PC-based Controller with Real-time Look-ahead NURBS Interpolator

Hong-Tzong Yau[1], Jun-Bin Wang[2], Chien-Yu Hsu[3] and Chih-Hua Yeh[4]

[1,3,4]National Chung-Cheng University, [1]imehty@ccu.edu.tw, [3]apache@pcc-server.ccu.edu.tw
[2]Nan-Jeon institute of technology, [2]jbwang@mail.njtc.edu.tw

ABSTRACT

This paper presents a real-time look-ahead NURBS interpolator implemented on a PC-based controller to solve the problems of traditional linear short NC segments. Taking advantage of the designs of the look-ahead function and multi-threads, the NURBS interpolator can obtain enough NC block information and complete feedrate planning before interpolation. Consequently, the computational load is reduced during an interpolation period and unnecessary feedrate acceleration and deceleration can be avoided. The continuous short blocks conforming to the CSB criterion can be fitted into NURBS curves in real-time. S-shaped federate profile for ACC/DEC planning can make the single block achieve C^1 continuity and jerk-limited capability. The simulation and experimental results show that the output contour profiles approach the input NC program models. This indicates that the proposed real-time look-ahead NURBS interpolator is able to provide a satisfactory performance

Keywords: NURBS, Interpolator, Real-time, Look-ahead, Continuous Short Blocks.

1. INTRODUCTION

There is a continuing trend toward using parametric forms to describe the contours of work-pieces in the areas of CAD/CAM technology and CNC machining. However, due to the rapid development of 3D scanning technology that employs linear NC segments for NC tool paths for digital CAD models, the use of linear short segments is still considered an issue. In high-speed and high-accuracy machining, the parametric interpolation is being substituted for conventional linear and circular interpolation. The commonly used parametric forms include Bezier, B-spline, and NURBS curves. NURBS curves have two advantages: (i) weighting control can locally change the contour profiles, and (ii) complicated contour representation does not need to increase the order of equation. NURBS curves have presently become a common and useful parametric form used by CAD/CAM engineers.

There have been many studies on NURBS interpolation[1]-[15]. The first issue is to compute successive reference-point parameter values. Farouki and Tsai [1] derived correct coefficients up to the third order for the Taylor series expansion. Cheng et al.[2] compared several numerical algorithms and suggested that Taylor's 2nd order approximation was a good choice for real-time NURBS command generators.

Machining with a constant feedrate can reduce machining profile errors that may result from cutter wear and vibrations. However, over-cutting will happen and lead to loss of accuracy when the cutter moves along high curvature profiles. This brought into consideration the use of adaptive feedrate. The related published works discussed compensatory parameter [3], confined chord errors [4], constant cutter contact velocity [5], constant material removal rate MMR [6][15], and system dynamics [7]. In addition, Yong and Narayanaswami [8] presented a speed-error controlled interpolator which employs an off-line detection of feedrate sensitive corners. Liu et al.[9] employed a filtering method to smooth the feedrate profile that could reduce feedrate and accelerate fluctuations.

On the other hand, imperfect feedrate profiles reduce the quality of machining because of great jerk amplitudes. Furthermore, exciting the natural modes of the mechanical structure or servo control system will incur serious resonance. Therefore, jerk-limited consideration was adopted in recently published works, for example, the jerk-limited profile generation method by Nam and Yang [10], and trapezoidal acceleration-based feedrate generation by Erkorkmaz and Altintas [11].

Finally, the implementation of the NURBS interpolator relies on a real-time environment, for example, DSP [2][5][6][11][13], or PC with a real-time operation system [4][7][10][12][15]. Zhiming et al.[12] showed the CPU time required for one interpolating operation using Taylor's 1st order and 2nd order approximation methods. Considering that the sampling interval of the servo system is 2ms, it should be feasible to implement this interpolator on-line. Tikhon et al.[6] compared CPU time for NURBS interpolators with different algorithms. The result showed that 4ms per one interpolation was the upper bond for each interpolator at that time. The same conclusion was given in Reference [5], too. Therefore, to reduce the interpolation period, multi-threads design with look-ahead facility had been employed by References [7][10].

