Sharp Corner PH Interpolation Algorithm for High Speed CNC Machining

Behnam Moetakef Imani, Amirmohammad Ghandehariun
Ferdowsi University of Mashhad,
Azadi Square, Mashhad-Iran
imani@um.ac.ir

Abstract: Various methods for parametric interpolation of NURBS curves have been proposed in the past. However, the errors caused by the approximate nature of the NURBS interpolator were rarely taken into account. This paper proposes an integrated look-ahead algorithm for parametric interpolation along NURBS curves. The algorithm interpolates the sharp corners on the curve with the Pythagorean-hodograph (PH) interpolation. This will minimize the geometric and interpolator approximation errors simultaneously. The algorithm consists of four different modules: a sharp corner detection module, a PH construction module, a feedrate planning module, and a dynamics module. Simulations are performed to show correctness of the proposed algorithm. Experiments on an X-Y table confirm that the developed method improves tracking and contour accuracies significantly compared to previously proposed algorithms.
Keywords: Numerical Control, CNC Interpolation, PH Curve

1. INTRODUCTION

The function of the real-time interpolator in a computer numerical control (CNC) machine is to convert prescribed tool path and feedrate data into reference points for each sampling interval of the servo system. The closed-loop position and speed control will be possible through comparing the actual machine position, measured by encoders on motor axes, with the reference point [Farouki, 2008].

Parametric curves are extensively being used in a wide range of CAD applications such as automotive, aerospace and dies/molds, etc. Different representations are available for parametric curves, mainly including Bezier, B-spline, cubic spline and non-uniform rational B-spline (NURBS). Among these representations, NURBS have gained wide popularity to gradually become the CAD standard [Liu *et al.,* 2005]. Controllers in modern CNC machines offer parametric interpolations in addition to linear/circular interpolations. The main task in parametric interpolation is to compute parameter values of successive reference points. Some researchers have stated that utilizing these interpolations will result in reduced feedrate fluctuations and chord errors

and shorter machining time in comparison with linear/circular interpolations [Tsai *et al.,* 2008].

Accurate feedrate adjustments become increasingly important in the context of high-speed machining, where extreme feed acceleration/deceleration rates are required. Furthermore, tool chatter or breakage is a plausible result of the interpolator's inability to properly maintain the commanded feedrates [Farouki and Tsai, 2001]. Interpolators for general B-spline/NURBS curves are typically obtained based on Taylor series expansions. Such schemes inevitably incur truncation errors, caused by omission of higher-order terms [Farouki and Tsai, 2001].

To overcome this problem, the tool path can be described in terms of the Pythagorean-hodograph (PH) curves [Farouki and Sakkalis, 1990]. The algebraic structure of PH curves admits a closed-form reduction of the interpolation integral, yielding real-time CNC interpolation equations for constant or variable feedrates with high accuracy, robustness and flexibility [Tsai *et al.,* 2001].

Many interpolation methods for parametric curves have been proposed. Yeh and Hsu [Yeh and Hsu, 1999] developed a parametric curve interpolator using a Taylor series expansion algorithm. Farouki and Tsai [Farouki and Tsai, 2001] derived the exact Taylor series coefficients for variable-feedrate interpolators up to the cubic terms. Tsai et al. [Tsai *et al.,* 2001] proposed an algorithm for parametric interpolation with time-dependent feedrates along PH curves. However, no methodology for adjusting the feedrate based on the curve's properties was used. Yeh and Hsu [Yeh and Hsu, 2002] proposed an adaptive-feedrate interpolation algorithm. The feedrate was adjusted based on confined chord errors. Zhiming et al. [Zhiming *et al.,* 2002] suggested a strategy of variable feedrate machining based on geometric properties of tool path. Since feedrate fluctuations were not considered in these algorithms, sudden changes in curve's curvature would cause the acceleration/deceleration and jerk to be beyond the limits. Tsai et al. [Tsai *et al.,* 2008] proposed an integrated look-ahead dynamics-based algorithm for interpolation along NURBS curves which considered geometric and servo errors simultaneously. Nevertheless, the errors caused by the approximate nature of the Taylor series interpolator were not included in this algorithm.

In this paper, an integrated look-ahead Pythagorean hodograph interpolation algorithm for NURBS curves is proposed. The algorithm consists of a sharp corner detection module, a PH construction module, a feedrate planning module, and a dynamics module in order to keep the constraints on chord errors, feedrate fluctuations, servo errors, and interpolation errors on sharp corners simultaneously. With the proposed algorithm, planning a higher feedrate profile while achieving better contour accuracy and shorter machining time is possible. Simulations and experiments are performed to show effectiveness of the proposed algorithm.

