Real-Time Optimum Feedrate Planning for Coplanar NURBS Curves Considering Motion Dynamics

Chin Sheng Chen[1], Cheng Yin Tsai[2] and Cheng Hsien Lai[3]

[1]National Taipei University of Technology, saint@ntut.edu.tw
[2]National Taipei University of Technology, t98618016@ntut.edu.tw
[3]National Taipei University of Technology, t97618001@ntut.edu.tw

ABSTRACT

This paper proposes a real-time optimum feedrate planning for NURBS (Non-Uniform Rational B-Spline) motion trajectory considering motion dynamics including maximum feedrate and the limitations of acceleration and jerk. First, the estimated feedrate, which satisfies the chord accuracy and maximum feedrate limitation, can be evaluated by adaptive feedrate. Then, to achieve optimum feedrate planning, the estimated feedrate is fed into our proposed lowest feedrate first priority planning (LFFPP) algorithm, which can adapt any kind of acceleration/deceleration (ACC/DEC) profile. The curvature of an NURBS curve is calculated, and the estimated feedrate corresponding to different curvature and chord error is evaluated by adaptive feedrate. In addition, the break points, which have the local maximum curvatures and where the corresponding estimated feedrate is smaller than the maximum feedrate limitation, are picked up. The NURBS curve is separated into many segments by these break points and the estimated feedrate in these break points will be the initial starting and ending feedrates in these segments. Furthermore,to obtain an optimum feedrate planning, these break points and estimated feedrate will be fed into our proposed LFFPP algorithm, which applies two kinds of S-Curve and S-L-Curve ACC/DEC profiles. The optimum feedrate planning simultaneously satisfies the specifications of chord accuracy, as well as the limitation of maximum velocity, acceleration and jerk in each segment.

Keywords: NURBS motion trajectory, feedrate, adaptive feedrate, break points.
DOI: 10.3722/cadaps.2011.873-887

1 INTRODUCTION

In modern motion control or CNC systems, there is increasing demand for moving continuous complex curve/surfaces designed by CAD systems. In contrast, the traditional motion control or CNC systems generally support the functions of only linear and circular paths. A continuous complex path can be approximated by a huge number of small piecewise linear or circular segments, which are sent to the servo system. However, using these tiny segmented curves will lead to feedrate discontinuities between segments. In contrast, the parametric curves generate a path directly without segmentation curve processing, and the NURBS curve is a parametric form in mathematics that is usually used in motion control systems [1].

The servo command corresponding to an NURBS curve is generated by NURBS interpolation. These interpolations rely mainly on parameter approximation methods using Taylor's expansions for only the desired feedrate. Consequently, feedrate fluctuations associated with the approximation truncation errors occur [2]. Tsai and Cheng [3, 4] proposed a real-time predictor-corrector interpolator to compensate for the feedrate fluctuations. Yeh and Hsu [5] proposed the adaptive feedrate which considers the relationship of chord error and curvature of NURBS curve to obtain the adaptive feedrate. However, the adaptive feedrate with chord error is not designed by motion dynamics, which means that the motion system cannot achieve feedrate and compensate for vibration in the machine. Du [6] proposed an adaptive parametric curve interpolator with a real-time look-ahead function developed for NURBS curves interpolation, which considers the maximum acceleration/deceleration of the machine tool. Sekar et al., [7~10] further considered the jerk limitation in high speed machining to obtain a smooth feedrate profile which can reduce the vibration effect. In practice, the value of normal acceleration is huge in the locations of high curvature, and that will dominate the maximum acceleration limitation. But these previous studies did not consider this critical factor. Our proposed algorithm, as noted below, will resolve this problem.

This paper proposes real-time optimum feedrate planning for coplanar NURBS motion trajectory. When compared with the conventional NC code, this new scheme not only describes the desired tool path accuracy, but also has smooth dynamics profiles considering the feedrate, acceleration and jerk of motion characteristics. Since the optimum feedrate planning, the machined surface could be better and the machine life could be higher. In the proposed strategies, the curvature of a NURBS curve is calculated, and the estimated feedrate corresponding to different curvature and chord error is evaluated by adaptive feedrate. In addition, the break points, which have the local maximum curvature and where the corresponding estimated feedrate is smaller than the maximum feedrate limitation, are picked up. The NURBS curve is separated into a number of segments by these break points and the estimated feedrate in these break points is the initial starting and ending feedrate in these segments. Furthermore, the above break points and estimated feedrate are fed into our proposed LFFPP algorithm, which applies two kinds of S-Curve and S-L-Curve ACC/DEC profiles, to obtain an optimum feedrate planning. The optimum feedrate planning simultaneously satisfies the specifications of chord accuracy, as well as the limitations of maximum velocity, acceleration and jerk in each segment.

2 REVIEW OF NURBS CURVE

2.1 NURBS Mathematic Modeling

In general, a 3D parametric curve can be represented as:

$$P(u) = P_x(u)\vec{i} + P_y(u)\vec{j} + P_z(u)\vec{k} \tag{2.1}$$

where u is a normalized free parameter between 0 and 1. The NURBS curve is a non-uniform rational B-spine parametric curve which is easy to program and to present, and so NURBS is commonly used in engineering problems. The NURBS mathematic modeling is given by:

$$P(u) = \frac{\sum_{i=0}^{n} N_{i,k}(u) W_i V_i}{\sum_{i=0}^{n} N_{i,k}(u) W_i} \tag{2.2}$$

where V_i s are the control points, W_i s are the weights, n is the number of control points and k is the order of the NURBS. $N_{i,k}(u)$ is called basis function or blending function of NURBS, which can be defined as:

$$N_{i,1}(u) = \begin{cases} 1 & for \quad u_i \le u \le u_{i+1} \\ 0 & otherwise \end{cases} \tag{2.3}$$

$$N_{i,k}(u)=\frac{u-u_i}{u_{i+k-1}-u_i}N_{i,k-1}(u)+\frac{u_{i+k}-u}{u_{i+k}-u_{i+1}}N_{i+1,k-1}(u) \tag{2.4}$$

where u_i is the knot of NURBS.

2.2 NURBS Interpolation Review

The purpose of NURBS interpolation is to distribute the NURBS curve into the motion command of each axis. The command considers the sampling time, feedrate and the current geometric status of any parametric curve. In the NURBS parametric curve, the position command is generated by the interpolation, which can satisfy the specification of desired feedrate. Fig. 1. presents the interpolation for the NURBS curve *P(u)* in 3D space. The incremental distance Δs_k from point A to point B should be based on equal sampling period T rather than equal increments of the Δu. According to differential geometry, the desired feedrate along the curve is defined as

$$V(t)=\left\|\frac{dP(u)}{dt}\right\|=\left\|\frac{dP(u)}{du}\right\|\left(\frac{du}{dt}\right) \tag{2.5}$$

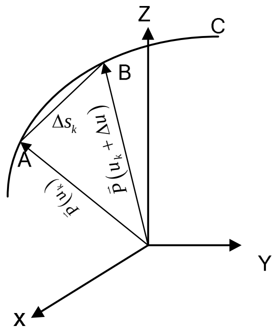


Fig. 1: The interpolation for the NURBS curve *P(u)* in 3D space.

Then, Eqn. (2.5) yields:

$$\frac{du}{dt}=\frac{V(t)}{\left\|\frac{dP(u)}{du}\right\|}=\frac{V(t)}{\sqrt{(x')^2+(y')^2+(z')^2}} \tag{2.6}$$

where

$$x'=\frac{dP_x(u)}{du},\quad y'=\frac{dP_y(u)}{du},\quad z'=\frac{dP_z(u)}{du} \tag{2.7}$$

Furthermore, the 2nd derivative of *u(t)* is obtained as

$$\frac{d^2u}{dt^2}=\frac{A(t)}{\sqrt{(x')^2+(y')^2+(z')^2}}-\frac{V^2(t)(x'x''+y'y''+z'z'')}{\left((x')^2+(y')^2+(z')^2\right)^2} \tag{2.8}$$

Eqn. (2.6) describes the relationship between the geometric properties of the NURBS curve and the motion properties. There are some existing methods that can be applied to solve it, and this paper uses a well-known method based on the 2nd order Taylor's expansion. The 2nd order approximation of $\frac{du}{dt}$, at the time instant of $t_k=kT$, is given by:

$$u_{k+1}=u_k+T\frac{du_k}{dt}+\left(\frac{T^2}{2}\right)\frac{d^2u_k}{dt^2} \tag{2.9}$$

Substituting Eqn. (2.6) and Eqn.(2.8) into Eqn. (2.9), the u_{k+1} can be re-written as:

$$u_{k+1} = u_k + \frac{V(t_k)T}{\sqrt{\left((\dot{x}_k)^2 + (\dot{y}_k)^2 + (\dot{z}_k)^2\right)}} + \frac{T^2}{2}\left(\frac{A(t_k)}{\sqrt{(\dot{x}_k)^2 + (\dot{y}_k)^2 + (\dot{z}_k)^2}} - \frac{V^2(t_k)(\dot{x}_k\ddot{x}_k + \dot{y}_k\ddot{y}_k + \dot{z}_k\ddot{z}_k)}{\left((\dot{x}_k)^2 + (\dot{y}_k)^2 + (\dot{z}_k)^2\right)^2}\right) \tag{2.10}$$

3 PROPOSED OPTIMUM FEEDRATE PLANNING OF NURBS CURVE

Fig. 2. shows the flow chart of our proposed optimum feedrate planning. First, the curvature of a NURBS curve is calculated, and the estimated feedrate corresponding to different curvature and chord error is evaluated by adaptive feedrate. Then the break points, which have the local maximum curvature and where the corresponding estimated feedrate is smaller than the maximum feedrate limitation, are picked up. The NURBS curve is separated into many segments by these N_{bp} break points. In addition, the corresponding length in each segment is calculated and the estimated feedrate in these break points will be the initial starting and ending feedrate in these segments. Furthermore, the length and estimated feedrate in the above segments and break points will be fed into our proposed LFFPP algorithm, which applies two kinds of S-Curve and S-L-Curve ACC/DEC profiles, to obtain an optimum feedrate planning. The optimum feedrate planning simultaneously satisfies the specifications of chord accuracy as well as, the limitations of maximum velocity, acceleration and jerk in each segment. The process will be finished when all the segments are planned.

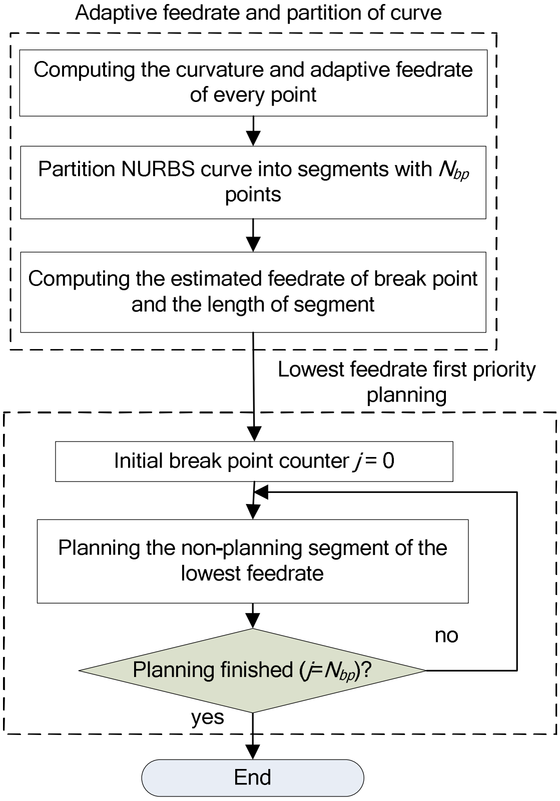
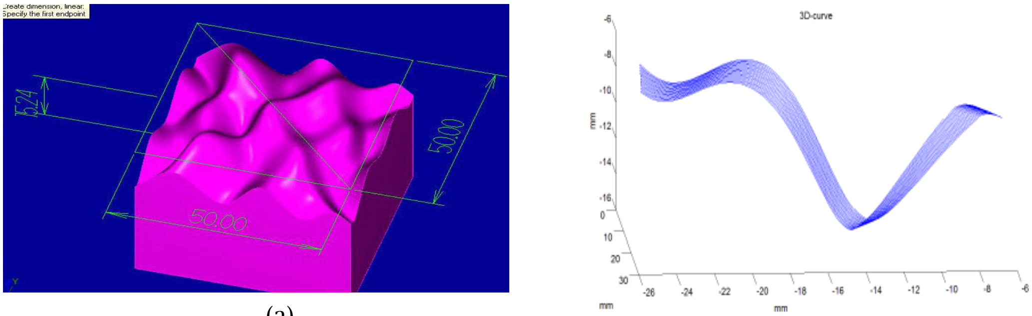


Fig. 2: The flow chart of optimum feedrate planning.

3.1 Adaptive Feedrate and Curve Partition

With the rapid development of computer graphic technology and 3D scanning technology, it is easy to design the freeform surface more complexly, as shown in Fig. 3(a). Many tool path planning assist systems of CAM for freeform surface machining have been presented [11, 12]. In general, the tool path is generated based on the feature of coplanar curves in the CAD model, and some specified tool paths, corresponding to the freeform surface in Fig. 3(a), are described in Fig. 3(b). This paper proposes a real-time optimum feedrate planning for coplanar NURBS curves considering motion dynamics including maximum feedrate and the limitations of acceleration and jerk. With the assisting of this developed system, freeform surface could be machined efficiently and accurately.



(a) (b)

Fig. 3: 3D freeform surface and the corresponding tool path: (a) 3D freeform surface, (b) some specified tool paths.

In real applications, high feedrate will induce high chord errors, as shown in Fig. 4. The interpolated position sequences corresponding to high feedrate and low feedrate are indicated as $\{PC_{ai}\}$ and $\{PC_{bi}\}$. The high feedrate yields chord error ER_1 in the first interpolated slice along from point PC_{a1} to PC_{a2}. In contrast, the low feedrate yields chord error ER_2 in the first interpolated slice along from point PC_{b1} to PC_{b3}. Obviously, the chord error ER_1 , yielded in high feedrate, is largger than ER_2, yielded in low feedrate. Similar effects will occur in different interpolated slices. This implies that whether the curve feedrate must be changed adaptively depends on the curvature during the interpolation process in order to satisfy the limitation of chord error within a tolerance range.

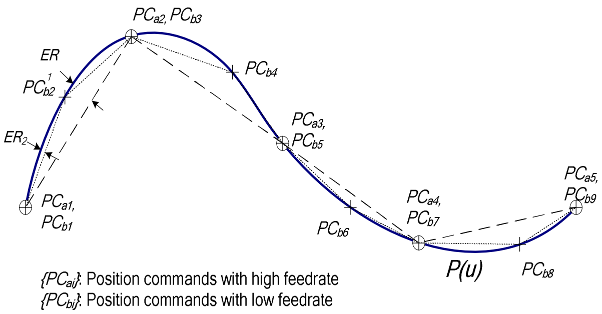


Fig. 4: Chord error comparison with different feedrates.

Yeh and Hsu [5] proposed an adaptive feedrate interpolation for parametric curves, computed based on the tolerance value of chord errors. Since the exact feedrate is difficult to compute by chord error of an NURBS curve, the feedrate here is computed by applying the circular approximation method, as shown in Fig. 5. Considering the interpolated slices both in NURBS and circle curves of $u \in [u_i \quad u_{i+1})$, the radius of curvature of NURBS curve at parameter $u = u_i$ could be approximated as the radius of a circle.

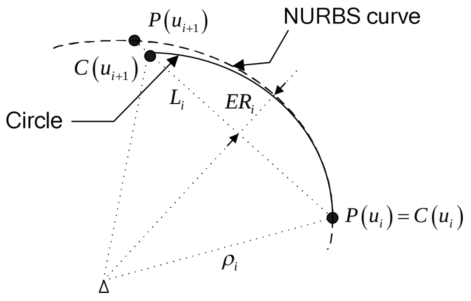


Fig. 5: Chord error by applying circular approximation.

The curvature and the radius of curvature of NURBS curve at parameter $u = u_i$ are given by:

$$k_i(u) = \left. \frac{|P'(u) \times P''(u)|}{|P'(u)|^3} \right|_{u=u_i}, \quad \rho_i = \frac{1}{k_i} \tag{3.1}$$

Therefore, the adaptive feedrate can be obtained according to chord error tolerance ER_i as

$$V(u_i) = \begin{cases} F, & \text{if } \frac{2}{T_s}\sqrt{\rho_i^2 - (\rho_i - ER_i)^2} > F \\ \frac{2}{T_s}\sqrt{\rho_i^2 - (\rho_i - ER_i)^2} & \text{if } \frac{2}{T_s}\sqrt{\rho_i^2 - (\rho_i - ER_i)^2} \le F \end{cases} \tag{3.2}$$

where F is the maximum limit feedrate of the system.

For the NURBS curve partition, the break points that is at the local maximum curvature, and these have the corresponding estimated feedrate smaller than the maximum feedrate limitation are picked up. From Eq. (3.1), the corresponding parameters with local maximum curvature can be obtained by

$$\frac{dk(u)}{du} = 0 \tag{3.3}$$

Substituting the specified parameters, determined by Eq. (3.3), into Eq. (2.2), the locations in NURBS curve with maximum curvature are obtained. If the feedrate of each location is smaller than maximum limit feedrate *F*, the location will be the break point. Here the total number of break points is indicated as N_{bp}, shown in Fig. 2. Then, the NURBS curve is separated into $N_{bp} - 1$ segments by these break points, and the corresponding length *Seg(a,b)* in each segment is calculated for two break points *a* and *b* of the specified segment. In addition the estimated adaptive feedrate in locations *a* and *b* are the initial starting and ending feedrates, respectively. Furthermore, the length and estimated feedrate in the above segments and the corresponding break points are fed into our proposed LFFPP algorithm to finish the optimal feedrate planning considering motion dynamics.