In recent years, due to the rapid development of 3D scanning technology, there has been a trend towards creating digital models using point-based or triangulated surface models. Linear NC segments are perhaps still the best NC representation form that is used to generate NC tool paths for digital CAD models because of the tool path generation speed and accuracy. Consequently, to deal with large amounts of linear short segments, Han et al.[13] presented a high speed machining algorithm based on the look-ahead interpolation technique for the machining of 3D surfaces obtained by CAD/CAM systems. The proposed algorithm improves the machining speed in a program consisting of small line blocks without any hardware support.

Yau and Kuo [14] took the lead in proposing a post-processing approach to convert G1 NC codes from most CAD/CAM systems to NURBS NC paths for high-speed contour machining. The NURBS interpolation strategy that took into account the optimized cutting feedrate based on machine dynamic response and curvature of the NURBS curve was also developed. Due to the superiority of cubic Bezier interpolation for fast computation, Yau et al.[7] proposed a real-time cubic Bezier interpolator with look-ahead function to deal with a large amount of linear short segments. But the cubic Bezier interpolator still has a C^1 discontinuity property at the interface of two G01 blocks, as shown in Fig. 1(b).

This paper presents a new PC-based real-time interpolated control system. A real-time look-ahead NURBS interpolator is used to process linear NC segments. In the NC interpreter, CSB Criterion is used to check the continuous short blocks to fit the NURBS curve, as shown in Fig. 1(c). Finally, S-shaped federate profile for ACC/DEC planning can make the single block achieve C^1 continuity and jerk-limited capability.

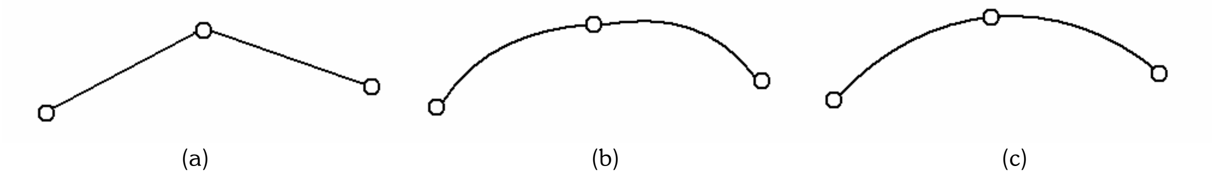

(a) (b) (c)

Fig. 1: (a) G01 linear interpolation. (b) cubic Bezier interpolation with C^0 continuity. (c) NURBS interpolation with C^1 continuity.

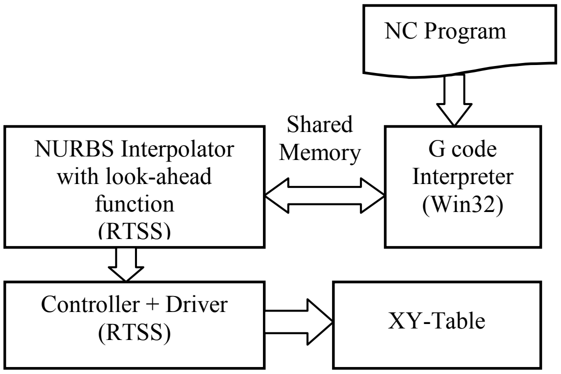


Fig. 2: The architecture of PC-based real-time NURBS interpolator.

2. LOOK-AHEAD FUNCTION

In order to achieve real-time interpolation control, a real time PC-based open control system is used in this study, in which the read file and interpreter are planned under the Win32 thread and the interpolator is planned on the RTSS (real-time subsystem) thread. The interpreter and interpolator can share the NC program code in real-time using the shared memory design. In Fig. 2, when the Interpreter decoder begins to read the NC program and transfer the data to the shared memory, the interpolator starts the look-ahead function and reads the data from the shared memory. Then the Continuous Short Blocks(CSB) Criterion [7] can quickly determine if the CSBs are G01 linear interpolation or NURBS interpolation and save the data in the look-ahead buffers 。

The flow chart of the NURBS interpolator with the look-ahead function is shown in Fig. 3. A NURBS curve with degree three was used to interpolate the NURBS curve in this paper. The line segment cannot be less than three. The maximum number of line segments is 30 because real-time look-ahead capability and computation of PC are considered.

