Parametric NURBS Curve Interpolators : A Review

Sekar Mohan[1], Sung-Hwan Kweon[1], Dong-Mok Lee[1] and Seung-Han Yang[1,#]

1 School of Mechanical Engineering, Kyungpook National University, 1370 Sankyuk-dong, Buk-gu, Deagu, South Korea, 702-701
Corresponding Author / E-mail: syang@knu.ac.kr, TEL: +82-53-950-6569 FAX: +082-53-950-6550

KEYWORDS : NURBS, CNC, Free-form machining, Parametric curve Interpolator

Free-form shapes which were once considered as an aesthetic feature are now an important functional requirement. CNC industries are looking for a compact solution for reproducing free-form shapes as conventional interpolation models are inadequate. The parametric curve interpolator developed in the last decade has clearly emerged as favorite among its contemporaries in recent years. At present intense research has been done on parametric curve interpolators and interesting developments are reported. Out of the various parametric representations for curves and surfaces, NURBS has been standardized and widely used in free-form shape design. This paper presents a review of various methods of parametric interpolation for NURBS and discusses the salient features, problems and solutions. Recent approaches on variable feedrate interpolation, parameter compensation are also reviewed and research trends are addressed finally.

Manuscript received: February 11, 2008 / Accepted: March 14, 2008

1. Introduction

The introduction of computers to control the machine tool has significantly improved automation of the machining process. The availability of low cost microprocessors with increased computing power and capabilities has led to major advancements in the design and implementation of Computer Numerical Control (CNC) systems. Nowadays major portion of processing the part data, interpolation and servo control are handled through software components. CNC interpolators have also witnessed significant advances. In contrast to the Numerical Control (NC) hardwired interpolators which were built on multiple Digital Differential Analyzers, CNC interpolators are implemented as software. The main function of the interpolator is to generate position commands for reproduction of a prescribed shape of the part with minimum errors. Modern CNC interpolators also have subroutines, macros, special cycles and look-ahead features. In addition to linear and circular interpolators, parabolic and elliptical interpolators are also developed for specialized processes like laser beam cutting, etc.[1,2]

Developments in CAD industry witnessed a very rapid pace of growth due to sophisticated and better mathematical models and computation facilities that can exactly describe the free-form shapes. Free-form shapes once used for aesthetics have become an essential functional requirement in modern product design. Although machining of complex free-form shapes became a reality with conventional interpolation methods, accurate reproduction of these could not achieved, since they are based only on standard motions: linear and circular.

This necessity motivated researchers to explore new methods for interpolation. One such interpolator which has received considerable attention over the last decade is the parametric interpolator. Long usage of parametric curves to represent the geometry in CAD systems is an important reason behind this development. It was at this stage, that, most of the CAD free-form shape design widely followed Non-Uniform Rational B-Spline (NURBS) representation. In recent years research on NURBS based parametric interpolators has intensified and many new algorithms and methods are proposed. This paper presents a discussion on problems with existing interpolation methods to appreciate the need of parametric interpolators. This review later focuses on the recent developments from the perspective of parametric NURBS curve interpolators. Accomplishments relevant to the above are also mentioned.

2. An Introduction to NURBS

In CAD/CAM, parametric curves are preferred over implicit representation due to their extension to three dimensional spaces and easier implementation in computer. Parametric representation is suitable to readily obtain cutter offsets for CNC and also lends itself to piecewise description of curves and surfaces.[3] Although these features enabled description of intricate shapes in CAD, a unified representation was not made until NURBS was proposed. NURBS provided this possibility because of its excellent features and adherence to existing CAD standards. NURBS permits manipulation of weight and control points thereby offering ease of design of both standard analytical shapes (e.g. conics) and free-form curves.[4] It is also independent of coordinate system which makes it suitable for easier and faster transformations. Closed shapes, large slopes, local control and geometric intuitiveness are some of the other advantages. Hence CAD/CAM had no hesitation in choosing NURBS as the common representation of parametric curves. However the NC technology originated well ahead of it and could not implement it straightforward. NURBS remained long as a fancy till the development of curve interpolators.

A NURBS curve $C(u)$ is a vector valued piecewise rational polynomial function of the form

$$C(u) = \sum_{i=0}^{n} R_{i,p}(u)P_i \quad (1)$$

$$\text{where } R_{i,p}(u) = \frac{w_i N_{i,p}(u)}{\sum_{j=0}^{n} N_{j,p}(u)w_j} \quad (2)$$

The B-spline basis function of degree p is defined recursively as

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \le u \le u_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (4)$$

where P_i are the control points forming a control polygon, w_i are the weights and $R_{i,p}(u)$ are the p^{th} degree rational basis functions defined over a non-uniform knot vector $u \in [0\ 1]$ with u as non-dimensional curve parameter. Degree p, number of knots $m+1$ and number of control points n can be related by

$$m = n + p + 1 \quad (5)$$

With another parameter v, Eq. 1 can easily be extended to define a NURBS surface. Piegl[5] and Rogers[6] give excellent introduction on the subject of NURBS and may be referred for more details.

3. Interpolation of Free Form Shapes Designed with NURBS

In this paper we presume that free-form curves and surfaces in CAD are designed with parametric NURBS though other forms such as explicit and implicit non-parametric forms of representation are purely admissible. In addition we assume that the NURBS tool path is available after performing necessary offsetting calculations. The strategies to generate position commands for NURBS can be classified into 1) approximation methods, 2) incremental methods and 3) direct curve interpolation methods.

3.1 Approximating with Linear and Circular Segments

In the approximation method, the complex CAD geometry is discretized into simpler entities offline and the corresponding G-codes are used for real-time interpolation. A more detailed description can be found in the report by Jensen *et al.*[7] A CAM program is usually engaged to automatically generate part programs after performing necessary offsets. The resulting part program consists of several lines of G codes or APT statements that correspond to consecutive linear segments or a combination of both.[8] Fig. 1 shows this approach. Understandably the number of segments must be maximized for better approximation and surface finish. On the other hand increase in number of segments will lead to the problems mentioned below.

3.1.1 Constraints on Resources

The stringent tolerance for precision machining results in a voluminous part program which usually is not possible to execute from memory of the CNC controller and hence is split into smaller part programs. This results in interruption of machining process and increase in machining time. Nowadays, a powerful CAM program executed from a remote computer is employed to continuously transfer the commands to the CNC controller. But this requires reliable communication facilities between the CAM and CNC I/O systems to avoid data starvation, data noise and data loss.