3.2 Lowest Feedrate First Priority Planning

The proposed Lowest Feedrate First Priority Planning contains two parts: 1) the acceleration/deceleration (ACC/DEC) profile planning according to the limitations of acceleration and jerk; and 2) the algorithm to choose the next non-planning break point with the lowest feedrate. In general, the conventional feedrate planning sequentially plans the feedrate of each segment form starting to ending segments. When compared to conventional feedrate planning, the LFFPP can dramatically reduce the re-planning iterations to let the distance of each segment is longer enough to move obeying the limitation of acceleration and jerk.

3.2.1 ACC/DEC Profile Planning according to the Limitations of Acceleration and Jerk

Every machine has its limitations of velocity and acceleration, and a rapid change of acceleration implies a large jerk value, which will induce vibrations of the machine. Hence, jerk limitation will be addressed in acceleration profile planning. In this paper, the ACC/DEC profile will be planed to consider the limitations of acceleration and jerk.

The acceleration has tangent and normal vectors that are given by:

$$\begin{cases} \vec{a}_t = \dfrac{d\vec{V}}{dt} \\ \vec{a}_n = \dfrac{\|\vec{V}\|^2}{\rho}\hat{n} = a_c\hat{n} \end{cases} \tag{3.4}$$

In Eqn. (3.4), $\vec{a}_n$ is decided by the radius of curvature and velocity magnitude (feedrate). Substituting the adaptive feedrate from Eqn. (3.2) into Eqn. (3.4), the magnitude of normal acceleration is given by:

$$a_n = \frac{8ER_i}{T_s^2} - \frac{4ER_i^2}{\rho_i T_s^2} \tag{3.5}$$

where the second term is positive, and it propositional decreases with the radius of curvature. When the radius of curvature approaches to infinite (linear curve), E.q. (3.5) will equal to $8ER_i/T_s^2$. Therefore, in acceleration planning, the maximum normal acceleration is conservational designed as:

$$a_{n-\max} = \frac{8ER_i}{T_s^2} \tag{3.6}$$

where $a_{n-\max}$ is surely larger than the magnitude of normal acceleration $\vec{a}_n$. Assuming $a_{system-\max}$ is the limitation acceleration of end effectors, the maximum value of tangent acceleration $a_{t-\max}$ can be described as:

$$a_{t-\max} = \sqrt{a_{system-\max}^2 - a_{c-\max}^2} \tag{3.7}$$

The S-curve and S-L-curve ACC/DEC profiles can be designed based on the constrains of $a_{t-\max}$ and jerk limitation $J_{\lim}$. Fig. 6(a). is the S-curve ACC/DEC profile planning of feedrate profile, 6(b). is the S-curve ACC/DEC profile planning of acceleration profile, 6(c). is the S-curve ACC/DEC profile planning of the jerk profile.

$$V(t) = V_{st} + C_1 t^2 \ , \ t \le T_s \tag{3.8a}$$

$$A(t) = 2C_1 t \ , \ t \le T_s \tag{3.8b}$$

and part II curve is given by:

$$V(t) = V_{st} + C_2 t - V_s \ , \ T_s \le t \le T_a + T_s \tag{3.9}$$

$$C_2 = \frac{V_{end} - 2V_s - V_{st}}{T_a} \ , \ T_s \le t \le T_a + T_s \tag{3.10}$$

Since the feedrates of part I and part II curves coincide at the time instant T_s , the C_1 can be derived as:

$$C_1 = \frac{V_{end} - 2V_s - V_{st}}{2T_s T_a} \ , \ t \le T_s \tag{3.11}$$

Fig. 7(a). is the S-L-curve ACC/DEC profile planning of feedrate profile, 7(b). is the S-L-curve ACC/DEC profile planning of acceleration profile, 7(c). is the S-L-curve ACC/DEC profile planning of jerk profile. In Fig. 7(a), the S-L-curve contains two second order polynomial curves and a line. The formula of part I curve is same as Eqn. (3.7), and part II and part III curves are given by:

$$V(t) = V_{st} + C_2 t - V_s \ , \ T_s \le t \le T_a + T_s \ , \ C_2 = \frac{V_{end} - 2V_s - V_{st}}{T_a} \ , \ T_s \le t \le T_a + T_s \tag{3.12}$$

$$V(t) = V_{end} - C_1 \cdot (T_a - t)^2 \ , \ T_a + T_s \le t \le T_a + 2T_s \tag{3.13}$$

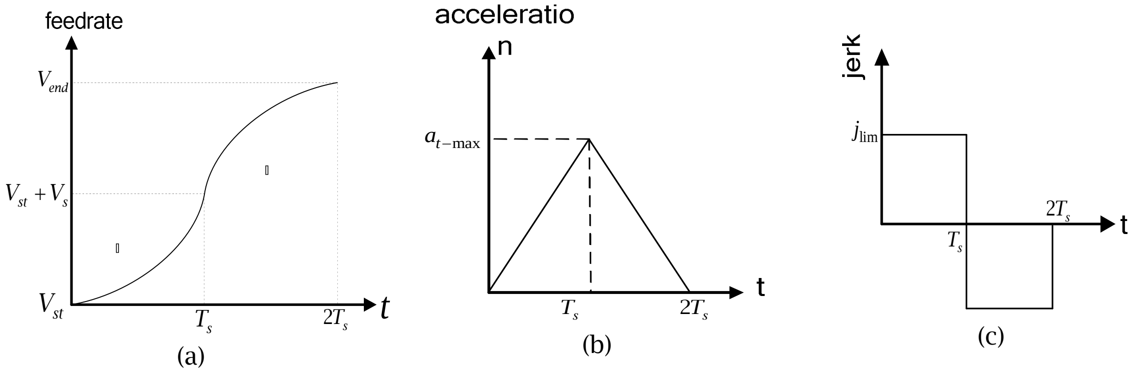


Fig. 6: S-curve ACC/DEC profile planning: (a) feedrate profile, (b) acceleration profile, (c) jerk profile.

