15th Triennial World Congress, Barcelona, Spain

REAL-TIME VARIABLE FEEDRATE PARAMETRIC INTERPOLATOR FOR CNC MACHINING

Chung-Wei Cheng[*], Mi-Ching Tsai[*] and Ming-Yang Cheng[**]

[*]*Department of Mechanical Engineering, National Cheng Kung University, Tainan 701, Taiwan*
[**]*Department of Electrical Engineering, National Cheng Kung University, Tainan 701, Taiwan*

Abstract: This study explores the feasibility of variable feedrate parametric interpolators for real-time motion command generation using a digital signal processor (DSP). Most of the existing parametric interpolators are developed for a constant federate due to the computational complexity. However in certain applications, in order reduce the deterioration in accuracy of machining, feedrate commands should be performed in advance to overcome excessive acceleration/deceleration of CNC machines. The aim of this study is to develop a real-time parametric interpolator with variable federate where acceleration/deceleration planning on the feedrate command is performed before the interpolation takes place. This approach that would provide new capabilities for advanced CAD/CAM systems, can compensate for the path command error caused by the acceleration/deceleration after interpolation. To justify the performance of the real-time control interpolator, an X-Y table driven by two servomotors is controlled to track a parametric curve represented in non-uniform rational B-Spline (NURBS) form.

Keywords: CAD/CAM; digital signal processor; parametric interpolator.

1. INTRODUCTION

In modern commercial CAD and systems, a wide variety of part shapes for dies and molds are usually represented in parametric curves (or surfaces) like the Bezier curve, B-spline, and NURBS. However, conventional CNC machines only provide line or circular interpolators. In order to perform the machining of molds, the shapes and desired tool paths designed using CAD/CAM systems are typically approximated with very small line or circular segments. Such approximation approaches inherit several disadvantages: (1) for accurate machining the data file size is usually quite large so that the data transmission load is substantially increased; (2) will lead to velocity and acceleration discontinuity at the junction of line segments and may cause a shock or variation in mechanical system. These drawbacks suggest that the conventional interpolators fail to meet the requirements of high-speed machining in modern industry. Therefore the need to develop new parametric interpolators for CNC machines cannot be overlooked.

The machining method used for parametric curves (or surfaces) is illustrated in Fig. 1 (Koren et al., 1993) where parametric interpolator converts the parametric curve (or surface) segments designed in CAD systems to each axis's servo command of CNC machines. Based on first- and second- order Taylor's expansion or Runge-Kutta method, several investigators have proposed different kinds of parametric interpolators to generate servo commands for applications in constant feedrate control problems (Koren et al., 1993; Shpitalni et al., 1994; Wang et al., 1998; Kuo et al., 2000). Nevertheless, the capability of performing variable feedrate control is invaluable in high-speed machining applications and deserves more study. Huang and Yang (1992) developed a cubic spline interpolator by using the first-order Euler approximation method to study the variable federate control problem. Their method yielded satisfactory performances as long as the curvature of the curve for machining was small. Yang and Kong (1994) proposed a parametric interpolator for variable feedrate based on the first- and second-order Taylor's expansion. This is a comparative study

of linear and parametric interpolators on several aspects such as memory size, feedrate fluctuation and CPU time is presented. Zhang and Greenway (1998) developed a NURBS curve interpolator for the motion controller of a manipulator by using the first-order Taylor's expansion, where as Yeh and Hsu (1999) proposed a speed-controlled interpolator for machining parametric curves. Farouki and Tsai (2001) have recently pointed out some problems with the Yang (1994) and Yeh (1999) interpolators, when formulating the second-order Taylor's expansion interpolator for variable federate. Then they obtained exact Taylor series coefficients using the Pythagorean-Hodograph (PH) curves that have the closed-form analytic reductions of the interpolation integral.