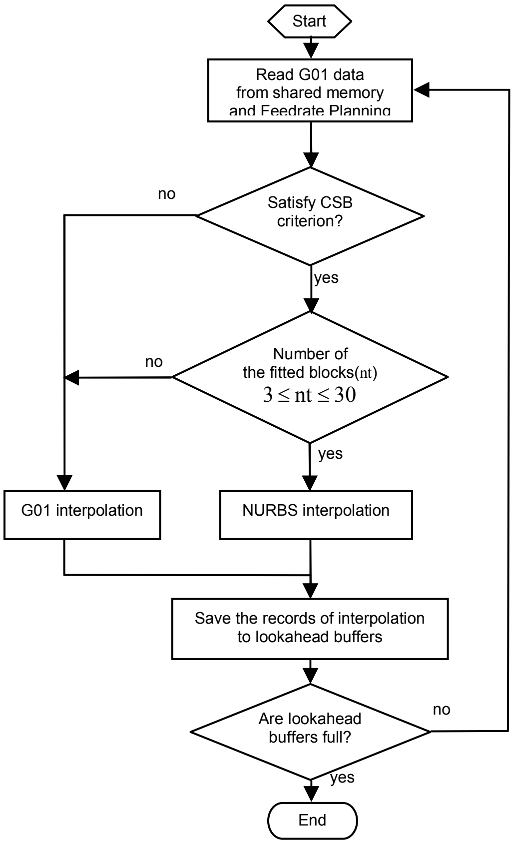


Fig. 3: Flowchart of the look-ahead function that determines G01 or NURBS interpolation.

3. REAL-TIME NURBS INTERPOLATOR

NURBS curves can describe very complex paths by changing the weights, knot vectors, and control points. In the look-ahead function, CSBs that satisfy the CSB criterion are used for NURBS interpolation. This section will introduce definition of NURBS curves and how to calculate control points. The interpolated tool path has jerk limit capability because the knot vectors are calculated according to the S-shaped ACC/DEC.

3.1 NURBS Curves

A pth-degree NURBS curve [16] is defined by

$$C(u)=\frac{\sum_{i=0}^{n} N_{i,p}(u)w_iP_i}{\sum_{i=0}^{n} N_{i,p}(u)w_i}=\sum_{i=0}^{n} R_{i,p}(u)P_i \qquad 0\le u\le 1 \tag{1}$$

where P_i are the control points, w_i are the weights, $N_{i,p}$ are the pth-degree B-spline basis functions defined on the non-periodic knot vector U

$$U = \{u_0, u_1, \ldots, u_{n+p+1}\} \tag{2}$$

$N_{i,p}$ are defined by equation (3) and $R_{i,p}$, the rational basis functions, are given by equation (4).

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

$$\mathrm{N}_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad i = 0, 1, \ldots, n \tag{3}$$

$$R_{i,p} = \frac{N_{i,p}(u) w_i}{\sum_{j=0}^{n} N_{j,p}(u) w_j} \tag{4}$$

$\mathrm{C}_0, \mathrm{C}_1, \ldots, \mathrm{C}_\mathrm{n}$ are the CSBs points that NURBS curves fitting will pass through. The NURBS curve is represented in matrix form. The following equation is the matrix form of the NURBS curve.