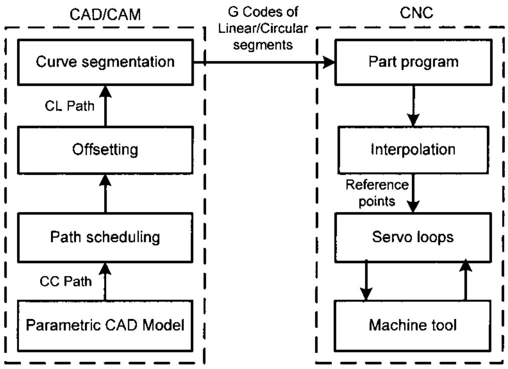

Fig. 1 Conventional approach to machine NURBS curve

3.1.2 Non-utilization of Maximum Feedrate

In a typical CNC machine, the G codes are executed in the controller in a sequential manner. Each block of code has its own inherent acceleration and deceleration phases consuming much of the machining time. In addition to overall increase of machining time, the prescribed feedrate V_c may never be reached and the feedrate fluctuates between the blocks as shown in Fig. 2. These effects are more pronounced with the decrease of segment length.

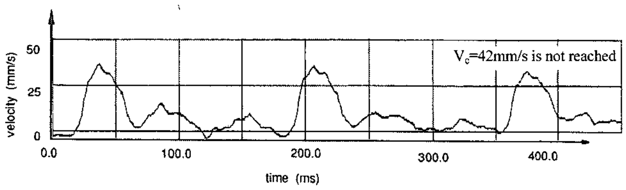

Fig. 2 Fluctuating feedrate due to segmentation[8]

3.1.3 Fluctuation of Feedrate

The actual command generation process of conventional linear interpolator repeats at a sampling interval of ΔT s and the commands are transmitted to the CNC system as a sequence of reference pulses or binary word. During each cycle the tool position is incremented by $\Delta T \times V$ BLUs where BLU (Basic Length Unit) is the length the machine can move when a control pulse is given. Hence the resulting number of intermediate commands (N) must be an integer value, given by,[9,10]

$$N = int\left(\frac{l}{V\Delta T} + 0.5\right) \quad (6)$$

This cannot be achieved in all the cases and hence the last increment pertaining to a shorter length is also carried out in the same sampling time thus reducing the overall feedrate. Otherwise the interpolator adjusts the feedrate such that the length of the line segment is an integer multiple of modified feedrate V_c'. This effect is practically negligible as long as the length of the line segment is big, which is the case of normal interpolation. For shorter segments, the above approach results in fluctuation of feedrate and causes poor surface finish. In most cases additional hand work such as polishing is required.

3.2 Approximating with Polynomials

The next choice is to fit the discrete data of cutter paths with polynomials. Parametric form of polynomials is preferred due to the advantages stated above. Although it is possible to fit an n^{th} order polynomial with a dimensionless parameter through data points, a piecewise approach to approximate the entire curve satisfying the continuity conditions at the connecting points is widely followed.[11] Important ones are the cubic spline and quintic spline interpolation methods where the parameterization is done based on the chord length or arc length. Cubic polynomials could represent the shape well but they suffer from feedrate fluctuations and hence quintic polynomials parameterized with near arc length are used.[12] The continuity constraints of the composite curve consisting of n quintic segments are obtained from pre-fitting the cubic spline. The arc length is approximated using a nonlinear integral equation solved on a least-squares basis. In another work Chou *et al.*[11] succeeded in relating the time domain with curve parameter to produce a feedrate profile derived from kinematic constraints. These approaches have limited application because some parts of the segment may not participate in machining at certain feedrates. This can be overcome by adding more data points that improve arc-length parameterization.[13] In the method proposed by Erkorkmaz *et al.*[14] a series of reference points are connected by quintic splines while the step size is based on the total travel length. The acceleration and deceleration planned reference trajectory can be reconstructed at the servo sampling frequency in real time. Though the feedrate fluctuations are comparatively less, the above methods require considerable offline processing and involve higher order polynomials to describe intricate curves.

3.3 Incremental Methods

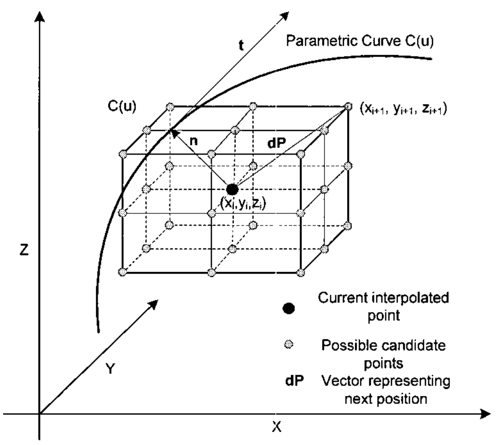

Fig. 3 Non-Orthogonal method searching for next best position[15]

In incremental methods, the next interpolation point is estimated from a set of nearby points such that it lies closest to the curve.[15] In this approach incremental moves of the machine tool are obtained from two conditions namely, direction and step size. The exact direction of the successive moves is chosen based on the tangent vectors. The movement can be done one axis at a time (orthogonal) or simultaneously (non-orthogonal). The non-orthogonal method (see Fig. 3) is preferred for machining NURBS. The closest point near the curve can be computed by imposing the other condition: the normal distance from the curve is a minimum ($n \bullet t = 0$). Integer programming method is used to solve the unknown steps in terms of BLU and the corresponding unknown parameter values are computed with Newton-Raphson method.[16,17]

In all these methods it is not guaranteed that the interpolated points lie on the originally designed curve. In case of 3D curve/surface interpolation, information of surface is required to avoid deviation of path and probable undercutting. Compared to the direct curve interpolator which follows this section, the algorithms are complex and require computation of arc-length for which no closed form solution is available except for lines.

4. Direct Parametric Curve Interpolation

Parametric curve interpolators (Fig. 4) proposed to overcome the disadvantages of conventional interpolators use Euler or Taylor's expansion to compute increments of parameter at subsequent time intervals.[9,10,18] Unlike the above interpolators direct curve interpolators generate real-time position commands that lie on the original curve. Since segmentation is not required additional resources of memory and communication bottlenecks can be avoided. A study done by Yang *et al.*[9] comparing the linear and parameter approaches shows that parametric interpolator behaves far better than its counterpart in terms of memory size requirement, feedrate fluctuation and position accuracy.

