, which has only one acceptable root lying between 0 and 1. Substituting this value for τ_1 in the expressions for $\boldsymbol{\kappa}$ and $\mathbf{c}$ in Eqs. (37) and (38), we can finally get the expressions for $\mathbf{x}(t)$ during the entire motion.

4. MAPPING FROM CARTESIAN SPACE TO JOINT SPACE

As we have already obtained the path in Cartesian space for unconstrained and curved (for obstacle avoidance) trajectories by minimizing the jerk during motion, we can now find joint profiles in joint space that will allow for the desired motion of the hand and simultaneously minimize jerk in Cartesian space.

We will use B-splines to represent joint displacements as a function of time, one for each joint (readers are referred to the Appendix for NURBS basic concepts). The B-spline curve of joint j can be obtained as

$$q_j(t) = \sum_{i=1}^{n_c} N_{i,3}(t) P_i^j, \qquad 0 \le t \le t_f, \qquad j = 1, 2, \ldots, l, \quad (43)$$

where $N_{i,3}(t)$ are the base functions, P_i^j are control points for joint j, and the total number of control points for joint j is n_c.[33]

When we have the start and end points, the optimization-based posture prediction algorithm[34] is first used to predict the natural postures at the start and end points, i.e., one can obtain P_1^j and P_m^j, where $j = 1, \ldots, l$. Therefore, the optimization problem is defined as

Find P_i^j, $i = 2, \ldots, n_c - 1$, $j = 1, \ldots, l$.

Minimize

$$f_{\text{cost}} = w_1 f_{\text{joint-displacement}} + w_2 f_{\text{inconsistency}} + w_3 f_{\text{non-smoothness}} + w_4 f_{\text{non-continuity}}. \quad (44)$$

Subject to

$$\|\mathbf{x}(\mathbf{q}(t)) - \boldsymbol{p}(t)\| = \left\|\mathbf{x}\left(\sum_{i=1}^{n_c} N_{i,3}(t)\mathbf{P}_i\right) - \mathbf{p}(t)\right\| < \varepsilon \quad (45)$$

$$q_j^{\text{L}} \le P_i^j \le q_j^{\text{U}}, \quad (46)$$

where $0 \le t \le t_f$, ε is a small positive number as the tolerance; $\mathbf{P}_i = [\, P_i^1 \ \ldots \ P_i^n \,]^T$; $\mathbf{p}(t)$ is the path obtained from the planning in Cartesian space phase; w_1, w_2, w_3, and w_4 are the weights added to each performance index, L in the superscript denotes lower limit and U denotes upper limit.

(1) The *joint-displacement function* of all joints:

$$f_{\text{joint-displacement}}(\mathbf{q}) = \sum_{j=1}^{n} \xi_j \left(q_j - q_j^N\right)^2 = \sum_{j=1}^{n} \xi_j \left(\sum_{i=1}^{n_c} N_{i,3}(t) P_i^j - q_j^N\right)^2, \quad (47)$$

where q_j^N is the neutral position of a joint measured from the starting home configuration and ξ_j is a weight function assigned to each joint in order to give more importance to joints that are typically more affected than others.

(2) *The inconsistency function*: By comparing the two postures (initial and end points), an overall changing trend of each joint (increasing or decreasing) can be predicted to avoid abrupt change in each joint's velocity. As a result, the consistency between the joint rate change (first derivative) and predicted overall trend is evaluated and will be added to the cost function. The detailed formulation of this consistency is as follows:

$$\left.\begin{matrix}\mathbf{x}_0 \to \mathbf{q}^0 \\ \mathbf{x}_f \to \mathbf{q}^f\end{matrix}\right\} \to \text{trend}_i = \begin{cases} 1 & \text{if } \left(q_j^f - q_j^0\right) \ge 0 \\ -1 & \text{if } \left(q_j^f - q_j^0\right) < 0 \end{cases} \quad (48)$$

and

$$f_{\text{inconsistency}} = \sum_{j=1}^{n} (|\text{sign}(\dot{q}_j(t)) - \text{trend}_j| + 1)|\dot{q}_j(t)|, \quad (49)$$

where

$$\text{sign}(\dot{q}_j(t)) = \begin{cases} 1 & \text{if } \dot{q}_j(t) \ge 0 \\ -1 & \text{if } \dot{q}_j(t) < 0, \end{cases} \quad (50)$$

and $\mathbf{q}^0$ and $\mathbf{q}^f$ are the joint vectors corresponding to the starting and ending position of the path, respectively. The $(+1)$ in Eq. (49) is to make sure the amplitude of the joint rate change still effectively optimizes a smooth joint trajectory when the first term within the parentheses is evaluated as zero. Multiplying the amplitude of this joint change rate enforces the underlying assumption that the smaller the joint angle change rate is, the smoother the joint trajectory will be. It also has a significant effect on the optimization process, by quantifying as well as qualifying the consistency so as to

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

avoid the zero gradient of this objective, that is characteristic of an ill-stated optimization problem statement.

(3) *The non-smoothness function*: The second derivative of the joint trajectory is considered in a non-smoothness (or smoothness) function as

$$f_{\text{non-smoothness}} = \sum_{j=1}^{n} (\ddot{q}_j(t))^2. \tag{51}$$

If joint (angular) accelerations $\ddot{q}_j$ exist, then $\dot{q}_j$ and q_j will exist and be continuous, and it is called that the q_j are smooth.

(4) *The non-continuity function*: The amplitudes of joint angle rates at the start and final target points are considered as the continuity function:

$$f_{\text{non-continuity}} = \sum_{j=1}^{n} |\dot{q}_j^0| + \sum_{j=1}^{n} |\dot{q}_j^{\text{f}}|. \tag{52}$$

With respect to multi-objective optimization (MOO), from a mathematical perspective, there is no "best" set of weights in Eqs. (44) and (47). There is no single "correct" solution. Technically, there are infinitely many solution points to an MOO problem, and these solutions form the Pareto optimal set. In order to determine a single final solution, the user must articulate preferences at some point in some form.