2. NURBS INTERPOLATOR ALGORITHM

Suppose $C(u)$ represents a NURBS curve and is given by [Piegl and Tiller, 1997]:

$$C(u) = \frac{\sum_{i=0}^{n} N_{i,p}(u) P_i W_i}{\sum_{i=0}^{n} N_{i,p}(u) W_i} \tag{1}$$

where P_i represents the control point, W_i is the weight of P_i, $n+1$ is the number of the control points, and p is the degree of NURBS. $N_{i,p}(u)$ is the B-spline basis function, and can be calculated using the recursive formulas given as follows:

$$N_{i,0}(u) = \begin{cases} 1 & u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \tag{3}$$

where $\{u_0, \ldots, u_{n+p+2}\}$ represents the knot vector and u is the interpolation parameter.

To implement NURBS interpolation, a second-order approximation interpolation algorithm is adopted here. Using the Taylor series expansion method, the curve approximation up to the second derivative is given as follows:

$$u_{k+1} = u_k + \left.\frac{du}{dt}\right|_{t=t_k} \Delta t + \left.\frac{d^2u}{dt^2}\right|_{t=t_k} \frac{(\Delta t)^2}{2!} \tag{4}$$

Substituting the derivatives into the Eq. (4), the second-order Taylor expansion can be expressed as [Farouki and Tsai, 2001]:

$$u_{k+1} = u_k + \frac{V_k T_s}{|C'(u_k)|} + \frac{T_s^2}{2}\left\{\frac{A_k}{|C'(u_k)|} - \frac{V_k^2 [C'(u_k).C''(u_k)]}{|C'(u_k)|^4}\right\} \tag{5}$$

where V_k, A_k, T_s, $C'(u_k)$ and $C''(u_k)$ are the feedrate, the acceleration, the sampling time, and the first and second derivatives of the NURBS curve, respectively. The De Boor algorithm has been used for calculating $C(u_k)$, $C'(u_k)$ and $C''(u_k)$ in real-time implementation [Tsai *et al.,* 2008].

3. PYTHAGOREAN-HODOGRAPH INTERPOLATOR ALGORITHM

A polynomial PH curve $r(\xi) = \big(x(\xi), y(\xi)\big)$ is defined by its parametric hodograph [Tsai *et al.,* 2001]:

$$x'(\xi) = u^2(\xi) - v^2(\xi) \quad y'(\xi) = 2u(\xi)v(\xi) \tag{6}$$

These forms guarantee that $x'(\xi)$ and $y'(\xi)$ are elements of a Pythagorean triple of polynomials which satisfy

$$x'^2(\xi) + y'^2(\xi) = \sigma^2(\xi) \tag{7}$$

where

$$\sigma(\xi) = |r'(\xi)| = u^2(\xi) + v^2(\xi) = \frac{ds}{d\xi} \tag{8}$$

is the parametric speed of $r(\xi)$. A PH quintic is obtained by substituting Bernstein-form quadratic polynomials

$$u(\xi) = u_0(1-\xi)^2 + 2u_1(1-\xi)\xi + u_2\xi^2$$
$$v(\xi) = v_0(1-\xi)^2 + 2v_1(1-\xi)\xi + v_2\xi^2 \quad (9)$$

into Eq. (6) and integrating [Tsai *et al.,* 2001].

Considering a time-dependent feedrate function $V(t)$ with the indefinite integral $F(t)$ imposed on the PH curve $r(\xi)$, the interpolation equation yields to the solution of the relation

$$s(\xi) = F(t) \quad (10)$$

where $s(\xi)$ is the arc length [Tsai *et al.,* 2001]. For a sampling interval Δt, the real-time CNC interpolator must compute the parameter values $\xi_1, \xi_2, \ldots$ of reference points at times $\Delta t, 2\Delta t, \ldots$ These values are roots of the polynomial equations

$$s(\xi_k) = F(k\Delta t) \qquad \text{for } k = 1, 2, \ldots \quad (11)$$

The above equations can be solved to obtain ξ_k using a few Newton-Raphson iterations [Tsai *et al.,* 2001],

$$\xi_k^{(r+1)} = \xi_k^{(r)} - \frac{s\left(\xi_k^{(r)}\right) - F(k\Delta t)}{\sigma\left(\xi_k^{(r)}\right)} \;\; ; \;\; r = 0, 1, \ldots \quad (12)$$

with starting approximation $\xi_k^{(0)} = \xi_{k-1}$.

4. THE LOOK-AHEAD PH-BASED ALGORITHM

4.1. **System Architecture**

The look-ahead PH-based algorithm, which acts as a CNC controller is described here in detail. The controller consists of three main programs: a CNC interpreter, a reference point generator, and a motion controller. The CNC interpreter reads NC commands from NC files to generate and store NC blocks in the NC FIFO (First-In-First-Out) memory. The reference point generator generates consecutive reference points based on the proposed algorithm. The motion controller will then control the X-Y table based on the generated reference points.

