Int J Adv Manuf Technol (2012) 58:597–605
DOI 10.1007/s00170-011-3403-6

ORIGINAL ARTICLE

Fast NURBS interpolation based on the biarc guide curve

Jichun Wu · Huicheng Zhou · Xiaoqi Tang · Jihong Chen

Received: 8 July 2010 / Accepted: 23 May 2011 / Published online: 9 June 2011
© Springer-Verlag London Limited 2011

Abstract In parametric spline interpolation, the real-time parameter update is a crucial step which will directly affect the processing performance such as the feed rate fluctuation, the contour error, the online computational effort, etc. The use of Taylor approximation interpolation method to identify the next interpolate point will cause large feed rate fluctuation due to the accumulation error and the truncation error, which will affect the machining quality. As there is no accurate analytic expression between the parameter u and arc length S and the mapping between them is nonlinear, and in order to reduce the feed rate fluctuation and light computation requirement for online interpolation, the paper first samples the tool path with step parameter and Gauss integration, with the sampled points being in the coordinate system defined by parameter u and arc length S. Then, the sampled points are fitted into the guide curve with the use of the biarc fitting method, and the analytic expression between parameter u and arc length S is established. The biarc so derived can be used to realize a fast NURBS interpolation and the simulation results validate the reliability and effectiveness of the proposed method.

Keywords Biarc · Guide curve · Gauss integration · Feed rate fluctuation · NURBS interpolation

1 Introduction

Traditional machining methods can no longer meet the requirements for today's modern manufacture which is expected to provide diversified, flexible, and efficient machining for workpieces in complicated shapes. Therefore, the past several decades have witnessed the rapid development and the wide application of the numerical control (NC) machining technique which can provide a perfect solution for the complex, precise, and different machining requirement parts in a small batch. So this technique radically revolutionized manufacturing engineering. A typical NC machining is implemented in three steps [1, 2], as shown in Fig. 1.

Firstly, the geometrical shape of the part is constructed in a CAD system. The common CAD-integrated software for commercial use includes AutoCAD, CATLA, Pro/ENGINEER, Unigraphics, etc.

Secondly, the NC machining command is achieved by the CAM system, and the tool path for the cutter is generated according to the tool path scheme. Then, the cutter contact points are generated by approximating with small line segments, and the cutter location (CL) points are gained with tool radius offset from CL points. Finally, post-processing converts the CL points into a G code program for NC machining.

Thirdly, by loading the G code program into the machine tool controller (CNC) for execution, the motor is driven by the output order through the servo unit.

The tool path trajectory in the form of small line segments is quite common in surface machining, but some practical problems are related to it:

1. Linearization of curves would result in a large amount of small line segments and approximate errors. The more complicated the surface is, the less approximate errors are allowed. A large amount of information is needed to be dealt with, which is time-consuming and brings troubles of storage and communication to the CAD/CAM system.

J. Wu (✉) · H. Zhou · X. Tang · J. Chen
National NC System Engineering Research Center, Huazhong University of Science and Technology,
Wuhan, China
e-mail: wujichun888@gmail.com

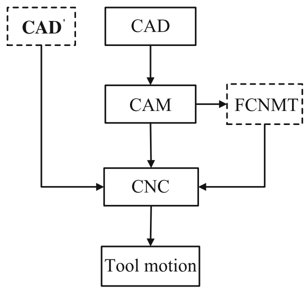


Fig. 1 Flowchart for NC machining

2. The tool path trajectory has discontinuous first and second derivatives on the junction of each G01 segment, which results in a rough tool path. In high-speed machining, the unsmooth tool path trajectory will affect the continuity of the feed rate and acceleration of the cutter and brings about abrupt changes in the motion direction and applied force. Inevitably, torque saturation and excitation of the machine tool's structural modes may occur, which have the effect of degrading the positioning accuracy and part quality and wearing the tool.
3. The frequent program processing in the CNC system will consume a lot of time and makes it hard to realize high speed.
4. When machining the intricate parts, the displacement defined by the NC code segment is so small that the cutter moving at high speed may cause an overshoot error or an undershoot error.

With the shortcomings listed above, the small line segment is not the best choice for tool path trajectory. Some researchers take it into consideration to display modeling of CAD to make service for NC machining when constructing a uniform model. From the perspective of NC machining, researchers propose to directly use the CAD′ model as the tool path trajectory; that is, the curves in CAD′ are suitable for tool paths, such as simple tool offset, the light computational effort, etc., which can directly input the CNC system (CAD′, see Fig. 1). Deng [3] proposed to build the digital model for the free-formed surface which is expressed in data point sets and presented the corresponding modeling method—the self-organizing curve/surface fitting method. By introducing the concept of self-organization into the modeling for surface and NC machining, the modeling and the machining process are unified, with the former serving for the latter. STPE-NC [4] is a programming language based on manufacturing features. It covers the description of all the machining processes for workpieces and all the operations from the original models to the finished products. It provides rich information to the CNC system and allows the two-way exchange of information about the geometric features between CNC and CAX. However, these proposed methods are often designed for machining specific parts rather than for general application. In order to make full use of the existing modeling methods and spline interpolation techniques by which parametric spline interpolation has been proven to be superior over linear and circular interpolation in terms of providing a smoother and a more continuous motion, we generate facing numerical control machining tool path trajectory (see Fig. 1) by fitting a large number of small line segments after post-processing into the spline tool path trajectory. The spline tool path trajectory of the proposed method has the following advantages:

1. As compared with small line segments, the generated spline tool path trajectory ensures machining precision as it closely follows the concaves and convexes of the part, minimizes the contour error, and restores to the greatest extent the shape designed by CAD.
2. The spline interpolation simplifies the G code without the need to transmit data to the CNC at high speed. And the information input into the NC in G code only relates to the control point, node vector, weight factor, process technical information, etc.
3. The generated tool path is smooth and provides a basis for the high-speed and high-precision NC machining, especially the mould processing. The workpiece machined in this way has a smooth surface and high quality.

The continuous feed rate modulation capability is crucial to the spline interpolation. In real-time interpolation, the feed rate may be constrained by a lot of factors, such as the shape of curve/surface, the kinematic characteristics of machine tool, the machining techniques, etc. Therefore, it is important to make the feed rate develop smoothly on condition that the machining precision is guaranteed. In-depth studies have been done into the spline interpolation, but the spline interpolation [5, 6] that has been made possible so far would cause a large feed rate fluctuation due to truncation error. In order to reduce the feed rate fluctuation, the NURBS interpolators are developed by regulating the feed rate using parameter compensatory [7, 8]. A number of parametric interpolators have been developed by taking into consideration the adaptive feed rate, a confined chord error, etc., constraints [9–12]. Yong and Narayanaswami [13] analyzed the feed rate with a full consideration of contour errors and the deceleration/acceleration ability of the machine tool. By utilizing off-line methods to optimize the feed rate-sensitive regions, a feed rate profile is obtained and applied to control the feed rate in real-time interpolation. By considering the kinematic characteristics of the machine tool, Wang and Yau [14] conducted a feed rate planning with CSB criteria and

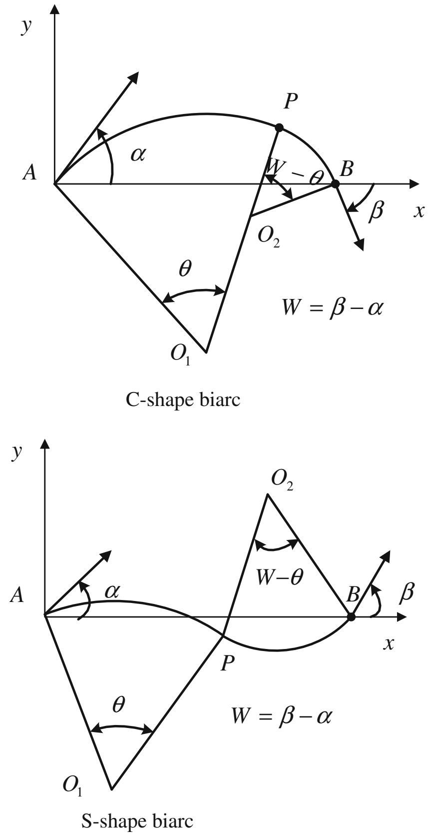


Fig. 2 Biarc curve

generated an optimized tool path trajectory to realize the continuous feed rate. Heng and Erkorkmaz [15] designed an interpolator which can control the feed rate smoothly with only a slight feed rate fluctuation.

The main cause for the unsmooth feed rate is the lack of accurate analytic expression between the parameter u and arc length S, which makes it hard to calculate the next interpolate point, and the lack of information about arc length makes it hard to plan the feed rate or control the motion of the cutter. Based on the above considerations, the sampled points are first gained from the tool path trajectory with the step parameter and Gauss integration. The coordinate of parameter u and arc length S is established by calculating the arc length between two sampled points with the Gauss integration. Then, the sampled points are fitted into the biarc guide curve to direct the real-time fast NURBS interpolation.

2 Biarc guide curve

2.1 The arc length between two consecutive sampled points

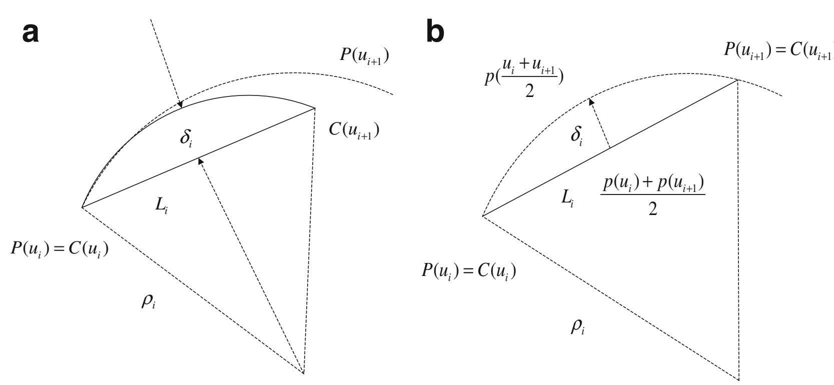
The spline curve is not the arc length parametric curve, so its arc length is hard to be accurately calculated. One common solution to parameterize the spline curve is to approximate the arc length by chord length. In order to calculate the arc length, an adaptive approach is used and the error tolerance is given [16]. This paper uses the Gauss integration to calculate the arc length. The Gauss integration is an efficient calculation method of high precision, computational stability, and convergence, which is often applied in engineering numerical calculations for the integration with hard to get analytic solution. The Gauss integration with N nodes can be expressed as:

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx \sum_{k=0}^{n} A_k f(x_k) \tag{1}$$

More nodes lead to high computation precision, and the value of node x_k and weight factor A_k is a lookup table (refer to [17]). Here, let n=3, $x_1 = -0.7745967$, $x_2 = 0$, $x_3 = 0.7745967$, and correspondingly $A_1 = 0.555556$, $A_2 = 0.888889$, $A_3 = 0.555556$. Let the tool path trajectory be $C(u)(0 \leq u \leq 1)$, and two consecutive points $C(u_i)$ and $C(u_{i+1})$ correspond to the parameter interval $[u_i, u_{i+1}]$. According to the differential geometry, the arc length between two consecutive sampled points can be derived with Gauss integration, which is expressed as follows:

$$l = \int_{u_i}^{u_{i+1}} \left\| C'(u) \right\| \mathrm{d}u \tag{2}$$

Fig. 3 Contour error



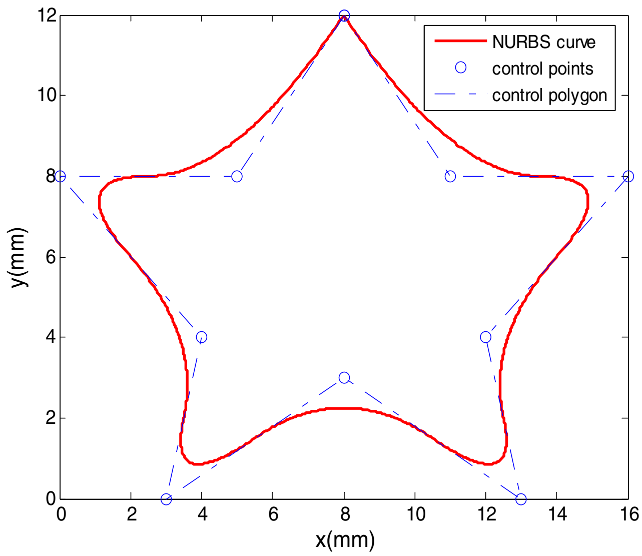


Fig. 4 Asteroid-shaped curve

Let $u = \frac{u_{i+1}-u_i}{2}t + \frac{u_{i+1}+u_i}{2}$ and substitute it into Eq. 2 as follows:

$$l = \frac{u_{i+1}-u_i}{2}\int_{-1}^{1}\left\|C'\left(\frac{u_{i+1}-u_i}{2}t + \frac{u_{i+1}+u_i}{2}\right)\right\|\mathrm{d}t \qquad (3)$$

Let the step parameter $\Delta u = u_{i+1} - u_i$. With the step parameter and Eq. 3, and suppose $m+1$ sampled points, we can calculate the coordinate of points $(u_i, S_i), (i = 0, 1, \ldots, m)$ whose coordinate components are the parameter u and arc length S.

2.2 The fitting of biarc guide curve

After the coordinate of the sampled points (u_i, S_i) on the whole tool path C(u) are listed, the most crucial step is the use of a biarc guide curve to facilitate real-time spline interpolation. The biarc guide curve is actually the result of fitting the sampled points (u_i, S_i) into the curve with biarc. The biarc is characterized by a lot of simple but important features, such as the geometrical invariability, the ease of implementation, and the G1 continuity on the junction points. It is well known that there are two types of biarc models: the C-shaped and the S-shaped biarcs, as shown in Fig. 2. The derivation given here is applicable to both types of biarcs. For ease of presentation, we describe the biarc model using a local coordinate system. In the formulation of the curve-fitting problem, a global coordinate system is needed. The conversion from one coordinate system to another is easily achieved by direct transformation.

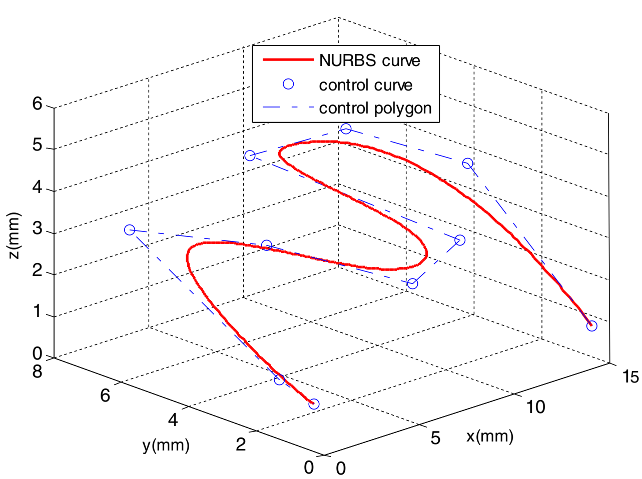


Fig. 5 Spatial curve

Table 1 Configuration of the asteroid-shaped curve

NURBS	
Numbers	11
Order	3
Knot vectors	[0 0 0 0.1110 0.2220 0.3330 0.4440 0.5550 0.6660 0.7770 0.8880 1 1 1]
Control points	(4,6) (2.5,4) (0,4) (2,2) (1.5,0) (4,1.5) (6.5,0) (6,2) (8,4) (5.5,4) (4,6)
Weights	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