A weighted sum provides one of many methods for articulating preferences *a priori*.[35] Investigating alternative methods is beyond the scope of this paper. Nonetheless, despite certain deficiencies,[36] the weighted sum provides a common and standard approach, sufficient for an initial MOO study in the context of the problem presented in this paper. Conceptually, the weights represent the relative importance of the criteria, and ideally, the magnitude for the weights should be similar to that of the objective-function values.[37] However, the weighted sum represents only a linear approximation (in terms of the criteria or objectives) of the user's preferences. The slope of this approximating function is modified by changing the weights. Consequently, it may be necessary to modify the weights once a corresponding solution is obtained. The solution does not always reflect the intended preferences with perfect accuracy. This can result in a trial-and-error approach to setting the weights and articulating preferences.

Once the control points of joint curves are selected by the iterative optimization algorithm, the cost function of Eq. (44) can be integrated (we integrate the first three terms and add the fourth term to it) 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. For simplicity, the cost function and distance constraints in our algorithm are evaluated by selecting representative points on the path where higher density is distributed close to the ends. Since each joint's profile has n_c control points, the total number of the design variables will initially be nn_c. In our calculation, the joint values at the start and end points have been obtained directly using the posture-prediction algorithm, where we only need to calculate the remaining $n_c - 2$ control points for each joint, i.e., the design variables for the optimization are reduced to $n(n_c - 2)$. A nonlinear optimization code DOT-Design Optimization Tools Program[38] is used to solve the optimization problem. From Eqs. (48) to (50), part 2 of Eq. (44) is not continuous. However, we use finite difference method for the gradient and the optimization problem always converges.

5. ILLUSTRATIVE EXAMPLES

Based on simulation experiments, a set of weights (50, 100, 1, 1000) are selected for w_1, w_2, w_3, and w_4, and the modified feasible direction method is used for the optimization. The overall calculation time is around 17–18 s on a 1.8 GHz Pentium 4 CPU with 512 MB RAM. An interface is implemented in 3D Studio Max, which can interact with the user, call the path-prediction algorithm to do the calculations, show results, and animate human motions in real time.

5.1 Point-to-point example

Figures 5–10 are snapshots of a predicted motion, where the digital human starts from point $[34.4\ -56.6\ 33.6]^T$

Fig. 5. Predicted motion at time 0 s.

Fig. 6. Predicted motion at time 0.30 s.

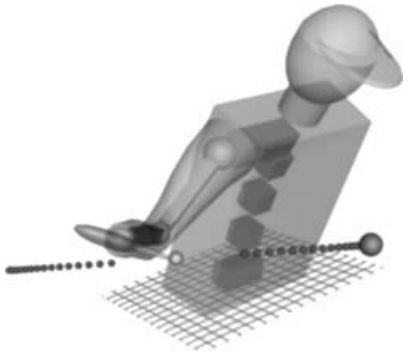

Fig. 7. Predicted motion at time 0.5 s.

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

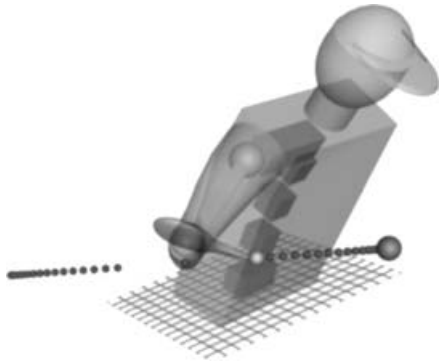

Fig. 8. Predicted motion at time 0.60 s.

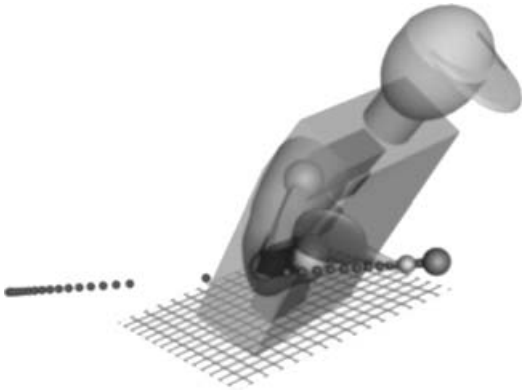

Fig. 9. Predicted motion at time 0.8 s.

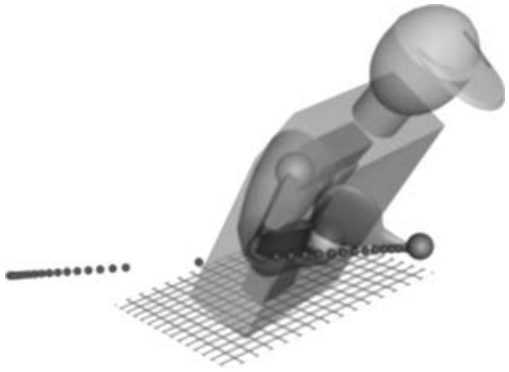

Fig. 10. Predicted motion at time 1.0 s.

and goes to the target point $[74.6\ -22.3\ 33.7]^T$, where the global coordinate is shown in Fig. 3 with $x_0y_0z_0$. The small spheres on the path are the constraints enforced on the hand position when predicting the joint B-splines. From the time stamps of the snapshots shown, it is easy to observe that the hand moves slower around the start and the end than in the middle. This is the so-called bell-shape velocity profile that is characteristic of a smooth and natural human arm movement,[8] and predictability of this profile is actually the strength of the minimum-jerk model. The predicted joint profiles for the 15 joints are shown in Figs. 11–13, where q_7 and q_8 are translational joints and are given in centimeters, and other joints are revolute joints and are given in radians. The (angular) velocities are shown in Figs. 14–16, where velocities $\dot{q}_7$ and $\dot{q}_8$ are in cm/s, and angular velocities

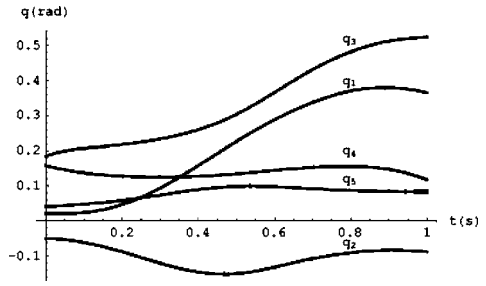

Fig. 11. Joint profiles for $q_1, \ldots, q_5$.