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ R_{0,p}(u_1) & R_{1,p}(u_1) & R_{2,p}(u_1) & \cdots & R_{\mathrm{n},p}(u_1) \\ R_{0,p}(u_2) & R_{1,p}(u_2) & R_{2,p}(u_2) & \cdots & R_{\mathrm{n},p}(u_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ \vdots \\ P_n \end{bmatrix} = \begin{bmatrix} C_0 \\ C_1 \\ C_2 \\ \vdots \\ C_n \end{bmatrix} \tag{5}$$

The knot vectors u_i are determined by the chord length method, where L is the total length of all line blocks. The parameter u_i of the knots vector, calculated by the ratio of single line block and total line blocks, are given in equation (6) and equation (7).

$$L = \sum_{i=1}^{n} \sqrt{\|C_i - C_{i-1}\|} \tag{6}$$

$$u_i = u_{i-1} + \frac{\|C_i - C_{i-1}\|}{L} \qquad i = 1, \ldots, n\text{-}1 \tag{7}$$

After getting knot vectors, the Cox-de Boor algorithm is used to calculate the rational basis function $R_{i,p}$. In equation (5), the control points of NURBS curve could be calculated by inversing the matrix.

3.2 S-shaped Feedrate Profile for ACC/DEC Planning

The look-ahead function not only runs CSB criterion to detect G01 linear interpolation or NURBS interpolation but also records end point velocity of every line segment. When corner error is $\varepsilon_{\max}$ and corner angle is θ, the equation corner feedrate F is as following according to Yau et al.[7].

$$F = \frac{K_p}{1\text{-}K_f} \cdot \frac{\varepsilon_{\max}}{\cos(\frac{\theta}{2})} \tag{8}$$

Before the NURBS interpolator runs theS-shaped feedrate planning process, the velocity of start and end points in every block are recorded in look-ahead buffers. Three continuous blocks in the look-ahead buffer are shown in Fig. 4, where block 1 and block 3 are NURBS blocks. $P_1 \sim P_7$ and $P_8 \sim P_n$ are CSBs that fit into NURBS curves whereas block 2 is a G01 block. P_7P_8 constructs a line segment of block 2. These three blocks are planned by S-shaped feedrate profile for ACC/DEC planning and the feedrate profile is shown in Fig. 5. The S-shaped feedrate profiling equations of block 1 in each step are shown as follows:

$$f(t) = f_S + \frac{1}{2} J_{\max} t^2 , \qquad (t_0 \le t \le t_1) \tag{9}$$

$$f(t) = f_1 + A_{\text{ref}}(t - t_1) - \frac{1}{2} J_{\max}(t - t_1)^2 , \quad (t_1 < t \le t_2) \tag{10}$$

$$f(t) = f_2 = F_{\text{NC}}, \qquad (t_2 < t \le t_3) \tag{11}$$

$$f(t) = F_{\text{NC}} - \frac{1}{2} J_{\max}(t - t_3)^2, \qquad (t_3 < t \le t_4) \tag{12}$$

$$f(t) = f_4 - D_{ref}(t - t_4) + \frac{1}{2} J_{\max}(t - t_4)^2, \ (t_4 < t \le t_5) \tag{13}$$

where f_s is the start feedrate of the NURBS block, F_{NC} is the maximum feedrate of the system, A_{ref} and D_{ref} are the maximum acceleration value of the system. $J_{\max}$ is the jerk limit.

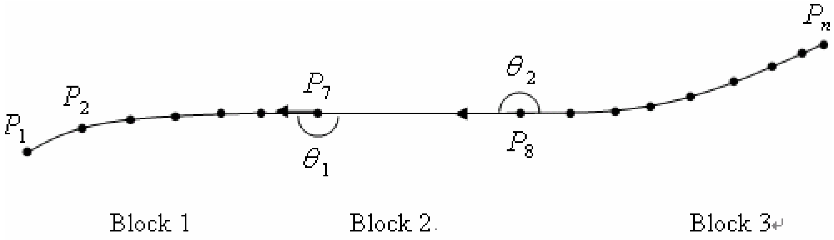


Fig. 4: Three successive blocks: Blockl and block3 are NURBS blocks, Block2 is a G01 block.