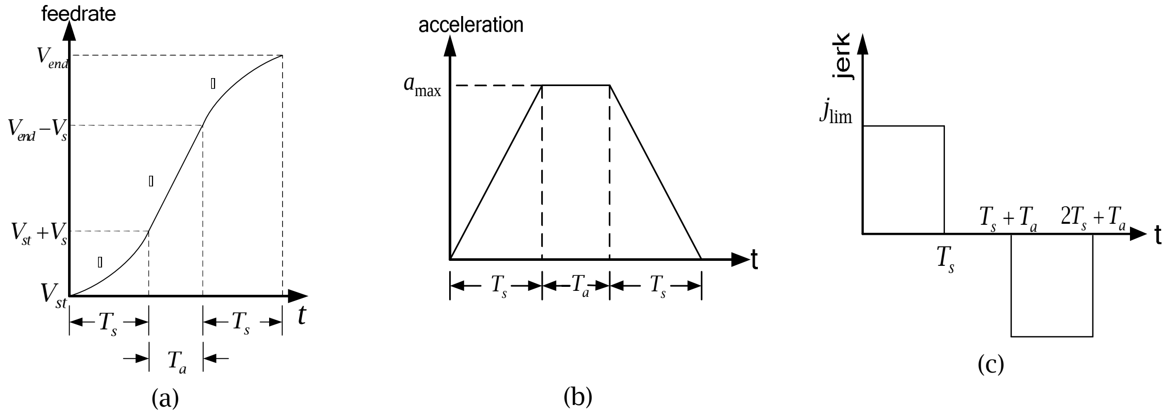


Fig. 7: S-L-curve ACC/DEC profile planning: (a) feedrate profile, (b) acceleration profile, (c) jerk profile.

Fig. 8 shows the flow chart of ACC/DEC feedrate profile planning considering the limitations of acceleration and jerk, where $j_{\lim}$ is the jerk limitation.

- Step 1: Computing the feedrate difference, $V_d = |V_{end} - V_{st}|$, from beginning point to ending point. Then the acceleration time T_s is given by:

$$T_s = \frac{a_{t-\max}}{j_{\lim}} \tag{3.14}$$

Then temporary feedrate increment, half of the feedrate difference in S-curve type, is given by:

$$V_s = \left(\frac{T_s^2}{2}\right) \times j_{\lim} \tag{3.15}$$

- Step 2: If V_d is smaller than $2V_s$, go to Step. 3(a), else branch to Step. 3(b).
- Step 3(a): Estimating the length $L_{est}(V_{st}, V_{end})$ from V_{st} to V_{end} in the S-curve ACC/DEC type.
- Step 3(b): Estimating the length $L_{est}(V_{st}, V_{end})$ from V_{st} to V_{end} in the S-L-curve ACC/DEC type.
- Step 4: If $Seg(P_{st}, P_{end}) > L_{est}$, go to Step 6 to finish the ACC/DEC profile planning, else go to Step 5 to renew the ending feedrate.
- Step 5: Renew the ending V_{end} under the constrain of $a_{t-\max}$ and $J_{\lim}$. Since the length is not long enough to plan the S-L-curve, the length in S-curve ACC/DEC type is planned to equal the curve partition lengh $Seg(P_{st}, P_{end}) > L_{est}$, described as:

$$Seg(P_{st}, P_{end}) = j_{\lim} T_{ns}^{\ 3} + 2V_{st} T_s \tag{3.16}$$

Then the renewed ending feedrate is given by:

$$V_{end} = V_{st} + T_s^2 \times j_{\lim} \tag{3.17}$$

Go to Step 6.

- Step 6: Since the length of the NURBS curve partition $Seg(P_{st}, P_{end})$ mentioned in section 3.1 is longer than the length L_{est} estimated under the constrain of $a_{t-\max}$ and jerk limitation $J_{\lim}$, we can directly plan the ACC/DEC feedrate profile according to V_{st} and V_{end}.

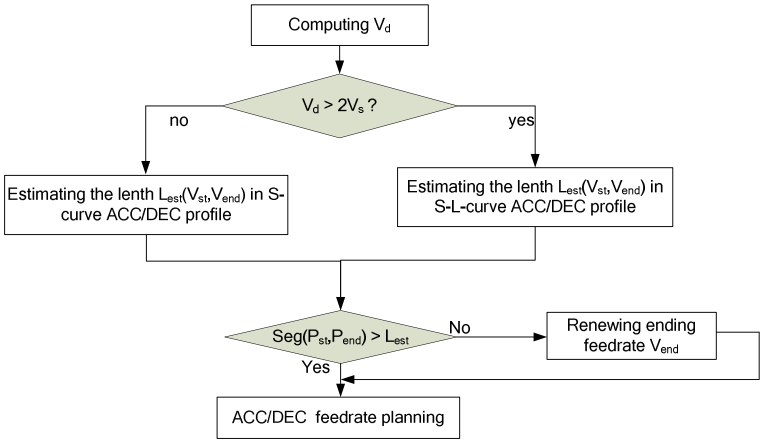


Fig. 8: ACC/DEC feedrate profile planning considering the limitations of acceleration and jerk.

3.2.2 Planning the Next Non-Planning Segment with Lowest Feedrate

For any specified segment from curve partition process is represented as $Seg(P_{st}, P_{end})$, where P_{st} and P_{end} indicate the beginning and ending points, the non-planning break point with lowest feedrate could be point P_{st} or P_{end}. In addition, the feedrate of the next non-planning break point P_i with lowest feedrate is represented as SegFeed[P_i]. Fig. 9 shows the process of planning the next non-planning segment with the lowest feedrate.

- **Step 1:** Find out the break point p_i with lowest feedrate.
- **Step 2:** If SegFeed[p_{i+1}]>SegFeed[p_{i-1}], go to Step3(b). Else, jump to Step3(a).
- **Step 3(a):** If $Seg(p_i, p_{i+1})$ is planned execute Step 4(b), else execute Step 4(a).