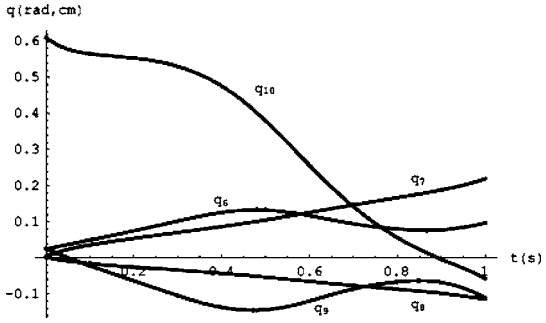

Fig. 12. Joint profiles for $q_6, \ldots, q_{10}$.

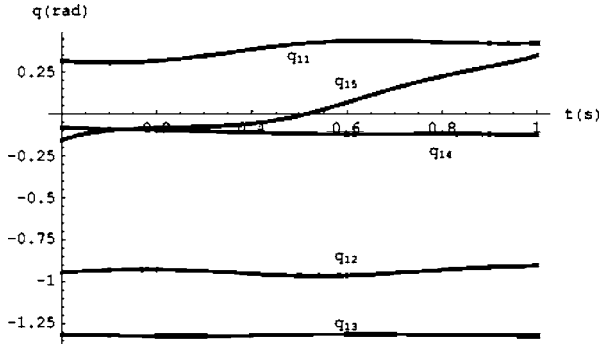

Fig. 13. Joint profiles for $q_{11}, \ldots, q_{15}$.

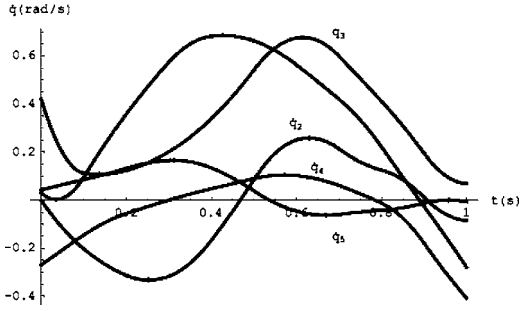

Fig. 14. Joint angular velocities for $\dot{q}_1 \sim \dot{q}_5$.

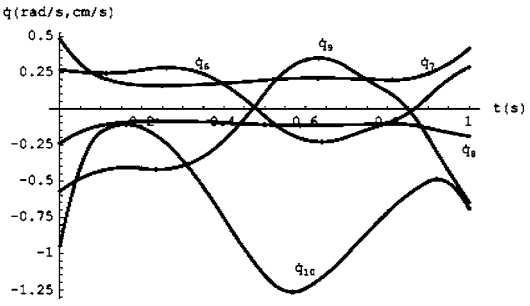

Fig. 15. Joint (angular) velocities for $\dot{q}_6 \sim \dot{q}_{10}$.

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

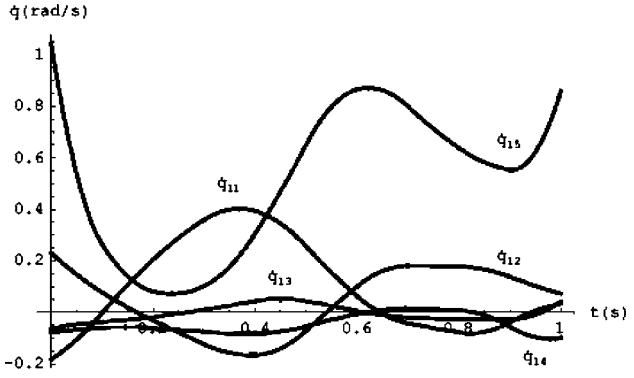

Fig. 16. Joint angular velocities for $\dot{q}_{11} \sim \dot{q}_{15}$.

for other angles are in rad/s, and the (angular) accelerations are shown in Figs. 17–19, where accelerations $\ddot{q}_7$ and $\ddot{q}_8$ are in cm/s^2, and angular accelerations for other angles are in rad/s^2. We see that each joint moves smoothly towards the final position.

From Figs. 5–10, the torso moves a lot more than expected and the arm moves too little than expected. This result shows that the selected weights for Eq. (47) are not accurate. The joints located on the torso should have larger weights relative to the joints on the arm. This conclusion comes from the fact that when minimizing the cost function (Eq. (47)), joints with larger weights will not change too much. Next example for via point will show that the selected weights are much more realistic. Figures 11–13 show that all joint profiles are smooth. Figures 14–19 show that although the hand velocity and acceleration are zero in the start and final target

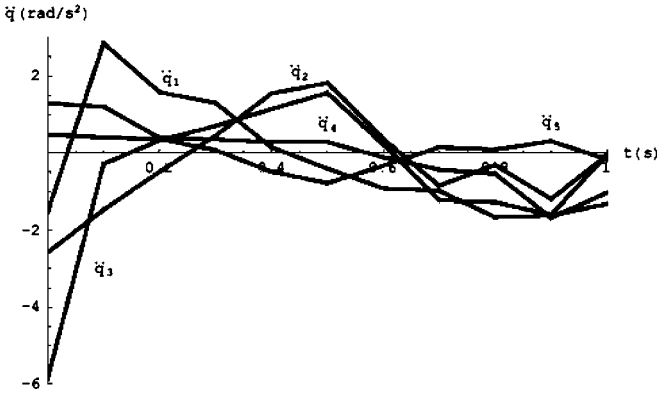

Fig. 17. Joint angular accelerations for $\ddot{q}_1 \sim \ddot{q}_5$.