When $\alpha\beta < 0$, two circular arcs have similar flexure, constituting a C shape, while when $\alpha\beta > 0$, two circular arcs have opposite flexure, constituting an S shape. In a local coordinate system, the center and radius of biarcs can be expressed as follows [18]:

$$\begin{aligned}
&\text{Radius of the left circle :}\\
&\quad R_1 = \mathrm{L}\sin\left(\frac{W+\theta}{2}+\alpha\right)/\left(2\sin\frac{W}{2}\sin\frac{\theta}{2}\right)\\
&\text{Coordinate of the center of the left circle :}\\
&\quad x_A = -R_1\sin\alpha, y_A = R_1\cos\alpha\\
&\text{Radius of the right circle :}\\
&\quad R_2 = -\mathrm{L}\sin\left(\frac{\theta}{2}+\alpha\right)/\left(2\sin\frac{W}{2}\sin\frac{\mathrm{W}-\theta}{2}\right)\\
&\text{Coordinate of the center of the right circle :}\\
&\quad x_B = L - R_2\sin\beta, y_B = R_2\cos\beta
\end{aligned} \qquad (4)$$

Table 2 Configuration of the spatial curve

NURBS	
Numbers	10
Order	4
Knot vectors	[0 0 0 0 1/7 2/7 3/7 4.2/7 5/7 5.4/7 1 1 1 1]
Control points	(3,2,0.2) (3,3,0.5) (4,8,2.5) (5,4.5,3) (6,0.75,3) (7.5,0.2,4) (5,5,5) (6.5,3,6) (12,2.5,4.5) (15,0.5,0.75)
Weights	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

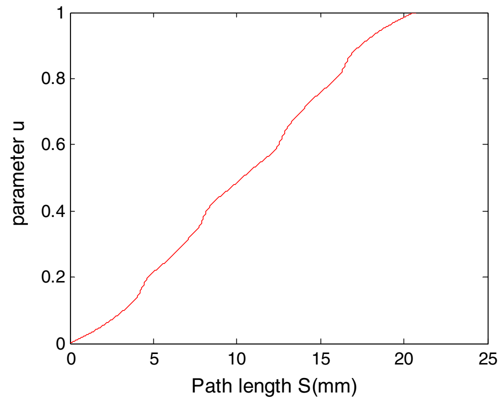


Fig. 6 Guide curve of the asteroid-shaped curve

Coordinate of the common point of tangent:

$$x_P = L\sin(\frac{W+\theta}{2}+\alpha)\cos(\frac{\theta}{2}+\alpha)/\sin\frac{W}{2}$$
$$y_P = L\sin(\frac{W+\theta}{2}+\alpha)\sin(\frac{\theta}{2}+\alpha)/\sin\frac{W}{2} \quad (5)$$

where $W=\beta-\alpha$, θ is the central angle of the left arc, and $W-\theta$ is that of the right arc. Let the counterclockwise direction be positive; then, the positive circle corresponds to the positive central angle and the negative circle to the negative central angle. $-\pi<\theta<\pi$, and when $\alpha\beta<0$, $\theta=-\alpha$; when $\alpha\beta>0$, $\theta=2\alpha-$ W/4. L is the length of segment AB.

Once the biarc guide curve is generated, we can establish the direct correspondence between the parameter u and arc length S, which can improve the efficiency of the real-time calculation for the next interpolate point without the truncation error or accumulation error. In the meantime, the biarc gives a detailed information about the arcs, based on which we can conduct a feed rate planning with taking constraint conditions into consideration, such as the shape of surface/curve, the kinematic characteristics of machine tool, etc. In this way, the feed rate can be controlled accurately and the machining precision can be guaranteed.

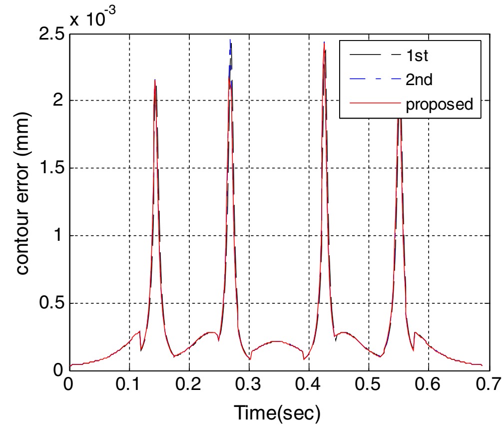


Fig. 8 Contour error (V_{ref}=30 mm/s)

3 The generation of NURBS tool path and the contour error

3.1 Definition of NURBS tool path

Piegl and Tiller [19] are an authority on NURBS modeling. The definition of NURBS tool path is similar to the definition of NURBS curve in the CAD system. Let the knot vector $U=\{u_0,u_1,\ldots,u_{n+p+1}\}$, the weight factor $W=\{w_0,w_1,\ldots,w_n\}$, and the control point $P_i(0\leq i\leq n)$. The tool path for the NURBS of order p is:

$$C(u)=\frac{\sum_{i=0}^{n}N_{i,p}(u)w_iP_i}{\sum_{i=0}^{n}N_{i,p}(u)w_i}=\frac{A(u)}{w(u)}0\leq u\leq 1 \quad (6)$$

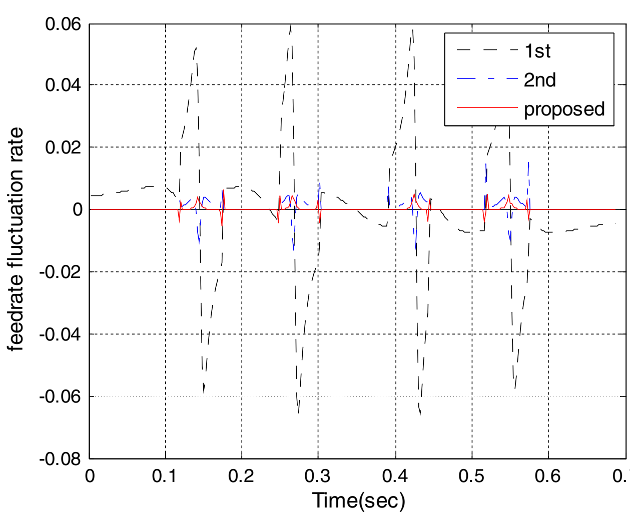


Fig. 7 Feed rate fluctuation (V_{ref}=30 mm/s)

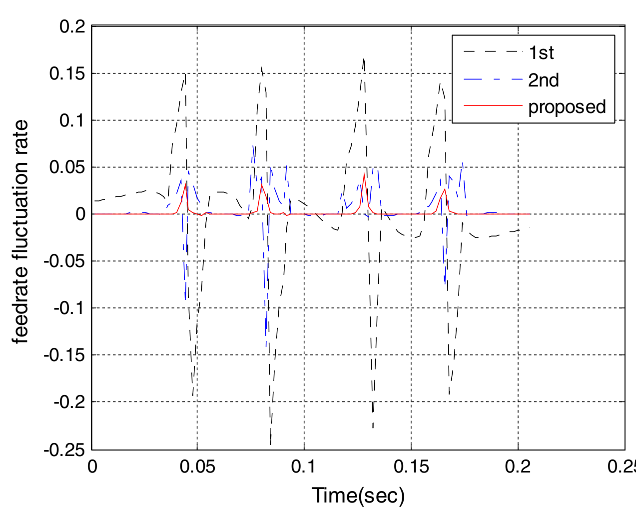


Fig. 9 Feed rate fluctuation (V_{ref}=100 mm/s)

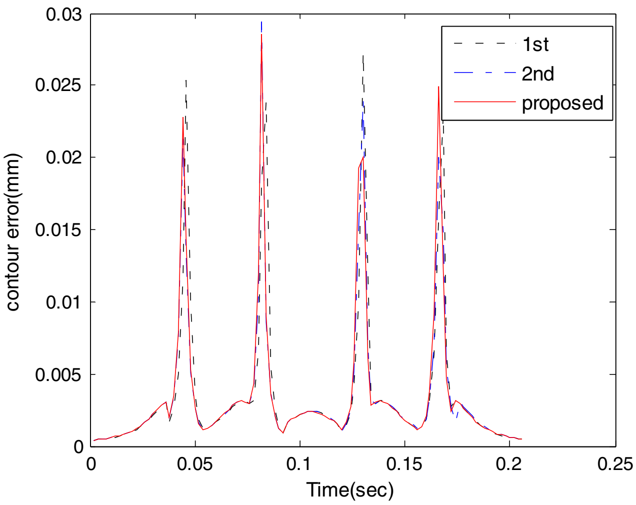


Fig. 10 Contour error (V_{ref}=100 mm/s)

The primary function $N_{i,p}(u)$ can be calculated by the following recurrence relations:

$$N_{i,0} = \begin{cases} 1 & u_i \le u \le u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \tag{7}$$

The first derivative and second derivative of the tool path $C^{'}(u)$ and $C^{'}{}'(u)$ are calculated respectively as:

$$C^{'}(u) = \frac{A^{'}(u) - w^{'}(u)C(u)}{w(u)} \tag{8}$$

$$C^{''}(u) = \frac{A^{''}(u) - 2w^{'}(u)C^{'}(u) - w^{''}(u)C(u)}{w(u)} \tag{9}$$