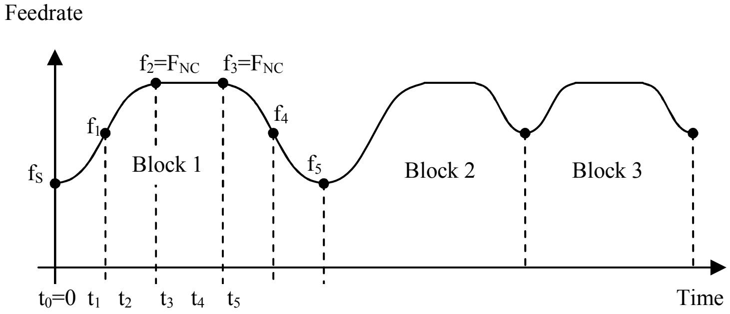


Fig. 5: S-shaped profiles for ACC/DEC feedrate planning.

3.3 NURBS Interpolation Algorithm

The definition of NURBS is changed to equation (14)

$$C(u)=\frac{\sum_{i=0}^{n}N_{i,p}(u)w_iP_i}{\sum_{i=0}^{n}N_{i,p}(u)w_i}=\frac{A(u)}{w(u)} \tag{14}$$

The knot of u_{i+1} can be obtained by using the Taylor series. Taylor's first-order expansion is used to speed up the interpolated process to 1ms. The following equation is used to calculate the knot of u_{i+1}.

$$u_{i+1}=u_i+\frac{V(u_i)\cdot T_s}{\left\|\frac{dC(u)}{du}\right\|_{u=u_i}} \tag{15}$$

where

$$\frac{dC(u)}{du}=C^{'}(u)=\frac{A^{'}(u)-w^{'}(u)C(u)}{w(u)} \tag{16}$$

$$w'(u)=\sum_{i=0}^{n}N'_{i,p}(u)w_i \tag{17}$$

$$A'(u)=\sum_{i=0}^{n}N'_{i,p}(u)P_i \tag{18}$$

$$N'_{i,p}(u)=P\left(\frac{N_{i,p-1}(u)}{u_{i+p}-u_i}-\frac{N_{i+1,p-1}(u)}{u_{i+p+1}-u_{i+1}}\right) \tag{19}$$

The procedure of the NURBS interpolator algorithm is as follows:

1. Determine the parameter u_i.
2. Calculate the first derivative of $C(u_i)$ using equation (16).
3. Update parameter u_{i+1} using equation (15).
4. Calculate the next interpolation point $C(u_{i+1})$ using equation (14).
5. Repeat the steps 1 to 4 until parameter $u_{i+1}>u_{\max}$,where $u_{\max}=\max\{u_j, j=0,1,\ldots,n+p+1\}$.

4. SIMULATION AND EXPERIEMENTAL RESULTS

This real-time NURBS interpolator is implemented using VC programming langrage and is run on a real-time OS (Windows 2000 + RTSS). The hardware includes a Pentium4 1.7G PC with Adventech PCI-1716 AD/DA card, PCI-1784 decoder card, and a XY plot table, as shown in Fig.6 (YASKAWA servo motors and drivers). Two NC codes with butterfly and shark shape are used for simulation. The detailed data of the NC codes are listed in Table.1. The motion parameters, such as feedrate, acceleration, jerk limit, and fitting conditions are listed in Table.2.

In the numerical simulation part, we pay special attention to the following three indications: (1) fitting rate of real-time NURBS interpolator to process large numbers of G01 code, (2) feedrate profiles, and (3) contour errors of the interpolated position command. The interpolated shapes of the curves are shown in Fig. 7. The NURBS interpolator fitting efficiency of these models is listed in Table.3. The fitting rate of the model Butterfly is 99.8% and that of the model Shark is 99.9%. The interpolated feedrate profiles are shown in Fig. 8. According to the result, the look-ahead function can eliminate unnecessary acceleration of single linear blocks. The S-shaped feedrate profile can make the velocity curve smoother, as shown with local magnification. The contour errors are shown in Fig. 8 and Fig. 10. The maximum contour error of the Butterfly model is 43.0 μm; its RMS is 8.8 μm. The maximum contour error of the Shark model is 34.4 μm, and its RMS is 16.6 μm (Table.4).