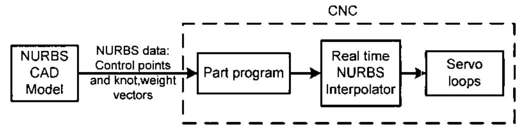

Fig. 4 Parametric approach does not require segmentation[10]

4.1 Derivation of Equations for Parametric Interpolator

In direct curve interpolation the successive parameter corresponding to the sampling period is evaluated using numerical integration. On substitution of the resulting parameter in curve equations such as Eq. 1, the coordinate points can be generated in real-time. The numerical approximation using Taylor's series is explained below.

Considering a parametric curve $C(u)$, the change of u in time domain is derived from the feedrate $V(u)$ as follows:

$$V(u) = \left\| \frac{dC(u)}{dt} \right\| \tag{7}$$

where ||.|| denotes the Euclidean norm in 3D space. Since $u(t)$ is a strictly monotonically increasing function, the feedrate upon application of chain rule becomes

$$\frac{du}{dt} = \frac{V(u)}{\left\| \frac{dC(u)}{du} \right\|} \tag{8}$$

$$\text{where } \left\| \frac{dC(u)}{dt} \right\| = \sqrt{\left[\left(\frac{dx}{du}\right)^2 + \left(\frac{dy}{du}\right)^2 + \left(\frac{dz}{du}\right)^2 \right]} \tag{9}$$

Since the derivative cannot be expressed in closed form, it is expanded using Taylor's series for time $t=kT_s$ where k denotes the sampling number.

$$u_{k+1} = u_k + T \frac{V(kt)}{\left\| \frac{dC(u)}{du} \right\|}\Bigg|_{u=u_k} + \frac{T^2}{2} \left[\frac{A(kt)}{\left\| \frac{dC(u)}{du} \right\|} - \frac{V^2(kt) \left[\frac{dC(u)}{du} \bullet \frac{d^2C(u)}{du^2} \right]}{\left\| \frac{dC(u)}{du} \right\|^4} \right]\Bigg|_{u=u_k} + HOT \tag{10}$$

For simplicity first order Taylor's' expression with a constant feedrate is widely used.

4.2 Problems with Parametric Approach

The parametric approach is not free of problems as it does not use the information about the intrinsic geometry of the curve. The approximation involved in Taylor's series truncation induces error during the evaluation of the successive parameter values. Although these errors are not cumulative, they cause feedrate error which is more pronounced when a small curve tolerance and sharp corners are negotiated. The feedrate error is defined as follows:

$$\varepsilon_{feed} = \frac{\frac{\Delta s}{\Delta t} - V_c}{V_c} \tag{11}$$

where V_c is the commanded feedrate and Δs is the actual distance traveled. Δs can be computed from the parametric values obtained from Taylor's series (Eq. 10).

Although feedrate errors are of important concern and researched to a great extent as explained later, the feedrate errors are less problematic compared to conventional linear interpolators.[9] Inclusion of higher order terms of Taylor's equation can significantly reduce the feedrate errors albeit at the cost of additional computational load.

The parametric approach fails when the curve derivatives are zero. This is a very rare case and arises in case of cusps which will not occur if the curve is a polynomial in terms of non-dimensional parameter.[10] When multiple curves are interpolated exact end may not be reached. This will cause similar problems posed by the linear interpolator, though the impact is very little.

5. Types of Parametric NURBS Interpolators

NURBS representation is fundamentally different from the parametric polynomials mentioned in section 3.2 though they can approximately represent a NURBS curve. NURBS uses the concept of weighed sum of the control points determined by the combination of basis functions. Free-form shapes designed earlier were composed of Bezier and B-Splines. In one of the pioneering work, Sata *et al.*[19] proposed an incremental procedure for interpolating Bezier curves using an approximating step function that can be performed with addition and bitwise operations. Another approach proposed by Hermann[20] used recursive subdivision algorithm to approximate the B-Spline curve piece-wise. However better approaches are available now.

NURBS parametric interpolators can be broadly classified into

1) Uniform increment
2) Constant feedrate
3) Variable feedrate interpolators

The uniform approach is the simplest of the above interpolation methods but not practicable. Though constant feedrate interpolator is widely preferred, new designs of variable feedrate interpolators are also favored.

5.1 Uniform Interpolators

The straightforward approach to implement a parametric interpolator for NURBS is to uniformly increase u . ($u_{k+1} = u_k + \Delta u$ where Δu is a constant increment to curve parameter u). Despite the simplicity of implementation of this method, it is difficult to obtain the optimal increment size. To overcome this problem a small value of Δu is preferred. Obviously this small value will increase the machining time. In addition, feedrate fluctuations are bound to happen as the uniform step size does not correspond to uniform length of travel in Cartesian space. The fluctuations ultimately decrease the surface finish of the machined part. Bedi *et al.*[21,22] used uniform increment to parameter for interpolating B-splines and suggested to position the control points of the curve in uniform distances to reduce fluctuations. But this method imposes severe restriction on the part of CAD designer and requires re-parameterization of the original curve.

5.2 Constant Feedrate Interpolators

Constant feedrate interpolators employ Taylor's series to compute successive parametric values according to Eq. 10 or its simplified form: first order Taylor's series. The constant feedrate must be selected such that it does not result in chord error deviation along the entire path. Cheng *et al.* [23] compared real time implementation of NURBS interpolators employing various methods such as first order, second order, R-K method etc., and suggested first order (execution time:1ms) for faster real-time applications and second order (execution time:2ms) for reduced feedrate error and accurate positioning. Zhang *et al.*[24] implemented a real time NURBS interpolator and used a look-up table for the calculation of derivatives by way of reduction in computation. It is also possible to reduce the computational effort by storing adjacent basis functions for calculating the higher order terms.[25] Extending these ideas real time surface interpolators for parametric and NURBS curves were developed.[26,27] The authors used iso-parametric curves where one of the parameter directions u or v is kept constant while parametric curve interpolation is carried on the other.

5.2.1 Trajectory Planning for Constant Feedrate Interpolators

Trajectory planning involves ACC/DEC planning for start and stop motions. G-Codes used in conventional interpolation have these features built-in but these must be planned explicitly for NURBS interpolation. For constant feedrate interpolators which start with the commanded feedrate, the machine stages can be directed to reach the starting point of the tool path from another point with the commanded feedrate. However it requires additional codes and careful planning. Hence it is ideal to plan the acceleration and deceleration phases during the interpolation process itself.

Different types of profiles for start and stop motions can be planned. A trapezoidal feedrate velocity profile is used by Zhang[24] where the time for acceleration and deceleration are computed by approximating the distance of the NURBS curve. A known profile of feedrate allows easier calculation of feedrate at every sampling travel and the same can be used to find successive parametric increments. If the curve length is insufficient to accommodate trapezoidal profile, a triangular profile can be followed. Alternative profiles such as bell and exponential feedrate profiles are also demonstrated.[25]

5.3 Curvature Dependent Interpolators

Curvature dependent interpolators also use the Taylor's series expansion. Instead of constant feedrate, a variable feedrate depending on the instantaneous curvature of the NURBS curve is used. In one such method[28] a linear relationship between the ratio of curvature corresponding to half the maximum commanded speed and the instantaneous current curvature is assumed. The commanded feedrate is followed at the flatter curve regions. Substituting the variable feedrate in the Taylor's expansion one can find the parameter increments at successive time intervals. But this method has no explicit control over chord error since the assumption does not take this into account. Chord error must be considered as it determines the size of the tolerance band and surface finish of the machined surface. To achieve this objective, the feedrate is adaptively adjusted so that the chord error does not exceed the allowable maximum value. In the adaptive interpolation method[29] the feedrate is expressed as a function of chord error and instantaneous curvature of the NURBS curve. The relationship between feedrate $V(u_k)$ and chord error is given by the following expression

$$V(u_k) = \frac{2}{T_s}\sqrt{\rho_k^{\,2} - (\rho_k - \delta_{max})^2} \tag{12}$$

where δ_{max} is the user specified maximum chord error. The instantaneous radius of curvature can be calculated from

$$\rho_k = \frac{1}{\kappa}\ ,\ \kappa = \frac{\left\| C^{(1)}(u_k) \times C^{(2)}(u_k) \right\|}{\left\| C^{(1)}(u_k) \right\|^3} \tag{13}$$

where κ is the curvature and $C^{(1)}(u)$ and $C^{(2)}(u)$ are the first and second derivatives respectively. Substituting the expression of Eq. 12 for $V(u_k)$ in Eq. 10, u_{k+1} can be found. Thus the adaptive interpolator computes feedrate $V(u_k)$ which is kept constant at most times and changed adaptively when the chord error deviates from the given tolerance δ_{max}.

Variable feedrate interpolator is also preferred for constant material removal and surface finish though the criteria to compute the feedrate differ from the above. Tikhon *et al.*[30] succeeded in expressing the feedrate at every sampling interval as function of curvature, tool radius and depth of cut so as to obtain constant removal of material. In another work which focused on surface finish, feedrate has been expressed in terms of roughness, radius of the cutter and pick feed.[31]

As the feedrate depends on radius of curvature, a large change in curvature will usually lead to abrupt change in velocity which may exceed the ACC/DEC and jerk limits of the machine tool. Unless they are controlled, the drives may saturate and excite the drive train. Feedrate planning for these limits is therefore essential to implement the variable feedrate interpolator. They are discussed in section 8.

5.4 Five-axis Interpolators

The above NURBS curve interpolators are primarily meant for 3 axis machining but can also be extended to 5-axis CNC machines with an additional inverse kinematics module.[32] The module computes the orientation of tool and work table. In contrast to the conventional approach, the orientation of the tool with the surface-normal enables production of smoother NURBS surface. These concepts can be extended to other complex applications also.[33]

5.5 Other Methods

Arc length parameterization gives exact distance of travel along the curve and is best suited for constant feedrate applications in CNC machining. Unfortunately for NURBS, there is no closed form analytical solution for arc length except for straight lines.[34] Though methods based on numerical quadrature exist they are time-consuming and unpredictable and hence cannot be applied for real-time CNC applications. However it is possible to use arc length parameterization when the parametric interval is sufficiently small. Utilizing this concept, the interpolated points generated at consecutive sampling time intervals with conventional linear interpolator can be mapped to the curve parameter by a quintic polynomial. Parametric curve interpolation with a near constant feedrate is realized upon substituting this value in Eq.1.[35] A preprocessing stage for conventional interpolation and a search for the segment for a corresponding arc length is unavoidable.

Generally, for complex curve/surface machining researchers develop specialized interpolation schemes for improved performance.[36] It is also possible to reconstruct NURBS curve from the linearly interpolated data though at the cost of discretization error. In one such method[37] the least-squares curve fitting technique is used to fit a series of discrete points, with predetermined number of control points. The tolerance is also constantly checked and in case of violation the limits of the control points are increased. In a recent development, Li *et al.*[38] developed 5-axis NURBS interpolator to interpolate trajectories generated with traditional interpolator in the form of NURBS. It is worth noting that existing G-codes which are available in conventional format can be interpolated in real-time in NURBS parametric form with this approach.

6. Computing Chord Error

In all the above mentioned types of interpolators, chord error plays an important role in the surface quality of the machining part. It is an important criterion to select feedrate (in case of curve interpolators) or step length (in case of conventional interpolators). The chord error deviation can be defined as the maximum orthogonal distance between the chord formed in the interval of two successive interpolated points and the desired NURBS curve. As the computation of chord error for NURBS is not straightforward, several methods to estimate chord are reported in literature.

6.1 Bisection Method

The classic method proposed by Faux *et al.*[3] considers that, when the chord error is at maximum, the curve tangent must be perpendicular to the perpendicular of chord segment. The segment under consideration (formed with successive interpolated points) is re-parameterized and the condition of orthogonality between the chord vector and the chordal deviation vector is used to find the chord error. The chord error vector with magnitude (chordal length) can be obtained from the commanded feedrate at a given sampling time. The resulting chord error corresponding to this length may however, be more than the desired value and a solution can be found iteratively. The number of iterations can be reduced by bisection method which employs an upper and lower bound. Though this may result in a quicker convergence, the number of iterations is unknown and hence is suitable for offline applications only.[39] Bisection results in error but can be controlled within limits by adjusting the bounds. Due to the complexity and increased computational loads this method is not preferred for NURBS.