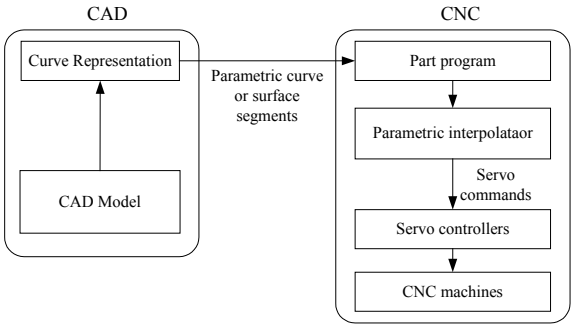


Fig. 1. Machining method used for parametric curve or surface segments

This paper presents numerical algorithms for realizing parametric interpolators that consider acceleration/deceleration planning on the desired velocity profile to achieve variable feedrate control. A test platform driven by two servomotors is used to evaluate the performance of the proposed parametric interpolator with variable feedrate in real-time, where the interpolation algorithms and servo controllers are implemented using a DSP. The powerful computation ability of the DSP makes it possible to eliminate the gap between theory and practice, which may provide an attractive cost-effective basis for modern control systems (Masten et al., 1997). In addition, the proposed interpolator is also applied to generate NURBS commands in several variable feedrate machining tasks.

The remainder of the paper is organized as follows. Numerical algorithms for the variable feedrate parametric interpolators are developed in Section 2. In Section 3, NURBS representation for a parametric curve is briefly introduced. The motion control structure used in this study is given in Section 4. Experimental results are presented in Section 5, and conclusions are given in Section 6.

2. VARIABLE FEEDRATE PARAMETRIC INTERPOLATORS

The general form of a parametric curve in 3-D space can be described as

$$P(u) = x(u)\vec{i} + y(u)\vec{j} + z(u)\vec{k} \qquad u \in [0,1] \tag{1}$$

where u is a dummy variable used in the spatial parameter of the curve. In general, the feedrate along a curve $P(u)$ is defined as

$$V(t) = \left\| \frac{dP(u)}{dt} \right\| \tag{2}$$

where $\|\cdot\|$ denotes the Euclidean norm in the 3-D space. Since $u(t)$ is a strictly monotone increasing function with respect to t and also by the chain rule, one can obtain

$$V(t) = \left\| \frac{dP(u)}{dt} \right\| = \left\| \frac{dP(u)}{du}\frac{du}{dt} \right\| = \left\| \frac{dP(u)}{du} \right\| \frac{du}{dt} \tag{3}$$

and therefore

$$\frac{du}{dt} = \frac{V(t)}{\left\| \frac{dP(u)}{du} \right\|} \tag{4}$$

Furthermore, it can be proved that the second derivative of u is given by

$$\frac{d^2u}{dt^2} = \frac{\frac{dV(t)}{dt}}{\left\| \frac{dP(u)}{du} \right\|} - \frac{V^2(t)\left(\frac{dP(u)}{du} \bullet \frac{d^2P(u)}{du^2} \right)}{\left\| \frac{dP(u)}{du} \right\|^4} \tag{5}$$

As can seen in Eq. (3), the feedrate $V(t)$ is a function of the dummy variable u. An important question is raised as how to compute appropriate values of $u(t)$ numerically to generate appropriate motion commands that yield desired feedrate along a parametric curve. To be qualified as a real-time interpolator for parametric curves, a computationally efficient solution of Eq. (4) is necessary. There are several existing algorithms that can be applied to solve du/dt numerically. One of the well known methods is based on the Taylor's expansion, in which its second order approximation of du/dt at the time instant of $t_k = kT$, is given by

$$u_{k+1} \approx u_k + T\,\dot{u}_k + \frac{T^2}{2}\ddot{u}_k \tag{6}$$

where $u_k = u(t_k)$ denotes the value of u at time t_k and the first and second derivatives of u, defined in Eqs. (4) and (5), at t_k are given by, respectively,

$$\dot{u}_k = \left.\frac{du}{dt}\right|_{t=t_k} \quad \text{and} \quad \ddot{u}_k = \left.\frac{d^2u}{dt^2}\right|_{t=t_k} \tag{7}$$