- **Step 3(b):** If *Seg(p_{i-1}, p_i)* is planned execute Step 4(a), else execute Step 4(b).
- **Step 4(a):** Set starting feedrate V_{st} =SegFeed[p_i], ending feedrate V_{end} =SegFeed[p_{i+1}].
- **Step 4(b)** Set starting feedrate V_{st} =SegFeed[p_i], ending feedrate V_{end} =SegFeed[p_{i-1}].
- **Step 5:** Plan the ACC/DEC feedrate profile of the above specified segment in step 4(a) or 4(b) considering the limitations of acceleration and jerk, as is described in section 3.2.1 in detail.
- **Step 6:** If V_{end} is renewed, then go to Step 7, else execute Step 9.
- **Step 7(a):** If $Seg(p_{i-2}, p_{i-1})$ is planned execute Step 8(a), else execute Step 9.
- **Step 7(b):** If $Seg(p_{i+1}, p_{i+2})$ is planned execute Step 8(b), else execute Step 9.
- **Step 8(a):** Replan $Seg(p_{i-2}, p_{i-1})$.
- **Step 8(b):** Replan $Seg(p_{i+1}, p_{i+2})$.
- **Step 9:** If *Seg(p_{i-1}, p_i)* and *Seg(p_i, p_{i+1})* are planned execute Step 10, else execute Step 11.
- **Step 10:** Make p_i is been planed, and break point counter *j* increase 1.
- **Step 11:** End.

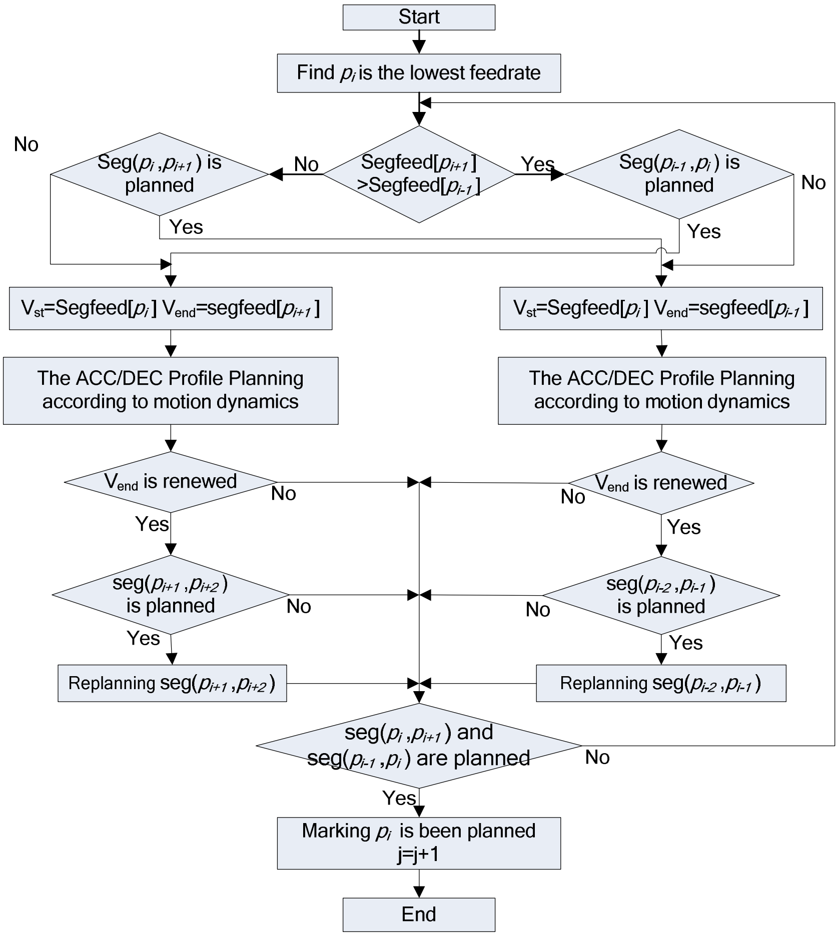


Fig. 9: Planning the next Non-Planning segment with Lowest Feedrate.

4 SIMULATION

One case of 2D face NURBS curve, which contains 9 curve partitions as shown in Fig. 10, is designed to verify the feedrate planning results for our proposed algorithms, with the sampling time of 1 ms. The parameters of the NURBS curve are shown in Tab. 1, and the motion dynamics and chord error specification are shown in Tab. 2. Fig. 11 shows the comparison of adaptive feedrate and our proposed optimal feedrate. Fig. 12(a) and Fig. 12(b) are the chord errors of the adaptive feedrate and of our proposed algorithm, respectively. Both cases fit the chord error tolerance specification shown in Tab. 2. The maximum normal acceleration $a_{n-\max}$ is obtained as $4000\, mm/\sec^2$, from eqn. (3.5), and the maximum value of tangent acceleration $a_{t-\max}$ can be derived as $1280\, mm/\sec^2$ from Eqn. (3.6). Fig. 13(a) shows the acceleration is beyond the specification of $a_{t-\max}$ for adaptive feedrate planning. In contrast, our proposed ACC/DEC planning satisfies the specification of $a_{t-\max}$, as shown in Fig. 13(b). In the other hand, all the jerk values in the ACC/DEC period are fixxed at the specification of 9000(mm/sec^3) for our proposed ACC/DEC planning, as shown in Fig. 14(b). However, Fig. 14(a) shows that some jerk values are beyond the jerk limitation, especially at the locations with large curvature. Fig. 15 presents the conventional feedrate planning, where the parameter u form 0.5 to 0.9. In this case, the length of *Seg(Ai,Bi)* is not enough to move under the limitation of accelerate and jerk. Therefore, the feedrates in locations *Ai* must be reduced to decrease the feedrate deviation between locations *Ai* and *Bi*. Fig. 15(a), (b) and (c) respectively indicate twice, three and four times re-planning. The re-planning amount is the summation of all the above re-planning times that results in 9 times. In addition, the replanned iterations will be reduced from 9 times to needing no replanning by using our proposed LFFPP algorithm. That means our proposed LFFPP can dramatically reduce the number of replanning iterations.

Number of control points	29
Degree of NURBS	2
Knot vector U	{0 , 0 , 0 , 0.083052 , 0.125272 , 0.164699 , 0.217898 , 0.261246 , 0.304345 , 0.352959 , 0.405095 , 0.458334 , 0.509451 , 0.556742 , 0.587608 , 0.610717 , 0.638227 , 0.663886 , 0.683049 , 0.699571 , 0.718443 , 0.738266 , 0.758913 , 0.782515 , 0.805031 , 0.833993 , 0.870327 , 0.905076 , 0.941849 , 1 , 1 , 1 }
Weights	{1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 }
Control points	(0,0) , (2.94502,-0.07689) , (11.97476,-1.39942) , (11.98783,0.38453) , (7.23852,3.49341) , (5.09557,7.93869) , (7.09882,9.92952) , (11.18372,12.11690) , (11.78317,16.49925) , (9.25978,22.08796) , (4.51671,23.13801) , (-0.21847,21.38230) , (-2.07897,18.54258) , (-1.97339,17.69132) , (-2.88781,16.76791) , (-4.35152,15.99191) , (-3.65731,15.08790) , (-3.55308,14.60622) , (-4.14139,14.20849) , (-3.38143,13.49837) , (-4.34644,13.26862) , (-4.05404,12.30382) , (-4.50403,11.26302) , (-4.79670,10.36289) , (-1.99430,9.48877) , (-1.39824,7.30463) , (-1.00626,5.11245) , (0.78235,2.14537) , (0,0)

Tab. 1: NURBS parameters.

Max feedrate (*F*)	50(mm/sec)
Max ACC/DEC ($a_{system-\max}$)	4200(mm/sec^2)
Jerk Limitation ($j_{\lim}$)	9000(mm/sec^3)
Chord error (*ER*)	0.5(μm)

Tab. 2: Specifications of motion dynamics and chord error.

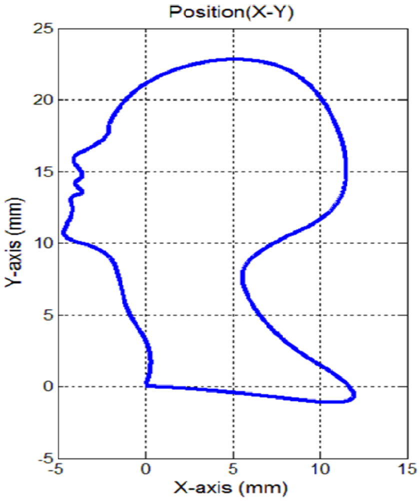


Fig. 10: 2D face NURBS curve.

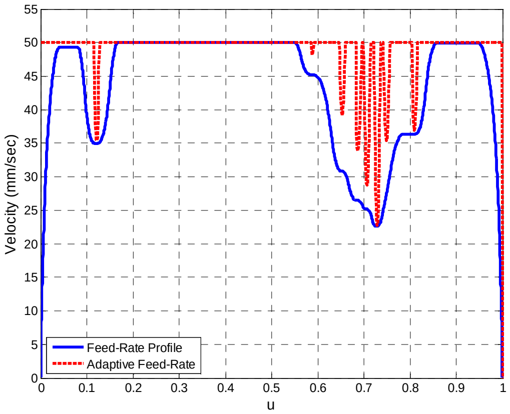
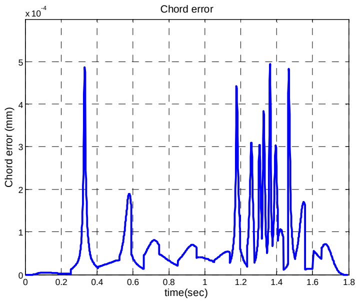
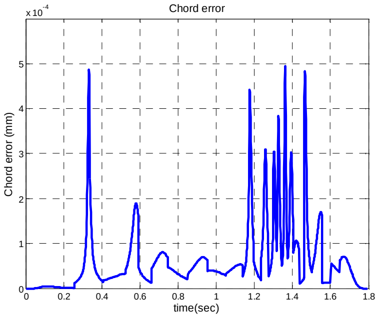


Fig. 11: Comparison of Adaptive Feedrate and our proposed optimal feedrate.

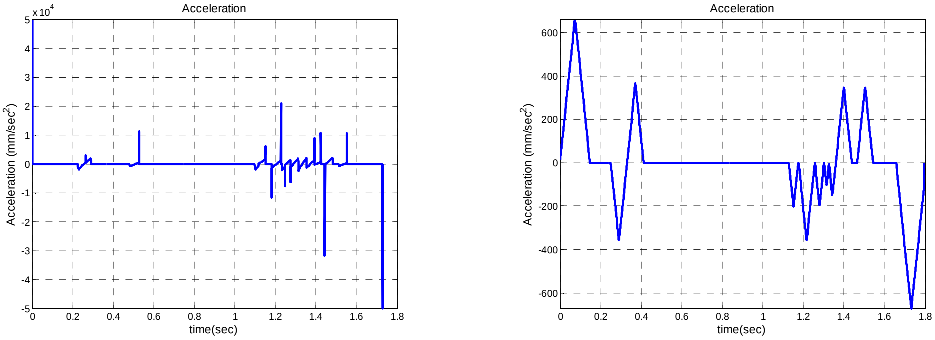


(a)



(b)

Fig. 12: Chord Error: (a) Adaptive Feedrate, (b) Proposed algorithm.



(a) (b)

Fig. 13: Acceleration: (a) Adaptive Feedrate, (b) Proposed algorithm.

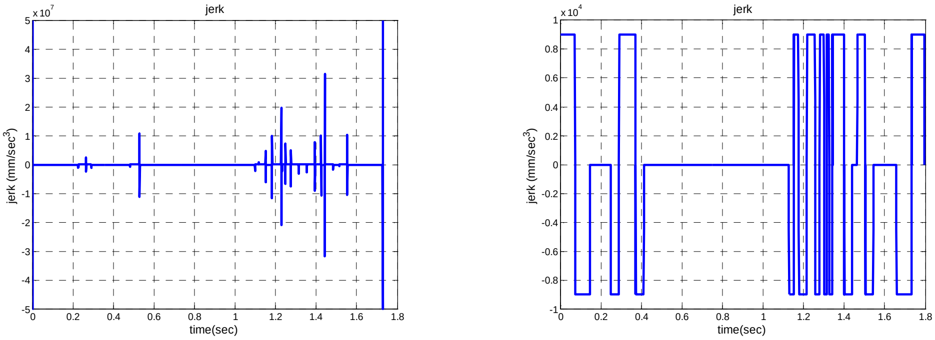
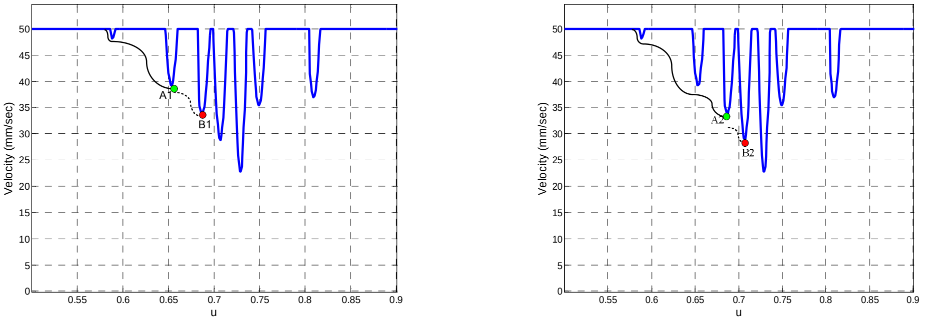
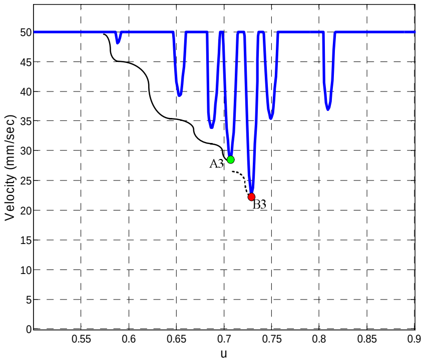


Fig. 14: Jerk: (a) Adaptive Feedrate, (b) Proposed algorithm.



(a) (b)



(C)

Fig. 15: Replanning times: (a) Twice, (b) Three times. (C) Four times.

5 CONCLUSIONS

The proposed optimum feedrate planning simultaneously satisfies the specifications of chord accuracy, as well as the limitation of maximum velocity, acceleration (both normal and tangent accelerations) and jerk in each segment. First, the estimated feedrate, which satisfies the chord accuracy and maximum feedrate limitation, can be evaluated by adaptive feedrate. In addition, to achieve optimum feedrate planning, the estimated feedrate is fed into our proposed lowest feedrate first priority planning (LFFPP) algorithm, which can adapt any kind of acceleration/deceleration (ACC/DEC) profile. The LFFPP can reduce the replanned iterations for a large number of segments that is beyond the machining dynamics. Simulation results show that the S-curve ACC/DEC can simultaneously satisfies the chord error tolerance, as well as the limitation of maximum velocity, acceleration and jerk in each segment.

REFERENCES

[1] Piegl, L.; Tiller, W.: The NURBS Book, 2nd edn., Springer, New York, 1997.

[2] Chou, J. –J.; Yang, D. C. H.: On the generation of coordinated motion of five-axis CNC/CMM machines, Journal of Engineering for Industry, Transactions of the ASME, 114, 1992, 15-22.

[3] Cheng, M.-Y.; Tsai, M.-C.; Kao, J.-C.: Real-time NURBS command generators for CNC servo controllers, International Journal of Machine Tools and Manufacture, 42, 2002, 801-813.

[4] Tsai, M.-C.; Cheng, C.-W.: A Real-Time Predictor-Corrector Interpolator for CNC Machining, Journal of Manufacturing Science and Engineering, 125, 2003, 449-460.

[5] Yeh S., Hsu P.: Adaptive feedrate interpolation for parametric curves with confined chord error. Comput. Aided Des, 34, 2002, 229–237.

[6] Du, D.-S.; Liu, Y.-D.; Yan, C.-L.; Li, C.-G.: An accurate adaptive parametric curve interpolator for NURBS curve interpolation, The International Journal of Advanced Manufacturing Technology, 32, 2007, 999-1008.

[7] Sekar, M.; Narayanan V.-N., Yang, S.-H.: Design of jerk bounded feedrate with ripple effect for adaptive nurbs interpolator, Int. J Adv Manuf Technol, 37, 2008, 545–552.

[8] Dong, J.; Stori, J.-A.: Optimal feed-rate scheduling for high-speed contouring, ASME Journal of Manufacturing Science and Engineering, 29, 2007, 63–67.

[9] Du, D.-S.; Liu, Y.-D.; Guo, X.-G.; Yamazaki Kazuo; Fujishima Makoto: An accurate adaptive NURBS curve interpolator with real-time flexible acceleration/deceleration control, Robotics and Computer-Integrated Manufacturing, 26, 2010, 273-281.

[10] Heng, M.; Erkorkmaz, K.: Design of a NURBS Interpolator with Minimal Feed Fluctuation and Continuous Feed Modulation Capability, International Journal of Machine Tools & Manufacture, 50, 2010, 281–293.
[11] Washeng, Z.; Yang, Y.; Zhenlong, W.; Yong, Z.: A CAD/CAM system for micro-ED-milling of small 3D freeform cavity, Journal of Materials Processing Technology, 149, 2004, 573–578.
[12] Steffen, H.; Claus, R.; Lothar, G; Lars, L.: Constant cusp toolpath generation in configuration space based on offset curves, Int. J Adv Manuf Technol, 53, 2011, 325–338.