The reference point generator algorithm consists of four different modules: a sharp corner detection module, a PH construction module, a feedrate planning module, and a dynamics module. The sharp corner detection module looks ahead NC blocks to identify sharp corners of the curve. Then the curve is divided into small segments according to sharp corners. The task of the PH construction module is to construct PH curves. Here, the region on the curve that is found to be a sharp corner will be approximated using a PH curve. The length of each segment is calculated and stored in the buffer. The feedrate planning module plans the feedrate profile of each segment based on

constraints on chord errors, feedrate, and acceleration/deceleration. In order to keep the constraints on the contour errors, the dynamics module will further modify the feedrate profile. Finally, the motion controller performs real-time control on the X-Y table. Algorithms for each module are detailed in the following sections.

4.2. **The Sharp Corner Detection Module**

The sharp corner detection module plays an important role in the look-ahead algorithm. In this study, a sharp corner is defined as the feedrate sensitive zone at which the feedrate should be reduced to maintain contour accuracy. There are two criteria in identifying sharp corners. The first criterion is that the derivative of the curvature at a sharp corner is equal to zero; it is given as:

$$\left.\frac{d\kappa(u)}{du}\right|_{u=u_k} = 0 \tag{13}$$

where $\kappa(u)$ is the curvature given as:

$$\kappa(u) = \frac{\left|\frac{dC}{du} \times \frac{d^2C}{du^2}\right|}{\left|\frac{dC}{du}\right|^3} \tag{14}$$

The first criterion is not sufficient in determining sharp corners as it does not include feedrate effects [Tsai *et al.,* 2008]. Therefore, in order to consider feedrate effects in identifying sharp corners, a second criterion is utilized. Based on this criterion, the curvature at the sharp corner zone should exceed the threshold value κ_{th}, which is defined as:

$$\kappa_{th} = \frac{A_{max}}{V_{max}^2} \tag{15}$$

where A_{max} is the maximum acceleration limit and V_{max} is the given feedrate in the NC block. In other words, the second criterion examines whether the centripetal accelerations of local max/min points exceed the maximum acceleration limit [Tsai *et al.,* 2008]. The region on the curve where this value is exceeded is identified as the sharp corner zone, and it should be replaced with a PH curve for a better supervision of the feedrate. These criterions can be illustrated by the curve shown in Figure 1 where segments AB and CD are identified as sharp corners.

4.3. **The PH Construction Module**

Utilizing Taylor series in interpolation along the NURBS curve will inherently cause interpolation errors and feedrate fluctuations, which will reduce the contour accuracy and surface quality. However, PH curves admit analytic reduction of the interpolation integral, which will result in an exact interpolation equation [Tsai *et al.,* 2001]. Therefore, approximating the sharp corner regions on the NURBS curve with PH curves will decrease feedrate fluctuations.

In order to approximate a segment on the NURBS curve by a PH quintic, the first order Hermite data of the segment, namely the start and end points and derivatives,

should be firstly computed. Using the complex representation for a planar PH curve [Moon *et al.,* 2001],

$$r(\xi) = x(\xi) + iy(\xi) \tag{16}$$

where ξ is a real parameter, the complex polynomial $w(\xi) = u(\xi) + iv(\xi)$ will be defined so that

$$r'(\xi) = w^2(\xi) \tag{17}$$

Considering r_0, r_1, d_0 and d_1 as complex representations of the start point, end point, start derivative and end derivative, respectively, $w(\xi)$ can then be expressed in Bernstein form [Moon *et al.,* 2001],

$$w(\xi) = w_0(1-\xi)^2 + 2w_1(1-\xi)\xi + w_2\xi^2 \tag{18}$$

where

$$w_0 = \pm\sqrt{d_0} \tag{19}$$

$$w_2 = \pm\sqrt{d_1} \tag{20}$$

$$w_1 = \frac{1}{4}\left[-3(w_0 + w_2) \pm \sqrt{120(r_1 - r_0) - 15w_0^2 - 15w_2^2 + 10w_0w_2}\right] \tag{21}$$

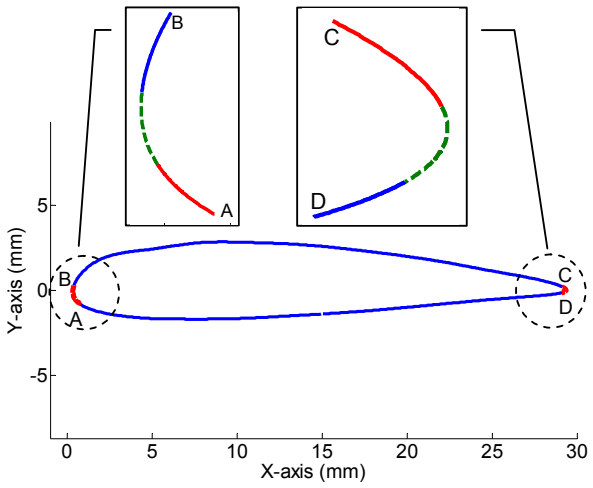

Figure 1; NURBS approximation of a NACA2415 airfoil profile

Table I; Comparison of error of PH approximation for the airfoil profile

Curve segment	Error (μm)			
	Without curve division		With curve division	
	RMS	Max	RMS	Max
AB	1.951	4.341	0.189	0.358
CD	5.627	12.54	0.539	1.363

The PH curve on the sharp corner should be constructed in the way that the error between the curve and the original NURBS curve is bounded. In order to achieve this, the module will calculate the root-mean-square of error between the two curves. If the calculated error exceeds the assigned bound of error, the sharp corner zone on the original curve will be divided into two segments and a PH curve will be constructed on each segment. This procedure will be repeated until the desired bound of error is achieved. This algorithm has been implemented for the airfoil in Figure 1 with the maximum allowed RMS of error $e = 1\mu m$. Table I shows the effect of dividing the

sharp corner segments on reduction of the error between the original curve and the approximated PH curve.

4.4. **The Feedrate Planning Module**

The first task of the feedrate planning module is to obtain the feedrate at sharp corners. To achieve this task, the proposed algorithm combines the adaptive-feedrate interpolation scheme [Yeh and Hsu, 2002] with the curvature-based feedrate interpolation algorithm [Zhiming *et al.,* 2002]. The algorithm for determining the feedrate at sharp corners is given as [Tsai *et al.,* 2008]:

$$V_{sp}(u_k) = \min\{V_{af}(u_k), V_{cf}(u_k)\} \tag{22}$$

where $V_{af}(u_k)$ and $V_{cf}(u_k)$ are given as follows:

$$V_{af}(u_k) = \frac{2}{T_s}\sqrt{\frac{2\delta}{\kappa(u_k)} - \delta^2} \tag{23}$$

$$V_{cf}(u_k) = \frac{\bar{\kappa}}{\kappa(u_k) + \bar{\kappa}} V_{max} \tag{24}$$

Here, V_{af}, V_{cf} and V_{max} are adaptive-feedrate, curvature-feedrate, and feedrate command, respectively. δ and κ are the chord tolerance and the curvature of the NURBS curve. $\bar{\kappa}$ maintains the derivative continuity of curvature-feedrate in Eq. (24).

Having obtained the feedrate at sharp corners, the second task of the feedrate planning module is to plan the feedrate profile of each segment such that the constraints on feedrate and acceleration/deceleration are satisfied. The polynomial quintic feedrate profile in the normalized form [Tsai *et al.,* 2001] will be generated based on the calculated length of each segment, feedrates at the corners, maximum feedrate, and maximum acceleration/deceleration. The quintic feedrate profile will ensure C^2 continuity on the joints of the curves on sharp corners.

Assuming T as the traversal time of each of the profile's phases, the normalized time variable is introduced as $\tau = t/T$. The velocity profile is expressed using the Bernstein basis functions on the unit interval $\tau \in [0,1]$ [Tsai *et al.,* 2001]:

$$b_k^n(\tau) = \binom{n}{k}(1-\tau)^{n-k}\tau^k \quad \text{for } k = 0,1,\dots,n \tag{25}$$

Therefore, the quintic feedrate profile will have the form [Tsai *et al.,* 2001]:

$$V(\tau) = \sum_{k=0}^{5} V_k b_k^5(\tau) \tag{26}$$

The parameters V_k will be calculated based on the length of each segment, feedrates at the corners, maximum feedrate, and maximum acceleration/deceleration.

4.5. **The Dynamics Module**

Application of the first three modules will not guarantee the satisfaction of the constraint on the contour error, which is because no contour error information is included in these modules.

In order to predict the contour errors, the AC servo control system for each axis should be modeled and simulated. The block diagram shown in Figure 2 is used as the dynamic model of the servo control system.

The transfer function between the tracking error $E(s)$ and the velocity $V(s)$ on each axis for the servo control system shown in Figure 2 is obtained as:

$$\frac{E(s)}{V(s)} = \frac{a_4 s^4 + a_3 s^3 + (a_2 - b_2)s^2 + (a_1 - b_1)s + (a_0 - b_0)}{a_4 s^5 + a_3 s^4 + a_2 s^3 + a_1 s^2 + a_0 s} \quad (27)$$

The parameters in Eq. (27) are given in Table II.

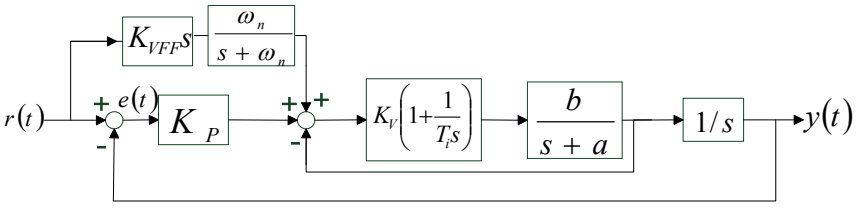

Figure 2; Block diagram of the servo control system [Tsai et al., 2008]

Table II; Parameters of the servo control transfer function [Tsai et al., 2008]

Parameter	X-axis	Y-axis
a_0	1.938×10^9	1.904×10^9
a_1	3.538×10^7	3.496×10^7
a_2	2.135×10^5	2.120×10^5
a_3	6.984×10^2	6.948×10^2
a_4	1.00	1.00
b_0	1.938×10^9	1.904×10^9
b_1	3.476×10^7	3.435×10^7
b_2	1.471×10^5	1.466×10^5

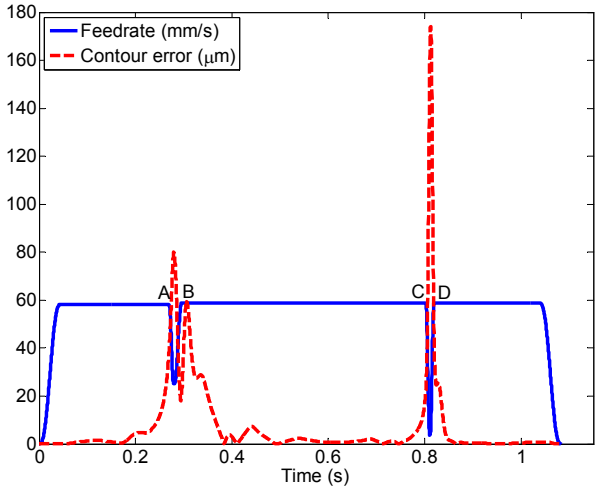

Figure 3; The feedrate profile and the contour errors after application of sharp corner detection, PH construction, and feedrate planning modules

Having developed the tracking error equation, the contour error equation can be approximated as [Tsai *et al.,* 2008]:

$$\varepsilon_e = -E_x \sin\phi + E_y \cos\phi \quad (28)$$

where E_x and E_y are tracking errors on the x-axis and y-axis, respectively. ϕ is the angle between the tool path tangent and the x-axis.

The dynamics module will use Eq. (27) and Eq. (28) to calculate the contour error. Figure 3 shows the contour errors along with the feedrate profile obtained by applying

the sharp corner detection, PH construction, and feedrate planning modules. Here, maximum feedrate and maximum acceleration/deceleration are set to 3500 mm/min and 2450 mm/s^2, respectively [Tsai *et al.,* 2008]. It is clear that major contour errors occur in the sharp corner areas of the curve, namely AB and CD.

Therefore, in order for the contour errors to be kept bounded to their limit value, the dynamics module will expand the sharp corner area and will plan the feedrate profile again. This will smoothen the feedrate profile on the sharp corner and thus, will reduce the contour errors is this area. This procedure will be repeated until the desired constraint on the contour error is satisfied. The flowchart of the dynamics module is shown in Figure 4.

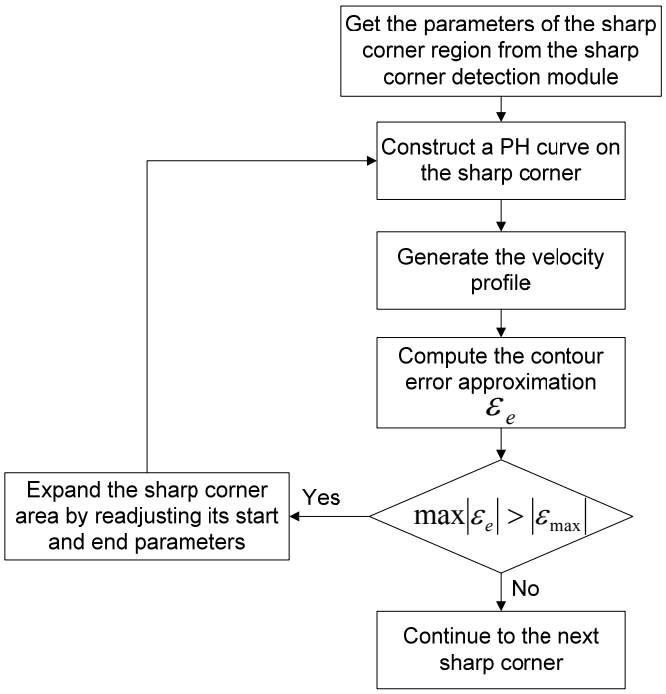

Figure 4; The flowchart of the dynamics module

Figure 5 shows the feedrate profile of the airfoil curve using the look-ahead PH-based algorithm with the contour error limit $\varepsilon_{max} = 30$ μm. Note the smoothness of the feedrate profile on the sharp corners as compared to Figure 3.