According to Eqs. (4)-(7), the first-order Taylor's expansion interpolators for variable feedrate to generate u_{k+1} is expressed as

$$u_{k+1} = u_k + \frac{T\,V_k}{\left\| \frac{dP(u)}{du} \right\|_{u=u_k}} \tag{8}$$

and the second-order Taylor's expansion interpolator is expressed as

$$u_{k+1} = u_k + \frac{T V_k}{\left\| \frac{dP(u)}{du} \right\|_{u=u_k}} + \frac{T^2}{2}\left(\frac{A_k}{\left\| \frac{dP(u)}{du} \right\|_{u=u_k}} - \frac{V_k^2 \left(\frac{dP(u)}{du} \bullet \frac{d^2P(u)}{du^2} \right)\Big|_{u=u_k}}{\left\| \frac{dP(u)}{du} \right\|^4_{u=u_k}} \right) \tag{9}$$

where $V_k = V(t_k)$ and $A_k = \left.\frac{dV(t)}{dt}\right|_{t=t_k}$.

Substituting u_{k+1}, obtained from Eqs. (8)~(9), into Eq. (1) will yield the next reference servo command $P(u_{k+1})$ for the servo controller at sampling time t_{k+1}.

3. NURBS REPRESENTATION FOR PARAMETRIC CURVE

Consider the NURBS represented parametrically by the following equations (Piegl, L. 1991)

$$P(u) = \frac{\sum_{i=0}^{n} N_{i,k}(u) W_i V_i}{\sum_{i=0}^{n} N_{i,k}(u) W_i} = \sum_{i=0}^{n} V_i R_{i,k}(u) \tag{10}$$

and

$$R_{i,k}(u) = \frac{N_{i,k}(u) W_i}{\sum_{i=0}^{n} N_{i,k}(u) W_i} \tag{11}$$

where V_i represents the control points, W_i is the weight vector, and *k* is the order of NURBS. $N_{i,k}(u)$ is called the blending function, which is also called the basis function, and $R_{i,k}(u)$ the single rational B-spline. From Eq. (10), the m[th] derivative of the $P(u)$ can be obtained as

$$P^{(m)}(u) = \sum_{i=0}^{n} V_i R_{i,k}^{(m)}(u) \tag{12}$$

where $R_{i,k}^{(m)}(u)$ represents the m[th] derivative of $R_{i,k}(u)$.

Equations (13) and (14) provide the recursive formulas of computing $N_{i,k}(u)$.

$$N_{i,1}(u) = \begin{cases} 1 & \text{for} \quad u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

$$N_{i,k}(u) = \frac{u - u_i}{u_{i+k-1} - u_i} N_{i,k-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_{i+1}} N_{i+1,k-1}(u) \tag{14}$$

where $[u_i, \ \ldots, \ u_{i+k}]$ represents the knot vector. To implement the first- and second-order Taylor's expansion interpolators, it is necessary to calculate the first and second derivatives of its NURBS representation. Eq. (15) and (16) show the 1[st] and 2[nd] derivatives of the rational B-spline:

$$R_{i,k}^{(1)}(u) = \frac{W_i N_{i,k}^{(1)}(u)}{\sum_{i=0}^{n} W_i N_{i,k}(u)} - \frac{W_i N_{i,k}(u) \sum_{i=0}^{n} W_i N_{i,k}^{(1)}(u)}{\left[\sum_{i=0}^{n} W_i N_{i,k}(u) \right]^2} \tag{15}$$

$$R_{i,k}^{(2)}(u) = \frac{W_i N_{i,k}^{(2)}(u)}{\sum_{i=0}^{n} W_i N_{i,k}(u)} - \frac{\left[2W_i N_{i,k}^{(1)}(u) \sum_{i=0}^{n} W_i N_{i,k}^{(1)}(u) + W_i N_{i,k}(u) \sum_{i=0}^{n} W_i N_{i,k}^{(2)}(u) \right]}{\left[\sum_{i=0}^{n} W_i N_{i,k}(u) \right]^2} + \frac{2W_i N_{i,k}(u) \left[\sum_{i=0}^{n} W_i N_{i,k}^{(1)}(u) \right]^2}{\left[\sum_{i=0}^{n} W_i N_{i,k}(u) \right]^3} \tag{16}$$