6.2 Circular Approximation Method

The most widely used approximation method is circular approximation method since it is reasonable to approximate a small NURBS segment with its osculating circle. Employing Pythagoras theorem [3, 29] chord error δ_k can be approximated as (Fig. 5)

$$\delta_k = \rho_k - \sqrt{\rho_k{}^2 - \left(\frac{L_k}{2}\right)^2} \tag{14}$$

where ρ_k is radius of the curvature and L_k is the distance between points $C(u_k)$ and $C(u_{k+1})$. Upon further simplification of the above equation the feedrate can be expressed as

$$v = \frac{2\sqrt{2\delta}}{T_s}\sqrt{\rho} \tag{15}$$

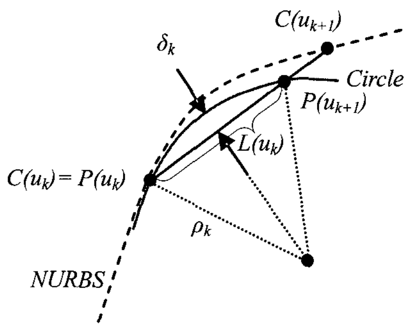

Fig. 5 Circular approximation to NURBS

6.3 Arc Substitution Method

The circular approximation may not yield appropriate results at sharp corners of NURBS tool path. In such cases, the radius can be approximated. A sharp corner is identified based on tangents[40] defined by two lines formed between the successive interpolation points. An arc whose radius is defined based on the angle between them and its geometric derivatives are used to compute the instantaneous radius as follows.

$$\rho_k = \frac{\delta \sin(\theta/2)}{\cos^2 \theta/2} \tag{16}$$

where θ is the angle between the two lines formed between two successive interpolation points and δ is the permitted chord error.[41]

7. Parameter Compensation Methods

One of the major drawbacks of curve interpolators is feedrate error caused by the approximation of the truncated Taylor's series. Although higher order methods exist[42] the recursive nature of NURBS rational functions make this unsuitable for real-time implementation and hence most of the works prefer to employ the first order Taylor's series. Feedrate errors are especially more at high curvature points and a feedrate error of 5% is obtained with first order methods and 0.2% with the second order methods.[9,40] The parameter compensation method is an alternate to higher order Taylor's series and is used to reduce the truncation error.

7.1 Predictor Method

The predictor method is based on the concept of feedback proposed by Lo[40] which uses finite differences to compute the derivatives of the parametric curve. Although the resulting algorithm is simple and fast, a large feedrate error is inevitable because of multiple approximations (finite differences and Taylor's expansion). To reduce this, a revised feedrate is computed corresponding to coordinate points obtained by updating the current parameter. However calculating intermediate coordinate points of NURBS requires computation of recursive rational functions which is as costly as computing derivatives.

7.2 Predictor –Corrector Method

Although feedrate error is reduced using the above approach, control over the feedrate error is not ensured. In the Predictor – Corrector approach the feedrate error can be brought within limits by repeatedly iterating the predicted parameter with an updated parameter. Since the number of such iterations is unknown, the corrector approach may result in unstable computation time.[23] In a later work Tsai *et al.*[43] derived convergence conditions for real time implementation.

7.3 Speed Controlled Interpolation

In this method, the interpolated points at $(k+1)^{th}$ time are also approximated in addition to the Taylor's expansion. This enables the feedrate at k_{th} interval to be expressed in terms of coordinates at u_{k+1}. Based on this concept, Yeh *et al.*[44] expressed a quadratic equation with the compensatory parametric value. Even though the feedrate is maintained constant at most parts of the curve, the feedrate deviates at certain parameter values. The presence of acceleration term prevents using second order Taylor's expansion.[42] To use acceleration further approximation is needed as the acceleration of NURBS curve cannot be computed explicitly unlike feedrate. However it can be used with Taylor's first order method as the acceleration term is not involved.[45]

7.4 Other Methods

A more practical approach is to compute a new feedrate instead of adjusting the parameter value with a compensatory value, thereby retaining the original parameter. The new commanded feedrate can be computed from distance of travel between the interpolated points. But it incurs additional computational load to find the distance and feedrate. A feed forward controller design is also required.[25] In another effort to compensate the parameter, Park *et al.*[46] related BLU and chord error as the ratio of angles between the actual and estimated interpolated points assuming near arc approximation.

8. Incorporating Machining Dynamics in the Variable Feedrate NURBS Interpolator

One of the latest improvements in CNC machining is high speed machining. Latest CNC machine tools with modern tooling enable machining at very high speeds and feeds resulting in enhanced process economics. While machining an intricate shape, non-optimal feedrates might be required in order to satisfy the chord error. This is usually overcome by means of look-ahead methods incorporated in the CNC controller. For a typical CNC machine using conventional linear interpolator, look-ahead method can foresee a certain number of blocks and plan the feedrate according to ACC/DEC limits of the machine tool. But in common parametric curve interpolation, the whole curve is executed as a single instruction. The parametric nature of NURBS prevents the usage of conventional ACC/DEC and jerk control methods as they distort the entire feedrate profile.

Various methods have been proposed to modify the profile to satisfy the ACC/DEC and jerk limits. Most of these methods rely on the concept of detecting the regions where the feedrate deviates from the commanded feedrate. Such regions are called as Feed Rate Sensitive (FRS) corners or key regions. A FRS corner consists of segments of deceleration and acceleration with the former following the latter. Depending upon the various chord error methods described in section 6, a constant feedrate segment may also occur in between.

In the offline look-ahead methods proposed by Yong *et al.*[39], an ACC/DEC limited feedrate profile simultaneously satisfying chord error is obtained by analyzing the change in feedrate between consecutive interpolated points. The feedrate profile is adjusted if the change violates the limiting values of ACC/DEC. This method is further improved by Liu *et al.*[47] by incorporating jerk control. They identified sharp points from the information of geometric derivatives and gradually reduced the feedrate to zero. Further, the corresponding jerk along the path is distributed based on a jerk reshaping method. With these methods it is not possible to reach the exact parameter values corresponding to start and stop point of FRS corner, since, Taylor's expansion cannot compute subsequent parameter increments so as to reach the predefined curve parameter u_e. Any deviation from the exact value of u_e will result in chord error as well as ACC/DEC and jerk values exceeding the limits. Hence it is imperative that, to reach the exact u_e value the constant feedrate segment feedrate segments must be recomputed. But such modification of profile is possible only if the arc length between the start and stop points is sufficient so that adequate number of iterations of u could be carried out. This can be overcome if the new parameter value corresponds to one of the previously interpolated points. A method using this idea is implemented by Du *et al.*[41] for real-time look-ahead implementation using a fixed number of look-ahead points due to restricted buffer size.

