Explicit Free-form Curve Interpolation and Error Analysis for NC Machining of Complex Surface Models

Yongfu Ren and Yuan-Shin Lee

North Carolina State University, yren@ncsu.edu and yslee@ncsu.edu

ABSTRACT

This This paper presents an explicit solution approach to calculate the exact maximal interpolation errors for cubic free-form curves and the offset curves. To solve the maximal interpolation errors, the exact locations of the maximal error points are found by solving polynomial functions explicitly. Compared with conventional approaches, the proposed interpolation method ensures interpolation accuracy and generates fewer interpolation points for free-form curves and their offset curves. The proposed method can be used for high-accuracy curve interpolation and NC tool-path generation in CAD/CAM systems. Computer implementation and practical examples are also presented in this paper.

Keywords: CAD/CAM, Curve interpolation, NC machining, Sculptured surface machining, Offset curves.

1. INTRODUCTION

Modern products usually have complex shapes that need to be machined by CNC machines with highly accurate tool paths. However, current curve interpolation methods of using approximation may not always guarantee tool-path accuracy [3,6]. As shown in Figure 1, a cutter is used to machine a cutter contact (CC) curve $\mathbf{r}(t)$ with a given tolerance τ. One of the traditional methods is to use circular arcs to approximate the local region of the curved CC path [1,2,3,4]. Figure 1 shows the problem of inaccurate interpolation of the offset curve between the current CC-point $\mathbf{P}_i$ and the next CC-point $\mathbf{P}_{i+1}$. Although the estimated error $\varepsilon_s \le \tau$, the actual error is $\varepsilon^o > \tau$, as shown in Figure 1. Therefore, the cutter will cut into the part surface with the actual error ε^0 larger than the given tolerance τ. In this case, the part may not be acceptable to meet the high accuracy requirement in manufacturing.

The problem arises from the incorrect calculation of the interpolation error (also called the chordal deviation error). Traditionally, chordal deviation is estimated either by circular approximation or convex hull properties. In circular approximation approaches, chordal deviation is estimated based on the approximation arc around the interpolation points [3].

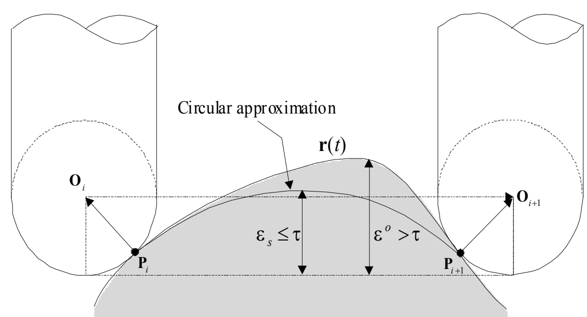


Fig. 1. The tool-path may not meet the machining tolerance

The second approach is the subdivision method based on the convex hull property of free-form curves such as Bezier curves and B-spline curves [7,8,9]. Despite its advantages of stability and efficiency, the subdivision method may encounter difficulties in interpolating offset curves, which are widely used in NC tool-path generation such as profiling and contour machining. The difficulties stem from the fact that the convex hull property may not be valid for the offset curves and it is hard to estimate their chordal deviations.

In high accuracy machining, it is important to ensure that the interpolation error is within a given tolerance [5,6].

In this paper, explicit solutions are proposed to calculate the exact chordal deviations of cubic free-form curves and their offset curves. Instead of being approximated, the choral deviation can be calculated explicitly by solving polynomial functions. In this paper, the most common cubic free-form curves are used for formulation and illustration.

The remainder of this paper is organized as follows. In Section 2, the maximum error conditions of curve interpolation are discussed. Section 3 presents the explicit solutions for evaluating the chordal deviation of non-rational free-form curves, including 2D planar curves and 3D spatial curves. In Section 4, the explicit solution of finding the chordal deviation of offset curves from planar non-rational free-form curves is discussed. Section 5 presents the solutions of computing the chordal deviation of planar NURBS curves and offset curves. Adaptive interpolation method is discussed in Section 6. Practical examples are presented in Section 7, followed by the concluding remarks in Section 8.

2. MAXIMUM ERROR CONDITIONS OF CURVE INTERPOLATION

As shown in Figure 2, an interpolating linear segment $\overline{\mathbf{P}_i\mathbf{P}_{i+1}}$ passes through two points $\mathbf{P}_i = \mathbf{r}(t_i)$ and $\mathbf{P}_{i+1} = \mathbf{r}(t_{i+1})$ on a curve $\mathbf{r}(t)$. For any given point on the curve $\mathbf{r}(t)$, let $\mathbf{Q}(t)$ be the projected point on the linear segment $\overline{\mathbf{P}_i\mathbf{P}_{i+1}}$, i.e., $\overline{\mathbf{Q}(t)\mathbf{r}(t)}$ is perpendicular to $\overline{\mathbf{P}_i\mathbf{P}_{i+1}}$. As shown in Figure 2, $\mathbf{h}(t)$ is the error vector from the point $\mathbf{Q}(t)$ to the correspondent point $\mathbf{r}(t)$ on the curve. Its magnitude $|\mathbf{h}(t)|$ denotes the interpolation error between a curve point $\mathbf{r}(t)$ and the linear segment $\overline{\mathbf{P}_i\mathbf{P}_{i+1}}$, where the parameter t of $\mathbf{r}(t)$ should be within the interval $t_i \le t \le t_{i+1}$. As shown in Figure 2, the error vector $\mathbf{h}(t)$ is defined as follows:

$$\begin{aligned}\mathbf{h}(t) &= \mathbf{r}(t) - \mathbf{Q}(t) \\ &= [\mathbf{r}(t) - \mathbf{P}_i] - \{[\mathbf{r}(t) - \mathbf{P}_i]\cdot\mathbf{E}\}\mathbf{E}\end{aligned} \tag{1}$$

where $\mathbf{E}$ is the unit chord vector defined as $\mathbf{E} = \dfrac{\mathbf{P}_{i+1} - \mathbf{P}_i}{\left|\mathbf{P}_{i+1} - \mathbf{P}_i\right|}$. The maximum curve interpolation error $\left|\mathbf{h}(t)_{\max}\right|$ occurs if the following maximum chordal error conditions are satisfied:

$$\dot{\mathbf{r}}(t)\cdot\mathbf{h}(t) = 0 \quad \text{and} \quad \mathbf{h}(t) \ne 0 \tag{2}$$

Equation (2) means that the maximal curve interpolation error $\left|\mathbf{h}(t)_{\max}\right|$ exists when the tangential vector $\dot{\mathbf{r}}(t)$ and the error vector $\mathbf{h}(t)$ are perpendicular to each other, as shown in Figure 2. Equation (2) also shows that, without losing generality, the non-trivial solution exists when the error $|\mathbf{h}(t)|$ is not zero, i.e., $\mathbf{h}(t) \ne 0$.

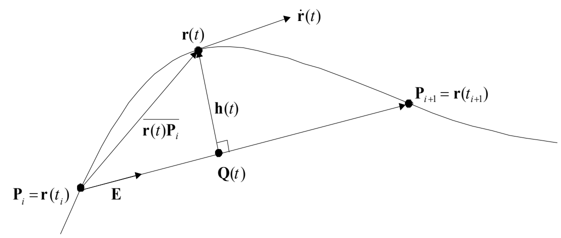


Fig. 2. Maximum error for curve interpolation

3. CHORDAL DEVIATION OF NON-RATIONAL FREE-FORM CURVES

A cubic non-rational free-form curve $\mathbf{r}(t)$ can be represented by a polynomial function as follows:

$$\mathbf{r}(t) = \mathbf{a}_0 + \mathbf{a}_1 t + \mathbf{a}_2 t^2 + \mathbf{a}_3 t^3 \tag{3}$$

where $\mathbf{a}_i, i = 0,1,2,3$, are coefficient vectors. The coefficients $\mathbf{a}_i, i = 0,1,2,3$, can be calculated from the control points of Bezier curves and B-spline curves. Combine Equation (1) and (3),one can get the following:

$$\begin{aligned}\mathbf{h}(t) &= \left[\mathbf{a}_3 - (\mathbf{a}_3\cdot\mathbf{E})\mathbf{E}\right]t^3 \\ &+ \left[\mathbf{a}_2 - (\mathbf{a}_2\cdot\mathbf{E})\mathbf{E}\right]t^2 \\ &+ \left[\mathbf{a}_1 - (\mathbf{a}_1\cdot\mathbf{E})\mathbf{E}\right]t + \left[\mathbf{a}_0 - (\mathbf{a}_0\cdot\mathbf{E})\mathbf{E}\right] \\ &- \left[\mathbf{r}(t_i) - (\mathbf{r}(t_i)\cdot\mathbf{E})\mathbf{E}\right]\end{aligned} \tag{4}$$

Since the interpolation linear segment passes through points $\mathbf{r}(t_i)$ and $\mathbf{r}(t_{i+1})$, $\mathbf{h}(t_i) = \mathbf{h}(t_{i+1}) = 0$. $\mathbf{h}(t)$ can also be represented as follows:

$$\mathbf{h}(t) = (t - t_i)(t - t_{i+1})(\boldsymbol{\beta}_1 t + \boldsymbol{\beta}_0) \tag{5}$$

where $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_0$ are coefficient vectors to be determined. By comparing Equation (4) with Equation

(5), we can find the coefficient vectors $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_0$ shown as follows:

$$\boldsymbol{\beta}_1 = \mathbf{a}_3 - (\mathbf{a}_3 \cdot \mathbf{E})\mathbf{E} \tag{6}$$

$$\begin{aligned}\boldsymbol{\beta}_0 &= (t_i + t_{i+1})[\mathbf{a}_3 - (\mathbf{a}_3 \cdot \mathbf{E})\mathbf{E}] \\ &+ [\mathbf{a}_2 - (\mathbf{a}_2 \cdot \mathbf{E})\mathbf{E}]\end{aligned} \tag{7}$$

Combine Equations (5) and (2)

$$(\boldsymbol{\beta}_1 t + \boldsymbol{\beta}_0) \cdot \dot{\mathbf{r}}(t) = 0 \tag{8}$$

Combining Equations (3) and (8), a three-degree polynomial function is obtained as follows:

$$\begin{aligned}&(3\boldsymbol{\beta}_1 \cdot \mathbf{a}_3)t^3 + (2\boldsymbol{\beta}_1 \cdot \mathbf{a}_2 + 3\boldsymbol{\beta}_0 \cdot \mathbf{a}_3)t^2 \\ &+ (\boldsymbol{\beta}_1 \cdot \mathbf{a}_1 + 2\boldsymbol{\beta}_0 \cdot \mathbf{a}_2)t + \boldsymbol{\beta}_0 \cdot \mathbf{a}_1 = 0\end{aligned} \tag{9}$$

Equation (9) is a key function to find the maximum curve interpolation error $|\mathbf{h}(t)_{\max}|$. Detailed procedures of solving the remaining three roots are described in [11].

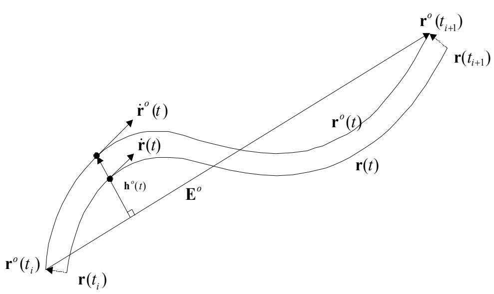


Fig. 3. A planar offset curve

4. CHORDAL DEVIATION OF OFFSET CURVES FROM PLANAR NON-RATIONAL FREE-FORM CURVES

In NC machining, offset curves are frequently used for profiling and contour machining. To ensure machining accuracy, it is important to calculate the chordal deviation of interpolating offset curves. Let $\mathbf{n}$ be the principal normal unit vector of a parametric curve $\mathbf{r}(t)$, respectively. For a 2D planar curve, $\dot{\mathbf{n}}$ can be represented as follows:

$$\dot{\mathbf{n}} = -\frac{\kappa \dot{\mathbf{r}}(t)}{|\dot{\mathbf{r}}(t)|} \tag{10}$$

where κ is the curvature of a point on the curve $\mathbf{r}(t)$. As shown in Figure 3, $\mathbf{r}^o(t)$ is the offset of a planar curve $\mathbf{r}(t)$. The offset curve $\mathbf{r}^o(t)$ can be defined as follows:

$$\mathbf{r}^o(t) = \mathbf{r}(t) + l \cdot \mathbf{n} \tag{11}$$

where l is a constant offset distance. Differentiating Equation (11) and using $\dot{\mathbf{n}}$ of Equation (10), one has the following:

$$\dot{\mathbf{r}}^o(t) = \left(1 - \frac{l\kappa}{|\dot{\mathbf{r}}(t)|}\right)\dot{\mathbf{r}}(t) \tag{12}$$

For an offset curve $\mathbf{r}^o(t)$, Equation (12) indicates that the tangent vector $\dot{\mathbf{r}}^o(t)$ of an offset curve has the same direction with the tangent vector $\dot{\mathbf{r}}(t)$ of the original curve $\mathbf{r}(t)$. As shown in Figure 3, $\overline{\mathbf{r}^o(t_i)\mathbf{r}^o(t_{i+1})}$ is an interpolating linear segment of the offset curve $\mathbf{r}^o(t)$. $\mathbf{E}^o$ is the unit vector of $\overline{\mathbf{r}^o(t_i)\mathbf{r}^o(t_{i+1})}$ and $\mathbf{h}^o(t)$ is the error vector from $\mathbf{r}^o(t)$ to $\overline{\mathbf{r}^o(t_i)\mathbf{r}^o(t_{i+1})}$, as shown in Figure 3. Since the error vector $\mathbf{h}^o(t)$ is perpendicular to the vector $\mathbf{E}^o$, one can find the following relationship:

$$\mathbf{h}^o(t) \cdot \mathbf{E}^o = 0 \tag{13}$$

According to Equation (2), the maximal error $|\mathbf{h}^o(t)_{\max}|$ of the offset curve $\mathbf{r}^o(t)$ exists if the following condition is satisfied:

$$\dot{\mathbf{r}}^o(t) \cdot \mathbf{h}^o(t) = 0 \tag{14}$$

Since $\mathbf{r}(t)$ and $\mathbf{r}^o(t)$ are planar curves, $\mathbf{h}^o(t)$ can be represented as follows:

$$\mathbf{h}^o(t) = \psi(t)\boldsymbol{\Omega}^o \tag{15}$$

where $\psi(t)$ is the magnitude of an error vector $\mathbf{h}^o(t)$, $\mathbf{\Omega}^o = \frac{\mathbf{N}\times\mathbf{E}^o}{\left|\mathbf{N}\times\mathbf{E}^o\right|}$ is the unit error vector perpendicular to the interpolating chord vector $\mathbf{E}^o$, and $\mathbf{N}$ is the unit normal vector of the plane on which the curve $\mathbf{r}(t)$ lies. In Equation (15), $\psi(t)$ has non-zero value at the maximum error point. One can get the following relationship:

$$3(\mathbf{\Omega}^o\cdot\mathbf{a}_3)t^2 + 2(\mathbf{\Omega}^o\cdot\mathbf{a}_2)t + \mathbf{\Omega}^o\cdot\mathbf{a}_1 = 0 \tag{16}$$

Equation (16) has at most two roots and the one with larger error $\left|\mathbf{h}^o(t)\right|$ is selected as the maximal error $\left|\mathbf{h}^o(t)_{\max}\right|$ of the offset curve $\mathbf{r}^o(t)$.

5. CHORDAL DEVIATION OF PLANAR NURBS CURVES

In the previous section, the explicit solution for finding the chordal deviation errors of interpolating non-rational cubic free-from curves is discussed, for example, Bezier curves and B-spline curve. Besides non-rational free-from curves, NURBS curves are also widely used in CAD/CAM systems. This section discusses the chordal deviation for interpolating cubic NURBS curves. A cubic NURBS curve is usually represented as follows [6]:

$$\mathbf{r}(t) = \frac{\sum_{i=0}^{3} N_i^3(t)w_i\mathbf{V}_i}{\sum_{i=0}^{3} N_i^3(t)w_i} \tag{17}$$

where $\mathbf{V}_i$ are control points; w_i are the correspondent weights for the control points $\mathbf{V}_i$. $N_i^3(t)$ are B-spline basis functions. Define two polynomial functions $\mathbf{A}(t)$ and $B(t)$ shown as follows:

$$\mathbf{A}(t) = \mathbf{a}_3 t^3 + \mathbf{a}_2 t^2 + \mathbf{a}_1 t + \mathbf{a}_0 \equiv \sum_{i=0}^{3} N_i^3(t)w_i\mathbf{V}_i \tag{18}$$

$$B(t) = b_3t^3 + b_2t^2 + b_1t + b_0 \equiv \sum_{i=0}^{3} N_i^3(t)w_i \tag{19}$$

where the coefficient vectors $\mathbf{a}_i, i = 0,1,2,3$, can be evaluated from Equations (17); the coefficient $b_i, i = 0,1,2,3$, can be evaluated by the similar procedure for $\mathbf{a}_i$ except that the weights w_i are used instead of control points $w_i\mathbf{V}_i$ in Equation (18). Therefore, a NURBS curve $\mathbf{r}(t)$ can be represented as:

$$\mathbf{r}(t) = \frac{\mathbf{A}(t)}{B(t)} \tag{20}$$

According to Equation (2), the maximal chordal deviation $\left|\mathbf{h}(t)_{\max}\right|$ occurs at:

$$\dot{\mathbf{r}}(t)\cdot\mathbf{h}(t) = \frac{\mathbf{h}(t)}{\left(B(t)\right)^2}\cdot\left(B(t)\dot{\mathbf{A}}(t) - \dot{B}(t)\mathbf{A}(t)\right) = 0 \tag{21}$$

Usually $B(t) \neq 0$ for NURBS curves, so Equation (21) can be simplified as:

$$B(t)[\dot{\mathbf{A}}(t)\cdot\mathbf{h}(t)] - \dot{B}(t)[\mathbf{A}(t)\cdot\mathbf{h}(t)] = 0 \tag{22}$$

Similar to the discussion of non-rational free-form curves, for a planar NURBS curve, $\mathbf{h}(t)$ can be represented as follows (see Figure 2):

$$\mathbf{h}(t) = \omega(t)\mathbf{\Omega} \tag{23}$$

where $\omega(t)$ is the magnitude of an error vector $\mathbf{h}(t)$; $\mathbf{\Omega} = \frac{\mathbf{N}\times\mathbf{E}}{\left|\mathbf{N}\times\mathbf{E}\right|}$ is the unit error vector perpendicular to the interpolating chord vector $\mathbf{E}$; $\mathbf{N}$ is the unit normal vector of the plane on which the curve $\mathbf{r}(t)$ lies. Substituting $\mathbf{h}(t)$ into Equation (22) yields:

$$\omega(t)\left\{B(t)[\dot{\mathbf{A}}(t)\cdot\mathbf{\Omega}] - \dot{B}(t)[\mathbf{A}(t)\cdot\mathbf{\Omega}]\right\} = 0 \tag{24}$$

The root of $\omega(t) = 0$ cannot be for the maximal error; otherwise $\mathbf{h}(t) = 0$ and the condition of Equation (2) is violated. Therefore, Equation (24) can be simplified as follows:

$$B(t)[\dot{\mathbf{A}}(t)\cdot\mathbf{\Omega}]-\dot{B}(t)[\mathbf{A}(t)\cdot\mathbf{\Omega}]=0 \tag{25}$$

The derivatives $\dot{\mathbf{A}}(t)$ and $\dot{B}(t)$ can easily be computed from Equations (18) and (19). Substituting $\mathbf{A}(t)$, $B(t)$, $\dot{\mathbf{A}}(t)$ and $\dot{B}(t)$ into Equation (25), the following equation can be obtained:

$$c_4t^4+c_3t^3+c_2t^2+c_1t+c_0=0 \tag{26}$$

where

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}=\mathbf{\Omega}\cdot\begin{bmatrix} b_0\mathbf{a}_1-b_1\mathbf{a}_0 \\ 2[b_0\mathbf{a}_2-b_2\mathbf{a}_0] \\ 3b_0\mathbf{a}_3+b_1\mathbf{a}_2-b_2\mathbf{a}_1+3b_3\mathbf{a}_0 \\ 2[b_1\mathbf{a}_3-b_3\mathbf{a}_1] \\ b_2\mathbf{a}_3-b_3\mathbf{a}_2 \end{bmatrix} \tag{27}$$

Equation (26) is a quartic polynomial function, which can be solved explicitly. At most four roots of t_j^*, $j\le 4$, can be found in Equation (26). The maximal chordal deviation $|\mathbf{h}(t)_{\max}|$ of a planar NURBS curve occurs at the root t_j^* that has the largest error $|\mathbf{h}(t_j^*)|$.

For 3D spatial NURBS curves, the error vector $\mathbf{h}(t)$ no longer has the constant orientation as the error vector of a 2D planar curve does. The function to find the parameters for the maximal chordal deviation is usually higher than four-degree. According to Abel theory, the general polynomial equations of degree higher than four cannot be solved by purely algebraic methods [10]. Some conservative estimation method using convex hull property can be used for estimating the chordal deviation of the 3D spatial NURBS curves [7].

Similar to the discussion in Section 4, the chordal deviation of a planar NURBS offset curve can be calculated. To find the maximal error parameter of the NURBS offset curve, the equation similar to Equation (26) can be deducted as follows:

$$e_4t^4+e_3t^3+e_2t^2+e_1t+e_0=0 \tag{28}$$

where

$$\begin{bmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}=\mathbf{\Omega}^o\cdot\begin{bmatrix} b_0\mathbf{a}_1-b_1\mathbf{a}_0 \\ 2[b_0\mathbf{a}_2-b_2\mathbf{a}_0] \\ 3b_0\mathbf{a}_3+b_1\mathbf{a}_2-b_2\mathbf{a}_1+3b_3\mathbf{a}_0 \\ 2[b_1\mathbf{a}_3-b_3\mathbf{a}_1] \\ b_2\mathbf{a}_3-b_3\mathbf{a}_2 \end{bmatrix} \tag{29}$$

Equations (28) and (29) are similar to Equations (26) and (27) except that $\mathbf{\Omega}$ is replaced by $\mathbf{\Omega}^o$. The procedures to solve Equation (28) are the same as those for Equation (26).

6. ADAPTIVE CURVE INTERPOLATION

In an interpolation process, a sequence of linear segments is generated to interpolate a curve $\mathbf{r}(t)$ under a given tolerance τ. Let $\mathbf{r}(t_i)$ be the start point of the current interpolation segment on a curve $\mathbf{r}(t)$. An adaptive interpolation method is proposed for finding the next interpolation point $\mathbf{r}(t_{i+1})=\mathbf{r}(t_i+\Delta t)$ satisfying the following condition:

$$\left|\mathbf{h}\left(\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t)}\right)\right|\le\tau \tag{30}$$

where Δt is the incremental parameter value from current parameter t_i; $\left|\mathbf{h}\left(\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t)}\right)\right|$ is the chordal deviation of the interpolating segment $\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t)}$ as discussed in the previous sections. Let $\mu\in[0,1]$ be a variation ratio for controlling the chordal deviation. Usually, to minimize the number of interpolation segments, a good interpolation method keeps the chordal deviation close to the interpolation tolerance τ. By combining the variation ratio μ, Equation (30) is converted to the equation shown as follows:

$$\left|\mathbf{h}\left(\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t)}\right)\right|\in[(1-\mu)\tau,\tau] \tag{31}$$

When μ has a small value, the chordal deviation $\left|\mathbf{h}\left(\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t)}\right)\right|$ is close enough to the given tolerance τ that the number of interpolation segments can be minimized. However, the incremental parameter Δt

usually cannot be found explicitly in Equation (31). It is noticed that the value $(1-\frac{\mu}{2})\tau$ is in the middle of $[(1-\mu)\tau,\tau]$. To find the incremental parameter Δ_t in Equation (31), an interpolation error function $f(\Delta t)$ is defined as follows:

$$f(\Delta_t) \equiv \left|\mathbf{h}\left(\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t)}\right)\right| - (1-\frac{\mu}{2})\tau = 0 \tag{32}$$

Since $f(\Delta t)$ usually cannot be solved explicitly, a numerical secant method [10] can be used in this paper to iteratively search for the roots of Equation (32) until the following condition is satisfied:

$$-\frac{\mu\tau}{2} \le f(\Delta t) \le \frac{\mu\tau}{2} \tag{33}$$

By combining Equations (32) and (33), the chordal deviation $\left|\mathbf{h}\left(\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t)}\right)\right|$ satisfies the condition in Equation (31) and the interpolation accuracy can be guaranteed. Using the discussed secant searching method, the adaptive interpolation algorithm for non-rational curves is detailed in the following Algorithm I. Similar algorithms can also easily be applied to NURBS curves and planar offset curves.

Algorithm I: Adaptive interpolation method

Input:

Polynomial Curve coefficients: $\mathbf{a}_0$, $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$

Interpolation tolerance: τ

Variation ratio: μ

Output:

The interpolation points $\{\mathbf{r}(t_i)\}$ that satisfy the interpolation tolerance τ.

Begin

Evaluate the interpolation starting point at parameter $t_i = 0$ and output the starting point $\mathbf{r}(0)$.

While { Interpolation starting parameter $t_i < 1$ }

Compute the maximal chordal deviation parameter by Equations (9) and its correspondent error $\left|\mathbf{h}(\overline{\mathbf{r}(t_i)\mathbf{r}(1)})\right|$.

If $\{\left|\mathbf{h}(\overline{\mathbf{r}(t_i)\mathbf{r}(1)})\right| \le \tau\}$

Output the ending point $\mathbf{r}(1)$ and terminate the while loop.

End if

Estimate the initial Δt_0 by the curvature information [3].

Compute the chordal deviation $\left|\mathbf{h}\left(\overline{\mathbf{r}(t_i)\mathbf{r}(t_i+\Delta t_0)}\right)\right|$ by Equations (9).

Compute $f(\Delta t_0)$ by Equation (32).

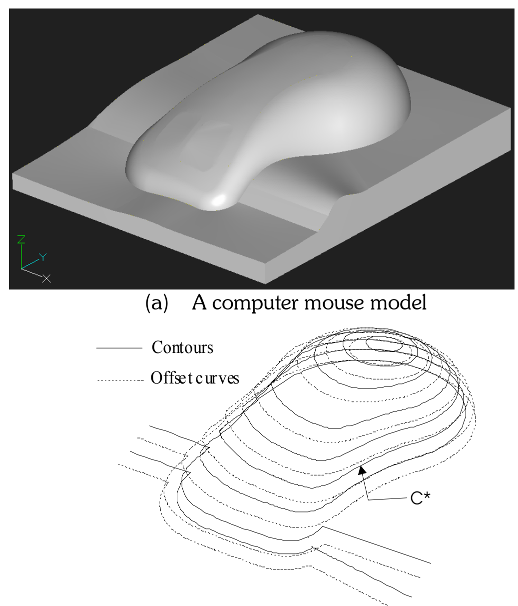
Find Δt^* by using the secant searching method that it can satisfy Equation (33).

Output $\mathbf{r}(t_i+\Delta t^*)$ as a new interpolating point.

Update the interpolation starting parameter as $t_i := t_i + \Delta t^*$.

End while

End.



(a) A computer mouse model

(b) The contours and their offset curves

Fig. 4. A computer mouse model and its contours for roughing

Algorithm I starts with the interpolation point located at the start point of the curve $\mathbf{r}(t)$. After finding the incremental parameter Δt, the parameter of the next point $\mathbf{r}(t_{i+1})$ is set to $t_{i+1} = t_i + \Delta t$ and a new searching iteration continues until the end point of the curve is reached. The following section shows some illustrative examples of using the proposed method for free-form curves interpolation and the applications for NC machining.

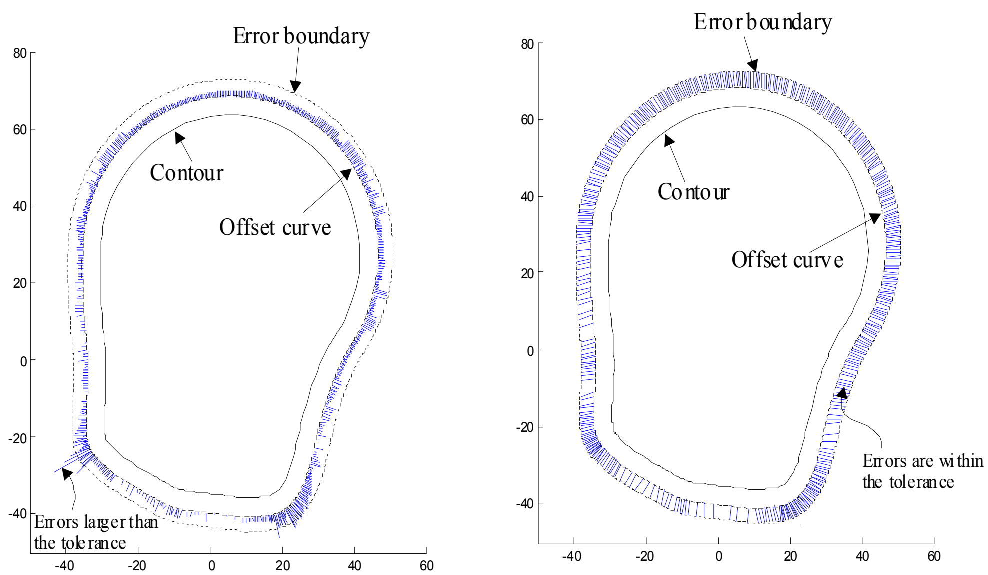
7. COMPUTER IMPLEMENTATION AND EXAMPLES

The proposed method and algorithm have been implemented to evaluate the performance of the proposed adaptive interpolation method. We also implemented the conventional subdivision method using convex hull property for comparison [8, 12].

Figure 4(a) shows an example part of a computer mouse. The size of the part surface is about $130 \times 150 \times 36$. To machine the example part, an endmill with size of 5.0 is used and the interpolation tolerance is set to $\tau = 0.001$. Shown in Figure 4(b) are the contours and their offset curves for the roughing process. A contour C^* of Figure 4(b) is used as an illustrative example to demonstrate the actual interpolation errors. The computation process of the adaptive interpolation method takes a few milli-seconds to compute the offset curve of each slice. The actual interpolation errors are scaled by 4,000 time, as shown in Figures 5(a) and 5(b) for the traditional subdivision method and the proposed adaptive interpolation method, respectively. In Figure 5(a), notice that some of the interpolation errors from the traditional subdivision method are larger than the given tolerance τ. On the other hand, the proposed adaptive interpolation method keeps the interpolation error within the tolerance τ, as shown in Figure 5(b).

8. CONCLUSIONS

This paper presents a method to find the maximum errors for interpolating free-form curves and the offset curves. Explicit solutions are presented for calculating the chordal deviations. Compared with the conventional subdivision and linear interpolation methods, the proposed interpolation method can greatly improve the interpolation accuracy and reduce the number of interpolation points of free-form curves and their offset curves. The computation process of the interpolation method is also fast (taking a few milli-seconds) and stable. The developed techniques of chordal deviation evaluation can be used for curve interpolation and NC tool-path generation in CAD/CAM systems. In the future research, accurate circular approximation of free-form curves will be investigated.



(a) The interpolation errors from the traditional subdivision method

(b) The interpolation errors from the proposed interpolation method

Fig. 5. The interpolation errors for one roughing tool-path

ACKNOWLEDGEMENTS

This work was partially supported by the National Science Foundation (NSF) CAREER Award (DMI-9702374), NSF Grant (DMI-0300297) and the Army Research Office (Grant #DAAG55-98-D-0003). Their support is greatly appreciated.

9. REFERENCES

[1] Choi B.K., *Surface Modeling for CAD/CAM*. Amsterdam: Elsevier, 1991.
[2] Choi B.K. and Jerard R.B., *Sculptured surface machining – Theory and applications*. Dordrecht: Kluwer Academic Publishers, 1998.
[3] Koc B., Ma Y., and Lee Y-S., Smoothing STL Files by Max-Fit Biarcs Curves for Rapid Prototyping, *Rapid Prototyping Journal*, 2000; 6(3):186-203.
[4] Lee Y-S. and Ji H., Surface Interrogation and Machining Strip Evaluation for Multi-axis CNC Die/Mold Machining, *International Journal of Production Research*, 1997; 35(1):225-252.
[5] Lee Y-S., Choi B.K., and Chang T.C., Cut Distribution and Cutter Selection for Sculptured Surface Cavity Machining, *International Journal of Production Research*, 1992;30(6):1447-1470.
[6] Piegl L. and Tiller W., *The NURBS book, Second edition*. Berlin, Germany: Springer-Verlag, 1997.
[7] Piegl L. and Tiller W., Geometry-based triangulation of trimmed NURBS surfaces. *Computer Aided Design*, 1998, 30(1):11-18.
[8] Zhu X., *Free-form curve & surface modeling techniques*. Beijing, China: Academy Press, 2000.
[9] Burden R.L. and Faires J.D., *Numerical Analysis, third edition*. Boston, MA: Prindle, Weber & Schmidt, 1985
[10] Pearson C.E., *Handbook of Applied Mathematics*. New York, NY: Van Nostrand Reinhold Company, 1974.
[11] Ren, H.T. Yau, and Y-S. Lee, "Contraction Tool Method and Clean-up Tool Path Generation for Machining Complex Polyhedral Models," (in print) Accepted for publication in the *Computers in Industry,* 2004.