Substituting Eq. (15) and Eq. (16) into Eq. (12) will yield the 1[st] and 2[nd] derivatives of the NURBS respectively. Nevertheless, to compute the derivatives of rational B-spline, the derivatives of blending function $N_{i,k}(u)$ should be calculated first. The reduced order form of the m[th] derivative of blending function $N_{i,k}(u)$ is expressed as

$$N_{i,k}^{(m)}(u) = (k-1)\left[\frac{N_{i,k-1}^{(m-1)}(u)}{u_{i+k-1} - u_i} - \frac{N_{i+1,k-1}^{(m-1)}(u)}{u_{i+k} - u_{i+1}} \right] \tag{17}$$

4. MOTION CONTROL STRUCTURE

The overall motion control structure of the experimental system is shown in Fig. 2, where the PMC32-6000 motion control card, equipped with a high performance TI TMS320C32 DSP, is used to perform real-time computations. These include acceleration/deceleration planning, real-time variable-feedrate parametric interpolator and servo controller. All of interpolation algorithms and servo controller are implemented in C-language and executed using DSP. An X-Y table shown in Fig. 3 is used for experimental study where the table is driven by two servomotors with built-in incremental encoders (2500x4 pulses/rev) for position feedback.

4.1 Acceleration/deceleration Planning

In order to reduce the deterioration in accuracy resulting from excessive acceleration/deceleration of CNC machines, specific acceleration/deceleration planning on desired velocity profile is used to obtain smooth motion. In this study, velocity profiles as shown in Fig. 4 will be used as velocity profiles to generate desired variable feedrate V_k and acceleration A_k at each time instant t_k. Thus, substituting V_k and A_k into Eqs. (8) and (9), parametric interpolator for variable feedrate control can be achieved. In addition, in this study the acceleration/ deceleration planning on the feedrate command is performed before the interpolation. This approach can effectively

compensate the path command error caused by the way that the "acceleration/deceleration" is made after interpolation (Kim et al., 1994).

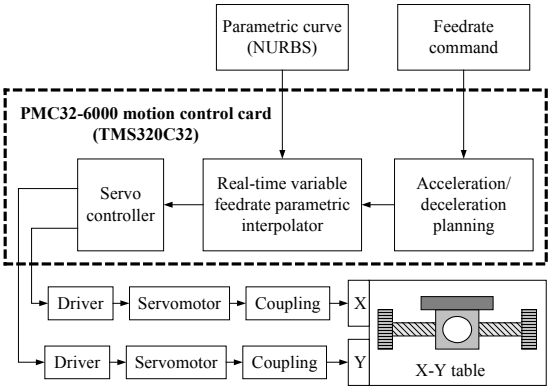


Fig. 2. Overall of motion control structure

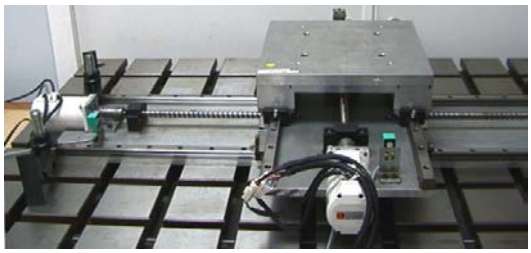


Fig. 3. The experimental system

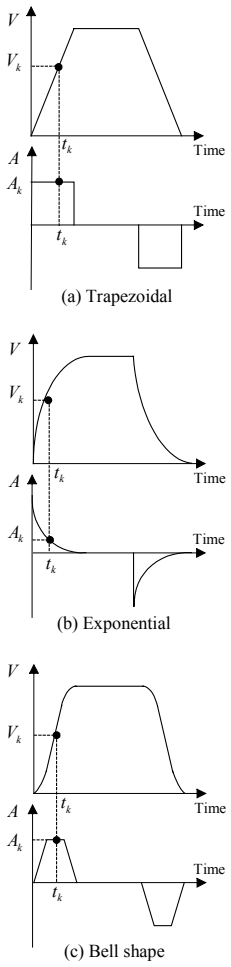


Fig. 4. Different velocity profiles for variable feedrate parametric interpolator

4.2 Servo Controller

The block diagram of the servo controller used in this experimental system is shown in Fig. 5, where Xc , Yc are the desired position commands; Xa , Ya are the actual positions and Kpx , Kpy are the position gains. In addition, Kvx , Kvy are the feed-forward coefficients, which can reduce the tracking errors. Note that position controllers for both X and Y–axis are only simple proportional gains with a feed-forward path (FAUNC, 1991).