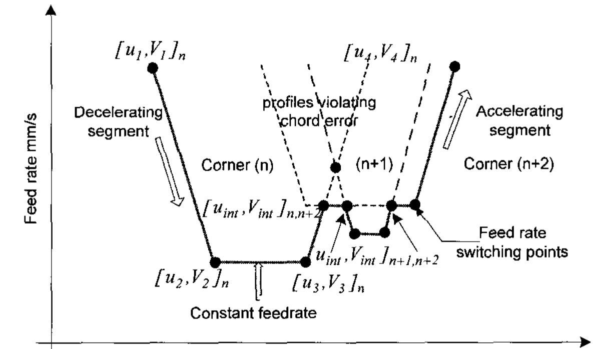

Fig. 6 Interaction between adjacent feedrate profiles

When a complex NURBS profile is adaptively interpolated, several key regions (FRS corners) will form depending on the commanded feedrate. When a NURBS profile is planned for ACC/DEC limits the resulting new ACC/DEC corrected segments may overlap with neighboring corners (shown as dotted lines in Fig. 6). The complex interaction of adjacent feedrate corners thus makes the computed look-ahead profile with the above procedures invalid. A new profile passing through the intersection points $[u_{int}, V_{int}]_{n,n+2}$ and $[u_{int}, V_{int}]_{n+1,n+2}$ can be used.

Sun *et al.*[48] proposed a method to compute the feedrate profile without using look-ahead method explicitly. The feedrate sensitive regions are found by relating acceleration, chord error and sampling time together to a permissible slope of a curve, defined by parameter and curvature. However, jerk control was not implemented in their method. It also requires the conversion of feedrate profile curve to a NURBS curve. Considering real time implementation, an additional NURBS curve for feedrate profile will increase the computational load significantly.

A cubic curve fitting procedure is proposed to overcome this problem.[49] In this method coefficients of the cubic feedrate curve are computed considering ACC/DEC and jerk limits. Expressing the feedrate profile as cubic curve and relating it to knot vectors of the NURBS curve the information can be easily embedded with the modified control points and passed to the controller so that a jerk limited feedrate profile can be realized in real-time. Since velocity can be computed based on the cubic feedrate profile, the CNC controller is relieved from computing the second derivative (Eq. 12). As every segment of a FRS corner must be checked with all the other FRS corners for possible interaction, the intersection algorithm however becomes computationally intense and new intersecting patterns may emerge. When the length of the cubic feedrate segment becomes short it becomes difficult to reach the feedrate at the intersection points. Therefore the feedrate profile continuity may be lost resulting in violation of ACC/DEC and jerk limits.

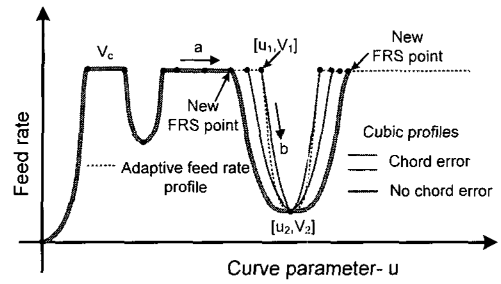

Fig. 7 Chord error free jerk limited cubic feedrate profile

In a modified approach the feedrate profile can be fitted first, using cubic curve with ACC/DEC and jerk constraints also being satisfied and later adjusted for chord error.[50] As the cubic curve passes through the region of chord error, the starting point $[u_1, V_1]$ is moved backwards to a previously interpolated value. In case of accelerating profile it is shifted forward. Fig. 7 shows this method. In the similar manner, cubic feedrate profiles are constructed with subsequent feedrate switching points and a complete feedrate profile satisfying the chord error and ACC/DEC and jerk limits is obtained.

In general, real-time systems for NURBS interpolation require multiple modules/stages to satisfy ACC/DEC and jerk limits.[46] Modules to obtain information on geometry, plan for look-ahead and interpolate are often executed as multiple threads of varying priority. Recently Lin *et al.*[51] proposed a look-ahead method integrating servo dynamics. In this approach the segments with sharp corners are identified considering instantaneous curvature and acceleration. The dynamic feedrates at sharp corners are computed from the system frequency that satisfies the steady state contour error tolerance. Further a jerk limited module is employed to optimize different types of feedrate profiles which arise due to path constraints. Nam and Yang[52] proposed a generalized variable feedrate interpolator which can handle NURBS and other curves also. The interpolator computes subsequent parameters of composite curves confining to path and kinematic constraints by estimating the curve length in real-time. The feedrate profile also satisfies the minimum time criterion by following a trapezoidal acceleration profile. With improved controller designs[53,54] real-time systems for NURBS can result in precise machining of complex shapes at high speeds.

9. Research Directives

At present the aerospace, automobile, die and mold and jewelry industries are the prominent users of freeform shapes designed with NURBS and in the future, this will extend to other applications also. Parametric NURBS curve interpolators are very useful and the implementation of G-Codes for NURBS in commercial controllers such as FANUC shows their success.[55]

Currently research is focused on removing an important drawback of parametric interpolators – the feedrate error with various models. Though preprocessing the NURBS curve offers a realistic estimation of parameter, it needs further refinement. Variable feedrate interpolation is given much importance as high speed machining became a reality. It has attracted much attention as it is unnecessary to command the machine at a lower constant feedrate than optimal. However the command generation process must consider the machine tool limits so that trajectory generated is smooth and does not saturate feed drives. A varying feedrate interpolator will also result in chatter vibration which causes rapid tool wear and poor surface finish. Enhanced variable feedrate interpolators with ACC/DEC and jerk control are also designed and evaluated and further improvements are expected.

The possibility of approximating NURBS with Pythagorean hodographs[34] and Quintic splines is interesting. PH curves have closed form solution to arc length and can be expressed in Bernstein form. On contrary to NURBS, Quintic splines[12,13,14,56] can satisfy dynamics of both the axes individually and show desirable properties. But they are not much popular like NURBS.

Another area requiring more attention is machining of curves and surfaces with 5-axis CNCs. For machining of such free-form shapes the CNC industry still depends largely on conventional linear segmentation as the procedures for generating tool paths are well defined. In 5-axis CNC the cutting tool must be treated as rigid body and must be driven with angular feedrate in addition to translation. Though the theory of linear parametric interpolator is well developed, parametric angular feedrate interpolation must be given more atttention.[57,58]

10. Conclusions

This paper has summarized the research results of parametric curve interpolators for NURBS from the aspect of command generation algorithms. A comparison with other types of interpolation approaches is also presented. Though the arc-length parameterized approach is less prone to feedrate error, the parametric curve approach permits real-time implementation of free-form curves and surface with reduced data transfer and contour error and is preferred. Parameter compensation methods, chord error computation methods and recent designs of variable feedrate interpolators are also discussed. These methods show promising results at the research level. This encourages to think that in the imminent future precise and high-speed machining of NURBS curves and surfaces will be realized.

