COMPUTER-AIDED DESIGN & APPLICATIONS, 2016
VOL. 13, NO. 3, 369–378
http://dx.doi.org/10.1080/16864360.2015.1114394

5D Cubic B-Spline Interpolated Compensation of Geometry-Based Errors in Five-Axis Surface Machining

Md Shafayet H. Bhuiya and O. Remus Tutunea-Fatan

Western University, Canada

ABSTRACT
Linear interpolation constitutes the most common type of interpolation currently employed in five-axis computer numerically-controlled (CNC) surface machining. However, despite its popularity, this type of interpolation constitutes one of the primary causes of the significant decreases in the overall productivity and accuracy of the five-axis machining process. This decline in efficiency can be explained – at least in part – through the discontinuities in velocity that are present in the joint space as well as through the geometry-based errors that are an expression of the unsynchronized translational and rotational motions associated with machine tool kinematics. To address these issues, the present study proposes an alternate interpolation scheme based on a 5D cubic B-Spline whose knot values are determined by means of ideal joint trajectories to be followed. Two supplementary end conditions are then added to make the problem at hand determined in terms of control point position. The implementation of the proposed technique revealed that in addition to its inherent C^1 continuity, the proposed 5D B-Spline interpolation enables significant reductions of the machining errors along the intended tool path.

KEYWORDS
CAD/CAM; five-axis CNC machining; sculptured surfaces; geometry-based error reduction; inverse kinematics; NC post-processing; 5D joint space; synchronized five-axis motions; B-Spline interpolation; 5D velocity continuity

1. Introduction

In computer numerically-controlled (CNC) machining technology, the interpolator is responsible for the translation in real time of the motion commands to be followed to generate a workpiece with a minimum deviation with respect to its digitally-designed shape. In the early days of numerically controlled (NC) systems, tool path interpolation was performed by means of linear and circular interpolations which represent in fact simple computations performed with a minimal amount of constraints. However, these straightforward interpolation schemes are presently regarded as being unable to keep up with modern accuracy and productivity demands associated with the fabrication of mechanical components whose shape is defined by multiple complex or freeform surfaces.

The approximation of three-dimensional tool paths/curves with a set of linear and circular piecewise segments results in a number of drawbacks, such as: velocity discontinuities, feed rate/acceleration fluctuations, increased controller memory requirements as well as heavier data transmission loads. To exemplify, in order to preserve a relatively constant feed rate along the intended tool path and to reduce the number of accelerations and decelerations that increase the overall machining time and overload machine tool drives, the number of approximating linear piecewise segments should be maximized. On the other hand, long linear segments typically mean large deviations from the natural curvature of the machined surface, and this in turn will decrease significantly the both the quality and accuracy of the machined surface. One of the solutions proposed to address the aforementioned challenges was related to the development of real-time interpolators attempting to determine "online" or "on-the-fly" the position of the "next" CC point along the intended tool path. However, with very limited exceptions [10], the vast majority of early studies on CNC interpolators were focused on three-axis machining instances.

By contrast, a growing number of recent research attempts were directed towards the problematics of five-axis machining. For instance, Rao and Sarma [21] have proposed an interpolation scheme based on a tool path discretization techniques aiming to maintain specific contour, feed rate and orientation errors within an allowable range. Furthermore, Sencer et al. [22] have proposed real-time feed rate scheduling algorithms for five-axis CNC systems aiming to minimize the machining time required for five-axis contour machining of sculptured surfaces. The combined offline and online (real-time)

CONTACT Remus Tutunea-Fatan rtutunea@eng.uwo.ca

method proposed by Fleisig and Spence [9] involves the computation of three different spline curves to control cutter position, orientation, and reparametrization. The issue of constant feed rate along the tool path can be elegantly solved by means of Pythagorean Hodograph (PH) curves that were initially proposed by Farouki and Sakkalis [7]. However, the theory of PH curves is still insufficiently developed since so far it has not been extended beyond the Euclidian 3D space [6] and PH curve fitting still poses serious computational difficulties [1]. The only attempt to apply the concept of PH curves in five-axis machining proved to be cumbersome and accompanied by a number of limiting assumptions with respect to the shape of the sculptured surface and/or initial discretization of the tool path [18].

Non-Uniform Rational B-Splines (NURBS) represent a class of parametric curves with a special applicability in CAD/CAM. As a result of the tight interdependence between CAD/CAM and CNC systems, NURBS constitute an attractive solution to five-axis interpolation issues. The benefits associated with NURBS interpolation are multiple, but they were generally linked to the robustness of their representations as well as their C^2 continuity. The surveyed literature documents a number of five-axis NURBS interpolators that were successfully developed in the past – most commonly in the context of open-architecture controllers [14,15,20,28,29,31]. It is worth mentioning here that sometimes the term NURBS is used inadequately since all control points are assumed of even weight, which in turn means that NURBS have been reduced to B-Spline formulations only.

As Makhanov [17] has correctly pointed out, one of the common drawbacks of the NURBS interpolators developed so far is that they tend to neglect the real cutter trajectory as imposed by machine tool kinematics. As a result, the interplay between the CAM-originated errors and interpolation scheme was relatively less investigated. One of the significant steps in this direction was made by Langeron et al. [11] who proposed a cubic polynomial B-Spline format for both cutter position and orientation curves and incorporated machine tool kinematics in tool path generation algorithm. Further developments of the idea were later proposed by Chiou and Lee [5] who developed a novel surface-based NURBS path interpolation approach in which both machine tool configuration and flat-end cutter geometry were taken into account. The idea of constant feed rate surface-based interpolator was further expanded by other researchers who have coupled it with the isoscallop tool path constraint [16]. Recent studies on the topic of five-axis interpolators advocate strongly for a more powerful CNC programming language to replace the overrestrictive G-code standard that forces the controller to rely on approximations, rather than exact representations of the intended tool path. A good example in this regard was presented by Beudaert et al. [2] who have integrated Sinumerik 840D with a custom-build controller and - by accounting for the kinematic and dynamic behavior of the five-axis machine – they were able to concurrently improve the accuracy and the productivity of the machining process.

When it comes to interpolation in five-axis machining, two primary options exist: (i) tool path calculation in workpiece coordinate system; and (ii) tool path calculation in machine control coordinate system. Langeron et al. [11] have commented on the relative difficulty associated with prioritization of these tasks in the context of the overall structure of five-axis CNC machining systems, since problem (i) can be solved before or after problem (ii). While most of the prior works have assumed that problem (i) should be solved before problem (ii), the current study practically advocates for solving (ii) prior to (i). The rationale behind this decision is twofold: 1) performing the inverse kinematics in real-time by means of the numerical controller will increase its computational load and this might have negative effects on the accuracy of the machining operations; and 2) since CAM represents the principal system tasked with tool path planning, it seems appropriate to add inverse kinematics computations to its activities, since this addition will increase the accuracy of the pre-machining simulations. While some previous attempts have already been made in this direction [3,26], the current study will provide a direct comparison between linear and cubic B-Spline interpolations. However, unlike the previous studies, which relied on cubic B-Spline fitting procedures based on conventional knot removal algorithms [19] or on quadratic approximations [3], the current work proposes an original cubic B-Spline calculation technique simultaneously enforcing C^2 continuity and 5D joint space of the five-axis machine tool.

2. 5D Interpolation in Five-axis Machining

2.1. Velocity Discontinuities in Linearly-Interpolated Motions

Due to the hardware limitations of the commercial controllers commonly installed on five-axis machine tools, the cutter is typically unable to accurately track a 3D tool path located on a sculptured surface. The negative impact of the frequent accelerations and decelerations required to instantaneously change cutter's direction of motion has been experimentally assessed by Vickers and Bradley [27], who have shown that the programmed feed rate is achieved on average for only 10% of the total machining time. Moreover, sudden changes in tool trajectory contribute to the generation of non-smooth surfaces, that require additional polishing and therefore increases the overall fabrication costs of the mechanical components whose shape is delimited by freeform geometries [23,30].

In five-axis CNC machining, the tool path is typically followed in a sequence of small discrete motions. The distance between consecutive CC points – that constitutes the core of the discretization algorithm – can be determined based on several different geometric constraints. Some of the possible options in this regard are: chordal deviation [8], geometry-based error [13], constant parameter increment [18] or other methods derived from them [32]. Tool path discretization could also be performed in such a way to enforce a constant feed rate along the intended tool path in manner similar to that suggested initially by Farouki and Sakkalis [7]. Regardless of the tool path discretization selected, an array of successive CC points ($\mathbf{P}_{\mathrm{CC}_i}$ and $\mathbf{P}_{\mathrm{CC}_{i+1}}$) will be generated along the intended tool path (Fig. 1).

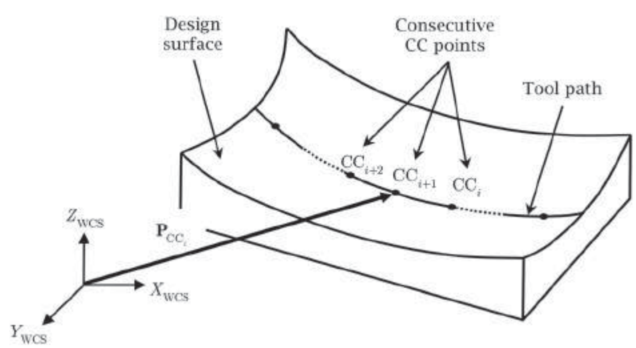

Figure 1. Discrete CC points along the intended tool path.

For each instantaneous CC point along the tool path, tool posture (i.e. position and orientation) can be determined from a number of machining constraints, including, but not being limited to maximization of the material removed, avoidance of the local surface gouging, generation of a constant scallop height on the machined surface, etc. However, regardless of the machining constraints used to determine the cutter posture associated with each of the discrete CC points chosen, the numerical controller of the machine has to be instructed how much to move each of the five axes (joints) of the machine in order to reach the desired tool posture. In this case, the conversion of tool posture from WCS to MCS has to be performed through an inverse kinematics technique [24]. A schematization of this process is illustrated in Fig. 2.

Then, for each pair of consecutive CC points (CC_i and CC_{i+1}), the intermediate tool postures are determined by employing an interpolation law that is available within the numerical controller. In case of the linear interpolation, this means that:

$$\mathbf{P}_{\mathrm{MCC}}(\lambda) = \mathbf{P}_{\mathrm{MCC}_i} + \lambda \cdot (\mathbf{P}_{\mathrm{MCC}_{i+1}} - \mathbf{P}_{\mathrm{MCC}_i}) \quad (1)$$

where $\mathbf{P}_{\mathrm{MCC}_i}$ and $\mathbf{P}_{\mathrm{MCC}_{i+1}}$ are 5D machine control coordinates associated with CC_i and CC_{i+1} respectively, and λ represents interpolation parameter.

For vertical spindle-rotating (SR) five-axis machines also known as wrist-type, the actual trajectory followed by CL point during linearly-interpolated motions has a nonlinear nature and deviates out of the bilinear surface delimited by $\mathrm{O}_{\mathrm{P}_{i+1}}$, CL_{i+1}, CL_i, and $\mathrm{O}_{\mathrm{P}_i}$ points as illustrated by Fig. 3. While performing these movements, the pivot point of the machine (O_{P}), located at the intersection of the two rotary axes of the machine, will follow linear trajectories. When represented in 3D/2D translational/rotational machine control (joint) spaces, these linearly-interpolated motions will approximate the ideal MCC trajectory (Ψ_{MCC}) with a succession of linear segments (Fig. 3). As such, the ideal 5D MCC trajectory is generated through conversion into MCC space of the tool postures corresponding to all tool path points.

Although from a strict algebraic perspective, the discontinuity analysis to be detailed below could be performed directly in 5D space, all subsequent considerations were made by separating the translational and rotational motions, since they represent measures with different meanings and units. In a generic five-axis machining operation, the discontinuities between two consecutive linearly-interpolated cutter movements as determined by CC_i, CC_{i+1} and CC_{i+2} tool path points can be assessed by analyzing the magnitude and orientation of the parametric speeds acquired along each of the two linear segments to be calculated as following (Fig. 4):

$$\mathbf{v}_{\mathrm{MCC}_{i+1}}^{\mathrm{T,R}} = \frac{\mathrm{d}[\mathbf{P}_{\mathrm{MCC}^{\mathrm{T,R}}}(\lambda)]}{\mathrm{d}\lambda} = \mathbf{P}_{\mathrm{MCC}_{i+1}^{\mathrm{T,R}}} - \mathbf{P}_{\mathrm{MCC}_i^{\mathrm{T,R}}} \quad (2)$$

where $\mathbf{v}_{\mathrm{MCC}_{i+1}}^{\mathrm{T,R}}$ represent the translational and rotational parametric speeds, respectively, acquired along the $(i+1)$-th linearly-interpolated segment. The two indices

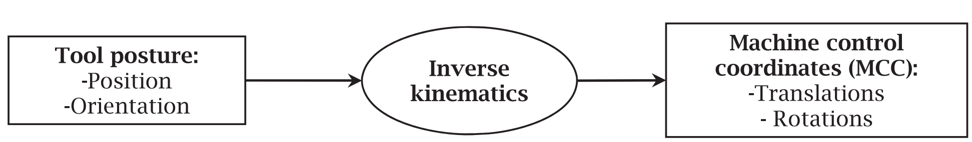

Figure 2. Inverse kinematic transformation.

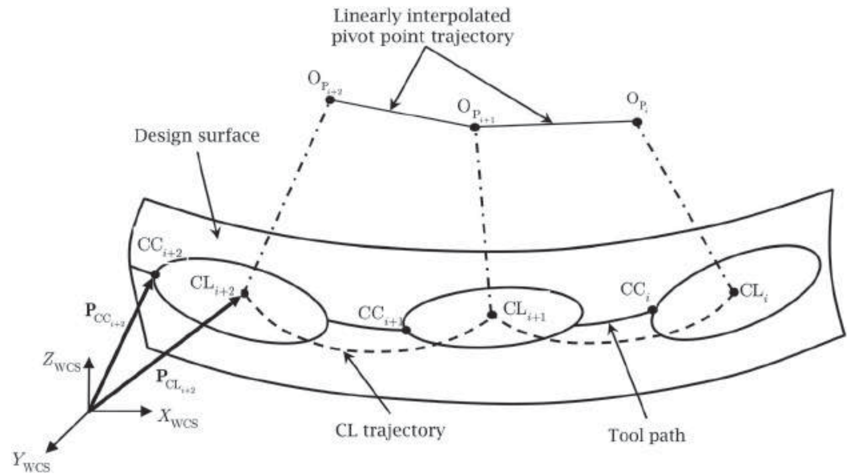

Figure 3. Real CL trajectories for two consecutive tool path segments.

to be used to quantify 5D MCC velocity discontinuities during linearly-interpolated motions along the tool path are defined as:

$$\Delta^{\mathrm{T,R}}_{\theta_{i+1,\,i+2}} = \mathrm{abs}\left(\cos^{-1}\left[\frac{\mathbf{v}^{\mathrm{T,R}}_{\mathrm{MCC}_{i+1}} \cdot \mathbf{v}^{\mathrm{T,R}}_{\mathrm{MCC}_{i+2}}}{\left|\mathbf{v}^{\mathrm{T,R}}_{\mathrm{MCC}_{i+1}}\right| \cdot \left|\mathbf{v}^{\mathrm{T,R}}_{\mathrm{MCC}_{i+2}}\right|}\right]\right)$$

$$\Delta^{\mathrm{T,R}}_{|\mathbf{v}|_{i+1,\,i+2}} = \mathrm{abs}\left(\frac{\left|\mathbf{v}^{\mathrm{T,R}}_{\mathrm{MCC}_{i+2}}\right|}{\left|\mathbf{v}^{\mathrm{T,R}}_{\mathrm{MCC}_{i+1}}\right|}\right) \tag{3}$$

where $\Delta^{\mathrm{T,R}}_{\theta_{i+1,\,i+2}}$ represents the absolute value of angular change in velocity direction for two consecutive linearly-interpolated segments. Velocity vectors are calculated separately in the 3D translational (T) and 2D rotational (R) machine joint spaces, respectively. Similarly, $\Delta^{\mathrm{T,R}}_{|\mathbf{v}|_{i+1,\,i+2}}$ represents the ratio of two consecutive velocity magnitudes as calculated in translational and rotational joint spaces, respectively.

2.2. 5D Cubic B-Spline Interpolation

In order to address the velocity discontinuities associated with linearly-interpolated motions, a novel interpolation technique based on 5D cubic B-Spline formulation will be developed in this section. Owed to its intrinsic definition, B-Spline curve is a C^2 continuous composite curve consisting of multiple piecewise curves joined together at knot points. Another major advantage of the 5D B-Spline representation is that no further reparametrization is required between the translational and rotational MCC trajectories in order to preserve the desired synchronization between them, since both curves share common a curve parameter and knot vector values. The idea of multi-dimensional B-Spline (NURBS) curves is not absolutely new in five-axis machining, but so far it was only considered from a control-related perspective [4].

The shape of the 5D cubic B-Spline is determined by the $(n + 1)$ known data points yielded from the tool path discretization algorithm. However, unlike generic B-Spline fitting algorithms based on knot removal techniques [19,26] or chord length parametrization [19], the algorithm used in this section is based on a knot vector whose components are synchronized with the set of given data points. This approach will provide a direct comparison basis between 5D linear and B-Spline interpolations, since both schemes approximate the ideal MCC curve with the same number of given data points.

Let MCC_i, $i \in \{0,\ 1,\ \ldots,\ n\}$ be the set of given points to be interpolated by the 5D B-Spline. The 5D coordinates of MCC_i are to be determined through inverse kinematics from discrete tool postures defined along the intended tool path. The objective is to determine an open non-periodic uniform cubic (degree $p = 3$) B-Spline that passes through all MCC_i points in their original sequential order. According to classical B-Spline theory [19], the general expression of this 5D curve is:

$$\mathbf{P}_{\mathrm{MCC_B}}(\lambda) = \sum_{i=0}^{m} N_{i,p+1}(\lambda) \cdot \mathbf{P}_{\mathrm{CP}_{\mathrm{MCC}_i}},\ \lambda \in [0,\ \lambda_{\max}] \tag{4}$$

where $\mathrm{CP}_{\mathrm{MCC}_i}$, $i \in \{0,\ 1,\ \ldots,\ m\}$ constitute the control points required to define the shape of the 5D cubic B-Spline curve (Fig. 5), and $N_{i,p+1}(\lambda)$ represent the corresponding blending functions associated with each of the $\mathbf{P}_{\mathrm{CP}_{\mathrm{MCC}_i}}$ control points.

As mentioned above, the number of non-periodic and uniform knots λ_i is purposely chosen to match the number of MCC_i points. Because of this condition, the most straightforward method to define the uniform knot values would be by demanding that their distinct values to be

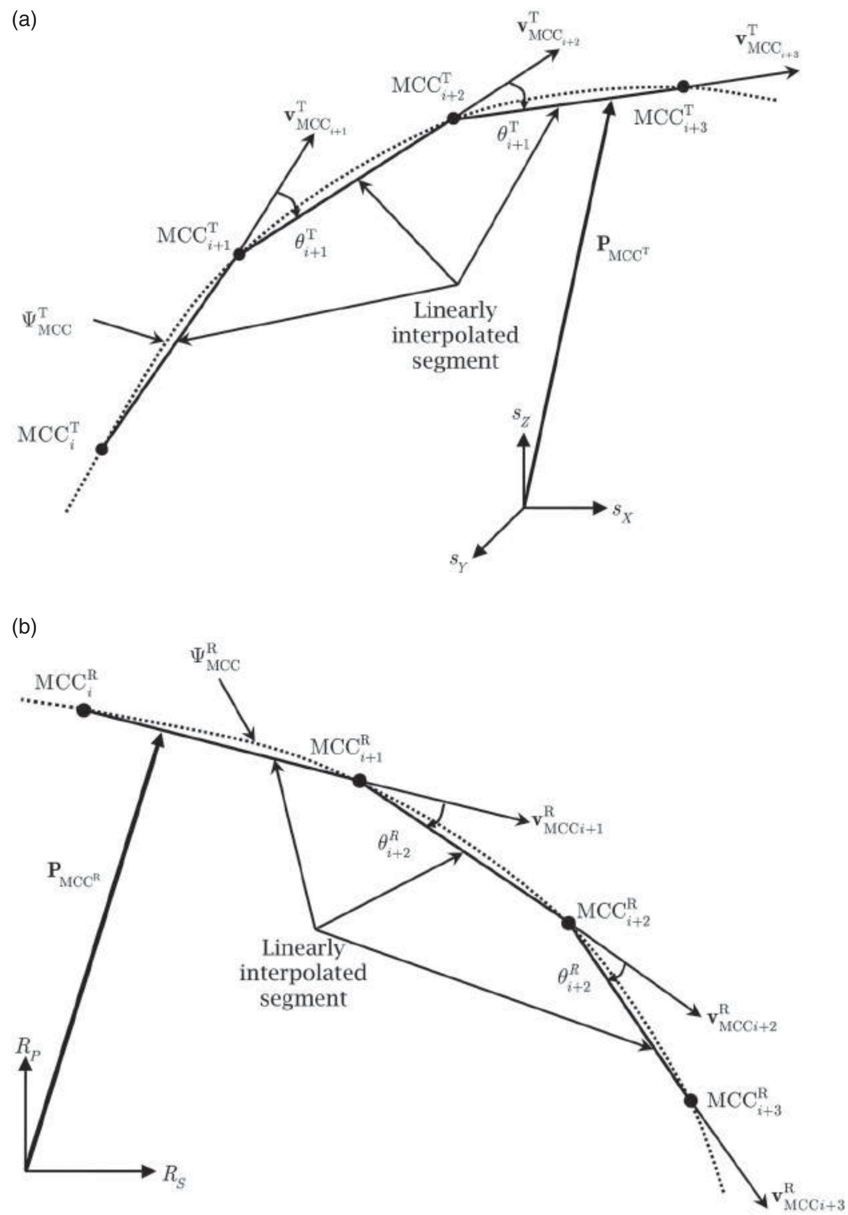

Figure 4. Magnitude and angular velocity discontinuities in: (a) 3D translational, and (b) 2D rotational machine tool joint spaces.

defined simply as $\lambda_i = i$, with $i \in \{0,\ 1,\ \ldots,\ n\}$. Since, according to common CAD/CAM practices, the 5D B-Spline has to pass through the first and last points of the control polygon and it has to be tangent to first and last segments of the control polygon, the complete knot vector associated with the sought curve yields as:

$$\boldsymbol{\Lambda} = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 2 \quad \ldots \quad n \quad n+1 \quad n+1 \quad n+1 \quad n+1] \tag{5}$$

The $(p+1)$ repeats of the first and last knot vector value ensures in fact the compliance with the two aforementioned geometric B-Spline properties. Based on these assumptions, the total length of $\boldsymbol{\Lambda}$ vector becomes $(n+7)$, from which it can be inferred that the number of control points $(m+1)$ required to completely determine the shape of the B-Spline curve is:

$$m + 1 = n + 7 - (p+1) \Rightarrow m = n + 2 \tag{6}$$

By substituting the values of m and $\lambda_{\max}$ from Eqn. (5) and (6) into (4), the expression of B-Spline becomes:

$$\mathbf{P}_{\mathrm{MCC_B}}(\lambda) = \sum_{i=0}^{n+2} N_{i,p+1}(\lambda) \cdot \mathbf{P}_{\mathrm{CP}_{\mathrm{MCC}_i}},\ \lambda \in [0,\ n+1] \tag{7}$$

where blending functions $N_{i,p+1}(\lambda)$ are defined in a recursive manner as follows:

$$N_{i,1}(\lambda) = \begin{cases} 1, & \lambda_i \leq \lambda \leq \lambda_{i+1} \\ 0, & \text{otherwise} \end{cases}$$

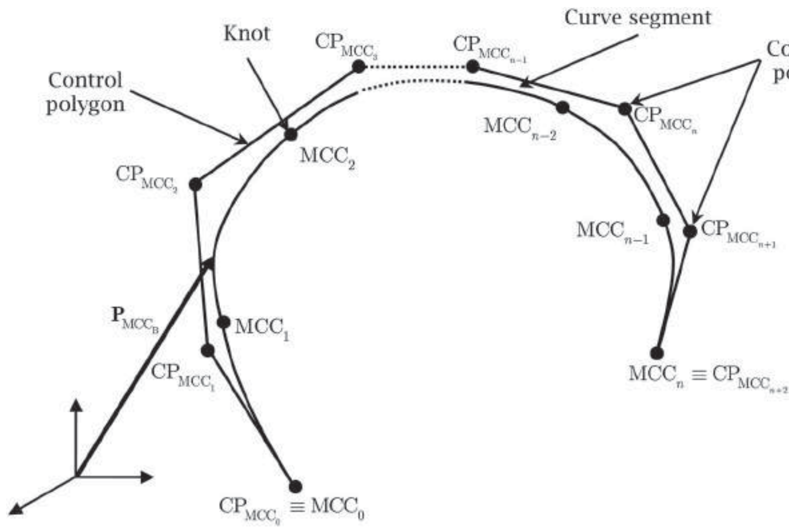

Figure 5. Characteristic elements of B-Spline curves.

$$N_{i,p+1}(\lambda) = \frac{(\lambda - \lambda_i)}{(\lambda_{i+p} - \lambda_i)} \cdot N_{i,p}(\lambda) + \frac{(\lambda_{i+p+1} - \lambda)}{(\lambda_{i+p+1} - \lambda_{i+1})} \cdot N_{i+1,p}(\lambda) \quad (8)$$

The coordinates of the unknown set of control points can be determined by forcing the known set of given points to become junction points between pairs of consecutive piecewise segments of the cubic B-Spline (with the exception of the two end points):

$$\mathbf{P}_{\mathrm{MCC}_i} = \mathbf{P}_{\mathrm{MCC_B}}(\lambda_i),\ i \in \{0,\ 1,\ \ldots,\ n\} \quad (9)$$

Obviously, the $(n+1)$ conditions available through Eqn. (9) are not sufficient to solve for the $(n+3)$ unknown control points required to establish the shape of the B-Spline curve. In this study, the two missing conditions will be assumed to be provided by two end second derivatives of the curve:

$$\left|\frac{d\mathbf{P}^2_{\mathrm{MCC_B}}(\lambda)}{d\lambda^2}\right|_{\lambda=0} = \left|\frac{d\mathbf{P}^2_{\mathrm{MCC_B}}(\lambda)}{d\lambda^2}\right|_{\lambda=n} = 0 \quad (10)$$

Other supplementary end conditions could be used to fully determine the shape of the B-Spline, but they would usually involve a preselection of two end first derivatives whose direction and magnitude are relatively difficult to surmise [19].

According to Lee [12], the general formula for r-th derivative of a B-Spline curve is:

$$\frac{d\mathbf{P}^r_{\mathrm{MCC_B}}(\lambda)}{d\lambda^r} = \sum_{i=l-p+r}^{l} \mathbf{P}^r_{\mathrm{CP_{MCC}}_i} \cdot N_{i,p+1-r}(\lambda),\ \lambda \in [\lambda_l,\ \lambda_{l+1}] \quad (11)$$

where

$$\mathbf{P}^r_{\mathrm{CP_{MCC}}_i} = (p+1-r) \cdot \frac{\mathbf{P}^{r-1}_{\mathrm{CP_{MCC}}_i} - \mathbf{P}^{r-1}_{\mathrm{CP_{MCC}}_{i-1}}}{\lambda_{i+p+1-r} - \lambda_i} \quad (12)$$

Particularization of Eqn. (11) and (12) for the desired end conditions specified by Eqn. (10), and derived for the knot vector of the cubic B-Spline indicated by Eqn. (5) yield the following relationships:

$$\begin{aligned} 2\mathbf{P}_{\mathrm{CP_{MCC_0}}} - 3\mathbf{P}_{\mathrm{CP_{MCC_1}}} + \mathbf{P}_{\mathrm{CP_{MCC_2}}} &= 0 \\ \mathbf{P}_{\mathrm{CP_{MCC_n}}} - 3\mathbf{P}_{\mathrm{CP_{MCC_{n+1}}}} + 2\mathbf{P}_{\mathrm{CP_{MCC_{n+2}}}} &= 0 \end{aligned} \quad (13)$$

With these two supplementary conditions, the location of the $(n+3)$ unknown control points can be determined by simply solving the linear system of equations formed through the merging of Eqn. (9) and (13).

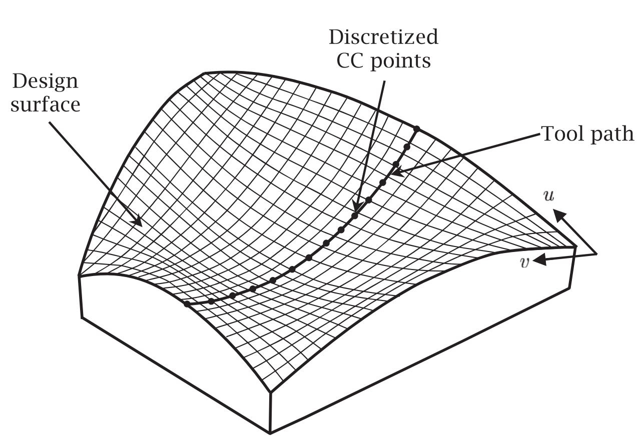

Figure 6. Tool path and discretized CC points on the Bezier surface.

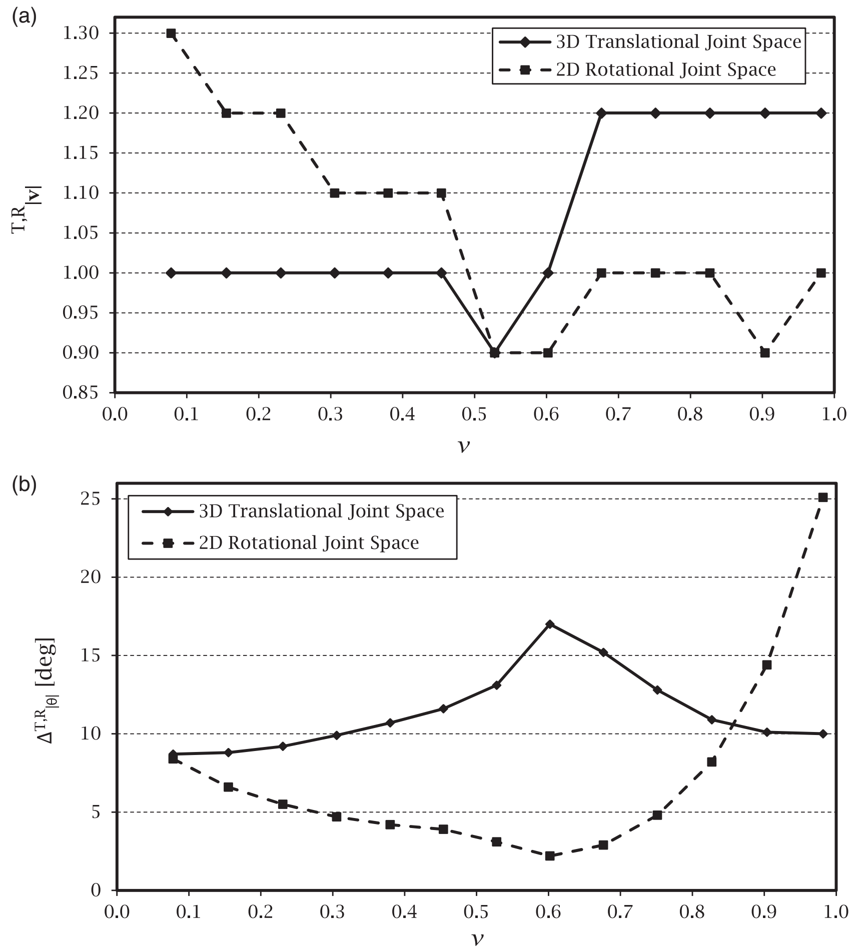

Figure 7. Joint space discontinuities in linearly-interpolated motions along the tool path: (a) velocity magnitude change, and (b) velocity direction change.

3. Implementation and Results

In order to demonstrate the utility of the proposed approach, the magnitude and directional velocity discontinuities were assessed while performing 5D linear interpolations along a sample isoparametric tool path (v = constant) placed on a bicubic Bezier surface patch $\mathbf{P}_S(u,\ v)$ shown in Fig. 6. The approximate dimensions of the surface are 110 × 140 mm. While more complex tool path generation techniques could also be employed, they are considered outside of the scope of the current study that is focused on tool path interpolation schemes.

The five-axis machine tool assumed for this simulation was a vertical spindle-rotating *BA* type, with *B*-axis defined as primary rotary axis. The rotational arm of this machine – defined as the distance from the intersection of the machine's rotary axes to tool's CL point – had a length of 200 mm. The flat-end cutter assumed for this operation had a diameter of 10 mm. The intended isoparametric tool path was discretized based on a chordal deviation of 0.08 mm. Tool postures at each of the discrete CC points were determined from maximum material removal and gouging avoidance constraints and then converted into machine control coordinates by employing an inverse kinematics algorithm.

The angular and magnitude variations of velocity during five-axis linearly-interpolated motions of the cutter along the isoparametric tool path are shown in Fig. 7. It can be noticed that while frequent updates in the direction of both translational and rotational velocities are necessary along the tool path, this is not the case for translational velocity magnitude that tends to remain relatively unchanged for multiple consecutive linearly-interpolated segments. The increased rate of changes in angular velocity can be attributed to the highly nonlinear interdependence between tool orientation and rotational machine control coordinates. On the other hand, the reduced rate of change in translational velocity magnitude can be explained by means of the quasi-constant arc length distances that exist between the successive CC points as are generated by the discretization technique used.

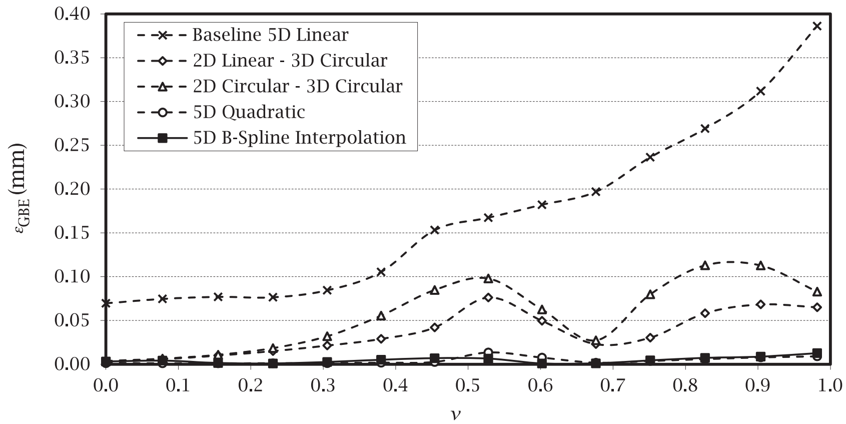

Figure 8. Comparison between geometry-based errors generated in the context of several different types of 5D interpolations.

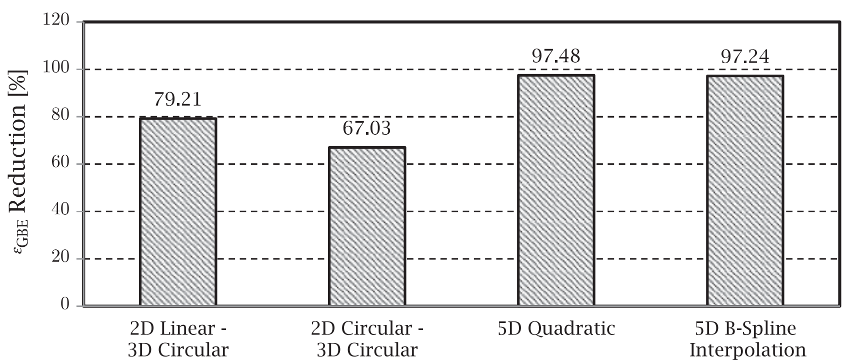

Figure 9. The effectiveness of various interpolation schemes with respect to 5D linear baseline.

Indeed, since machined surface is characterized by relatively low curvature values, chordal deviation algorithm generates CC points that are quasi-equidistant in space and therefore their associated pivot points will also be located at almost equal distances both in workpiece and machine control coordinate systems. As a result, the changes in translational velocity magnitudes required to pass through these points in 3D translational joint space will be minimal. However, as shown in Fig. 7b the direction of the translational velocity has to be tuned continuously, since the selected tool path is a 3D spatial curve that is far from being isoplanar.

All discontinuities outlined by 7 were practically eliminated through introduction of 5D cubic B-Spline interpolation scheme implementing the method described in Section 2.2, above. For this particular tool path, the 15 given 5D MCC space points were interpolated with a B-Spline whose shape was determined by 17 control points.

According to Eqn. (5), the range of variation for curve parameter λ is [0, 14], with a total of 21 knots required. Each distinct knot value is associated with one of the 15 given 5D MCC points.

By comparing the size of the geometry-based errors that were generated by means of the proposed 5D cubic interpolation with three others that were previously [3] analyzed (*e.g.* 2D linear – 3D circular, 2D circular – 3D circular, and 5D quadratic) alongside with the baseline 5D linear technique [25] (Fig. 8), it can be noticed that in addition to C^2 continuity, the new scheme is capable of important machining error reductions with respect to the baseline (Fig. 9).

The explanation of this significant reduction in the magnitude of geometry-based errors resides in the superior approximation of the ideal MCC curve that is enabled by the use of proposed 5D cubic B-Spline interpolation. Furthermore, as Fig. 8 suggests, while no virtually no difference between 5D quadratic and 5D cubic B-Spline exists in terms of machining error, the latter remains superior because of its inherent C^2 continuity. However, the existence of small geometry-based errors indicates that although the 15 targeted ideal tool postures were reached in a synchronized translational and rotational manner, the developed 5D B-Spline interpolation scheme still causes cutter deviations from the desired tool path between the given MCC points.

4. Conclusions

This study proposes an enhanced 5D B-Spline interpolation technique for five-axis sculptured surface machining. Due to the inherent definition of B-Spline curves, a continuous curve is generated in 5D machine control coordinate space, such that no translational or rotational velocity discontinuities are further generated in joint space. In addition to the increased productivity achieved as a result elimination of sudden changes in the magnitude and direction of 5D MCC velocity during linearly-interpolated motions along the tool path, the developed higher-order interpolation scheme enables significant accuracy increases that were quantified through geometry-based error amounts. Indeed, for a certain tool path discretization, 5D B-Spline interpolation introduces significantly lower geometry-based errors along the intended tool path. This notable CAM-originated error reduction/compensation represents in fact a consequence of the improved approximation of 5D ideal MCC trajectory that is enabled by the implementation of the proposed B-Spline interpolation scheme. To the best of authors's knowledge, the proposed approach represents a first attempt to fit a C^2 continous 5D cubic B-Spline to the ideal joint motions represented in the 5D MCC space of a five-axis machine tool.

Further work will attempt to further increase the precision of the developed technique by investigating the possibility to extend the problem into the context of 5D NURBS. This could lead to significant reductions in the number of given data points required for ideal MCC trajectory approximation to be compensated by means of nonuniform weights assigned to the control points of the 5D NURBS curve to be tracked by the numerical controller.

ORCID

O. Remus Tutunea-Fatan http://orcid.org/[0000-0002-1016-5103]

References

[1] Aigner, M.; ir, Z.; Juttler, B.: Evolution-based least-squares fitting using Pythagorean hodograph spline curves, *Computer Aided Geometric Design*, 24(6), 2007, 310–322. http://dx.doi.org/10.1016/j.cagd.2007.04.001

[2] Beudaert, X.; Lavernhe, S.; Tournier, C.: Direct trajectory interpolation on the surface using an open CNC, *International Journal of Advanced Manufacturing Technology*, 75(1–4), 2014, 535–546. http://dx.doi.org/10.1007/s00170-014-6134-7

[3] Bhuiya, M. S. H.; Tutunea-Fatan, O. R.: Reduction of geometry-based errors in five-axis machining through enhanced 5D interpolation, *International Journal of Advanced Manufacturing Technology*, 64(1–4), 2013, 305–317. http://dx.doi.org/10.1007/s00170-012-4009-3

[4] Cheatham, R. M.; Red, W. E.; Jensen, C. G.: Direct process control using n-dimensional NURBS curves, *Computer-Aided Design and Applications*, 2(6), 2005, 825–834. http://dx.doi.org/10.1080/16864360.2005.10738346

[5] Chiou, J. C. J.; Lee, Y.-S.: Five-axis high speed machining of sculptured surfaces by surface-based NURBS path interpolation, *Computer-Aided Design and Applications*, 4(1–6), 2007, 639–648. http://dx.doi.org/10.1080/16864360.2007.10738498

[6] Farouki, R. T.; Han, C. Y.: Algorithms for spatial Pythagoreanhodograph curves, in “Geometric Properties for Incomplete data,” 2006, Springer-Verlag, Secaucus, NJ, USA, 43–58. http://dx.doi.org/10.1007/1-4020-3858-8_3

[7] Farouki, R. T.; Sakkalis, I.: Pythagorean hodographs, *IBM Journal of Research and Development*, 34(5), 1990, 736–752. http://dx.doi.org/10.1147/rd.345.0736

[8] Faux, I. D.; Pratt, M. J.: *Computational Geometry for Design and Manufacture*, Halsted Press, New York, NY, USA, 1979.

[9] Fleisig, R. V.; Spence, A. D.: Constant feed and reduced angular acceleration interpolation algorithm for multi-axis machining, *Computer Aided Design*, 33(1), 2001, 1–15. http://dx.doi.org/10.1016/S0010-4485(00)00049-X

[10] Koren, Y.; Lin, R.-S.: Five-axis surface interpolators, *CIRP Annals - Manufacturing Technology*, 44(1), 1995, 379–382. http://dx.doi.org/10.1016/s0007-8506(07)62346-4

[11] Langeron, J. M.; Duc, E.; Lartigue, C.; Bourdet, P.: A new format for 5-axis tool path computation, using Bspline curves, *Computer Aided Design*, 36(12), 2004, 1219–1229. http://dx.doi.org/10.1016/j.cad.2003.12.002

[12] Lee, K.: *Principles of CAD/CAM/CAE Systems*, Addison-Wesley, Boston, MA, USA, 1999.

[13] Li, H.; Tutunea-Fatan, O. R.; Feng, H.-Y.: An improved tool path discretization method for five-axis sculptured surface machining, *International Journal of Advanced Manufacturing Technology*, 33(9–10), 2007, 994–1000. http://dx.doi.org/10.1007/s00170-006-0529-z

[14] Li, Y.; Wang, Y.; Feng, J.; Yang, J.: The research of dual NURBS curves interpolation algorithm for high-speed five-axis machining, in “Lecture Notes in Computer Science,” 2008, Springer-Verlag, Wuhan, China, 983–992. http://dx.doi.org/10.1007/978-3-540-88518-4_105

[15] Liang, H. B.; Li, X.: Five-axis STEP-NC controller for machining of surfaces, *International Journal of Advanced Manufacturing Technology*, 68(9–12), 2013, 2791–2800. http://dx.doi.org/10.1007/s00170-013-4871-7

[16] Liangji, C.: Algorithm for surface directly interpolated used in 5-axis computer numerical controlled system, In: Proceedings of International Conference on Measuring Technology and Mechatronics Automation (ICMTMA 2009), Zhangjiajie, Hunan, China, IEEE Society, 2009, 175–178. http://dx.doi.org/10.1109/ICMTMA.2009.173

[17] Makhanov, S.: Adaptable geometric patterns for five-axis machining: a survey, *The International Journal of Advanced Manufacturing Technology*, 47(9), 2010, 1167–1208. http://dx.doi.org/10.1007/s00170-009-2244-z

[18] Muller, M.; Erdos, G.; Xirouchakis, P.: High accuracy spline interpolation for 5-axis machining, *Computer Aided Design*, 36, 2004, 1379–1393. http://dx.doi.org/10.1016/j.cad.2004.02.007

[19] Piegl, L.; Tiller, W.: *The NURBS book* (2nd ed.), Springer-Verlag, New York, NY, USA, 1997.

[20] Quan, L.; Yongzhang, W.: The studies of spline interpolation for five-axis machining, In: Proceedings of 4th IEEE Conference on Industrial Electronics and Applications (ICIEA 2009), Xi'an, China, IEEE Society, 2009, 2770–2777. http://dx.doi.org/10.1109/ICIEA.2009.5138714

[21] Rao, A.; Sarma, R.: On local gouging in five-axis sculptured surface machining using flat-end tools, *Computer Aided Design*, 32(7), 2000, 409–420. http://dx.doi.org/10.1016/S0010-4485(99)00105-0

[22] Sencer, B.; Altintas, Y.; Croft, E.: Feed optimization for five-axis CNC machine tools with drive constraints, *International Journal of Machine Tools and Manufacture*, 48(7–8), 2008, 733–745. http://dx.doi.org/10.1016/j.ijmachtools.2008.01.002

[23] Shpitalni, M.; Koren, Y.; Lo, C. C.: Realtime Curve Interpolators, *Computer-Aided Design*, 26(11), 1994, 832–838. http://dx.doi.org/10.1016/0010-4485(94)90097-3

[24] Tutunea-Fatan, O. R.; Feng, H.-Y.: Configuration analysis of five-axis machine tools using a generic kinematic model, *International Journal of Machine Tools and Manufacture*, 44(11), 2004, 1235–1243. http://dx.doi.org/10.1016/j.ijmachtools.2004.03.009

[25] Tutunea-Fatan, O. R.; Feng, H. Y.: Determination of geometry-based errors for interpolated tool paths in five-axis surface machining, *Journal of Manufacturing Science and Engineering-Transactions of the Asme*, 127(1), 2005, 60–67. http://dx.doi.org/10.1115/1.1831285

[26] Tutunea-Fatan, O. R.; Feng, H. Y.: On the B-spline interpolated tool trajectories for five-axis sculptured surface machining, *International Journal of Manufacturing Research*, 1(2), 2006, 198–212. http://dx.doi.org/10.1504/ijmr.2006.011353

[27] Vickers, G. W.; Bradley, C.: Curved surface machining through circular arc interpolation, *Computers in Industry*, 19(3), 1992, 329–37. http://dx.doi.org/10.1016/0166-3615(92)90070-4

[28] Wang, Y.; Liu, Y.; Han, Z.; Shao, Z.: Integration of a 5-axis Spline Interpolation Controller in an Open CNC System, *Chinese Journal of Aeronautics*, 22(2), 2009, 218–224. http://dx.doi.org/10.1016/S1000-9361(08)60090-4

[29] Wang, