High speed CNC system design. Part I: Jerk limited trajectory generation and quintic spline interpolation

Article *in* International Journal of Machine Tools and Manufacture · July 2001

DOI: 10.1016/S0890-6955(01)00002-5

CITATIONS
493

READS
10,775

2 authors:

Kaan Erkorkmaz
University of Waterloo
69 PUBLICATIONS **2,512** CITATIONS

SEE PROFILE

Yusuf Altintas
University of British Columbia - Vancouver
286 PUBLICATIONS **25,456** CITATIONS

SEE PROFILE

Pergamon International Journal of Machine Tools & Manufacture 41 (2001) 1323–1345

INTERNATIONAL JOURNAL OF MACHINE TOOLS & MANUFACTURE DESIGN, RESEARCH AND APPLICATION

High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation

Kaan Erkorkmaz, Yusuf Altintas *

The University of British Columbia, Department of Mechanical Engineering, 2324 Main Mall, Vancouver, BC, Canada V6T 1Z4

Received 30 October 2000; accepted 18 December 2000

Abstract

Reference trajectory generation plays a key role in the computer control of machine tools. Generated trajectories must not only describe the desired tool path accurately, but must also have smooth kinematic profiles in order to maintain high tracking accuracy, and avoid exciting the natural modes of the mechanical structure or servo control system. Spline trajectory generation techniques have become widely adopted in machining aerospace parts, dies, and molds for this reason; they provide a more continuous feed motion compared to multiple linear or circular segments and result in shorter machining time, as well as better surface geometry. This paper presents a quintic spline trajectory generation algorithm that produces continuous position, velocity, and acceleration profiles. The spline interpolation is realized with a novel approach that eliminates feedrate fluctuations due to parametrization errors. Smooth accelerations and decelerations are obtained by imposing limits on the first and second time derivatives of feedrate, resulting in trapezoidal acceleration profiles along the toolpath. Finally, the reference trajectory generated with varying interpolation period is re-sampled at the servo loop closure period using fifth order polynomials, which enable the original kinematic profiles to be preserved. The proposed trajectory generation algorithm has been tested in machining a wing surface on a three axis milling machine, controlled with an in house developed open architecture CNC.

1. Introduction

Modern CNC machine tools need to operate at feedrates up to 40 m/min with accelerations up to 2 g, in order to deliver the rapid feed motion required for high speed machining. At such high speeds, small discontinuities in the reference toolpath can result in undesirable high frequency

* Corresponding author. Tel.: +1-604-822-2182; fax: +1-604-822-2403.
E-mail addresses: kaane@mech.ubc.ca (K. Erkorkmaz); altintas@mech.ubc.ca (Y. Altintas).

0890-6955/01/$ - see front matter
PII: S0890-6955(01)00002-5

harmonics in the reference trajectory, which end up exciting the natural modes of the mechanical structure and the servo control system. Such high frequency components may also saturate the actuators, as well as degrading the axis tracking, and therefore contouring accuracy. On the other hand, employing only linear and circular interpolation techniques to machine more complex shapes has serious limitations in terms of achieving the desired part geometry and productivity. To address these problems, a significant amount of effort has been put in the recent years, in developing new trajectory generation algorithms that provide smooth feed motion to high speed machining systems.

Butler et al. [1] have recommended the modification of the feedrate profile for a given path of second order continuity, in order to avoid actuator saturations for a general feedforward–feedback control scheme. They have also approximated sharp corners with arcs of smaller radius than the position feedback resolution, and let the feedrate modification algorithm with preview action take care of accelerations and decelerations at the corners. Weck and Ye [2] have used a low-pass filter which they have referred to as the *Inverse Compensation Filter* (IKF) with zero phase delay characteristics, to filter out the high frequency components of reference trajectories, and make them easier to track. This filter was used before a zero phase error tracking controller (ZPETC) [3], which is based on inverting the dynamics of the axis control loop in a stable manner. They have reported accurate corner tracking with this strategy. However, the tuning of the IKF was closely dependent on the drive dynamics, and for operations of different feedrate, readjustment to the filter was necessary. A similar approach of using an additional low pass filter with zero phase characteristics before the ZPETC was also adopted by Tung and Tomizuka in [4].

Pritschow [5] has recommended jerk limited trajectories with trapezoidal or sine square acceleration profiles, favoring the latter for better continuity. On the other hand, Wang and Yang [6] have implemented trajectory generation via cubic and quintic splines using chord length and nearly arc length parameterization. The nearly arc length parametrization resulted in smaller feedrate fluctuations compared to chord length parametrization. This was followed by the work of Wang and Wright [7], who have included an extra jerk continuity condition into the solution of the quintic spline, and also recommended using more points in fitting the spline at high curvatures, in order to reduce the feedrate fluctuations due to the nearly arc length parametrization errors. Weck et al. [8] have implemented cubic spline interpolation where feedrates for the spline segments are adjusted based on the physical limitations of the drives, and smooth transitions are obtained using fourth order acceleration profiles.

Makino and Ohde [9] have used a universal cam curve to generate jerk continuous profiles. Tomita et al. [10] have used trigonometric functions for the same purpose. Simon and Isik [11] have proposed using trigonometric splines in trajectory generation for robotic manipulators, because of their high order continuity, ease of computing parameters from boundary conditions, and less oscillatory behavior between distantly spaced knots, compared to high order algebraic splines. They have also performed optimization of such trajectories for minimum jerk.

This work presents a scheme for generating continuous quintic spline toolpaths, using a recursive interpolation technique that maintains a constant position increment (Δs) at each step [Fig. 1(a)]. This way, feedrate fluctuations arising from the difference between the spline parameter (typically the chord length) and the arc length are avoided. Smooth accelerations are then obtained by varying the interpolation period (T^{i}) between the fixed spaced position references [Fig. 1(b)]. The generated reference trajectory is then reconstructed at the servo loop closure period (T_{s}),

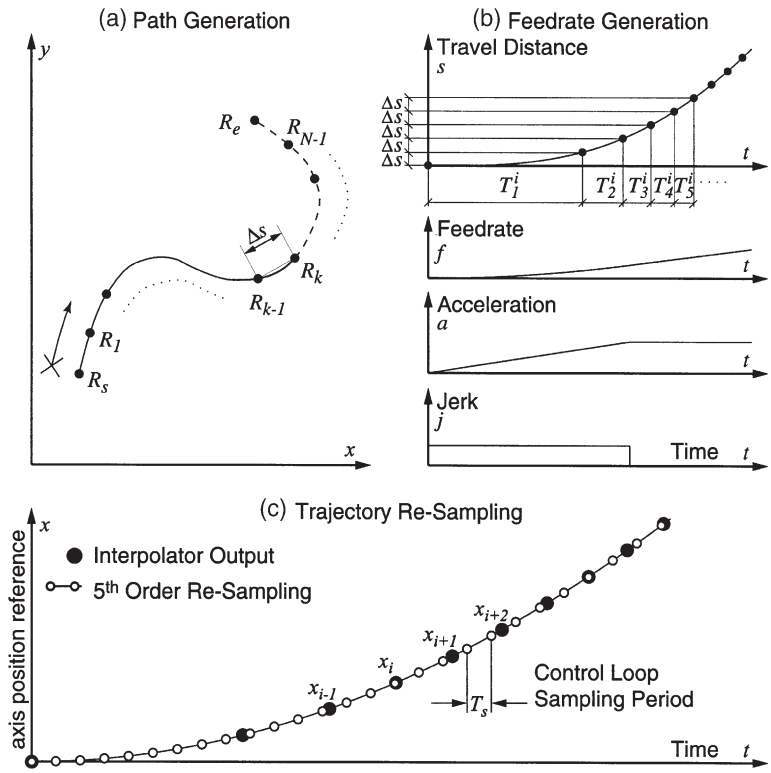

Fig. 1. Trajectory generation steps: (a) path generation in spatial coordinates, (b) feedrate profiling for an existing path, (c) re-sampling of the generated trajectory at the control loop frequency.

while maintaining the original acceleration and feedrate characteristics, using a fifth order re-sampling technique [Fig. 1(c)].

The rest of this paper is organized as follows: Toolpath generation using quintic splines is presented in Section 2. This is followed by jerk limited feedrate generation in Section 3, and fifth order re-sampling in Section 4. Simulation and experimental results obtained with the proposed algorithm are presented in Section 5, and the conclusions are summarized in Section 6.

2. Quintic spline toolpath generation

In CNC machine tools, the tool motion is realized by feeding a sequence of reference points that constitute the desired toolpath, to the servo control system. The task of generating this sequence is called *interpolation*. Two basic interpolation techniques common to most CNCs are linear and circular interpolation [12,13]. In linear interpolation, the objective is to move the tool from one point to another following a linear path, as shown in Fig. 2(a). In circular interpolation, the objective is to move the tool along a defined circular arc, as shown in Fig. 2(b). In both figures, R_s denotes the starting point, R_e denotes the end point, and R_1, …, R_{N-1} are the intermediate points, and Δs is the constant path increment. In circular interpolation, R_c denotes the arc center,

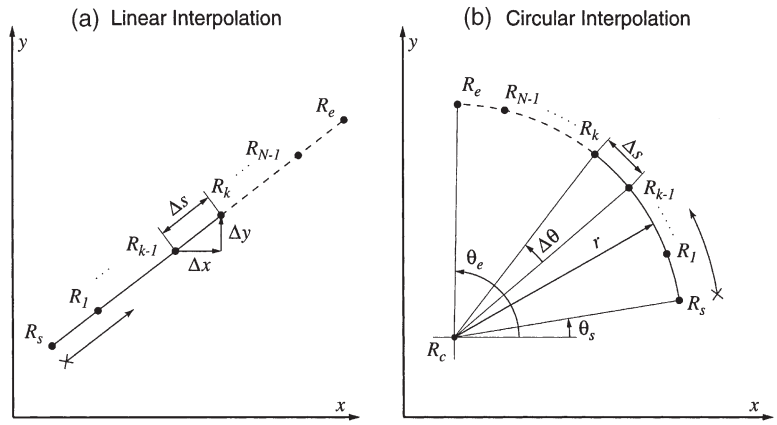

Fig. 2. Basic interpolation types common to most CNCs: (a) linear interpolation, (b) circular interpolation.

r denotes the radius, θ_s, and θ_e, are the start and end angles respectively, and $\Delta\theta$ is the angle increment which corresponds to the path increment Δs. There are also other interpolation types that employ various parametric description techniques, such as cubic and quintic splines [6,8].

In quintic spline toolpath generation, the objective is to connect a series of N reference knots, shown as $P_1, \ldots, P_N$ in Fig. 3(a), with $N-1$ fifth-order splines $S_1, \ldots, S_{N-1}$. These splines are fit in such a way that continuity up to the second derivative is preserved along the overall composite curve. This is done by estimating the first and second derivatives at the knots with respect to a chosen spline parameter, such as the chord length between the two consecutive knots. Then, a quintic spline is fit between successive knots, such that position, as well as first and second derivative boundary conditions are met at both ends of the spline. The toolpath is then generated by interpolating along the spline such that the magnitude of the resultant step size is always constant.

A common problem with parametric interpolation techniques is the difficulty in parametrizing the toolpath in terms of the arc length [6]. Instead, close approximations which have simpler

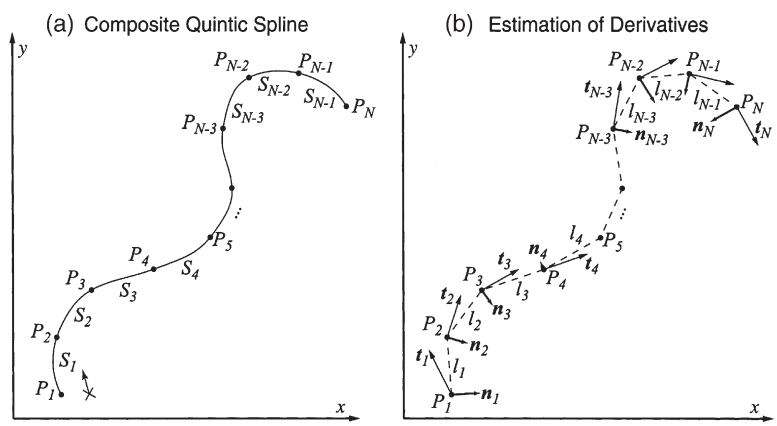

Fig. 3. (a) Composite quintic spline defined by N reference points, (b) estimation of first and second derivatives with respect to chord length as the spline parameter.

solutions are employed, at the expense of feedrate fluctuations that occur due to the difference between the actual arc length and the parameter that is being used. In this work, a new quintic spline interpolation technique is introduced where the spline parameter is readjusted at every interpolation step in order to avoid such feedrate fluctuations. In the following, the basic steps of the proposed spline interpolation scheme are presented. For simplicity, the formulation will be presented for only two dimensions, but it can easily be extended to the three dimensional case.

2.1. Estimation of derivatives

Considering Fig. 3(b), the first and second derivatives (i.e. tangent $\boldsymbol{t}_i$ and normal $\boldsymbol{n}_i$) at the knot P_i, are estimated by fitting a cubic polynomial,

$$\boldsymbol{Q}_i(u)=\boldsymbol{a}_i u^3+\boldsymbol{b}_i u^2+\boldsymbol{C}_i u+\boldsymbol{d}_i \tag{1}$$

through P_{i-1}, P_{i+1}, P_i and P_{i+2}. The parameter u corresponds to the chord (i.e. linear segment) length between the knots. For a two dimensional case, $\boldsymbol{Q}_i$, $\boldsymbol{a}_i$, $\boldsymbol{b}_i$, $\boldsymbol{c}_i$, and $\boldsymbol{d}_i$ can be written as,

$$\boldsymbol{Q}_i=\begin{bmatrix}Q_{xi}\\ Q_{yi}\end{bmatrix},\ \boldsymbol{a}_i=\begin{bmatrix}a_{xi}\\ a_{yi}\end{bmatrix},\ \boldsymbol{b}_i=\begin{bmatrix}b_{xi}\\ b_{yi}\end{bmatrix},\ \boldsymbol{c}_i=\begin{bmatrix}c_{xi}\\ c_{yi}\end{bmatrix},\ \boldsymbol{d}_i=\begin{bmatrix}d_{xi}\\ d_{yi}\end{bmatrix} \tag{2}$$

Noting that the chord length l_{i-1} between two consecutive points $P_{i-1}(x_{i-1},y_{i-1})$ and $P_i(x_i, y_i)$ can be expressed as,

$$l_{i-1}=\sqrt{(x_i-x_{i-1})^2+(y_i-y_{i-1})^2} \tag{3}$$

and also defining the notation $l_{i-1,i}$ and $l_{i-1,i+1}$, which represents the summation of consecutive chord lengths as,

$$l_{i-1,i}\underline{\underline{\triangle}}=l_{i-1}+l_i,\ l_{i-1,i+1}\underline{\underline{\triangle}}=l_{i-1}+l_i+l_{i+1} \tag{4}$$

the coefficients $\boldsymbol{a}_i$, $\boldsymbol{b}_i$, $\boldsymbol{c}_i$, and $\boldsymbol{d}_i$ can be computed as,

$$\boldsymbol{a}_i=\frac{1}{\Delta}\begin{bmatrix}\Delta_{ax}\\ \Delta_{ay}\end{bmatrix},\ \boldsymbol{b}_i=\frac{1}{\Delta}\begin{bmatrix}\Delta_{bx}\\ \Delta_{by}\end{bmatrix},\ \boldsymbol{c}_i=\frac{1}{\Delta}\begin{bmatrix}\Delta_{cx}\\ \Delta_{cy}\end{bmatrix},\ \boldsymbol{d}_i=\begin{bmatrix}x_{i-1}\\ y_{i-1}\end{bmatrix} \tag{5}$$

where,

$$\Delta=\begin{vmatrix} l_{i-1}^3 & l_{i-1}^2 & l_{i-1}\\ l_{i-1,i}^3 & l_{i-1,i}^2 & l_{i-1,i}\\ l_{i-1,i+1}^3 & l_{i-1,i+1}^2 & l_{i-1,i+1}\end{vmatrix},\ \Delta_{ax}=\begin{vmatrix} x_i-x_{i-1} & l_{i-1}^2 & l_{i-1}\\ x_{i+1}-x_{i-1} & l_{i-1,i}^2 & l_{i-1,i}\\ x_{i+2}-x_{i-1} & l_{i-1,i+1}^2 & l_{i-1,i+1}\end{vmatrix},$$

$$\Delta_{bx}=\begin{vmatrix} l_{i-1}^3 & x_i-x_{i-1} & l_{i-1}\\ l_{i-1,i}^3 & x_{i+1}-x_{i-1} & l_{i-1,i}\\ l_{i-1,i+1}^3 & x_{i+2}-x_{i-1} & l_{i-1,i+1}\end{vmatrix},\ \Delta_{cx}=\begin{vmatrix} l_{i-1}^3 & l_{i-1}^2 & x_i-x_{i-1}\\ l_{i-1,i}^3 & l_{i-1,i}^2 & x_{i+1}-x_{i-1}\\ l_{i-1,i+1}^3 & l_{i-1,i+1}^2 & x_{i+2}-x_{i-1}\end{vmatrix}$$

Δ_{ay} Δ_{by}, and Δ_{cy} are obtained by replacing the $x_k - x_i$ terms with $y_k - y_i$. Once the polynomial coefficients are calculated, the first and second derivatives (i.e. the tangent $\boldsymbol{t}_i$ and normal $\boldsymbol{n}_i$ at knot P_i) are estimated as,

$$\left.\begin{aligned} \boldsymbol{t}_i \triangleq \begin{bmatrix} t_{xi} \\ t_{yi} \end{bmatrix} &= \frac{\mathrm{d}\boldsymbol{Q}_i}{\mathrm{d}u}\Big|u=l_{i-1} = (3\boldsymbol{a}_i u^2 + 2\boldsymbol{b}_i u + \boldsymbol{c}_i)\big|_{u=l_{i-1}} \\ \boldsymbol{n}_i \triangleq \begin{bmatrix} n_{xi} \\ n_{yi} \end{bmatrix} &= \frac{\mathrm{d}^2 Q_i}{\mathrm{d}u^2}\Big|u=l_i - 1 = (6\boldsymbol{a}_i u + 2\boldsymbol{b}_i)\big|_{u=l_{i-1}} \end{aligned}\right\} \tag{6}$$

Due to the lack of extra neighboring knots, the tangents and normals at points P_1, P_{N-1}, and P_N are estimated using the cubic polynomials fit for the available knots, i.e. $\boldsymbol{t}_1$ and $\boldsymbol{n}_1$ are estimated using $\boldsymbol{Q}_2(u)|_{u=0}$; $\boldsymbol{t}_{N-1}$, $\boldsymbol{n}_{N-1}$ and $\boldsymbol{t}_{N,}$ $\boldsymbol{n}_n$ are estimated using $\boldsymbol{Q}_{N-2}(u)|_{u=l_{N-3}+l_{N-2}}$ and $\boldsymbol{Q}_{N-2}(u)|_{u=l_{N-3}+l_{N-2}+l_{N-1}}$ as follows,

$$\left.\begin{aligned} &\boldsymbol{t}_1 = 3\boldsymbol{a}_2 u^2 + 2\boldsymbol{b}_2 u + \boldsymbol{c}_2,\ \boldsymbol{n}_1 = 6\boldsymbol{a}_2 u + 2\boldsymbol{b}_2 \text{ for } u=0 \\ &\boldsymbol{t}_{N-1} = 3\boldsymbol{a}_{N-2} u^2 + 2\boldsymbol{b}_{N-2} u + \boldsymbol{c}_{N-2},\ \boldsymbol{n}_{N-1} = 6\boldsymbol{a}_{N-2} u + 2\boldsymbol{b}_{N-2} \text{ for } u = l_{N-3} + l_{N-2} \\ &\boldsymbol{t}_N = 3\boldsymbol{a}_{N-2} u^2 + 2\boldsymbol{b}_{N-2} u + \boldsymbol{c}_{N-2},\ n_N = 6\boldsymbol{a}_{N-2} u + 2\boldsymbol{b}_{N-2} \text{ for } u = l_{N-3} + l_{N-2} + l_{N-1} \end{aligned}\right\} \tag{7}$$

2.2. Fitting of the quintic spline

Once the first and second derivatives at the knots are determined, a quintic spline in the form,

$$\boldsymbol{S}_i(u) = \boldsymbol{A}_i u^5 + \boldsymbol{B}_i u^4 + \boldsymbol{C}_i u^3 + \boldsymbol{D}_i u^2 + \boldsymbol{E}_i u + \boldsymbol{F}_i \tag{8}$$

is fit between consecutive knots, to satisfy the boundary conditions of second order continuity. For the two axis case, $\boldsymbol{S}_i$, $\boldsymbol{A}_i$, $\boldsymbol{B}_i$, …, $\boldsymbol{F}_i$ are in the form,

$$\boldsymbol{S}_i = \begin{bmatrix} S_{xi} \\ S_{yi} \end{bmatrix},\ \boldsymbol{A}_i = \begin{bmatrix} A_{xi} \\ A_{yi} \end{bmatrix},\ \boldsymbol{B}_i = \begin{bmatrix} B_{xi} \\ B_{yi} \end{bmatrix},\ \ldots,\ \boldsymbol{F}_i = \begin{bmatrix} F_{xi} \\ F_{yi} \end{bmatrix} \tag{9}$$

For the quintic spline $\boldsymbol{S}_i(u)$ fit between points $P_i(x_i, y_i)$ and P_{i+1} (x_{i+1}, y_{i+1}) (which are denoted with the position vectors $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i+1}$, respectively), the following boundary conditions are used.

$$\left.\begin{aligned} &\boldsymbol{S}_i(u)|_{u=0} = \boldsymbol{p}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix},\ \boldsymbol{S}_i(u)|_{u=l_i} = \boldsymbol{p}_{i+1} = \begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} \\ &\frac{\mathrm{d}\boldsymbol{S}_i(u)}{\mathrm{d}u}\Big|_{u=0} = \boldsymbol{t}_i = \begin{bmatrix} t_{xi} \\ t_{yi} \end{bmatrix},\ \frac{\mathrm{d}\boldsymbol{S}_i(u)}{\mathrm{d}u}\Big|_{u=l_i} = \boldsymbol{t}_{i+1} = \begin{bmatrix} t_{x,i+1} \\ t_{y,i+1} \end{bmatrix} \\ &\frac{\mathrm{d}^2\boldsymbol{S}_i(u)}{\mathrm{d}u^2}\Big|_{u=0} = \boldsymbol{n}_i = \begin{bmatrix} n_{xi} \\ n_{yi} \end{bmatrix},\ \frac{\mathrm{d}^2\boldsymbol{S}_i(u)}{\mathrm{d}u^2}\Big|_{u=l_i} = \boldsymbol{n}_{i+1} = \begin{bmatrix} n_{x,i+1} \\ n_{y,i+1} \end{bmatrix} \end{aligned}\right\} \tag{10}$$

where $\boldsymbol{t}_i$, $\boldsymbol{n}_i$, $\boldsymbol{t}_{i+1}$, and $\boldsymbol{n}_{i+1}$ are calculated from Eq. (6). The $\boldsymbol{A}_i$, $\boldsymbol{B}_i$,..., $\boldsymbol{F}_i$ coefficients are obtained for the boundary conditions in Eq. (10). The solution for the x axis is presented below. The y axis solution can be obtained by replacing occurrences of x with y.

$$\left.\begin{aligned}
A_{xi} &= \frac{1}{l_i^5}[6(x_{i+1}-x_i)-3(t_{x,i+1}+t_{xi})l_i+0.5(n_{x,i+1}-n_{xi})l_i^2] \\
B_{xi} &= \frac{1}{l_i^4}[15(x_i-x_{i+1})+(7t_{x,i+1}+8t_{xi})l_i+(1.5n_{xi}-n_{x,i+1})l_i^2] \\
C_{xi} &= \frac{1}{l_i^3}[10(x_{i+1}-x_i)-(4t_{x,i+1}+6t_{xi})l_i-(1.5n_{xi}-0.5n_{x,i+1})l_i^2] \\
D_{xi} &= 0.5n_{xi} \\
E_{xi} &= t_{xi} \\
F_{xi} &= x_i
\end{aligned}\right\} \tag{11}$$

Once the coefficients for all segments are determined, the total length of the composite spline is calculated by numerically integrating the arc length for each segment, and summing the segment lengths as,

$$L=\sum_{i=1}^{N-1} s_i=\sum_{i=1}^{N-1}\int_0^{s_i} \mathrm{d}s \tag{12}$$

where L is the total composite spline length and s_i is the length the ith spline segment. The numerical value of the total travel length is required in order to plan the feedrate generation, as will be done in Section 3. The spline arc length s_i is evaluated by dividing the chord length l_i for each spline segment into M_i increments, and for each chord increment the corresponding points on the spline are estimated. The spatial increments between consecutive spline points are then-summed up in order to obtain s_i. A reasonable choice for M_i is,

$$M_i=\mathrm{round}\left(\frac{l_i}{F\cdot T_\mathrm{s}}\right) \tag{13}$$

where F is the planned feedrate of travel, T_s is the position control loop sampling period, and l_i is the chord length of the ith spline, as already defined in Eq. (3). Hence the chord increment becomes,

$$dl_i=l_i/M_i \tag{14}$$

Considering $j=1, 2,\ldots, M_i$ to be the chord increment counter, the corresponding point $[x_{ij}y_{ij}]^\mathrm{T}$ on the ith spline is computed as,

$$\begin{bmatrix} x_{ij} \\ y_{ij} \end{bmatrix}=\boldsymbol{A}_i(j\cdot dl_i)^5+\boldsymbol{B}_i(j\cdot dl_i)^4+\boldsymbol{C}_i(j\cdot dl_i)^3+\boldsymbol{D}_i(j\cdot dl_i)^2+\boldsymbol{E}_i(j\cdot dl_i)+\boldsymbol{F}_i \tag{15}$$

where $\boldsymbol{A}_i$, $\boldsymbol{B}_i$, …, $\boldsymbol{F}_i$ were defined in Eqs. (9) and (11). The arc length between two successive points can be expressed as,

$$ds_{ij} \cong \sqrt{(dx_{ij})^2+(dy_{ij})^2} = \sqrt{(x_{ij}-x_{i,j-1})^2+(y_{ij}-y_{i,j-1})^2} \tag{16}$$

Hence, the total length of travel L can be evaluated as,

$$L = \sum_{i=1}^{N-1} \int_0^{s_i} ds \cong \sum_{i=1}^{N-1} \sum_{j=1}^{M_i} ds_{ij} = \sum_{i=1}^{N-1} \sum_{j=1}^{M_i} \sqrt{(x_{ij}-x_{i,j-1})^2+(y_{ij}-y_{i,j-1})^2} \tag{17}$$

2.3. Interpolation without feedrate fluctuation

The quintic spline is parametrized with respect to the chord length between two consecutive reference knots. If interpolated with constant parameter increment, due to the difference between the chord and arc lengths, feedrate fluctuations will occur, which in return will lead to undesirable acceleration and jerk fluctuations. Ideally, to avoid this situation, the fit curve would have to be parametrized with respect to its arc length. This approach was implemented in an approximate manner in [6,7] using nearly arc-length spline parametrization. Although this improved the correctness of the resultant feedrate profile, significant fluctuations still occurred at high curvatures.

In this work, the chord length is retained as the spline parameter. However, the parameter increment is adjusted recursively to realize a constant displacement magnitude at each step of interpolation, hence avoiding feedrate fluctuations.

The path increment Δs is determined in such a way that, when T^i reaches its smallest value, which is the control loop sampling period T_s, the highest value of feedrate f_{max} is achieved. Hence, the step size is chosen assuming that the total travel length L would be encompassed at the highest expected feedrate f_{max}, which requires the selection of the total number of interpolation steps as,

$$N_i = \text{round}\left(\frac{L}{f_{max} T_s}\right) \tag{18}$$

resulting in a path increment of,

$$\Delta s = L/N_i \tag{19}$$

at each interpolation step. It is assumed that the commanded motion originates at an initial feedrate of f_s, then accelerates or decelerates to the desired feedrate F, and terminates by reaching a final feedrate of f_e. Allowing nonzero initial and final feedrates enables consecutive motion segments to be smoothly connected with each other. In planning the path increment size, the highest feedrate should be considered as,

$$f_{max} = \max(f_s, F, f_e). \tag{20}$$

For generating reference position commands along the spline $\boldsymbol{S}_i$, let $R_{ij}(x_{ij}, y_{ij})$ be the last reference point that was computed, and $R_{i,j+1}(x_{i,j+1}, y_{i,j+1})$ be the next one to follow. In this case, the resultant path increment can be written as,

$$\Delta s = \sqrt{(\Delta x)^2+(\Delta y)^2} \tag{21}$$

where the axis increments Δx and Δy are,

$$\left.\begin{aligned}\Delta x &= x_{i,j+1} - x_{ij} = A_{xi}u^5 + B_{xi}u^4 + C_{xi}u^3 + D_{xi}u^2 + E_{xi}u + F_{xi} - x_{ij} \\ \Delta y &= y_{i,j+1} - y_{ij} = A_{yi}u^5 + B_{yi}u^4 + C_{yi}u^3 + D_{yi}u^2 + E_{yi}u + F_{yi} - y_{ij}\end{aligned}\right\} \tag{22}$$

Since the previous position (x_{ij}, y_{ij}) and the spline coefficients $\boldsymbol{A}_i, \boldsymbol{B}_i, \ldots, \boldsymbol{F}_i$, are known, the problem is to find the new chord parameter u, so that the fixed arc increment Δs is realized. Combining Eqs. (21) and (22), the new chord parameter can be obtained by solving the root of the below tenth-order polynomial:

$$g(u) = \alpha_0 u^{10} + \alpha_1 u^9 + \ldots + \alpha_{10} \tag{23}$$

where the α coefficients are expressed as,

$$\left.\begin{aligned}
\alpha_0 &= A_{xi}^2 + A_{yi}^2 \\
\alpha_1 &= 2(A_{xi}B_{xi} + A_{yi}B_{yi}) \\
\alpha_2 &= B_{xi}^2 + B_{yi}^2 + 2(A_{xi}C_{xi} + A_{yi}C_{yi}) \\
\alpha_3 &= 2(B_{xi}C_{xi} + B_{yi}C_{yi} + A_{xi}D_{xi} + A_{yi}D_{yi}) \\
\alpha_4 &= C_{xi}^2 + C_{yi}^2 + 2(A_{xi}E_{xi} + A_{yi}E_{yi} + B_{xi}D_{xi} + B_{yi}D_{yi}) \\
\alpha_5 &= 2(A_{xi}F'_{xi} + A_{yi}F'_{yi} + B_{xi}E_{xi} + B_{yi}E_{yi} + C_{xi}D_{xi} + C_{yi}D_{yi}) \\
\alpha_6 &= D_{xi}^2 + D_{yi}^2 + 2(B_{xi}F'_{xi} + B_{yi}F'_{yi} + C_{xi}E_{xi} + C_{yi}E_{yi}) \\
\alpha_7 &= 2(D_{xi}E_{xi} + D_{yi}E_{yi} + C_{xi}F'_{xi} + C_{yi}F'_{yi}) \\
\alpha_8 &= E_{xi}^2 + E_{yi}^2 + 2(D_{xi}F'_{xi} + D_{yi}F'_{yi}) \\
\alpha_9 &= 2(E_{xi}F'_{xi} + E_{yi}F'_{yi}) \\
\alpha_{10} &= F'^2_{xi} + F'^2_{yi} - (\Delta s)^2
\end{aligned}\right\} \tag{24}$$

In the above expression, $F'_{xi} = F_{xi} - x_{ij}$, $F'_{yi} = F_{yi} - y_{ij}$. $A_{xi}, B_{xi}, \ldots, F_{xi}$, and $A_{yi}, B_{yi}, \ldots, F_{yi}$ are the spline coefficients of the ith quintic spline. The solution of Eq. (23) is obtained through Newton–Raphson's iterative method. Although $g(u)$ has a relatively high order, the result converges in less than three iterations, making this approach feasible for real-time implementation. Furthermore, the coefficients $\alpha_0, \ldots, \alpha_4$ in Eq. (23) are only calculated in the beginning of the spline segment. The terms involving F'_{xi} and F'_{yi}, which are $\alpha_5, \ldots, \alpha_{10}$ need to be updated at every interpolation step.

With this approach, the feedrate fluctuations arising from approximate parametrization are avoided. Step size constancy is accurately maintained even in toolpaths with relatively high curvature. This allows jerk limited feedrate profiles to be realized as they are as planned, without degradations arising from unwanted feedrate fluctuations.

3. Jerk limited feedrate profiling

This section presents a procedure for imposing a smooth feed motion along the quintic spline toolpath, which is done by modulating the time duration (i.e. interpolation period T^i) between

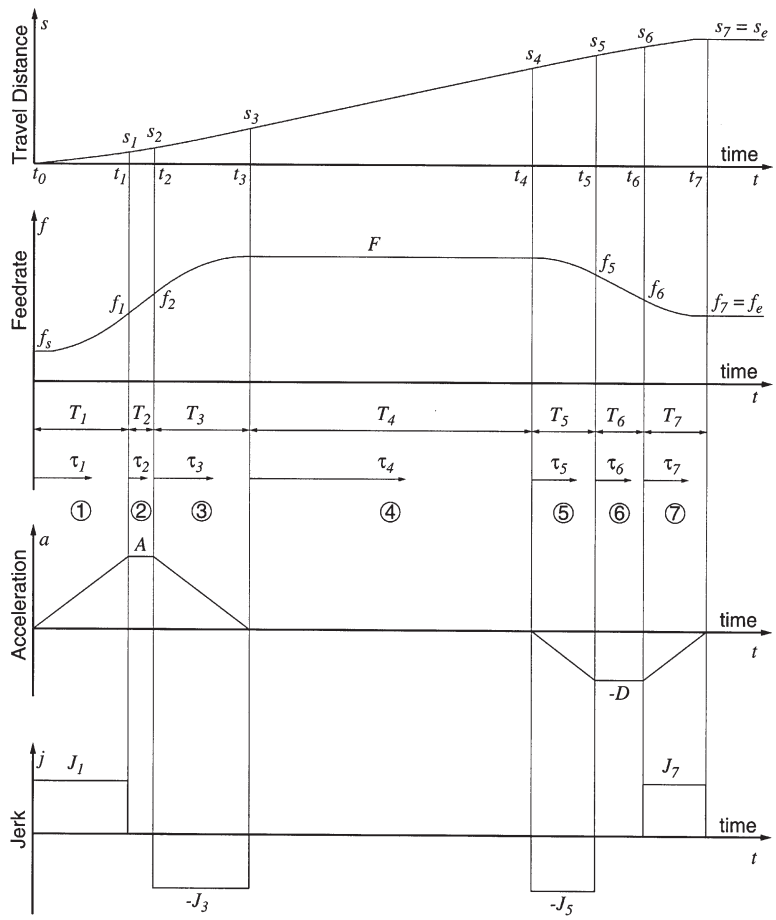

Fig. 4. Kinematic profiles for trapezoidal acceleration based feedrate generation.

evenly spaced position reference points. Using such an approach in trajectory generation allows the feedrate to be easily modified on the fly, just by rescaling the interpolation period with a desired override factor.

The kinematic profiles used in feedrate generation are illustrated in Fig. 4. For motion along the toolpath, accelerations have trapezoidal profiles with prespecified slopes (i.e. jerk values). Acceleration profiles are linear, feedrate profiles are parabolic and displacement profiles are cubic for regions 1, 3, 5, and 7 where accelerations and decelerations with constant jerk occur. Acceleration values are constant and jerk is zero for regions 2 and 6, where velocity profiles are linear and displacement profiles are parabolic (as in trapezoidal feedrate generation). In region 4, jerk and acceleration values are zero, feedrate is constant and displacement is linear.

In the following, the formulation and implementation of jerk limited feedrate profiling are presented.

3.1. Formulation

Considering Fig. 4, if the initial conditions for displacement, feedrate and acceleration at time $t_i (i=0, 1,\ldots, 6)$ are known, and the jerk profile is known, the acceleration $a(t)$, feedrate $f(t)$, and displacement $s(t)$ profiles can be obtained by integrating the jerk profile $j(t)$ as,

$$a(t)=a(t_i)+\int_{t_i}^{t} j(\tau_i)d\tau_i,\ f(t)=f(t_i)+\int_{t_i}^{t} a(\tau_i)d\tau_i,\ s(t)=s(t_i)+\int_{t_i}^{t} f(\tau_i)d\tau_i \tag{25}$$

The jerk profile in Fig. 4 can be written as,

$$j(\tau)=\begin{cases} J_1, & 0\le t<t_1 \\ 0, & t_1\le t<t_2 \\ -J_3, & t_2\le t<t_3 \\ 0, & t_3\le t<t_4 \\ -J_5, & t_4\le t<t_5 \\ 0, & t_5\le t<t_6 \\ J_7, & t_6\le t\le t_7 \end{cases} \tag{26}$$

where t denotes absolute time, $t_1, t_2,\ldots, t_7$ denote the time boundaries of each phase; J_1, J_3, J_5, and J_7 are the magnitudes of jerk in regions 1, 3, 5, and 7.

Integrating Eq. (26) with respect to time, and denoting the acceleration and deceleration magnitudes at phases 2 and 6 with A and D respectively, the acceleration profile can be expressed as,

$$a(\tau)=\begin{cases} J_1\tau_1, & 0\le t<t_1 \\ A, & t_1\le t<t_2 \\ A-J_3\tau_3, & t_2\le t<t_3 \\ 0, & t_3\le t<t_4 \\ -J_5\tau_5, & t_4\le t<t_5 \\ -D, & t_5\le t<t_6 \\ -D+J_7\tau_7, & t_6\le t\le t_7 \end{cases} \tag{27}$$

where τ_k is the relative time parameter that starts at the beginning of the kth phase, as shown in Fig. 4. Integrating Eq. (27) with respect to time, the feedrate profile is obtained as,

$$f(\tau)=\begin{cases} f_s+\frac{1}{2}J_1\tau_1^2, & 0\le t<t_1,\ f_s\text{: initial feedrate} \\ f_1+A\tau_2, & t_1\le t<t_2,\ f_1=f_s+\frac{1}{2}J_1T_1^2 \\ f_2+A\tau_3-\frac{1}{2}J_3\tau_3^2, & t_2\le t<t_3,\ f_2=f_1+AT_2 \\ f_3, & t_3\le t<t_4,\ f_3=f_2+AT_3-\frac{1}{2}J_3T_3^2=F \\ f_4-\frac{1}{2}J_5\tau_5^2, & t_4\le t<t_5,\ f_4=f_3 \\ f_5-D\tau_6, & t_5\le t<t_6,\ f_5=f_4-\frac{1}{2}J_5T_5^2 \\ f_6-D\tau_7+\frac{1}{2}J_7\tau_7^2, & t_6\le t\le t_7,\ f_6=f_5-DT_6 \end{cases} \tag{28}$$

where f_s is the initial feedrate, F is the desired feedrate which needs to be achieved at the end of the third phase. $T_k(k=1, 2,\ldots, 7)$ is the duration of the kth phase and f_k is the feedrate reached at the end of the corresponding phase. Again, integrating Eq. (28) with respect to time yields the displacement profile as,

$$s(\tau)=\begin{cases} f_s\tau_1+\frac{1}{6}J_1\tau_1^3, & 0\le t<t_1 \\ s_1+f_1\tau_2\frac{1}{2}A\tau_2^2, & t_1\le t<t_2,\ s_1=s_s+f_sT_1+\frac{1}{6}J_1T_1^3 \\ s_2+f_2\tau_3+\frac{1}{2}A\tau_3^2-\frac{1}{6}J_3\tau_3^3, & t_2\le t<t_3,\ s_2=s_1+f_1T_2+\frac{1}{2}AT_2^2 \\ s_3+f_3\tau_4, & t_3\le t<t_4,\ s_3=s_2+f_2T_3+\frac{1}{2}AT_3^2-\frac{1}{6}J_3T_3^3 \\ s_4+f_4\tau_5-\frac{1}{6}J_5\tau_5^3, & t_4\le t<t_5,\ s_4=s_3+f_3T_4 \\ s_5+f_5\tau_6-\frac{1}{2}D\tau_6^2, & t_5\le t<t_6,\ s_5=s_4+f_4T_5-\frac{1}{6}J_5T_5^3 \\ s_6+f_6\tau_7-\frac{1}{2}D\tau_7^2+\frac{1}{6}J_7\tau_7^3, & t_6\le t\le t_7,\ s_6=s_5+f_5T_6-\frac{1}{2}DT_6^2 \end{cases} \tag{29}$$

where $s_k(k=1, 2,\ldots, 7)$ is the displacement reached at the end of the kth phase. Hence the distance traveled during each phase can be written as,

$$l_k=\begin{cases} l_1=s_1+f_sT_1+\frac{1}{6}J_1T_1^3 \\ l_2=s_2-s_1=f_1T_2+\frac{1}{2}AT_2^2 \\ l_3=s_3-s_2=f_2T_3+\frac{1}{2}AT_3^2-\frac{1}{6}J_3T_3^3 \\ l_4=s_4-s_3=f_3T_4 \\ l_5=s_5-s_4=f_4T_5-\frac{1}{6}J_5T_5^3 \\ l_6=s_6-s_5=f_5T_6-\frac{1}{2}DT_6^2 \\ l_7=s_7-s_6=f_6T_7-\frac{1}{2}DT_7^2+\frac{1}{6}J_7T_7^3 \end{cases} \tag{30}$$

Furthermore, from the trapezoidal nature of the acceleration profile,

$$\left.\begin{aligned} A&=J_1T_1=J_3T_3 \\ D&=J_5T_5=J_7T_7 \end{aligned}\right\} \tag{31}$$

should hold. Considering that the desired feedrate F is reached at the end of the 3rd stage Eq. (28)

$$f_3=F\rightarrow T_2=\frac{1}{A}\left[F-f_s-\frac{1}{2}J_1T_1^2-AT_3+\frac{1}{2}J_3T_3^2\right] \tag{32}$$

and similarly, considering that the final feedrate f_e is reached at the end of the 7th stage,

$$f_7=f_6-DT_7+\frac{1}{2}J_7T_7^2=f_e\rightarrow T_6=\frac{1}{D}\left[F-f_e-\frac{1}{2}J_5T_5^2-DT_7+\frac{1}{2}J_7T_7^2\right] \tag{33}$$

Furthermore, the total distance traveled at the end of the seventh phase should be equal to the total path distance L,

$$s_7=s_6+f_6T_7-\frac{1}{2}DT_7^2+\frac{1}{6}J_7T_7^3=L \tag{34}$$

3.2. Initialization

The inputs to the jerk-limited feedrate generation algorithm are the control loop sampling period (T_s), the total distance of travel (L), the total number of interpolation steps (N_i), initial, desired, and final feedrates (f_s, F, and f_e, respectively), desired acceleration and deceleration magnitudes (A and D), and the desired jerk magnitude (J).

Initially, the correct signs for acceleration, deceleration and jerk values are determined from the given feedrate conditions.

$$\left.\begin{aligned} A&=\mathrm{sgn}(F-f_s)\cdot|A|, \quad J_1=J_3=\mathrm{sgn}(A)\cdot|J| \\ D&=\mathrm{sgn}(F-f_e)\cdot|D|, \quad J_5=J_7=\mathrm{sgn}(D)\cdot|J| \end{aligned}\right\} \tag{35}$$

A negative value for A indicates deceleration instead of acceleration in the beginning of motion. Similarly, a negative value for D indicates acceleration instead of deceleration. If the value of A or D is zero, this indicates the absence of an acceleration or a deceleration stage. The jerk limited feedrate generation algorithm is designed to function in the most general case and allow smooth transition between joined paths for all feasible boundary conditions of feedrate. This is the reason that special cases are also considered.

The total number of interpolation steps is also checked. If both acceleration and deceleration stages exist ($A\neq0$ and $D\neq0$), then $N_i\geq4$ has to hold in order to allow at least one step for phases 1, 3, 5, and 7. If only an acceleration or deceleration stage is missing (i.e. ($A=0$ and $D\neq0$) or ($A\neq0$ and $D=0$)), then $N_i\geq2$ has to hold in order to allow at least phases 1 and 3, or 5 and 7 to be realized. If no acceleration or deceleration is required, which would happen when the initial, desired, and final feedrates are set equal to each other, $N_i\geq1$ has to hold for a nonzero travel length of L. Upon violation of the above conditions, N_i is set to its minimum feasible value, and the path increment Δs is recalculated using Eq. (19).

The requested kinematic values are then checked for compatibility, and modified if necessary as follows:

3.2.1. Jerk condition

The requested value of jerk should not be larger than the achievable value for given acceleration, deceleration and sampling period values. This can be expressed as,

$$J\leq\min\left(\frac{|A|}{T_s},\frac{|D|}{T_s}\right) \tag{36}$$

for $|A|>0$ and $|D|>0$. If this condition does not hold, the desired jerk value is reduced to its maximum possible value as

$$J=\min\left(\frac{|A|}{T},\frac{|D|}{T}\right) \tag{37}$$

If acceleration is absent, only the deceleration value is considered, and vice versa. If both acceleration and deceleration are absent, this condition is not checked.

3.2.2. Acceleration condition

If an acceleration stage exists in the motion, the desired feedrate F should be reached at the end of the third stage of interpolation, either including a constant acceleration stage, or without one. Considering the expression for T_2 in Eq. (32), this implies that $T_2\geq0$. Also considering that $J_1=J_3$ [Eq. (35)], and $T_3=A/J_3$ Eq. (31), the acceleration condition becomes,

$$T_2=\frac{F-f_s}{A}-\frac{A}{J_1}\geq0 \tag{38}$$

for $A \neq 0$. If Eq. (38) does not hold, then the magnitude of acceleration is reduced to its maximum possible limit as,

$$A = \mathrm{sgn}(A) \cdot \sqrt{J_1(F - f_s)} \tag{39}$$

and T_2 is set to zero.

3.2.3. Deceleration condition

If a deceleration stage exists in the motion, the final feedrate f_e should be reached at the end of the seventh stage of interpolation, either including a constant deceleration stage, or without one. Considering the expression for T_6 in Eq. (33), this implies that $T_6 \geq 0$. Also considering that $J_5 = J_7$ [Eq. (35)], and $T_7 = D/J_7$ [Eq. (31)], the deceleration condition becomes,

$$T_6 = \frac{F - f_e}{D} - \frac{D}{J_5} \geq 0 \tag{40}$$

if $D \neq 0$. If Eq. (40) does not hold, then the magnitude of deceleration D is reduced to its maximum possible limit as,

$$D = \mathrm{sgn}(D) \cdot \sqrt{J_5(F - f_e)} \tag{41}$$

and T_6 is set to zero.

3.2.4. Travel length condition:

The total length of travel L should be encompassed through the seven stages of interpolation, either including a constant feedrate stage or without one. Therefore, the constant feedrate interval T_4 should be $T_4 \geq 0$.

By considering that $J_1 = J_3$, $J_5 = J_7$ (Eq. (35)), which leads to $T_1 = T_3 = A/J_1$, $T_5 = T_7 = D/J_5$ [Eq. (31)], the total travel length in Eq. (34) can be rewritten, using Eqs. (28) and (29), and substituting T_2 and T_6 from Eqs. (32) and (33) as,

$$L = \left(\frac{1}{2A} + \frac{1}{2D}\right)F^2 + \left(\frac{A}{2J_1} + \frac{D}{2J_5} + T_4\right)F + \left(\frac{Af_s}{2J_1} + \frac{Df_e}{2J_5} - \frac{f_s^2}{2A} - \frac{f_e^2}{2D}\right) \tag{42}$$

If an acceleration or deceleration stage does not exist, its corresponding terms should not be considered in the above equation. For $T_4 \geq 0$ to hold,

$$T_4 = \frac{1}{F}\left[L - \left\{\left(\frac{1}{2A} + \frac{1}{2D}\right)F^2 + \left(\frac{A}{2J_1} + \frac{D}{2J_5}\right)F + \left(\frac{Af_s}{2J_1} + \frac{Df_e}{2J_5} - \frac{f_s^2}{2A} - \frac{f_e^2}{2D}\right)\right\}\right] \geq 0 \tag{43}$$

should be satisfied. If Eq. (43) does not hold, then T_4 is set to zero and the magnitude of feedrate is reduced to its maximum possible value as,

$$F = \frac{-\beta + \sqrt{\beta^2 - 4\alpha\gamma}}{2\alpha} \tag{44}$$

where $\alpha=\frac{1}{2A}+\frac{1}{2D}$, $\beta=\frac{A}{2J_1}+\frac{D}{2J_5}$, $\gamma=\frac{Af_s}{2J_1}+\frac{Df_e}{2J_5}-\frac{f_s^2}{2A}-\frac{f_e^2}{2D}-L$. If Eq. (44) possesses complex roots, then initial and final feedrates f_s and f_e are set to zero. Path generation initialization is re-invoked to use the new maximum feedrate; and the other kinematic parameters J, A, D and F are readjusted if necessary to obtain a realizable case.

For interpolation initialization to continue, all above conditions must be realized. If any of the conditions require a parameter adjustment, then the dependent conditions are also checked.

3.2.5. Determination and quantization of travel lengths

In the next part of initialization, feedrates ($f_1, f_2,\ldots, f_6$) to be reached at the end of each stage and the distances ($l_1, l_2,\ldots, l_6$) to be travelled in each stage are calculated from Eqs. (28) and (30).

The number of interpolation steps for each phase are calculated as follows:

$$\left.\begin{aligned} &N_1=\text{round}(l_1/\Delta_s),\ N_3=\text{round}(l_3/\Delta_s)\\ &N_5=\text{round}(l_5/\Delta_s),\ N_7=\text{round}(l_7/\Delta_s)\end{aligned}\right\} \tag{45}$$

if any of N_1, N_3, N_5, or N_7 are zero for corresponding nonzero l_1, l_3, l_5, and l_7 they are set to one. Then, the total number of steps for the acceleration and deceleration stages are calculated as,

$$N_{acc}=\text{round}((l_1+l_2+l_3)/\Delta_s),\ N_{dec}=\text{round}((l_5+l_6+l_7)/\Delta_s) \tag{46}$$

and the number of steps for jerkless acceleration and deceleration stages are obtained as,

$$N_2=N_{acc}-(N_1+N_3),\ N_6=N_{dec}-(N_5+N_7) \tag{47}$$

which yield the number of constant feedrate steps as,

$$N_4=N-(N_{acc}+N_{dec}) \tag{48}$$

Using the path increment Δs and number of interpolation steps $N_1, N_2,\ldots, N_7$ for each stage, the travel lengths are quantized as,

$$l_k'=N_k\cdot\Delta_s \text{ where } k=1,2,\ldots,7 \tag{49}$$

3.2.6. Readjustment of acceleration and jerk values

After travel length quantizations, acceleration, time and jerk values are readjusted to maintain the specified feedrates (f_s, F, and f_e) for the new travel lengths.

For the acceleration stage, the expressions of l_1, l_2, and l_3, in Eq. (30) are used to readjust the values of A, T_1, and T_3, which would also result in the adjustment of J_1 and J_3, due to Eq. (31).

For the case $T_2>0$, the expression for T_2 is substituted from Eq. (32) and l_1, l_2, and l_3, are substituted by l'_1, l'_2,and l'_3, respectively, resulting in the following equations to be solved for A, T_1, and T_3,

$$\left.\begin{aligned} &f_sT_1+\frac{1}{6}AT_1^2-l'_1=0\\ &-\frac{1}{8}AT_1^2+\frac{1}{8}AT_3^2-\frac{1}{2}f_sT_1-\frac{1}{2}FT_3+\frac{F^2-f_s^2}{2A}-l_2'=0\\ &FT_3-\frac{1}{6}AT_3^2-l'_3=0\end{aligned}\right\} \tag{50}$$

If T_2=0, the l_2 expression in Eq. (30) cannot be used. Instead, the feedrate condition for the end of the acceleration stage (the expression for f_3 in Eq. (28)) is used as the second equation. In this case, the equations to be solved for A, T_1, and T_3 become,

$$\left.\begin{aligned} &f_s T_1+\frac{1}{6}AT_1^2-l_1'=0\\ &\frac{1}{2}AT_1+\frac{1}{2}AT_3+f_s-F=0\\ &\frac{1}{3}AT_3^2-\frac{1}{2}AT_1T_3+f_sT_3-l_3'=0 \end{aligned}\right\} \tag{51}$$

Similarly, for the deceleration stage, the expressions of l_5, l_6, and l_7, in Eq. (30) are used to readjust the vales of D, T_5, and T_7, which would also result in the adjustment of J_5 and J_7, due to Eq. (31).

For the case $T_6>0$, the expression for T_6 is substituted from Eq. (33) and l_5, l_6 and l_7, are substituted by l'_5,l'_6, and l'_7, respectively, resulting in the following equations to be solved for D, T_5, and T_7,

$$\left.\begin{aligned} &FT_5-\frac{1}{6}DT_5^2-l_5'=0\\ &\frac{1}{8}DT_5^2-\frac{1}{8}DT_7^2-\frac{1}{2}FT_5-\frac{1}{2}f_eT_7+\frac{F^2-f_e^2}{2D}-l_6'=0\\ &f_eT_7+\frac{1}{6}DT_7^2-l_7'=0 \end{aligned}\right\} \tag{52}$$

If T_6=0, the l_6 expression in Eq. (30) cannot be used. Instead, the feedrate condition at the end of deceleration (the expression of $f(\tau)$ for $\tau_7=T_7$ in the interval of $t_6\le t\le t_7$ in Eq. (28)) is used as the second equation. In this case, the equations to be solved for D, T_5, and T_7 become,

$$\left.\begin{aligned} &FT_5-\frac{1}{6}DT_5^2-l_5'=0\\ &\frac{1}{2}DT_5+\frac{1}{2}DT_7+f_e-F=0\\ &-\frac{1}{3}DT_7^2-\frac{1}{2}DT_5T_7+FT_7-l_7'=0 \end{aligned}\right\} \tag{53}$$

The nonlinear equation systems Eqs. (50)–(53) are solved iteratively using Newton–Raphson's algorithm. Once the vales of A, T_1, T_3, and D, T_5, T_7 converge, the jerk values J_1, J_3, J_5, and J_7 are updated using Eq. (31). Furthermore, the final feedrates $f_1, f_2,\ldots, f_6$ reached at the end of each phase are recalculated from Eq. (28), to be used in the continuously executed part of the algorithm.

3.3. Continuously executed part

After the initialization is completed, the continuously executed part of jerk limited feedrate generation is invoked at each step of interpolation.

For any of the seven phases in Fig. 4, the displacement profile can be written as,

$$s(\tau_k)=\frac{1}{6}j_{0k}\tau_k^3+\frac{1}{2}a_{0k}\tau_k^2+f_{0k}\tau_k+s_{0k},\ 0\le\tau_k\le T_k \tag{54}$$

where j_{0k}, a_{0k}, f_{0k}, and s_{0k} are the jerk, acceleration, feedrate and displacement values at the beginning of the kth phase, respectively, and τ_k is the time parameter that starts at the beginning of that phase. The coefficients j_{0k}, a_{0k}, f_{0k}, and s_{0k} were already calculated during the initialization.

Since the displacement step size Δs is predetermined and constant, the objective is to compute the interpolation period between consecutive reference points such that the kinematic profiles in Fig. 4 are realized. The total distance traveled from the beginning of the kth phase, until the nth interpolation step in the same phase is,

$$s_{kn}(\tau_{kn})=n\cdot\Delta s=\frac{1}{6}j_{0k}\tau_{kn}^3+\frac{1}{2}a_{0k}\tau_{kn}^2+f_{0k}\tau_{kn}+s_{0k} \tag{55}$$

The interpolation period T_{kn}^i is calculated by solving τ_{kn} from Eq. (55), and then subtracting the incremental time $\tau_{k,\,n-1}$ calculated in the previous step as,

$$T_{kn}^i=\tau_{kn}-\tau_{k,n-1} \tag{56}$$

Although Eq. (55) can be solved analytically, the solution involves the computation of trigonometric functions and cubic roots. Since the cubic equations that are to be solved in successive interpolation cycles have similar coefficients, Newton–Raphson's iterative algorithm is preferred for its efficiency [14]. It is also possible to use Eq. (55) to directly estimate the interpolation period for each cycle, by replacing the initial conditions with the final conditions reached in the previous step. However, doing so results in the accumulation of round-off errors. Since jerk limited trajectory generation requires very accurate computation of the interpolation period, this approach is not suitable.

4. Reconstruction of the reference trajectory at the servo control loop frequency

Reference trajectories generated with varying interpolation period need to be reconstructed at the control loop closure period in order to realize smooth feedrate and acceleration profiles received by each digital axis controller. This reconstruction technique is illustrated in Fig. 1(c) for a single axis. In this work, fifth degree polynomials are used for this purpose, which are fit in realtime with the boundary conditions of position, velocity, and acceleration continuity. Furthermore, the first and second derivative estimates are obtained using third degree polynomials, to smoothen out undesirable jerks at the beginning of acceleration and end of deceleration stages.

The scheme for estimating first and second derivatives is similar to the one presented in Section 2.1. In this case instead of using chord length, time is used as the parameter. Using cubic splines,

instead of parabolas or first order differentiation techniques yields smoother derivative estimates. Since more reference points are taken into account, the spikes that occur in acceleration estimates due to the interpolation period being relatively large at the beginning of acceleration and end of deceleration stages are filtered out.

Once the time derivative estimates are obtained for consecutive reference points $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$, a fifth degree polynomial is fit between them in the form,

$$\tilde{\boldsymbol{x}}(\tau)=\boldsymbol{A}\tau^5+\boldsymbol{B}\tau^4+\boldsymbol{C}\tau^3+\boldsymbol{D}\tau^2+\boldsymbol{E}\tau+\boldsymbol{F}. \tag{57}$$

The vector coefficients $\boldsymbol{A}$, $\boldsymbol{B}$,..., $\boldsymbol{F}$ have the same number of dimensions as the number of axes the trajectory is being generated for, and are calculated using the boundary conditions,

$$\left.\begin{aligned} &\tilde{\boldsymbol{x}}(0)=\boldsymbol{x}_i, && \tilde{\boldsymbol{x}}(T^i_{i+1})=\boldsymbol{x}_{i+1}\\ &d\tilde{\boldsymbol{x}}(0)/d\tau=\hat{\dot{\boldsymbol{x}}}_i, && d\tilde{\boldsymbol{x}}(T^i_{i+1})/d\tau=\hat{\dot{\boldsymbol{x}}}_{i+1}\\ &d^2\tilde{\boldsymbol{x}}(0)/d\tau^2=\hat{\ddot{\boldsymbol{x}}}_i, && d^2\tilde{\boldsymbol{x}}(T^i_{i+1})/d\tau^2=\hat{\ddot{\boldsymbol{x}}}_{i+1} \end{aligned}\right\} \tag{58}$$

where T^i_{i+1} is the interpolation period corresponding to the reference point $\boldsymbol{x}_{i+1}$ (i.e. $T^i_{i+1}=t_{i+1}-t_i$) and $0\le\tau\le T^i_{i+1}$. $\hat{\dot{\boldsymbol{x}}}_i$, $\hat{\ddot{\boldsymbol{x}}}_i$, $\hat{\dot{\boldsymbol{x}}}_{i+1}$, and $\hat{\ddot{\boldsymbol{x}}}_{i+1}$ are the velocity and acceleration estimates for points $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$, respectively. The solution for the polynomial coefficients has already been presented in Section 2.2.

For each new reference point coming out of the interpolator, the polynomial coefficients are recalculated, and the re-sampled reference points at the control loop closure period T_s are generated by interpolating along the polynomial. This technique is implemented in a recursive manner in real-time, making use of derivative estimates available from previous the step. Hence, position, velocity, and acceleration continuity are achieved for all reference trajectories, while undesirable jerks due to having to accelerate and decelerate within a finite number of steps are smoothened out.

5. Simulation and experiment results

In this section, the advantage of using a jerk limited feedrate profile, over trapezoidal velocity profiling is demonstrated. This is followed by a quintic spline interpolation example which exploys trapezoidal acceleration profiling along the toolpath. As a practical application, a sculptured wing surface is machined on a three axis machining center using the interpolation technique presented in this paper.

Trapezoidal acceleration profiled trajectories have smoother feedrate, acceleration and jerk characteristics compared to trapezoidal velocity profiled trajectories, which are commonly used in industrial motion control systems. This can be seen in Fig. 5, where two reference trajectories with the same travel distance, feedrate, and acceleration values are compared. For both cases, a linear displacement of 10 mm in one axis is commanded with a feedrate of 100 mm/s and an acceleration/deceleration value of 3000 mm/s^2. For the trapezoidal acceleration case, a jerk value of 100 000 mm/s^3 was also imposed. Both trajectories were re-sampled with fifth order poly-

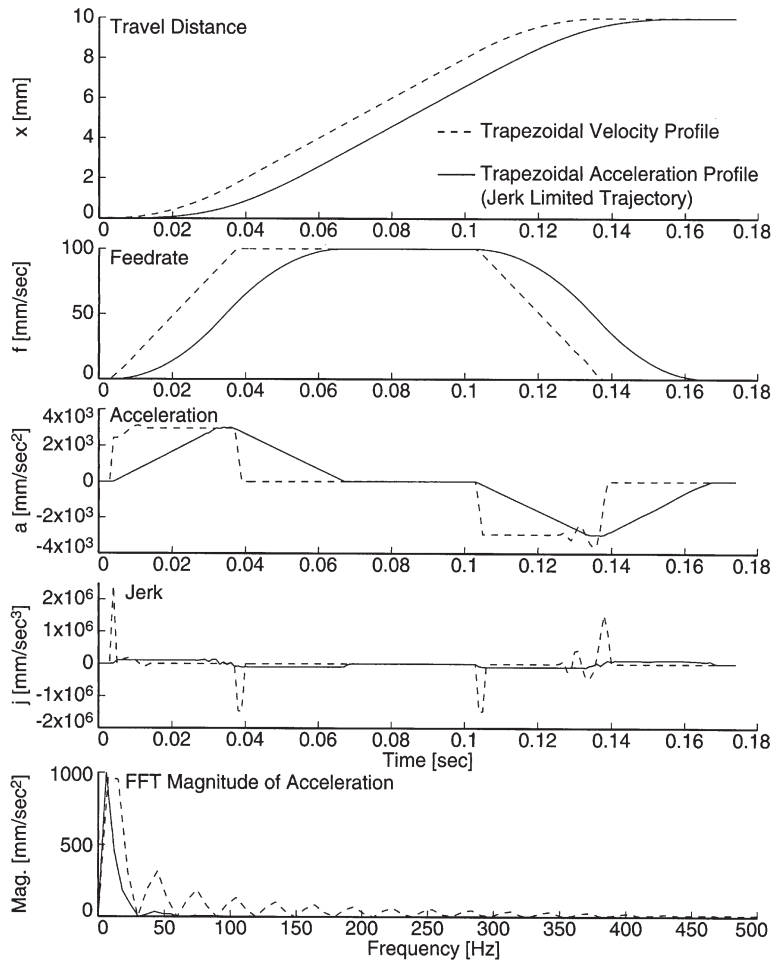

Fig. 5. Comparison of trapezoidal feedrate and acceleration profiled trajectories.

nomials. As can be seen, the acceleration and deceleration profiles for the jerk limited trajectory are smoother. The control signals resulting from the utilization of such a reference trajectory will also be smoother, hence reducing the risk of overheating the drives, or exciting the machine tool's structural dynamics. This can be seen by comparing the frequency content of the acceleration profiles for the two trajectories, which directly relate to motor current and therefore. actuation torque delivered by the drives to the mechanical structure. The trapezoidal velocity profile produces high frequency harmonics in acceleration, whereas the jerk limited trajectory's signal content is mainly in the low frequency range, making this trajectory also easier to track by the limited bandwidth of the servo controller.

In Fig. 6, an example of quintic spline interpolation with trapezoidal acceleration profiling is presented, where a wing profile made up of two composite splines has been generated. The reference knots and the spline toolpath are shown in Fig. 6(a). After imposing a trapezoidal acceleration profile to the spline toolpath, for a maximum feedrate of 100 mm/s with tangential accelerations

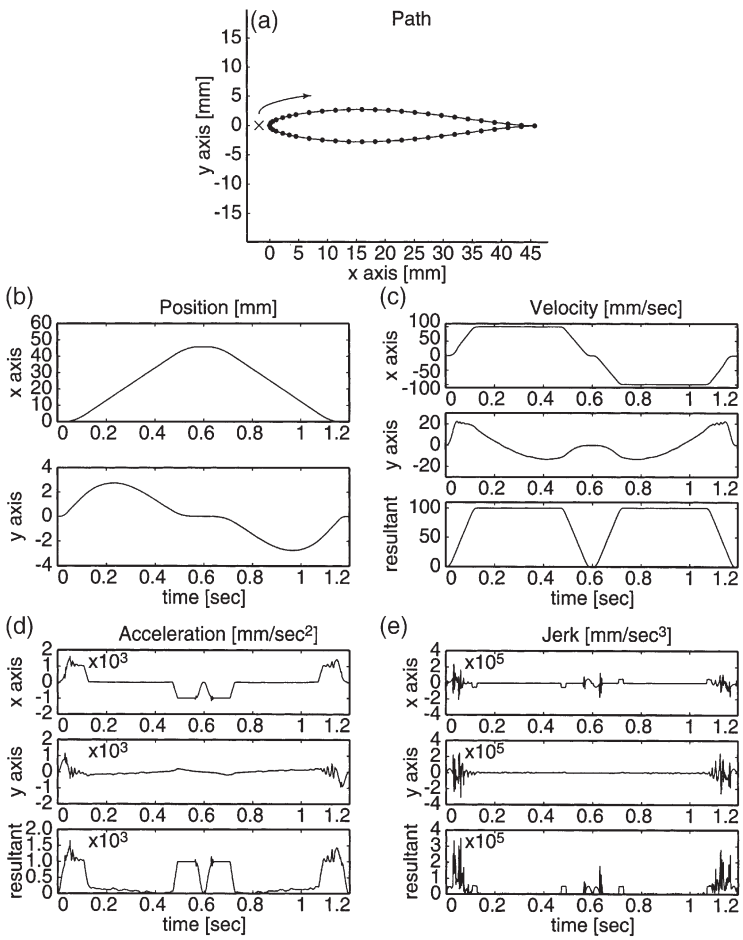

Fig. 6. Wing profile generated with quintic spline interpolation and trapezoidal acceleration profiling along the toolpath.

and decelerations of 1000 mm/s^2 and tangential jerk limit of 50 000 mm/s^3, the obtained position, velocity, acceleration and jerk profiles are shown in Fig. 6(b)–(e), respectively. It can be seen that the resultant feedrate profile in Fig. 6(c) is quite smooth, without any fluctuations during the constant feed motion. The acceleration and jerk fluctuations, which exceed the specified limits as seen in Fig. 6(d) and (e) occur due to curvatures in the path. These fluctuations cannot be avoided if the desired feedrate profile is to be imposed on top of the fit spline geometry. In this work, limits only on the first and second time derivatives of feedrate were considered. However, the problem of handling geometric effects in smooth trajectory generation have already been well covered in literature [1,8].

The top half of the wing profile shown in Fig. 6(a) was machined on a three axis machining center controlled with an in house developed open architecture CNC [15], that runs on a TMSC32 based DSP board. The CAD model of the wing surface and the generated toolpath are shown in

Fig. 7(a). The machined part is shown in Fig. 7(b). The toolpath was generated with tool offset compensation for a helical ball endmill with a tip radius of 2.7 mm. During the actual machining, the feed increment in the x axis was 1 mm, instead of 12.5 mm which was the illustration increment. The workpiece was machined from Medium Density Fiberboard (MDF) with a five fluted helical ball endmill at a spindle speed of 1500 rpm. The tool trajectory was generated at a feedrate of 40 mm/s, with acceleration/deceleration and jerk limits of 50 mm/s^2 and 1000 mm/s^3, respectively. The machined workpiece demonstrates the feasibility of the interpolation scheme proposed in this work. The relatively low feedrate and acceleration values were used due to the limitations of the machine tool drives and structure.

6. Conclusions

Using high performance drives has become a necessity for high speed machine tools. This in return requires smooth trajectory generation algorithms capable of providing continuous position, velocity, and acceleration profiles. This paper has presented such a trajectory generation algorithm

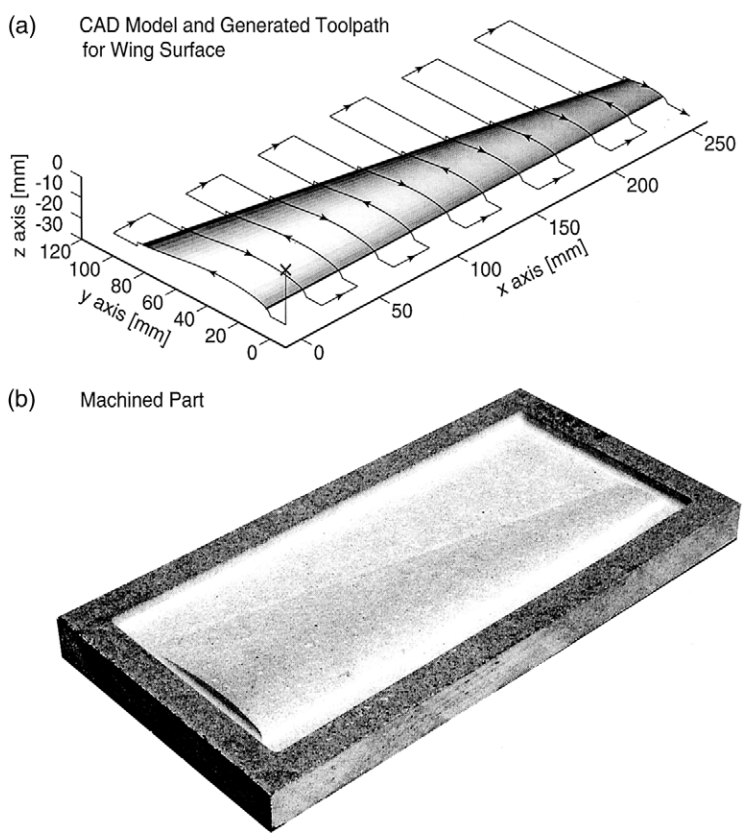

Fig. 7. Wing surface machined using quintic spline interpolation with jerk limited feedrate generation: (a) CAD model and generated toolpath, (b) machined workpiece.

which employs a quintic spline interpolation technique, as well as trapezoidal acceleration profiling. The feasibility of the approach was demonstrated by machining a free form surface on a three axis milling machine controlled with an in house developed open architecture CNC. During the implementation, it was seen that although some initialization routines required extensive computation, the core algorithms could easily be implemented in real-time on a TMSC32 DSP board, which was also used to accommodate the axis controllers.

References

[1] J. Butler, B. Haack, M. Tomizuka, Reference generation for high speed coordinated motion of a two axis system, Symposium on Robotics, Winter Annual Meeting of the American Society of Mechanical Engineers—1988, Chicago, IL, DSC 11, 1988, pp. 457–470.
[2] M. Weck, G. Ye, Sharp corner tracking using the IKF control strategy, Annals of CIRP 39 (1) (1990) 437–441.
[3] M. Tomizuka, Zero phase error tracking algorithm for digital control, Journal of Dynamic Systems, Measurement, and Control 109 (1987) 65–68.
[4] E.D. Tung, M. Tomizuka, Feedforward tracking controller design based on the identification of low frequency dynamics, Journal of Dynamic Systems, Measurement and Control 115 (3) (1993) 348–356.
[5] G. Pritschow, Course notes: Steuerungtechnik der Werkzeugmaschinen und Industrieroboter (control techniques of machine tools and industrial robots), Institute of Control Technology for Machine Tools and Manufacturing Units, Stuttgart University, Germany, 1997.
[6] F.-C. Wang, D.C.H. Yang, Nearly arc-length parameterized quintic-spline interpolation for precision machining, Computer Aided Design 25 (5) (1993) 281–288.
[7] F.-C. Wang, P.K. Wright, Open architecture controllers for machine tools Part 2. A real time quintic spline interpolator, Journal of Manufacturing Science and Engineering 120 (2) (1998) 425–432.
[8] M. Weck, A. Meylahn, C. Hardebusch, Innovative algorithms for spline-based CNC controller, Production Engineering Research and Development in Germany; Annals of the German Academic Society for Production Engineering VI (1) (1999) 83–86.
[9] H. Makino, T. Ohde, Motion control of the direct drive actuator, Annals of CIRP 40 (1) (1991) 375–378.
[10] Y. Tomita, K. Makino, M. Sugimine, N. Taniguchi, High-response *X–Y* stage system driven by in-parallel linear motors, Annals of CIRP 45 (1) (1996) 359–362.
[11] D. Simon, C. Isik, Optimal trigonometric robot joint trajectories, Robotica 9 (4) (1991) 379–386.
[12] Y. Altintas, Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design, Cambridge University Press, Cambridge, 2000.
[13] Y. Koren, Computer Control of Manufacturing Systems, McGraw-Hill, New York, 1983.
[14] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes in C, Cambridge University Press, New York, 1988.
[15] N.A. Erol, Y. Altintas, Open architecture modular tool kit for motion and process control, Proceedings of the ASME Manufacturing Engineering Division, 1997 ASME International Mechanical Engineering Congress and Exposition, MED 6 (1) (1997) 15–22.