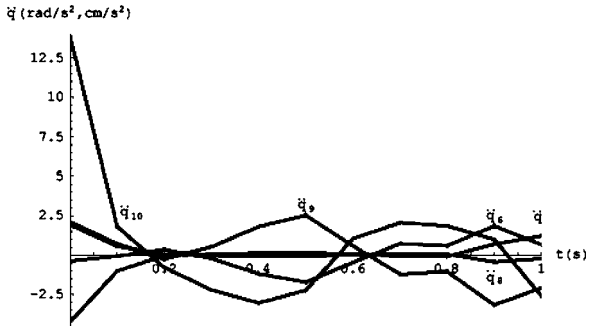

Fig. 18. Joint (angular) accelerations for $\ddot{q}_6 \sim \ddot{q}_{10}$.

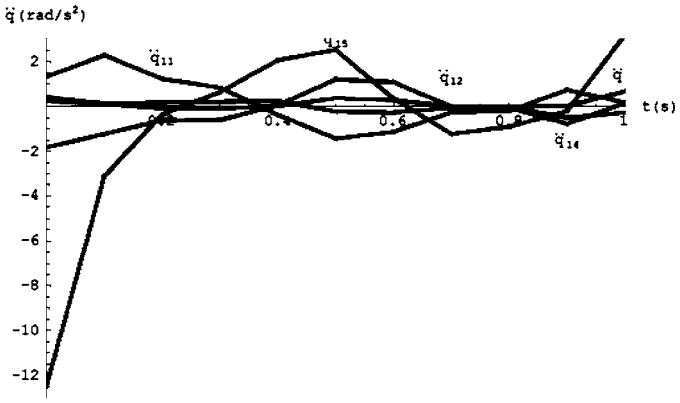

Fig. 19. Joint angular accelerations for $\ddot{q}_{11} \sim \ddot{q}_{15}$.

positions, joint (angular) velocities and accelerations are not. The reason is that when $\dot{\mathbf{x}} = J\dot{\mathbf{q}}$, where J is the Jacobian matrix of the position vector of the end-effector, there exist $\dot{\mathbf{x}}|_{t=0} = \mathbf{0}$ and $\dot{\mathbf{x}}|_{t=t_f} = \mathbf{0}$; however, J is not a diagonal matrix, therefore $\dot{\mathbf{q}} \neq \mathbf{0}$ in the initial and final target points. The similar situation is for the joint (angular) accelerations.

5.2 Curved and obstacle-avoidance example

For curved and obstacle-avoidance movements, it is assumed that the hand is required, in the motion between the end points, 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). So given the start and end points, and a third via point, a curved path in Cartesian space can be first generated,[8] while the time taken in passing through the via point is first solved. Figures 20–25 give the snapshots of such movements while the digital human begins moving from the initial point $[23.7\ -68.5\ 43.0]^T$ to the end point $[66.5\ 4.3\ 34]^T$, passing through the via point $[34.0\ -45.5\ 38.7]^T$. The big green sphere is the via point and the curve is the Cartesian path predicted using the minimum-jerk model. The small spheres are where distance constraints are enforced during the optimization for joint splines. The predicted joint profiles for the 15 joints are shown in Figs. 26–28, (angular) velocities in Figs. 29–31, and accelerations in Figs. 32–34, from which

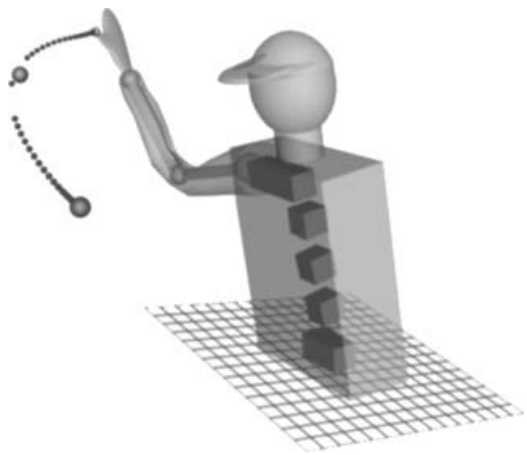

Fig. 20. Predicted motion at time 0 s.

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

Fig. 21. Predicted motion at time 0.30 s.

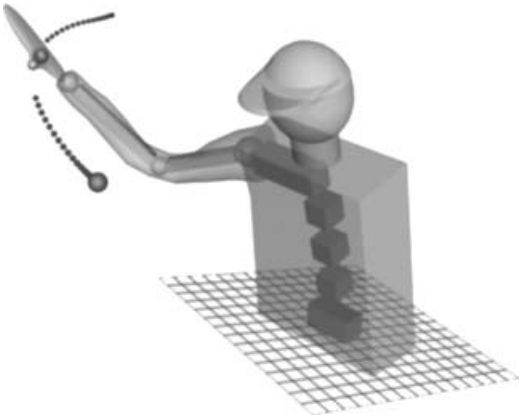

Fig. 22. Predicted motion at time 0.5 s.

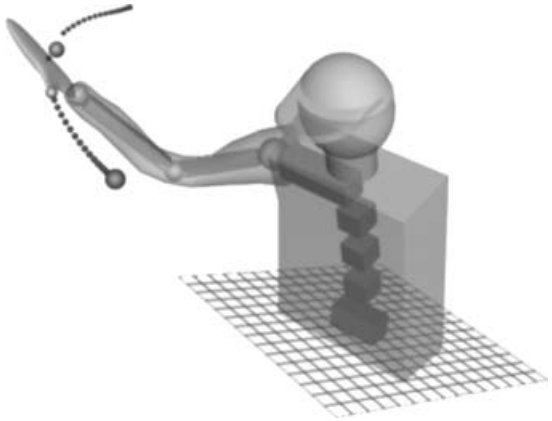

Fig. 23. Predicted motion at time 0.60 s.

we can see that each joint moves smoothly towards the final position where the units for the joint profiles and their first and second derivatives are the same as in the previous example.

As shown in the figures given earlier, the proposed method and algorithm can predict smooth and graceful movements of the upper body even for point-to-point and nonlinear (curved) paths. Figures 20–25 show that the predicted motion for the torso part is much more realistic than the previous example. It means that the weights in Eq. (47) for this example are good weights. Figures 26–28 show that the joint profiles

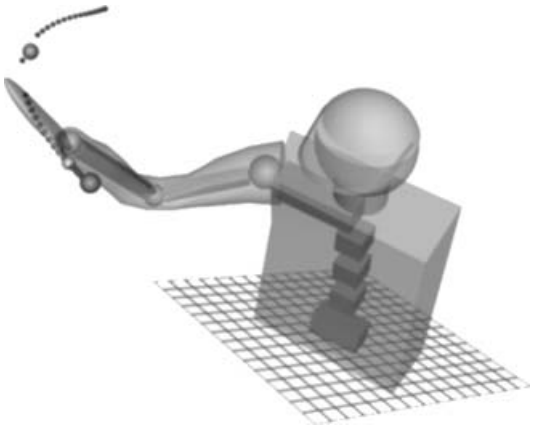

Fig. 24. Predicted motion at time 0.8 s.

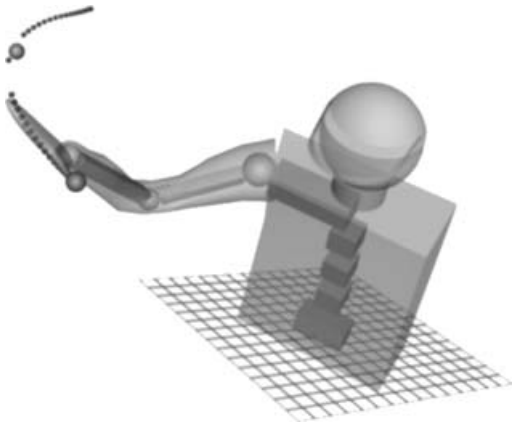

Fig. 25. Predicted motion at time 1.0 s.

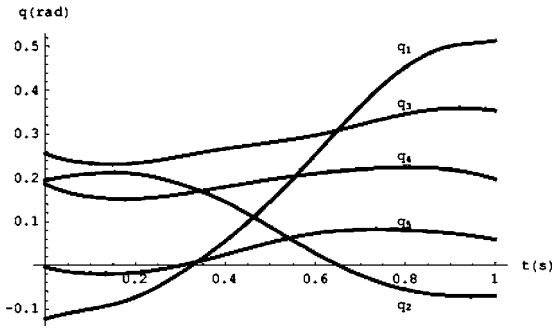

Fig. 26. Joint profiles for $q_1, \ldots, q_5$.

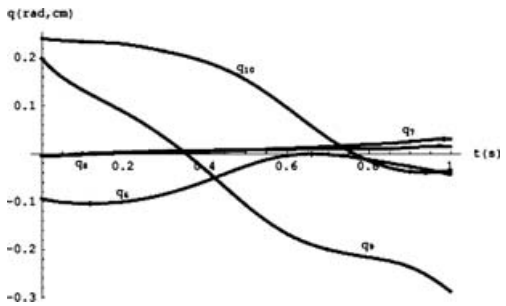

Fig. 27. Joint profiles for $q_6, \ldots, q_{10}$.

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

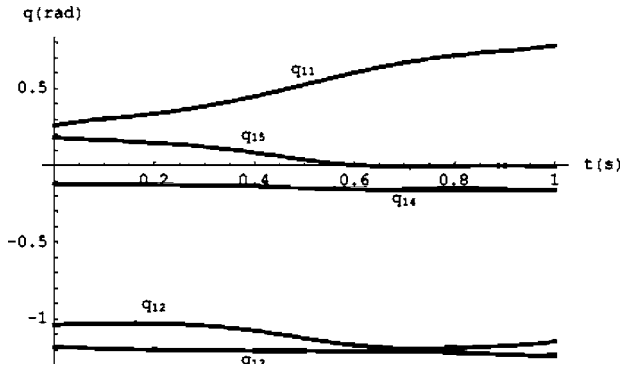

Fig. 28. Joint profiles for $q_{11}, \ldots, q_{15}$.

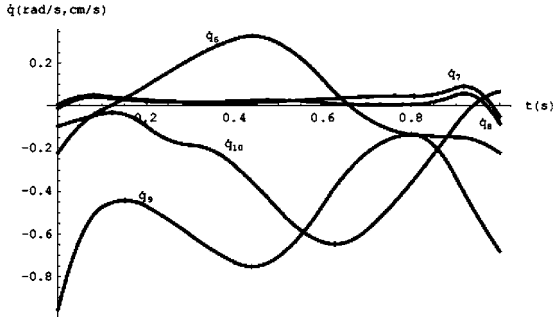

Fig. 29. Joint angular velocities for $\dot{q}_1 \sim \dot{q}_5$.

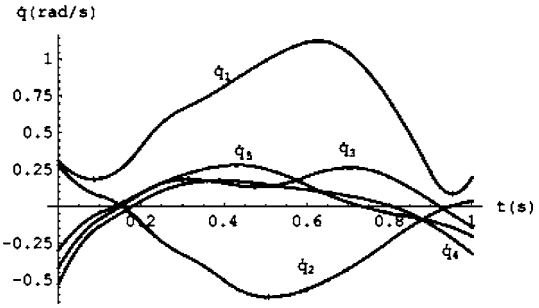

Fig. 30. Joint (angular) velocities for $\dot{q}_6 \sim \dot{q}_{10}$.