5. NUMERICAL SIMULATION AND EXPERIMENTAL VERIFICATION

In this section, numerical simulations and experiments are performed on the airfoil curve. Parameters of the interpolator for simulations and experiments are as listed in Table III.

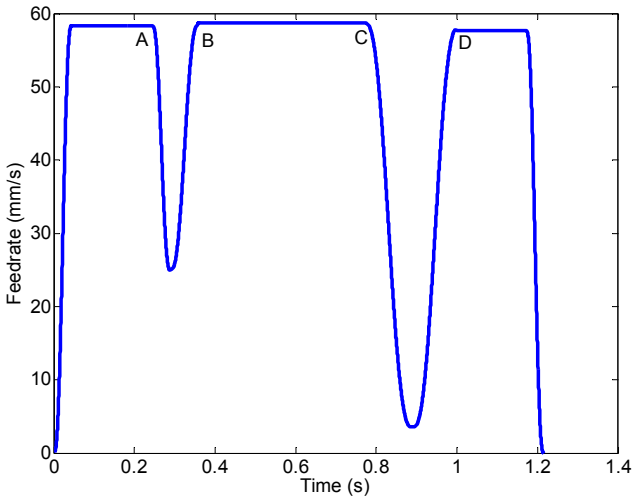

Figure 5; Feedrate profile of the airfoil using the look-ahead PH-based algorithm

5.1. **Numerical Simulation**

Simulations are conducted to compare the performance among adaptive-feedrate [Yeh and Hsu, 2002], curvature-feedrate [Zhiming *et al.,* 2002], the look-ahead NURBS-based [Tsai *et al.,* 2008], and the look-ahead PH-based interpolation algorithms. The servo control system as shown in Figure 2 is included in the simulation. Performances of the four interpolation algorithms are listed in Table IV.

According to Table IV, it is clear that the contour errors of the look-ahead PH-based interpolation are significantly smaller than those of the adaptive-feedrate and curvature-feedrate interpolations. Also it can be noted that both the look-ahead NURBS-based and the look-ahead PH-based algorithms keep the contour errors bounded to their maximum value of 30 μm. However, the look-ahead PH-based algorithm decreases the total motion time by 12%.

Table III; Parameters of the interpolator for numerical simulations and experiments

Maximum feedrate	V_{max}	3500 mm/min
Maximum acceleration	A_{max}	2450 mm/s^2
Chord error limitation	δ	1μm
Reference curvature	$\bar{\kappa}$	1 mm^{-1}
Contour error limit	ε_{max}	30 μm
PH approximation error limit	e	1 μm

Table IV; Simulation of the performance of different interpolation algorithms

Interpolation algorithm			Adaptive -feedrate	Curvature-feedrate	Look-ahead NURBS-based	Look-ahead PH-based
Tracking error (μm)	X-axis	Max	337.66	196.70	57.97	82.91
		RMS	41.50	27.88	21.69	23.98
	Y-axis	Max	84.32	22.66	26.54	24.99
		RMS	13.91	6.26	5.72	6.24
Contour error (μm)	Max		113.89	41.08	27.12	29.98
	RMS		19.60	8.83	6.35	8.15
	Mean		7.42	4.85	3.40	4.35
Time (s)			1.0295	1.1395	1.3615	1.2158

Figure 6 shows the contour errors of the above mentioned algorithms. It is seen that the proposed algorithm decreases the feedrate on the sharp corners just enough to limit the errors to their set maximum value while keeping the total motion time relatively short.

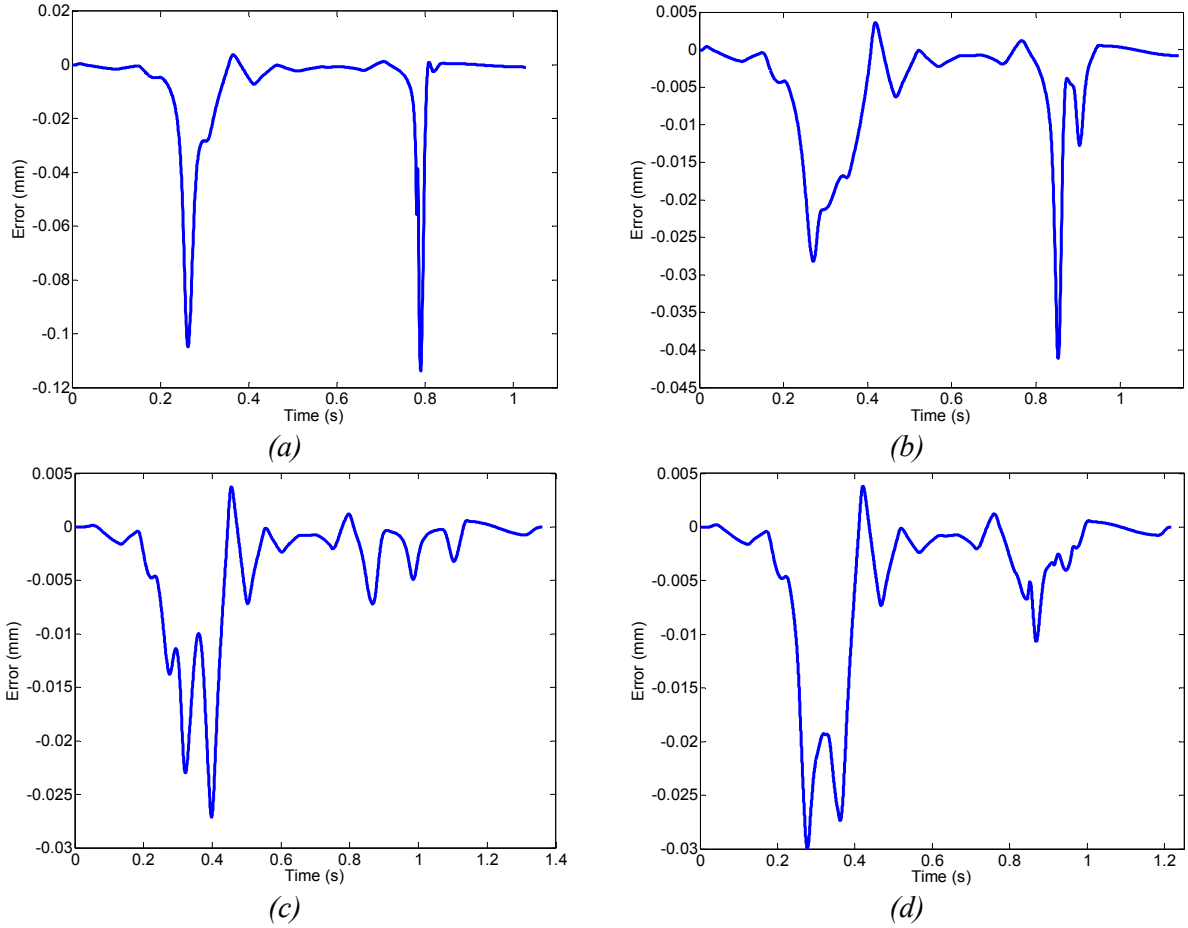

Figure 6; Comparison of contour errors: (a) Adaptive-feedrate; (b) Curvature-feedrate; (c) Look-ahead NURBS-based; (d) Look-ahead PH-based

5.2. **Experimental Results**

Figure 7 shows the hardware setup, which includes an X-Y table driven by two AC servomotors. The resolution of the encoders is 2500 pulses/rev. The PC interface utilized a motion control card to send the command pulses and receive encoder feedback signals. The Bit-Pattern Interpolation feature [Advantech, 2001] is used for controlling the servo motors. Two linear scales with 1 μm resolution are mounted on the X-Y table and are used for measuring the errors. The interpolator was implemented using MATLAB and Visual Basic.NET programming on a PC platform. Details of the experimental setup are reported in Table V.

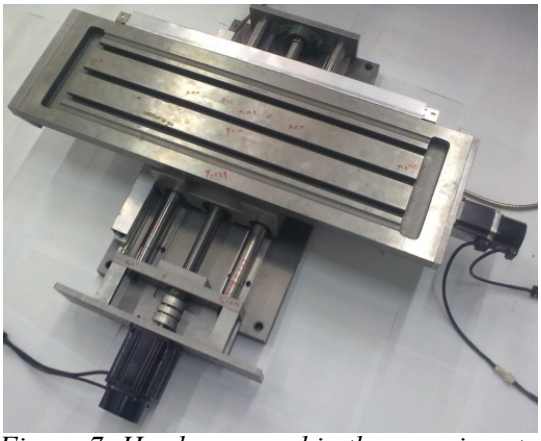

Figure 7; Hardware used in the experiments

Table V; Details of the experimental setup

Servo Drives	TECO TSTA-20C AC Servo Drive
Servo Motors	TECO TST0640 and TSB0845 AC Servo Motor
Motion Control	ADVANTECH PCI-1240 Motion Control Card
Linear Scale	CARMAR Linear Scales

The airfoil curve, shown in Figure 1, is tested under the feedrate command equal to $3500\ \mathrm{mm/min}$. The performances of the adaptive-feedrate, curvature-feedrate, the look-ahead NURBS-based, and the proposed algorithms are listed in Table VI. It is clear that the proposed algorithm can improve contour accuracy as compared to the adaptive-feedrate and curvature-feedrate interpolations. The machining time will also be shorter in the look-ahead PH-based algorithm than the look-ahead NURBS-based algorithm.

Table VI; Comparison of experimental performances of the interpolation algorithms

Interpolation algorithm			Adaptive-feedrate	Curvature-feedrate	Look-ahead NURBS-based	Look-ahead PH-based
Tracking error (μm)	X-axis	Max	514	516	505	512
		RMS	447.43	414.48	386.77	411.31
	Y-axis	Max	486	310	303	317
		RMS	109.87	86.17	77.81	86.51
Contour error (μm)	Max		314.64	59.28	51.08	58.78
	RMS		31.13	16.89	12.37	15.20
	Mean		8.63	5.62	3.79	5.29
Time (s)			1.0295	1.1395	1.3615	1.2158

6. CONCLUSION

A novel NURBS-PH interpolator was proposed in this paper. Sharp corners of the curve are identified not only by the derivative of the curvature but also by the feedrate criterion. A PH curve is then constructed on the corner to reduce feedrate fluctuations. The error between the constructed curve and the original NURBS curve is bounded to its set maximum value. The curve is divided into small segments and a smooth quintic feedrate profile is planned on sharp corners. The dynamics module further modifies the

feedrate profile to keep the contour errors limited. Simulations were performed to validate the proposed algorithm. Experiments confirm that the look-ahead PH-based algorithm can increase contour accuracy significantly compared with adaptive-feedrate and curvature-feedrate interpolation algorithms. It will also shorten the machining time as compared to the look-ahead NURBS-based algorithm. This research demonstrates the effectiveness of the look-ahead PH-based algorithm.

REFERENCES

[Advantech, 2001] "PCI-1240 4-Axis Stepping/Pulse-type Servo Motor Control Card User's Manual"; Advantech Co., Ltd; 2001.

[Farouki, 2008] Farouki, R. T.; "Pythagorean-Hodograph Curves: Algebra and Geometry Inseparable"; Springer; 2008.

[Farouki and Sakkalis, 1990] Farouki, R. T.; Sakkalis, T.; "Pythagorean hodographs"; In: *IBM Journal of Research and Development*, Vol. 34, pp. 736-752; 1990.

[Farouki and Tsai, 2001] Farouki, R. T.; Tsai, Y. F.; "Exact Taylor series coefficients for variable-feedrate CNC curve interpolators"; In: *Computer-Aided Design*, Vol. 33, pp. 155-165; 2001.

[Liu et al., 2005] Liu, X.; Ahmad, F.; Yamazaki, K.; Mori, M.; "Adaptive interpolation scheme for NURBS curves with the integration of machining dynamics"; In: *International Journal of Machine Tools and Manufacture*, Vol. 45, pp. 433-444; 2005.

[Moon et al., 2001] Moon, H. P.; Farouki, R. T.; Choi, H. I.; "Construction and shape analysis of PH quintic Hermite interpolants"; In: *Computer Aided Geometric Design*, Vol. 18, pp. 93-115; 2001.

[Piegl and Tiller, 1997] Piegl, L.; Tiller, W.; "The NURBS Book"; 2nd Ed.; Springer, New York; 1997.

[Tsai et al., 2001] Tsai, Y. F.; Farouki, R. T.; Feldman, B.; "Performance analysis of CNC interpolators for time-dependent feedrates along PH curves"; In: *Computer Aided Geometric Design*, Vol. 18, pp. 245-265; 2001.

[Tsai et al., 2008] Tsai, M. S.; Nien, H. W.; Yau, H. T.; "Development of an integrated look-ahead dynamics-based NURBS interpolator for high precision machinery"; In: *Computer-Aided Design*, Vol. 40, pp. 554-566; 2008.

[Yeh and Hsu, 1999] Yeh, S. S.; Hsu, P. L.; "The speed-controlled interpolator for machining parametric curves"; In: *Computer-Aided Design*, Vol. 31, pp. 349-357; 1999.

[Yeh and Hsu, 2002] Yeh, S. S.; Hsu, P. L.; "Adaptive-feedrate interpolation for parametric curves with a confined chord error"; In: *Computer-Aided Design*, Vol. 34, pp. 229-237; 2002.

[Zhiming et al., 2002] Zhiming, X.; Jincheng, C.; Zhengjin, F.; "Performance evaluation of a real-time interpolation algorithm for NURBS curves"; In: *International Journal of Advanced Manufacturing Technology*, Vol. 20, pp. 270-276; 2002.