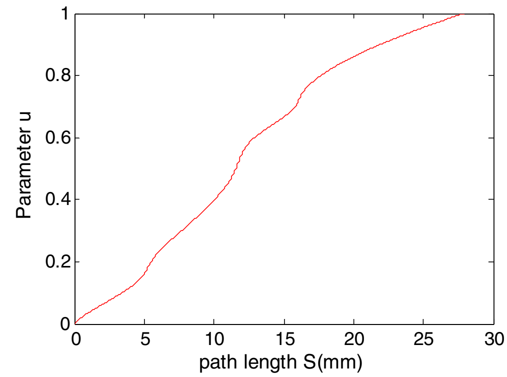


Fig. 11 Guide curve of the spatial curve

Among which, $A^{'}(u), A^{''}(u), w^{'}(u), w^{''}(u)$ can be quickly calculated with the Cox–de Boor algorithm.

3.2 Contour error

The increment of parameter u in each sampling period is calculated in real time, and this parameter value is mapped on the coordinates for the tool path trajectory to make the machining proceed continually. There is no radial error throughout the interpolation process. However, in one sampling period, the use of the small line segment to approximate the interpolate curve would give rise to the contour error, which can be approximately considered as the chord error. In order to ensure the machining precision, the chord error must be controlled under certain tolerance in the machining. As shown in Fig. 3, there are two common approaches to calculate the chord error. The first approach, as shown in Fig. 3a, is to approximate the distance from the midpoint of the curve to the midpoint of the chord line as the chord error [20].

$$\delta_i = \left| p\left(\frac{u_i + u_{i+1}}{2}\right) - \frac{p(u_i) + p(u_{i+1})}{2} \right| \tag{10}$$

Table 3 Simulation results of the asteroid-shaped curve [16]

	Methods	Feed rate fluctuation rate		Contour error (μm)	
		Max.	RMS	Max.	RMS
V_{ref}=30 mm/s	First-order approximation	0.066315	0.022007	2.433003	0.537127
	Second-order approximation	0.015246	0.002480	2.455950	0.534861
	The proposed guide interpolation	0.006279	0.001085	2.425250	0.520550
V_{ref}=100 mm/s	First-order approximation	0.247550	0.072906	27.274738	6.313128
	Second-order approximation	0.142023	0.026364	29.457224	6.008556
	The proposed guide interpolation	0.041551	0.007143	28.578412	6.202869

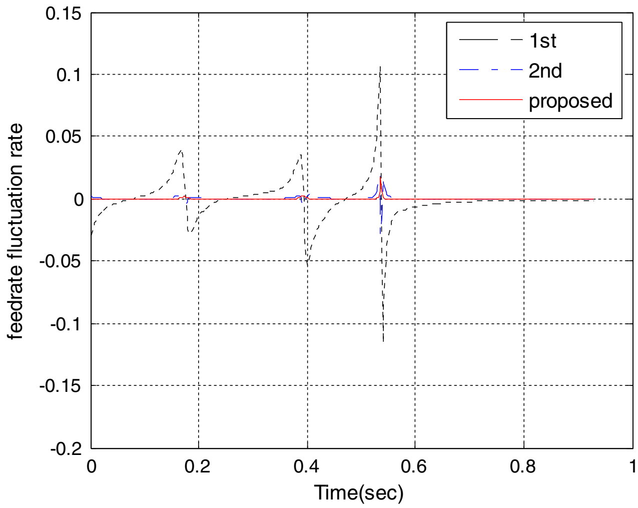


Fig. 12 Feed rate fluctuation (V_{ref}=30 mm/s)

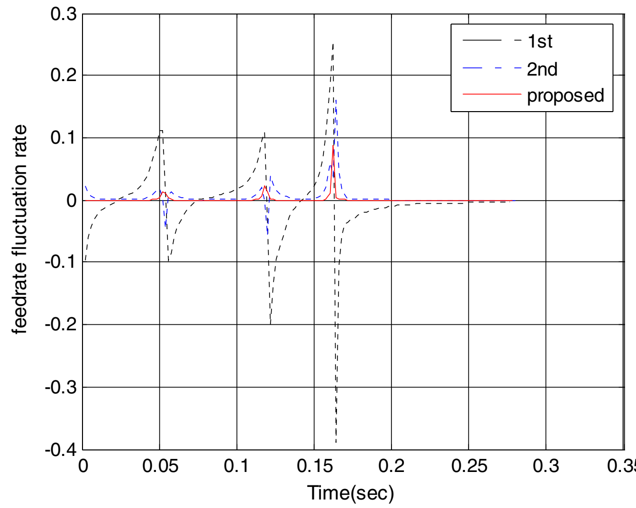


Fig. 14 Feed rate fluctuation (V_{ref}=100 mm/s)

The second approach is to exploit the curvature at the previous interpolate point to calculate the chord error, as shown in the equation below:

$$\delta_i = \rho_i - \sqrt{\rho_i^2 - \left(\frac{L_i}{2}\right)^2} \tag{11}$$

Among which, ρ (millimeters) is the curvature radius at the previous interpolate point, which equals to the reciprocal of the curvature κ, that is, $\rho = 1//\kappa$; L (millimeters) can be approximately seen as a desired interpolate length in one sampling period, the local curvature of which can be derived by the following equation:

$$K_i = \frac{\left\| \frac{\mathrm{d}C(u)}{\mathrm{d}u} \times \frac{\mathrm{d}^2 C(u)}{\mathrm{d}u^2} \right\|}{\left\| \frac{\mathrm{d}C(u)}{\mathrm{d}u} \right\|^3_{u=u_i}} \tag{12}$$

After making a comparison between the aforementioned two approaches, it can be found that the first approach is

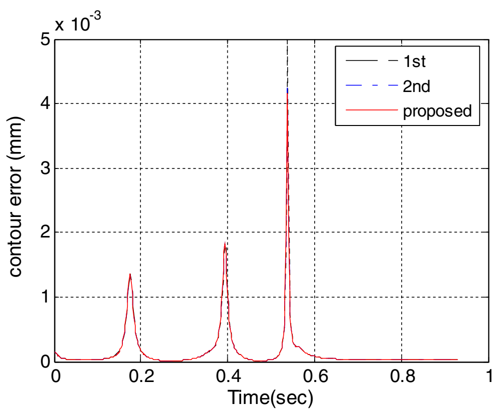


Fig. 13 Contour error (V_{ref}=30 mm/s)

not suitable for the real-time interpolate calculation because it has to calculate the interpolate points and the midpoints of the interpolate points before calculating the chord error, and it has to redetermine the interpolate points if a given chord error is violated. As for the second approach, the calculation for the curvature of the previous interpolate point can be conducted simultaneously with the calculation for interpolate point, and the machining precision can be guaranteed by directly adjusting the feed rate. Therefore, the paper chose the second approach (as shown in Fig. 3b) to calculate the chord error.

4 Simulation

The simulation is conducted in MATLAB7.0 with PC basic frequency of 2.0 GHz. The first-order and second-order

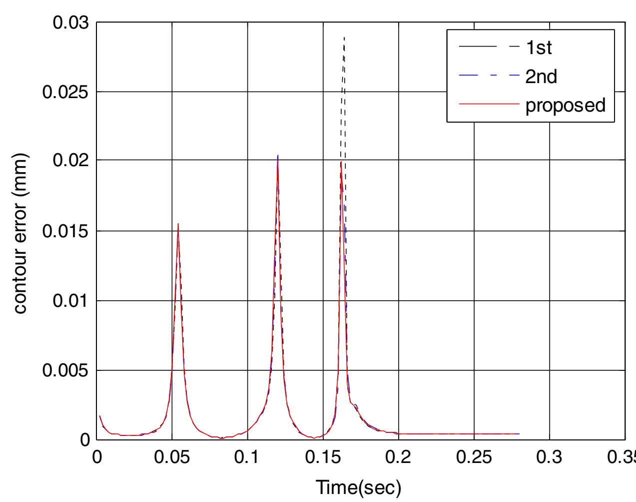


Fig. 15 Contour error (V_{ref}=100 mm/s)

Table 4 Simulation results of the spatial curve

	Methods	Feed rate fluctuation rate		Contour error (μm)	
		Max.	RMS	Max.	RMS
V_{ref}=30 mm/s	First-order approximation	0.115966	0.017070	4.983108	0.426685
	Second-order approximation	0.029663	0.002420	4.322127	0.419637
	The proposed guide interpolation	0.017277	0.000946	4.172228	0.370061
V_{ref}=100 mm/s	First-order approximation	0.389948	0.059666	28.805416	4.475131
	Second-order approximation	0.159905	0.017736	20.819668	3.780699
	The proposed guide interpolation	0.088043	0.008002	20.041845	3.714444

Taylor approximation interpolation methods are used for comparison with the interpolation method based on the biarc guide curve (called hereafter as guide interpolation) generated by a step parameter $\Delta u = 0.005$. The rate of speed fluctuation $\delta_i = \frac{V_s - \|C(u_{i+1}) - C(u_i)\|/T_s}{V_s}$. The NURBS curves [19], both asteroid-shaped curve in Fig. 4 and spatial curve in Fig. 5, are selected as the tool path trajectory, the curve parameters for which are listed in Tables 1 and 2, respectively. To simplify, the desirable feed rates are set as $V_{ref} = 30\text{and}100\text{mm}/s$ and the sampling period $T_s = 0.002\text{s}$. The simulation is divided into three groups: the first-order and second-order Taylor approximation interpolation methods and the proposed guiding interpolation. For simplicity, they are denoted as first, second, and proposed in the picture, respectively.

For the asteroid-shaped NURBS curve, Fig. 6 demonstrates the biarc guide curve and Figs. 7, 8, 9, and 10 give the feed rate fluctuation and the contour error involved in three interpolation approaches, respectively; Table 3 shows the simulation results. As for the spatial curve, the generated biarc guiding curve is shown in Fig. 11; the feed rate fluctuation and the contour errors involved in three interpolation approaches are listed in Figs. 12, 13, 14, and 15, and the simulation results are shown in Table 4.

For each interpolation computation, the first-order Taylor interpolation needs to calculate the first derivative and curve coordinate once, and the second-order Taylor interpolation needs to calculate curve coordinate and the first and second derivatives once; however, the proposed method only needs to calculate the curve coordinate, square root for next parameter.

As seen from the simulation results, compared with the first-order and second-order Taylor expansion methods, the proposed method only has a slight feed rate fluctuation and perfectly complies with the arc as parameter for interpolation. As for the proposed interpolation algorithm, it has established an accurate mapping relationship between the arc and the parameter. When the tool arrives at a region of great curvature, the speed will not fluctuate violently. The first- and second-order Taylor approximation methods, in contrast, would cause a large feed rate fluctuation due to the truncation error, and the contour error brought by the proposed interpolation method is similar to or smaller than that brought by the first- and second-order Taylor approximation methods. What is more is that the guide curve gives detailed information about the arc length, which helps plan the feed rate for the moving cutter and controlling the motion of the cutter.

5 Conclusion

After a brief introduction of the spline interpolation of NC machining, the paper pointed out that the real-time parameter update is a crucial step which will directly affect the processing performance and the weakness of the Taylor approximation interpolation methods, such as the truncation error and the failure to plan the feed rate of the tool path trajectory, because of the lack of arc length information. Based on this, the paper proposed a novel real-time interpolation method based on a biarc guide curve for NURBS tool path trajectory. The working principle of this newly proposed algorithm is as follows: First, the sampled points are achieved by step parameter and Gauss integration from the interpolation curve and then fitting the sampled points into a guide curve by the biarc fitting, which helps calculate the next interpolate point in real time and to plan the feed rate. Simulation is carried out to test the performance of the proposed method, and the results show that the proposed method has high stability and precision, which saves computation effort and is easy to be realized.

Acknowledgment The authors would like to thank the support of National Natural Science Foundation (grant no. 50875099) and the National Important Science and Technology Specialized Fund (grant nos. 2009ZX04009-011 and 2009ZX04009-021).

References

1. Ding G (2008) Research on modeling and interpolation of the toolpath based on cylindrical helix spline. Dissertation, Huazhong University of Science and Technology
2. Li J (2000) Development of a 5-axis CNC milling machine with an open-architecture controller and a real-time NURBS surface interpolator. Dissertation, University of Kansas
3. Deng C (2000) Research on digital modeling and self-organization fitting for freedom surface. Dissertation, Huazhong University of Science and Technology
4. Martin H, David L (2001) STEP into NC. Manuf Eng 6(38):50
5. Bedi S, Ali I, Quan N (1993) Advanced interpolation techniques for NC machines. Trans ASME J Eng Ind 115:329–336
6. Shpitalni M, Koren Y, Lo CC (1994) Real-time curve interpolators. Comput Aided Des 26(11):832–838
7. Yeh S, Hsu P (1999) The speed-controlled interpolator for machining parametric curves. Comput-Aided Des 31:349–357
8. Yan C, Du D, Li C (2007) Design of a real-time adaptive interpolator with parameter compensation. Int J Adv Manuf Technol 35:169–178
9. Zhou K, Wang G, Jin H, Tan Z (2008) NURBS interpolation based on exponential smoothing forecasting. Int J Adv Manuf Technol 39:1190–1196
10. Yeh S, Su H (2009) Implementation of online NURBS curve fitting process on CNC machines. Int J Adv Manuf Technol 40:531–540
11. Xu Z, Cao J (2002) Performance evaluation of a real-time interpolation algorithm for NURBS curves. Int J Adv Manuf Technol 20:270–276
12. Yeh S, Hsu P (2002) Adaptive-feedrate interpolation for parametric curves with a confined chord error. Comput-Aided Des 34(3):229–237
13. Yong T, Narayanaswami R (2003) A parametric interpolator with confined chord errors, acceleration, deceleration for NC machining. Comput Aided Des 35:1249–1259
14. Wang J, Yau H (2009) Real-time NURBS interpolator: application to short linear segments. Int J Adv Manuf Technol 41:1169–1185
15. Heng M, Erkorkmaz K (2009) Design of a NURBS interpolator with minimal feed fluctuation and continuous feed modulation capability. Int J Mach Tools Manuf 50:281–293. doi:10.1016/j.ijmachtools.2009.11.005
16. Jeong S, Choi Y, Park P (2006) Parametric interpolation using sampled data. Comput Aided Des 38:39–47
17. Yu Y (2001) Advanced engineering mathematics. Huazhong University of Technology Press, Wuhan
18. Wang Q (1997) Optimal biarc fitting for plant curve of a set of points. Mini-micro Sys 18(8):38–42
19. Piegl L, Tiller W (2003) The NURBS book, 2nd edn. Springer, Berlin
20. Liua X, Ahmada F, Yamazakia K, Morib M (2005) Adaptive interpolation scheme for NURBS curves with the integration of machining dynamics. Int J Mach Tools Manuf 45:433–444