Transfer functions $G_X(s)$ and $G_Y(s)$ of the controlled X-Y table are obtained by measuring the frequency responses from the power drive input to the motor shaft velocity. The transfer functions obtained from the control system analyzer HP3563A are

$$G_X(s) = \frac{112570.543}{s^2 + 601.677s + 113727.10} \quad \text{(mm/s-voltage)}$$

$$G_Y(s) = \frac{319514.874}{s^2 + 1649.133s + 342115.44} \quad \text{(mm/s-voltage)}$$

With specifying desired damping ratio and natural frequency, the gain constants of the servo controller can be determined using MATLAB/SIMULINK such as $Kpx = 40$, $Kpy = 70$ and $Kvx = Kvy = 0.6$ for the desired feedrate of 100 mm/s.

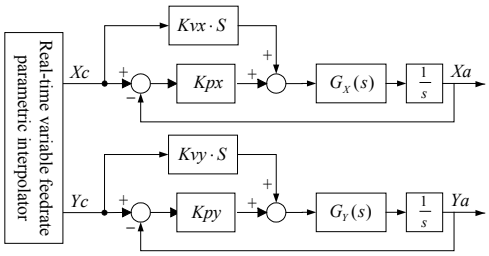


Fig. 5. Block diagram of servo controller

5. EXPERIMENTAL RESULTS

An illustrative example is given in the following to evaluate the performance of the proposed real-time variable feedrate parametric interpolator. The tasks for evaluation include: (1) computational efficiency analysis; (2) variable feedrate control for acceleration/deceleration planning. In this illustrated example the NURBS curve to represent a circle with radius 50 mm is shown in Fig.6, where the associated control points, weight vector and knot vector are assigned as below:

The control points: (0, 0), (0, 50), (100, 50), (100, 0), (100, -50), (0, -50), (0, 0) (mm)

The associated weight vector is [1.0, 0.5, 0.5, 1.0, 0.5, 0.5, 1.0]

The associated knot vector is [0, 0, 0, 0.25, 0.5, 0.5, 0.75, 1, 1, 1]

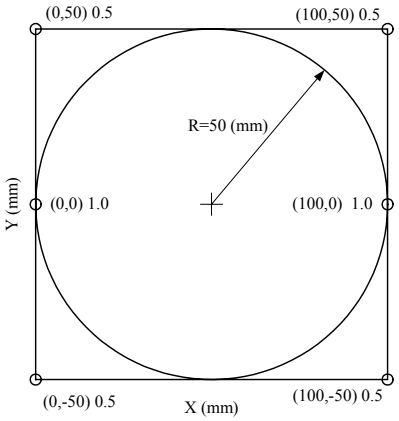


Fig. 6. The NURBS curve with control points

5.1 Computational Efficiency Analysis

Since parametric interpolators should be implemented in a real-time environment, the issue of computational efficiency is of great concern in this study. In the efficiency analysis experiments, the desired feedrate is set to 100 mm/s and the acceleration/deceleration time for acceleration/deceleration planning is set to 0.5 s. Table 1 lists the total computation time for both (1) parametric interpolator with different acceleration /deceleration planning and (2) servo controller. According to experimental results, one can find that the first-order parametric interpolator can be implemented in real-time for the sampling period T=1ms. However, the second-order parametric interpolator is only suitable for T=2 ms.

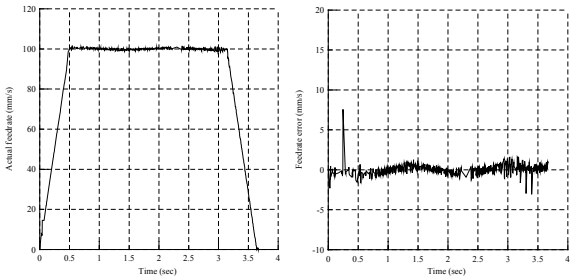
Table 1. Computation time for real-time control (μs)

Parametric interpolator	Acceleration/ deceleration planning type	Computation time	Servo controller
First-order	Trapezoidal	540	45
	Exponential	580	45
	Bell shape	600	45
Second-order	Trapezoidal	1020	45
	Exponential	1040	45
	Bell shape	1140	45

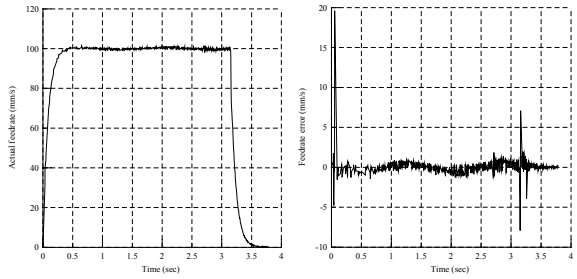
5.2 Variable Feedrate Control for Acceleration/deceleration Planning

To perform acceleration/deceleration motion planning, the length of the parametric curve has to be calculated first. Then the total motion time corresponds to the desired feedrate and acceleration (deceleration) is computed. Consider the NURBS circle shown in Fig. 6. The total length of the circle is $2\pi R$ = 314.1593 mm, where R=50 is the radius. Given the desired feedrate of 100 mm/s and acceleration (deceleration) time of 0.5s, then the total motion time is around 3.6416s. Several experiments based on different acceleration/deceleration planning have been performed using the first-order and second-order interpolators respectively. However, due to the paper length limit, only the parts related to the second-order interpolator are presented here.

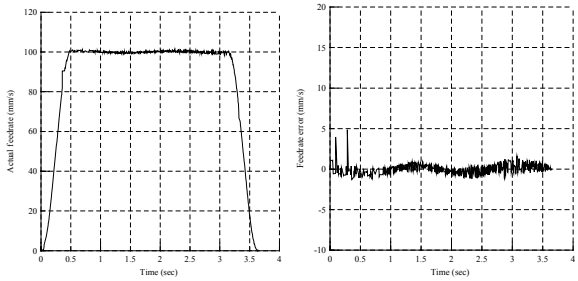
Figure 7 shows the actual feedrate and feedrate error along the parametric curve for three different acceleration/deceleration planning. It can be seen that the largest feedrate error (see Fig.7 (b)) is caused by the exponential acceleration/deceleration and the smallest feedrate error (see Fig.7 (c)) is by the bell shape acceleration/deceleration. Measured results of position tracking for using trapezoidal acceleration/deceleration planning for the cases before and after interpolation are illustrated in Figure 8. As can be seen, the contouring error for the approach of acceleration/deceleration planning method before interpolation is smaller than that done by the approach using acceleration/deceleration planning method after interpolation. These experimental results have revealed that the proposed approach is able to effectively achieve the X-Y motion with variable feedrate control.



(a) Trapezoidal acceleration/deceleration



(b) Exponential acceleration/deceleration



(c) Bell shape acceleration/deceleration

Fig. 7. Experimental results for acceleration/deceleration planning

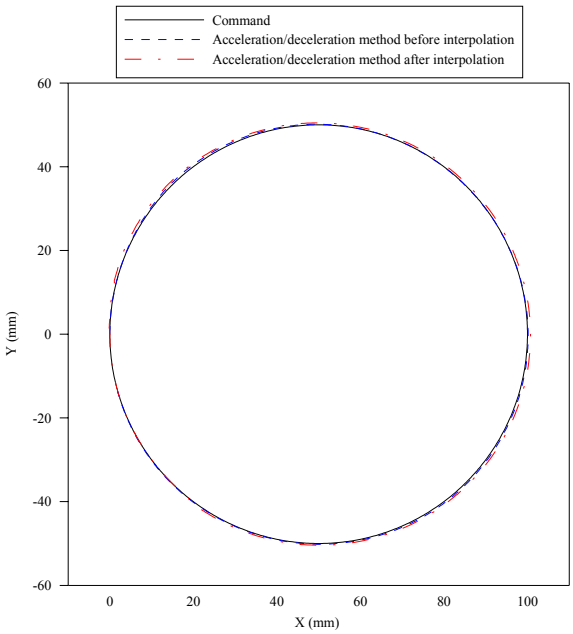


Fig. 8. Experimental results for position tracking

6. CONCLUSIONS

In this study, a new interpolator based on variable feedrate parametric curves such as NURBS has developed to produce servo commands for real-time control CNC machining. Unlike most of existing parametric interpolators based on constant feedrate, the proposed interpolator is capable of generating motion commands for servo controllers to achieve the desired variable feedrate. Experimental results obtained from real-time DSP implementation have shown that the proposed approach is promising when dealing with the variable feedrate control problem. Also it should be noted that the contour error resulting from using the acceleration/deceleration method before interpolation is smaller than when using it after interpolation.

ACKNOWLEDGEMENTS

The author is grateful to the National Science Council of the Republic of China for supporting this research under the grant NSC89-2212-E-006-110.

REFERENCES

FANUC Corp. (1991). *FANUC AC Servo Amplifier-Maintenance Manual* (6th ed.).

Farouki, R.T., and Tsai, Y.-F. (2001). Exact Taylor series coefficients for variable-feedrate CNC curve interpolators. *Computer Aided Design*, **33**, 155-165.

Huang, J.T. and Yang, D.C.H. (1992). A generalized interpolator for command generation of parametric curves in computer-controlled machines. *ASME JAPAN/USA Symposium on Flexible Automation* ,393-399.

Kim, D.II., Song, J.II., and Kim, S. (1994). Dependence of machining accuracy on acceleration/deceleration and interpolation methods in CNC machine tools. *IEEE Industry Applications Society Annual Meeting*, 1898-1905.

Koren, Y., Lo, C.C. and Shpitalni, M. (1993). CNC interpolators: algorithms and analysis. *ASME Journal of Manufacturing Science and Engineering*, **64**, 83-92.

Kuo, J.C., Tsai, M.C. and Cheng, M.Y. (2000). Real-time NURBS interpolator for precision command generation. *The Sixth International Conference on Automation Technology*, Taipei, Taiwan, 503-508.

Masten, M. K. and Panahi, I. (1997). Digital signal processors for modern control systems. *Control Engineering Practice*, **5(4)**, 449-458.

Math Works (1993). *MATLAB High-Performance Numeric Computation and Visualization Software*, Math Works Inc.

Mortenson, M.E. (1997). *Geometric Modeling*. John Wiley & Sons, Inc., USA.

Nakamura, S. (1993). *Applied Numerical Methods in C*. Prentice-Hall International, Inc., USA.

Piegl, L. (1991). On NURBS: A Survey. I*EEE Computer Graphics & Application*, **11**, 55-71.

Shpitalni, M., Koren, Y. and Lo, C.C. (1994). Realtime curve interpolators. *Computer Aided Design*, **26**, 832-838.

TEXAS Instruments (1991). *TMS320C3x User's Guide*, TEXAS Instruments Inc.

Wang, F.C. and Wright, P.K. (1998). Open architecture controllers for machine tools, Part 2: a real time quintic spline interpolator. *ASME Journal of Manufacturing Science and Engineering*, **120**, 425-432.

Yang, D.C.H. and Kong, T. (1994). Parametric interpolator versus linear interpolator for precision CNC machining. *Computer Aided Design*, **26**, 225-234.

Yeh, S.-S., and Hsu, P.-L. (1999). The speed-controlled interpolator for machining parametric curves. *Computer Aided Design*, **31**, 349-357.

Zhang, Q.G., Greenway, R.B. (1998). Development and Implementation of a NURBS Curve Motion Interpolator. *Robotics and Computer-Integrated Manufacturing*, **14**, 27-36.