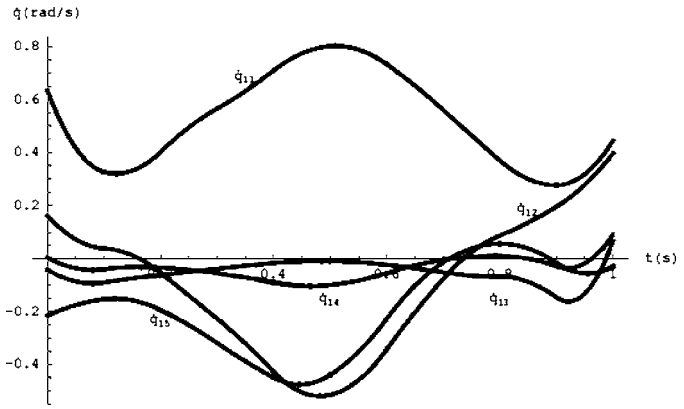

Fig. 31. Joint angular velocities for $\dot{q}_{11} \sim \dot{q}_{15}$.

are smooth although the hand passes through the via point. Figures 29–34 have the same characteristic as Figs. 14–19 for the start and final target positions.

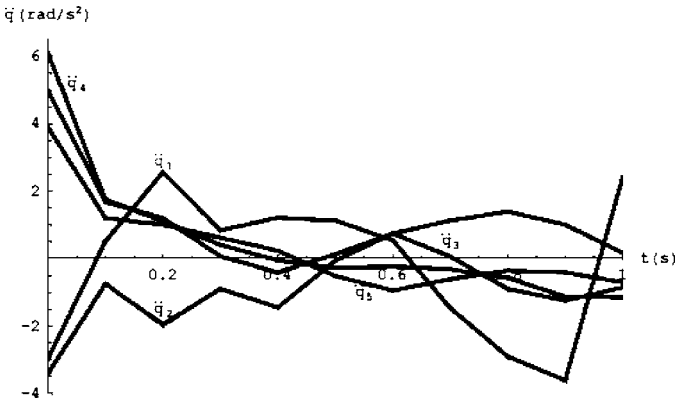

Fig. 32. Joint angular accelerations for $\ddot{q}_1 \sim \ddot{q}_5$.

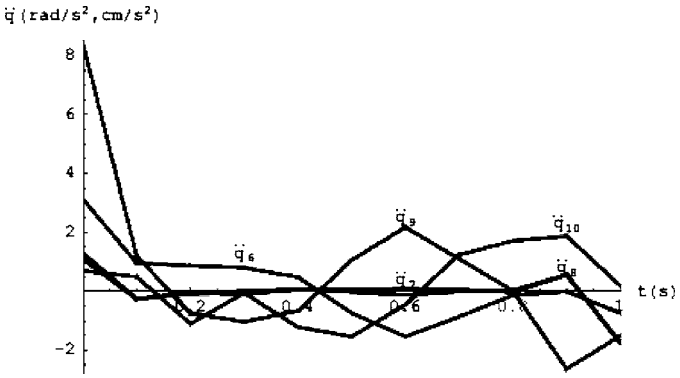

Fig. 33. Joint (angular) accelerations for $\ddot{q}_6 \sim \ddot{q}_{10}$.

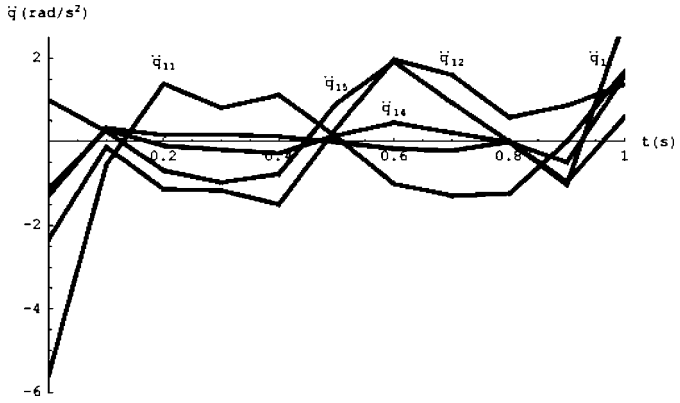

Fig. 34. Joint angular accelerations for $\ddot{q}_{11} \sim \ddot{q}_{15}$.

6. CONCLUSIONS

The proposed method for predicting joint profiles is broadly applicable to any type of path, i.e., to linear (straight) or nonlinear (curved) path trajectories. Nonlinear paths are applicable to obstacle-avoidance problems, where trajectories are deviated from the typical linear point-to-point motion with an extended minimum-jerk model. It is shown that a mathematical formulation applicable to any number of DOFs has been developed and demonstrated. Each of the joint profile has been defined by a smooth B-spline, where control points are calculated using a novel direct optimization-based algorithm. A Cartesian path (including the time to traverse through the via point) is first predicted based on the extended minimum-jerk model (within an iterative optimization algorithm), and is followed by the calculation of

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

joint profiles characterized by B-splines, where the objective is to minimize a joint displacement function, non-consistency function, non-smoothness function, and non-continuity function. It is also shown that the experimental code associated with this formulation was implemented in a graphical real-time simulation interface. It is shown that the weights are an important factor to make sure the results are realistic.

Acknowledgments

This research was funded by the US Army TACOM project: Digital Humans and Virtual Reality for Future Combat Systems (FCS) with contract No. DAAE 07-03-D-L003.