REFERENCES

1. Koren, Y., "Computer Control of Manufacturing Systems," McGraw-Hill, pp. 115-142, 1998.
2. Koren, Y., "Control of Machine Tools," ASME Journal of Manufacturing Science, Vol. 119, Issue 4B, pp. 749-755, 1997.
3. Faux, L. D. and Pratt, M. J., "Computational Geometry for Design and Manufacture," Ellis Horwood, pp. 261-277, 1979.
4. Piegl, L., "On NURBS: A Survey," IEEE Computer Graphics and Applications, Vol. 11, No. 1, pp. 55-71, 1991.
5. Piegl, L. and Tiller. W., "The NURBS Book," Springer, pp. 117-138, 1997.
6. Rogers, D. F., "An Introduction to NURBS: with Historical Perspective," Morgan Kaufmann Publishers Inc., pp. 129-150, 2001.
7. Jensen, C. G. and Anderson, D. D., "A Review of Numerically Controlled Methods for Finish Sculptured-Surface Machining," IIE Transactions, Vol. 28, No. 1, pp. 30-39, 1996.
8. Vickers, G. W. and Bradley, C., "Curved Surface Machining Through Circular Arc Interpolation," Computers in Industry, Vol. 19, No. 3, pp. 29-37, 1992.

9. Yang, D. C. H. and Kong, T., "Parametric Interpolator versus Linear Interpolator for Precision CNC Machining," Computer-Aided Design, Vol. 26, No. 3, pp. 225-234, 1994.

10. Shiptalni, M., Koren. Y. and Lo, C. C., "Real-Time Curve Interpolators," Computer-Aided Design, Vol. 26, No. 11, pp. 832-838, 1994.

11. Chou, J. J. and Yang, D. C. H., "Command Generation for Three Axis CNC Machining," ASME Journal of Engineering for Industry, Vol. 113, No. 3, pp. 306-310, 1991.

12. Wang, F. C. and Yang, D. C. H., "Nearly Arc Length Parameterized Quintic Spline Interpolation for Precision Machining," Computer-Aided Design, Vol. 5, No. 5, pp. 281-288, 1993.

13. Wang, F. C. and Wright, P. K., "Open Architecture Controllers for Machine Tools, Part 2: A Real Time Quintic Spline Interpolators," ASME Journal of Manufacturing Science, Vol. 120, No. 2, pp. 425-432, 1998.

14. Erkorkmaz, K. and Altintas, Y., "High Speed CNC System Design Part 1: Jerk Limited Trajectory Generation and Quintic Spline Interpolation," International Journal of Machine Tool and Manufacture, Vol. 41, No. 9, pp. 1323-1345, 2002.

15. Papaioannou, S. G., "Interpolation Algorithms for Numerical Control," Computers in Industry, Vol. 1, No. 1, pp. 27-40, 1979.

16. Kritsis, D., "High Precision Interpolation Algorithm for 3D Parametric Curve Generation," Computer-Aided Design, Vol. 26, No. 11, pp. 850-856, 1994.

17. Omirou, S. L., "Space Curve Interpolation for CNC Machines," Journal of Materials Processing Technology, Vol. 141, No. 3, pp. 343-350, 2003.

18. Huang, J. T. and Yang, D. C. H., "A Generalized Interpolator for Command Generation of Parametric Curves in Computer Controller Machines," ASME Japan-USA symposium on Flexible Automation, pp. 393-399, 1992.

19. Sata, T., Kimura, F., Okada, N. and Hosaka, M., "A New Method of NC Interpolation for Machining Sculptured Surfaces," Annals of CIRP, Vol. 30, No. 1, pp. 369-372, 1980.

20. Hermann, G., "Patch Programming: The Integration of Motion Planning into Numerical Control," Computers in Industry, Vol. 5, No. 4, pp. 351-359, 1984.

21. Bedi, S., Ali, I. and Quan, N., "Advanced Interpolation Techniques for NC Machines," ASME Journal of Engineering for Industry, Vol. 115, No. 3, pp. 329-336, 1993.

22. Bedi, S. and Quan, N., "Spline Interpolation Technique for NC Machines," Computers in Industry, Vol. 18, No. 3, pp. 307-313, 1992.

23. Cheng, M. Y., Tsai, M. C. and Kuo, J. C., "Real time NURBS Command Generators for CNC Servo Controllers," International Journal of Machine Tool and Manufacture, Vol. 42, No. 7, pp. 801-813, 2001.

24. Zhang, Q. G. and Greenway, R. B., "Development and Interpolation of a NURBS Curve Motion Interpolator," Robotics and Computer-Integrated Manufacturing, Vol. 14, No. 1, pp. 27-36, 1998.

25. Cheng, C. W. and Tsai, M. C., "Real-time Variable Feedrate NURBS Curve Interpolators for CNC Machining," International Journal of Advanced Manufacturing Technology, Vol. 23, pp. 865-873, 2004.

26. Lin, R. S., "Real-time Surface Interpolator for 3D Parametric Surface Machining on 3-Axis Machine Tools," International Journal of Machine Tool and Manufacture, Vol. 40, No. 10, pp. 1513-1526, 2000.

27. Tsai, M. C., Cheng, C. W. and Cheng, M. Y., "A Real-time Surface Interpolator for Precision Three Axis CNC Machining," International Journal of Machine Tool and Manufacture, Vol. 43, No. 12, pp. 1217-1227, 2003.

28. Zhiming, X., Jincheng. C. and Zhengjin, F., "Performance Evaluation of a Real-Time Interpolation Algorithm for NURBS Curve," International Journal of Advanced Manufacturing Technology, Vol. 20, No. 4, pp. 270-276, 2002.

29. Yeh, S. S. and Hsu, P. L., "Adaptive Feedrate Interpolation for Parametric Curves with Confined Chord Error," Computer-Aided Design, Vol. 34, No. 3, pp. 229-237, 2002.

30. Tikhon, M., Ko, T. J., Lee, S. H. and Kim, H. S., "NURBS Interpolator for Constant Material Removal Rate in Open NC Machine Tools," International Journal of Machine Tool and Manufacture, Vol. 44, No. 2, pp. 237-245, 2004.

31. Choi, I. H., Yang, M. Y., Hong, W. P. and Jung, T. S., "Curve Interpolation with Variable Feedrate for Surface Requirement," International Journal of Advanced Manufacturing Technology, Vol. 25, No. 3/4, pp. 325-333, 2005.