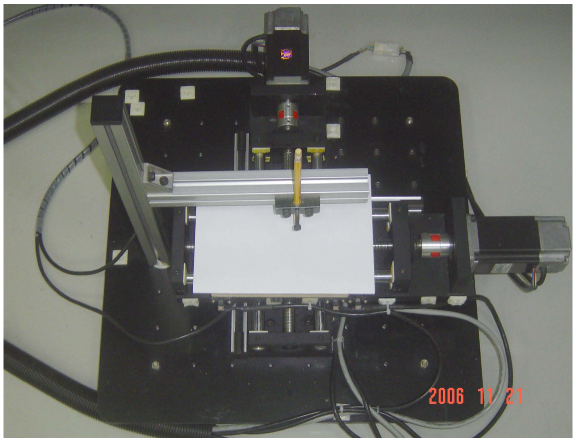


Fig. 6: Photograph of the X-Y table.

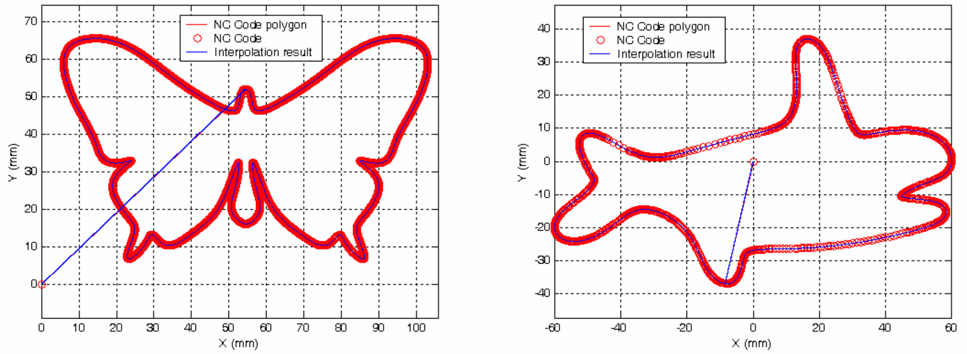
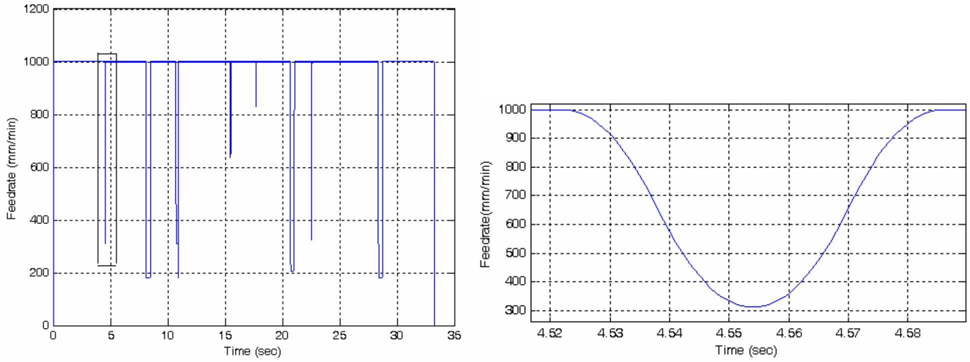
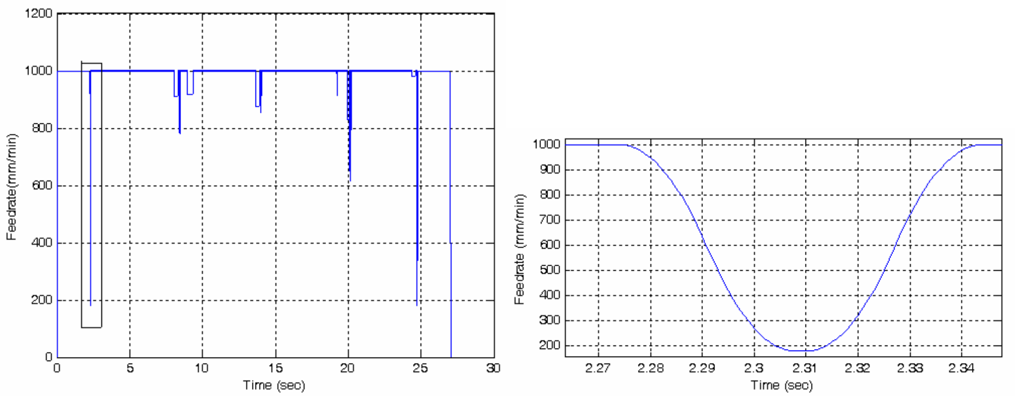


Fig. 7: Simulation result: (a) contour profile of the Butterfly. (b) contour profile of the Shark



(a) (b)



(c) (d)

Fig. 8: Simulation results: (a) Feedrate profiles of the Butterfly (b) Local magnification (c) Feedrate profiles of the Shark (d) Local magnification.

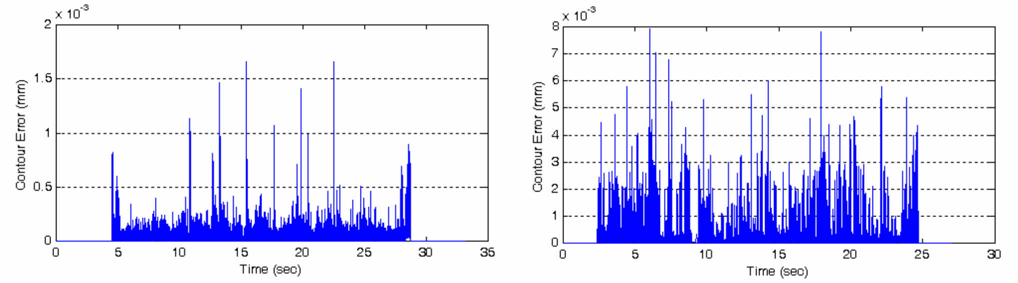
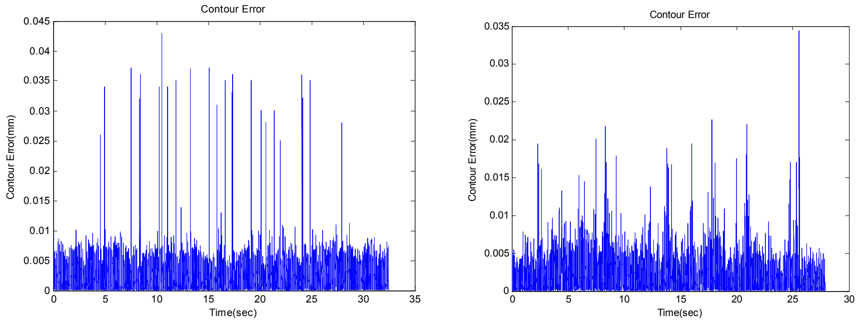


Fig. 9: Simulation: (a) Contour error of the butterfly, RMS=0.1028µm, Max=1.663µm. (b) Contour error of the Shark, RMS=0.8549µm, Max=7.919µm.



(a) (b)

Fig. 10: Experimental results: (a) Contour errors of the Butterfly (b) Contour errors of the Shark.

Cases	Number of G01 blocks	Max. length (mm)	Min. length (mm)	Mean length (mm)
Butterfly	5299	0.077	0.015	0.072
Shark	1004	1.799	0.012	0.370

Tab. 1: Data statistics of the NC programs in the simulations and experiments

Field	Velocity Planning				CSB Criterion	
Item	Feedrate (mm/min)	A_{max} (mm/s^2)	J_{max} (mm/s^3)	Corner error (mm)	Critical corner angle (degree)	Chord error (mm)
Value	1000	4900	50000	0.01	157	0.01

Tab. 2: The values of parameters for the simulations and experiments

	Classification	Before NURBS-fitting (blocks)	After NURBS-fitting (blocks)
Butterfly	G01	5299	11
	NURBS	0	185
	Total	5299	196
Shark	G01	1004	3
	NURBS	0	39
	Total	1004	42

Tab. 3: Simulation results: the comparisons between before and after NURBS fitting.

