Conformal Interpolating Algorithm Based on Cubic NURBS in Aspheric Ultra-Precision Machining

C G Li¹, Q R Zhang¹, C G Cao² and S L Zhao²

¹ School of Instrument Science and Opto-electronics Engineering, Beihang University, Beijing 100083, China
² Beijing Machine Tool Institute Precision Mechatronic Co., Ltd., Beijing 100102, China

E-mail: cgli@buaa.edu.cn,

Abstract. Numeric control machining and on-line compensation for aspheric surface are key techniques in ultra-precision machining. In this paper, conformal cubic NURBS interpolating curve is applied to fit the character curve of aspheric surface. Its algorithm and process are also proposed and imitated by Matlab 7.0 software. To evaluate the performance of the conformal cubic NURBS interpolation, we compare it with linear interpolations. The result verifies this method can ensure smoothness of interpolating spline curve and preserve original shape characters. The surface quality interpolated by cubic NURBS is higher than by line. The algorithm is benefit to increasing the surface form precision of workpieces in ultra-precision machining.

1. Introduction
Aspheric surfaces are non-spherical surface having rotation symmetry about the machining optical axis. Traditionally, these surfaces are generated by grinding followed by polishing, single point diamond cutting, wheel transverse grinding, or free-form machining et al [1]. The quality of the surfaces generated relies largely on craftsmanship and equipments, and is affected by the different cutting approach. This results in expensive manufacturing cost and low production yield.

Traditionally ultra-precision CNC interpolations support only linear or circular paths, because of a lower computational requirement which was important in the past when computers were slower. Today, computing speed has increased drastically and research on free-form curve interpolators has become quite active [2-3]. On the other hand, for conventional computer numerical control (CNC) ultra-precision machines to perform complex curve mold machining, the curve mold shapes and desired tool paths designed by the CAD/CAM systems are typically approximated with piecewise line or circular segments.

Among the techniques used for representing and designing specific curves and surfaces, the Non-Uniform Rational B-Spline (NURBS) [2-3] is one that currently attracts a lot of attention. NURBS offers a common mathematical form for representing and designing both standard analytical shapes and free-form curves (surfaces). By manipulating the values of weights, knot vectors and control points, a wide variety of shapes can be designed using NURBS. In addition, since the evaluation of NURBS is reasonably fast and computationally stable, it is not surprising that NURBS has been employed by many CAD/computer aided manufacturing (CAM) systems as a fundamental
geometry representation and has also gained tremendous popularity in the ultra-precision CNC system and computer graphics community.

The objective of this paper is to create an efficient and accurate tool path planning interpolating algorithm for aspheric surfaces in terms of planar cubic NURBS curves. In section 2 introduces some definitions and basic algorithms of cubic NURBS, and describes the procedure to interpolate the tool path by a conformal cubic NURBS curve. Sections 3 provides a simulating example of hyperbola aspheric curve, furthermore compares it with the conventional method. Finally, the conclusions of this study are given.

2. Conformal NURBS Interpolation Algorithm

2.1. NURBS Algorithm

There are two forms of representation for curves: implicit and parametric. Parametric representation is preferred in CAD systems because of its ease of programming and computability. In parametric form, a three-dimensional curve is described as

\[ x = x(u) \quad y = y(u) \quad z = z(u) \]

where \( u \) is an arbitrary parameter, and \( 0 < u < 1 \). The parametric form is very convenient for controlling multiaxis CNC machine tools and robotics, where each axis is driven individually. A Non-Uniform Rational B-Spline Curve, or simply a NURBS curve, is similar to a non-uniform B-spline curve in that it uses the same blending functions, derived from non-uniform knots, as those of non-uniform B-spline curves. NURBS are represented parametrically by the following equation

\[
P(u) = \frac{\sum_{i=0}^{n} N_{i,k}(u) w_i C_i}{\sum_{i=0}^{n} N_{i,k}(u) w_i}
\]

\( C_i \) represents the control point, \( w_i \) is the corresponding weight of \( C_i \), \((n+1)\) is the number of control points and \( k \) is the order of NURBS. In addition, \( N_{i,k}(u) \) is called the blending function of B-spline. Recursive formulas for computing \( N_{i,k}(u) \) can be found from [2,3] as

\[
N_{i,0}(u) = \begin{cases} 1 & \text{for } u_i \leq u \leq u_{i+1} \\ 0 & \text{otherwise} \end{cases}
\]

and

\[
N_{i,k}(u) = \frac{u - u_i}{u_{i+k} - u_i} N_{i,k-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_{i+1}} N_{i+1,k-1}(u)
\]

and

\[
\begin{align*}
0 &= 0 \\
0 &= 0
\end{align*}
\]

where \([u_0, \ldots, u_{i+k}]\) represents the knot vector and \( u \) is the interpolation parameter. \( P(u) \) is the vector to a point defined at some value of \( u \).

2.2. NURBS Interpolator

Many interpolation algorithms consider speed uniformity. In particular, the first-order approximation interpolation algorithm is adequate to generate commands in general motion systems. Suppose \( P(u) \) is the parametric curve function, and the time function \( u \) is the curve parameter \( u(t) = t \) and \( u(t_{i+1}) = t_{i+1} \). However, it is difficult to obtain the \( u(t) \). In practice, alternatively, a numerical solution can be used at each time instant of \( t = iT \). For example, using a Taylor series expansion, the approximation up to the first derivative is [2,3]
\[ u_{i+1} = u_i + \frac{du}{dt} \bigg|_{t_{i+1}} (t_{i+1} - t_i) + \text{H.O.T.} \]  \hspace{1cm} (6)

where \( t_{i+1} - t_i = T \), and \( T \) is the sampling time for the CNC; H.O.T. denotes higher-order terms.

Consider the feedrate \( v(t) \) along the curve \( s \) (namely \( \tilde{P}(u) \)) in a three-dimensional space. By definition, \( v(t) \) can be expressed as

\[ v(t) = \frac{dp(u)}{dt} = \frac{dp(u)}{du} \cdot \frac{du}{dt} \bigg|_{u = u_t} \]  \hspace{1cm} (7)

and hence

\[ \frac{du}{dt} \bigg|_{t = t_i} = \frac{v(t)}{dp(u)/du} \]  \hspace{1cm} (8)

where

\[ \frac{dp(u)}{du} \bigg|_{u = u_t} = \sqrt{(\dot{x})^2 + (\dot{y})^2 + (\dot{z})^2} \]  \hspace{1cm} (9)

Substituting Eq.(9) into Eq.(8) will yield

\[ \frac{du}{dt} \bigg|_{t = t_i} = \frac{v(t)}{\sqrt{(\dot{x})^2 + (\dot{y})^2 + (\dot{z})^2}} \]  \hspace{1cm} (10)

If \( t_{i+1} - t_i = T \) is small enough, the coordinate to be moved to in the space is calculated by

\[ u_{i+1} = u_i + \frac{v(t)}{\sqrt{(\dot{x})^2 + (\dot{y})^2 + (\dot{z})^2}} (t_{i+1} - t_i) \]  \hspace{1cm} (11)

where \( v(u) \) is the cutting feedrate of the CNC.

2.3. Conformal Cubic NURBS Interpolation Algorithm

Ordinarily, on the supposition that a set of initial control points \( d_i (i=0,1, \ldots, n) \) are given in a plane, and the corresponding weight are \( w_i (i=0,1, \ldots, n) \), then one cubic NURBS curve can be drawn in terms of equation (2).

If the control points \( d_i \) are parameterized, we can obtain its vector

\[ U = [u_0, u_1, \ldots, u_{n+4}] \]  \hspace{1cm} (12)

so the cubic NURBS piecewise curve can be expressed as

\[ P_j(u) = \frac{\sum_{i=j}^{j+3} N_{1,3}(u)w_i C_i}{\sum_{i=j}^{j+3} N_{1,3}(u)w_i} \quad u \in [u_{j+3}, u_{j+4}], \quad j = 0,1, \ldots, n-3 \]  \hspace{1cm} (13)

The conformal is defined as the polygon composed by the character points is convex, the interpolating curve is convex also; otherwise there exists only one inflexion in the curve. In particular, the cubic NURBS approximation interpolation algorithm is adequate to generate commands in general CNC systems. As this regard, here a conformal cubic NURBS interpolating algorithm is investigated as follows[4].

For the determinate cutting character points \( v_0, v_1, \ldots, v_n \) on the tool path, some accessoril control points should be interpolated between the adjacent character points. These control points and character
points composed a series of new control points. On the following conditions, the curve formed by the new control points is conformal.

(a) the new cubic NURBS interpolated curve accords with the character points \( v_0, v_1, \ldots, v_n \).

(b) the inflexion number on the cubic NURBS curve are less than on the character polygon.

The character polygon curve is shown in Figure 1, the tangent vectors \( T_i \) is

\[
\begin{align*}
T_0 &= t_0(-a_z) + (1-t_0)a_i \\
T_i &= t_i a_i + (1-t_i)a_{i+1} \quad (i = 1,2,\ldots,n-1) \\
T_n &= t_n a_n + (1-t_n)(-a_{n-1})
\end{align*}
\]

where \( 0 < t_i < 1 \) are the adjusting parameters for the tangent vector, and \( a_i \) (\( i=1,2,\ldots,n \)) are the tangent vector of polygon \( v_0v_1\ldots v_n \), as

\[
a_i = v_i - v_{i-1} \quad (i = 1,2,\ldots,n-1)
\]

the tangent equation at point \( v_i \) is

\[
V_i = v_i + \mu_i T_i \quad -\infty < \mu_i < +\infty
\]

now let

\[
d_{3i} = v_i \quad w_{3i} = w_i \quad (i = 0,1,\ldots,n)
\]

then the data points \( d_{3i}(v_i) \) are parameterized as

\[
\begin{align*}
 u_0 &= u_1 = u_2 = u_3 = 0 \\
 u_{3i} &= \sum_{j=0}^{n} \left| d_{3i} - d_{3(j-1)} \right| \\
 u_{3i+1} &= u_{3i+2} = u_{3i+3} = u_{3i+4} = 1
\end{align*}
\]

two new knot points are interpolated between \( [u_{3i}, u_{3(i+1)}] (i = 0,1,\ldots,n) \) as followed

\[
\begin{align*}
 u_{3i+1} &= \frac{2}{3} u_{3i} + \frac{1}{3} u_{3(i+1)} \\
 u_{3i+2} &= \frac{1}{3} u_{3i} + \frac{2}{3} u_{3(i+1)}
\end{align*}
\]

and the new knot vector is obtained

\[
 U = [u_0, u_1, u_2, u_3, \ldots, u_{3n}, u_{3n+1}, u_{3n+2}, u_{3n+3}, u_{3n+4}]\]

if two accessorial control points are interpolated between \( d_{3i} \) and \( d_{3(i+1)} \) (\( i=0,1,\ldots,n-1 \)), then a new control points group is composed of these accessorial points and all the character points.

Hence the cubic NURBS segmented curve equation is expressed as
and the accessorial interpolated control points are calculated by

\[ d_{3i} = v_i \quad (i = 0, 1, \cdots, n) \] (22)

\[ d_{3i+1} = v_i + \frac{\lambda_i}{w_{3i+1}N_{3i+3}(u_{3i+2})} \quad (0 < \lambda_i < w_{3i+1}N_{3i+1,3}(u_{3i+2})t_i, \quad i = 0, 1, \cdots, n) \] (23)

\[ d_{3i+2} = v_{i+1} + \frac{\lambda_{i+1}}{w_{3i+4}N_{3i+4,3}(u_{3i+5})t_{i+1}} \quad (0 < \lambda_{i+1} < w_{3i+4}N_{3i+4,3}(u_{3i+5})t_{i+1}, \quad i = 0, 1, \cdots, n-1) \] (24)

Thus we obtained a new series of control points \( d_i \) \( (i=0,1,\cdots,3n) \). If the cubic NURBS curve determined by the new control points is fitted the all character points \( v_i(d_i) \) \( (i=0,1,\cdots,n) \), then the interpolating curve is of conformal.

3. Simulation Study

There are two forms of representation for cutting curves: implicit and parametric. Parametric representation is preferred in CAD systems because of its ease of programming and computability. Ordinarily the aspheric surfaces are formed by hyperbola, parabola, ellipse, or helical line et al. In parametric form, a symmetric aspheric surface can be described by its meridian equation. Let \( z \) denote the optical axis, i.e. the symmetric axis of the aspheric surface, then the aspheric equation is

\[ z = \frac{\rho x^2}{1 + \sqrt{1 - (\beta + 1)\rho^2 x^2}} \] (25)

where \( \rho = 1/R \), and \( R \) is the basic circle radius of cutting curve. \( \beta \) is an arbitrary parameter associated with the aspheric curve form. As \( \beta<1 \), the aspheric curve is a hyperbola, and parabola when \( \beta=1 \), and so on.

To evaluate the performance of the conformal cubic NURBS interpolator proposed above, a simulation study was performed using a two-dimensional hyperbola curve as expressed in equation(25). Figure 2 shows the curve interpolated by conformal cubic NURBS algorithm, the cutting depth of tool path is not fixed. The dash line is a cutting tool path interpolated by proposed method in this paper, whereas the solid line is an ideal design tool path.

For comparing the performance of the proposed algorithm with the conventional, the tool path interpolated by line is illustrated in Figure 3. The conventional algorithms interpolated with invariable cutting depth, as shown the solid lines in the Figure. As shown in figures, cutting depth of the proposed conformal cubic NURBS algorithm varies according to the curvature. Therefore, the system
can assure constant surface quality along the surface. However, in the linear interpolation, the cutting depths are constant at any curvature as shown in Figure 3. This makes the machined surface rough.

4. Conclusion
In this paper, we have presented a novel method to generate an ultra-precision CNC tool path for an aspheric surface in terms of conformal cubic NURBS curve. The machinability of the proposed interpolator is imitated. For the purpose of comparison, a linear interpolator interpolator with constant feedrate was used. The self-adaptive cutting depth for the cubic NURBS interpolator improved the smoothness of the curve and confirmed the improvement in surface quality.

Acknowledgements
This research was supported by the Beijing Nature Science Foundation in 2005. (No.3052012)

References
[1] Tae Jo Ko, Hee Sool Kim and Sung Ho Park 2005 Machinability in NURBS interpolator considering constant material removal rate[J] International Journal of Machine Tools & Manufactures 45 665-671
[2] M.-Y. Cheng, M.-C.Tsai and J.-C.Kuo 2002 Real-time NURBS command generator for CNC servo controllers[J] International Journal of Machine Tools & Manufactures 42 801-813
[3] C.Lartigue, F.Thiebaut and T.Maekawa 2001 CNC tool path in term of B-spline curves[J] Computer- Aided Design 33 307-319
[4] Jun Zhao, Junqiang Song and Anxiang Sun 1999 C^2 shape-preserving interpolation curve of cubic NURBS[J] Mini-Micro Computer System 20 25-28