32. Koren, Y. and Lin, R. S., "Five-Axis Surface Interpolators," Annals of CIRP, Vol. 44, No.1, pp. 379-382, 1995.

33. Kang, C. G., "Online Trajectory Planning for a PUMA Robot," International Journal of Precision Engineering and Manufacturing, Vol. 8, No. 4, pp. 51-56, 2007.

34. Farouki, R. T. and Shah, S., "Real-time CNC interpolators for Pythagorean Hodograph curves," Computer-Aided Geometric Design, Vol. 13, No. 7, pp. 583-600, 1996.

35. Jeong, S. Y., Choi, Y. J. and Park, P. G., "Parametric Interpolation using Sampled Data," Computer-Aided Design, Vol. 38, No. 1, pp. 39-47, 2006.

36. Kim, H. T., Yang, H. J. and Kim, S. C., "Control Method for the Tool Path in Aspherical Surface Grinding and Polishing," International Journal of Precision Engineering and Manufacturing, Vol. 7, No. 4, pp. 51-56, 2006.

37. Yau, H. T. and Kuo, M. J., "NURBS Machining and Feed Rate Adjustment for High-Speed Cutting of Complex Sculptured Surfaces," International Journal of Prod. Research, Vol. 39, No. 1, pp. 21- 41, 2001.

38. Li, W., Liu, Y. Yamazaki, K. Fujisima, M. and Mori, M., "The Design of a NURBS Pre-Interpolator for Five-Axis Machining," International Journal of Advanced Manufacturing Technology, in print (DOI 10.1007/s00170-006-0905-8).

39. Yong, T. and Narayanasami, R., "A Parametric Interpolator with confined Chord Errors, Acceleration and Deceleration for NC Machining," Computer-Aided Design, Vol. 35, No. 13, pp. 1249-1259, 2003.

40. Lo, C., "Feedback Interpolators," ASME Journal of Manufacturing Science, Vol. 119, No. 4, pp. 587-592, 1997.

41. Du, D., Liu, Y., Yan, C. and Li, C., "An Accurate Adaptive Parametric Curve Interpolator for NURBS Curve Interpolator," International Journal of Advanced Manufacturing Technology, Vol. 32, No. 9/10, pp. 999-1008, 2007.

42. Farouki, R. T. and Tsai, Y. F., "Exact Taylor Series Coefficients for Variable Feedrate CNC Curve Interpolators," Computer-

Aided Design, Vol. 35, No. 2, pp. 155-165, 2001.

43. Tsai, M. C. and Cheng, C. W., "Real-time Predictor Corrector Interpolator For CNC Machining," ASME Journal of Manufacturing Engineering and Science, Vol. 125, No. 3, pp. 449-460, 2003.

44. Yeh, S. S and Hsu, P. L., "The Speed Controlled Interpolator for Machining Parametric Curves," Computer-Aided Design, Vol. 31, No. 5, pp. 349-357, 1999.

45. Yau, C., Du, D. and Li, C., "Design of a Real-time Adaptive Interpolator with Parameter Compensation," International Journal of Advanced Manufacturing Technology, Vol. 35, No. 1/2, pp. 169-178, 2007.

46. Park, J. H., Nam, S. H. and Yang, M. Y., "Development of a Real-time Trajectory Generator for NURBS Interpolator based on a 2 Stage Interpolation Method," International Journal of Advanced Manufacturing Technology, Vol. 26, No. 4, pp. 359-365, 2005.

47. Liu, X., Ahamd, F., Yamazaki, K. and Mori, M., "Adaptive Interpolation Scheme for NURBS Curves with the Integration of Machining Dynamics," International Journal of Machine Tool and Manufacture, Vol. 45, No. 4/5, pp. 433-444, 2005.

48. Sun, Y., Wang, J. and Guo, D., "Guide Curve based Interpolation Scheme of Parametric Curves for Precision Machining," International Journal of Machine Tool and Manufacture, Vol. 46, No. 3/4, pp. 235-242, 2006.

49. Sekar, M., Narayanan, V. N. and Yang, S. H., "Design of Jerk Bounded Feedrate with Ripple Effect for Adaptive NURBS Interpolators," International Journal of Advanced Manufacturing Technology, in print (DOI: 10.1007/s00170-007-992-1).

50. Lee, D. M., Sekar, M. and Yang, S. H., "NURBS Interpolator using Minimum Time Method for High Speed Machining," Proceeding of Leading Edge Manufacturing-21, pp. 187-190, 2007.

51. Lin, M. T., Tsai, M. S and Yau, H. T., "Development of a Dynamics-Based NURBS Interpolator with Real-Time Look-Ahead Algorithm," International Journal of Machine Tool and Manufacture, Vol. 47, No. 15, pp. 2246-2262, 2007.

52. Nam, S. H. and Yang, M. Y., "A Study on a Generalized Parametric Interpolator with Real-Time Jerk Limited Acceleration," Computer-Aided Design, Vol. 36, No. 1, pp. 27-36, 2004.

53. Lee, M. H., Yang, S. H., Kim, Y. S., Roh, Y. R. and Kim, C., "A Multi-Axis Contour Error Controller for Free Form Curves," JSME International Journal Series C, Mechanical Systems, Machine Elements and Manufacturing, Vol. 47, No. 1, pp. 144-149, 2004.

54. Jee, S. C. and Lee, H. C., "An Integrated Approach to the Analysis and Design of a Three-Axis Cross-Coupling Control System," International Journal of Precision Engineering and Manufacturing, Vol. 8, No. 2, pp. 59-63, 2007.

55. Shima, A., Sasaki, T., Ohtsuki, T. and Wakinotani, Y., "64-bit RISC-based Series 15 NURBS Interpolation," FANUC Technical Review, Vol. 9, No. 1, pp. 23-28, 1996.

56. Erkorkmaz, K. and Altintas. Y., "Quintic Spline Interpolation with Minimal Feed Fluctuation," ASME Journal of Manufacturing Science and Engineering, Vol. 127, No. 2, pp. 339-349, 2005.

57. Sarma, R. and Rao, A., "Discretizors and Interpolators for Five-Axis CNC Machines," ASME Journal of Manufacturing Science and Engineering, Vol. 122, No. 1, pp. 191-197, 2000.

58. Xu, H. Y., Dai, J. S. and Tam, H. Y., "Angular Interpolations for Planar Implicit Curves," International Journal of Prod. Research, Vol. 40, No. 9, pp. 2125-2140, 2002.