	Simulations		Experiments	
Contour errors	RMS	Max.	RMS	Max.
Butterfly	0.1	1.6	8.8	43.0
Shark	0.8	7.9	16.6	34.4

Tab. 4: Simulation and experimental results: The contour error (unit: µm)

5. CONCLUSION

This paper proposed a convincing solution, a real-time look-ahead NURBS interpolator, to solve the problems of linear short NC segments. Taking advantage of the designs of the look-ahead function and multi-threads, the NURBS interpolator can obtain enough NC block information and complete feedrate planning before interpolation. Consequently, the computational load is reduced in an interpolation period and unnecessary feedrate acceleration and deceleration can be avoided.

In addition, the continuous short blocks conforming to the CSB criterion can be fitted into NURBS curves in real-time. S-shaped federate profile for ACC/DEC planning can make the single block achieve C^1 continuity and jerk-limited capability. The simulation and experimental results show that the output contour profiles approach the input NC program models. This indicates that the proposed real-time look-ahead NURBS interpolator is able to perform satisfactorily.

6. REFERENCES

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

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

[3] Yeh, S. S.; Hsu, P. L.: The speed-controlled interpolator for machining parametric curves, Computer-Aided

Design, 31, 1999, 349-357.

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

[5] Tsai, M. C.; Cheng, C. W.; Cheng, M. Y.: A real-time NURBS surface interpolator for precision three-axis CNC machining, International Journal of Machine Tools and Manufacture, 43, 2003, 1217-1227.

[6] Tikhon, M.; Ko, T. J.; Lee, S. H.; Kim, H. S.: NURBS interpolator for constant material removal rate in open NC machine tools, International Journal of Machine Tools and Manufacture, 44, 2004, 237-245.

[7] Yau, H. T.; Wang, J. B.; Chen, W. C.: Development and Implementation for Real-Time Look-Ahead Interpolator by using Bezier Curve to Fit CNC Continuous Short Block, Proceedings of the 2005 IEEE International Conference on Mechatronics, Taipei, Taiwan, 2005, 78-83.

[8] Yong, T.; Narayanaswami, R.: A parametric interpolator with confined chord errors, acceleration and deceleration for NC machining, Computer-Aided Design, 35, 2003, 1249-1259.

[9] Liu, X.; Ahmad, F.; Yamaaki, K.; Mori, M.: Adaptive interpolation scheme for NURBS curves with the integration of machining dynamics, International Journal of Machine Tools and Manufacture, 45, 2005, 433-444.

[10] Nam, S. H.; Yang, M. Y.: A study on a generalized parametric interpolator with real-time jerk-limited acceleration, Computer-Aided Design, 36, 2004, 27-36.

[11] Erkorkmaz, K.; Altintas, Y.: High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation, International Journal of Machine Tools and Manufacture, 41, 2001, 1323-1345.

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

[13] Han, G. C.; Kim, D. I.; Kim, H. G.; Nam, K.; Choi, B. K.; Kim, S. K.: A high speed machining algorithm for CNC machine tools," IECON Proceedings (Industrial Electronics Conference), 1999, 1493-1497.

[14] Yau, H. T.; Kuo, M. J.: NURBS machining and feed rate adjustment for high-speed cutting of complex sculptured surfaces, International Journal of Production Research, 39(1), 2001, 21-41.

[15] Farouki, R. T.; Manjunathaiah, J.; Nicholas, D.; Yuan, G. F.; Jee, S.: Variable-feedrate CNC interpolators for constant material removal rates along Pythagorean-hodograph curves, Computer-Aided Design, 30(8), 631-640, 1998.

[16] Piegl, L.; Tiller, W.: The NURBS Books, Berlin Heidelberg: Springer, 1997.