References

1. K. Shin and N. McKay, "A dynamic programming approach to trajectory planning of robot manipulators," *IEEE Trans. Autom. Control* **AC-31**(6), 491–500 (1986).
2. J. E. Bobrow, "Optimal robot path planning using the minimum-time criterion," *IEEE J. Robot. Autom.* **4**(4), 443–450 (1988).
3. Z. Shiller and S. Dubowsky, "Robot path planning with obstacles, actuator, gripper, and payload constraints," *Int. J. Robot. Res.* **8**(6), 3–18 (1989).
4. Z. L. Zhou and C. C. Nguyen, "Globally optimal trajectory planning for redundant manipulators using state space augmentation method," *J. Intell. Robot. Syst.* **19**(1), 105–117 (1997).
5. G. Antonelli and S. Chiaverini, "Task-priority redundancy resolution for underwater vehicle-manipulator systems," *Proceedings of IEEE International Conference on Robotics and Automation*, Leuven, Belgium (1998), pp. 756–761.
6. A. R. Hirakawa and A. Kawamura, "Trajectory planning of redundant manipulators for minimum energy consumption without matrix inversion," *Proceedings of the IEEE International Conference on Robotics and Automation*, Albuquerque, New Mexico (1997), pp. 2415–2420.
7. Y. Yamamoto and S. Fukuda, "Trajectory planning of multiple mobile manipulators with collision avoidance capability," *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, DC (2002), pp. 3565–3570.
8. T. Flash and N. Hogan, "The coordination of arm movements: An experimentally confirmed mathematical model," *J. Neurosci.* **5**(7), 1688–1703 (1985).
9. Y. Uno, M. Kawato and R. Suzuki, "Formation and control of optimal trajectory in human multijoint arm movement," *Biol. Cybern.* **61**, 89–101 (1989).
10. M. Kawato, M. Isobe, Y. Maeda and R. Suzuki, "Coordinates transformation and learning control for visually-guided voluntary movement with iteration: A Newtonlike method in a function space," *Biol. Cybern.* **59**, 161–177 (1988).
11. J. Barraquand and J. C. Latombe, "Robot motion planning: A distributed representation approach," *Int. J. Robot. Res.* **10**(6), 628–649 (1991).
12. K. Schneider and R. F. Zernicke, "Jerk-cost modulations during the practice of rapid arm movements," *Biol. Cybern.* **60**, 221–230 (1989).
13. D. S. Hoffman and P. L. Strick, "Step-tracking movements of the wrist in humans. II. EMG analysis," *J. Neurosci.* **10**(1), 142–152 (1990).
14. H. Nagasaki, "Asymmetrical trajectory formation in cyclic forearm movements in man," *Exp. Brain Res.* **87**(3), 653–661 (1991).
15. A. W. Wiegner and M. M. Wierzbicka, "Kinematic models and human elbow flexion movements: Quantitative analysis," *Exp. Brain Res.* **88**(3), 665–673 (1992).
16. A. Hreljac and P. E. Martin, "The relationship between smoothness and economy during walking," *Biol. Cybern.* **69**(3), 213–218 (1993).
17. A. Hreljac, "The relationship between smoothness and performance during the practice of a lower limb obstacle avoidance task," *Biol. Cybern.* **68**(4), 375–379 (1993).
18. T. Furuna and H. Nagasaki, "Trajectory formation of vertical arm movements through a via-point: A limit of validity of the minimum-jerk model," *Percept. Motor Skills* **76**(3 Pt 1), 875–884 (1993).
19. W. M. Yun and Y. G. Xi, "Optimum motion planning in joint space for robots using genetic algorithms," *Robot. Auton. Syst.* **18**, 373–393 (1996).
20. D. Constantinescu and E. A. Croft, "Smooth and time-optimal trajectory planning for industrial manipulators along specified paths," *J. Robot. Syst.* **17**(5), 233–249 (2000).
21. S. F. P. Saramago and J. V. Steffen, "Optimization of the trajectory planning of robot manipulators taking into account the dynamics of the system," *Mech. Mach. Theory* **33**(7), 883–894 (1998).
22. S. F. P. Saramago and J. V. Steffen, "Optimal trajectory planning of robot manipulators in the presence of moving obstacles," *Mech. Mach. Theory* **35**(8), 079–1094 (2000).
23. S. F. P. Saramago and M. Ceccarelli, "An optimum robot path planning with payload constraints," *Robotica* **20**, 395–404 (2002).
24. S. Pugazhenthi, T. Nagarajan and M. Singaperumal, "Optimal trajectory planning for a hexapod machine tool during contour machining," *Proc. Inst. Mech. Eng. C J. Mech. Eng. Sci.* **216**(12), 1247–1256 (2002).
25. H. Li and D. Ceglarek, "Optimal trajectory planning for material handling of compliant sheet metal parts," *J. Mech. Des., Trans. ASME* **124**(2), 213–222 (2002).
26. R. M. Alexander, "A minimum energy cost hypothesis for human arm trajectories," *Biol. Cybern.* **76**, 97–105 (1997).
27. K. Ohta, M. M. Svinin, Z. W. Luo and S. Hosoe, "On the trajectory formation of the human arm constrained by the external environment," *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, Taipei, Taiwan (2003), pp. 2884–2891.
28. K. Abdel-Malek, W. Yu, E. Tanbour and M. Jaber, "Posture prediction versus inverse kinematics," *Proceedings of the ASME Design Automation Conference*, Pittsburgh, PA (2001).
29. J. Yang, K. Abdel-Malek and K. Nebel, "Reach envelope of a 9 degree of freedom model of the upper extremity," *Int. J. Robot. Autom.* **20**(4), 240–259 (2005).
30. J. Denavit and R. S. Hartenberg, "A kinematic notation for lower-pair mechanisms based on matrices," *J. Appl. Mech.***77**, 215–221 (1955).
31. A. E. Bryson and Y. C. Ho, *Applied Optimal Control* (Hemisphere Publ. Co., New York, (1975).
32. L. S. Pontryagin, V. Boltyanskii, R. Gamkrelidze and E. Mishchenko, *The Mathematical Theory of Optimal Processes* (Interscience Publishers Inc., New York, (1962).
33. L. Piegl and W. Tiller, *The NURBS Book* (Springer-Verlag, Berlin, (1997).
34. Z. Mi, J. Yang and K. Abdel-Malek, "Real-time inverse kinematics for humans," *Proceedings of the 2002 ASME Design Engineering Technical Conference*, Montreal, Canada (2002).
35. R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Struct. Multidiscip. Optim.* **26**(6), 369–395 (2004).
36. I. Das and J. E. Dennis, "A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems," *Struct. Optim.* **14**, 63–69 (1997).
37. R. T. Marler and J. S. Arora, "Transformation methods for multi-objective optimization," *Eng. Optim.* **37**(6), 551–569 (2005).

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press

38. G. Vanderplaats, *DOT-Design Optimization Tools Program-Users Manual* (Vanderplaats Research and Development, Inc., Colorado, 1995).
39. R. A. Brooks and T. Lozano-Perez, "A subdivision algorithm in configuration space for findpath with rotation," *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, W. Germany (1983), pp. 799–806.
40. L. E. Kavraki, P. Svestka, J. C. Latombe and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Autom.* **12**(4), 566–580 (1996).
41. W. L. Nelson, "Physical principles for economies of skilled movements," *Biol. Cybern.* **46**, 135–147 (1983).
42. R. Plamondon, "A kinematic theory of rapid human movements: Part I. Movement representation and generation," *Biol. Cybern.* **72**, 295–307 (1995).
43. R. Plamondon, "A kinematic theory of rapid human movements: Part II. Movement time and control," *Biol. Cybern.* **72**, 309–320 (1995).
44. R. Plamondon, "A kinematic theory of rapid human movements: Part III. Kinetic outcomes," *Biol. Cybern.* **78**, 133–145 (1998).
45. D. M. Wolpert, Z. Ghahramani and M. I. Jordan, "Are arm trajectories planned in kinematic or dynamic coordinates? An adaptation study," *Exp. Brain Res.* **103**, 460–470 (1995).

APPENDIX

There are a number of ways to define the B-spline basis functions; the most useful for computer implementation is the recurrence formula. We shall use the recurrence formula to represent a B-spline, such that its control points will be calculated as a result of the iterative numerical algorithm and will be based on optimizing the minimum-jerk formula. Let $\mathbf{U} = \{u_1, \ldots, u_m\}$ be a non-decreasing sequence of real numbers, i.e., $u_i \leq u_{i+1}$, $i = 1, \ldots, m-1$. The u_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 $N_{i,p}(u)$, is defined as

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u). \tag{A1}$$

A pth-degree B-spline curve is defined by

$$C(u) = \sum_{i=1}^{n} N_{i,p}(u) P_i \quad a \leq u \leq b, \tag{A2}$$

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

$$U = \{\underbrace{a, \ldots, a}_{p+1}, u_{p+1}, \ldots, u_{m-p-1}, \underbrace{b, \ldots, b}_{p+1}\} (m \text{ knots}).$$

The polygon formed by $\{P_i\}$ 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 u value: (1) Find the knot span in which u lies; (2) compute the basis functions; (3) multiply the values of the basis functions with the corresponding control points. A degree three B-spline with seven control points is shown in Fig. A1.

The following is the formulation to calculate the kth derivative of the basis function $N_{i,p}(u)$ in terms of the functions $N_{i,p-k}, \ldots, N_{i+j,p-k}$ which is

$$N_{i,p}^{(k)} = \frac{p!}{(p-k)!} \sum_{j=0}^{k} a_{k,j} N_{i+j,p-k} \tag{A3}$$

with

$$\begin{aligned} a_{0,0} &= 1 \\ a_{k,0} &= \frac{a_{k-1,0}}{u_{i+p-k+1} - u_i} \\ a_{k,j} &= \frac{a_{k-1,j} - a_{k-1,j-1}}{u_{i+p+j-k+1} - u_{i+j}} \qquad j = 1, \ldots, k-1 \\ a_{k,k} &= \frac{-a_{k-1,k-1}}{u_{i+p+1} - u_{i+k}}. \end{aligned} \tag{A4}$$

Then the kth derivative of a pth-degree B-spline curve is given by

$$C^{(k)}(u) = \sum_{i=1}^{n} N_{i,p}^{(k)}(u) P_i \qquad a \leq u \leq b, \tag{A5}$$

where k should not exceed p (all higher derivatives are zero).

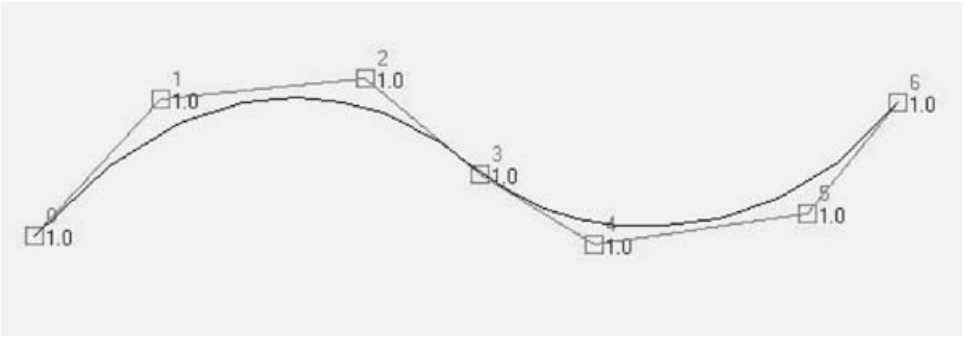

Fig. A1. A B-spline.

https://doi.org/10.1017/S0263574706002852 Published online by Cambridge University Press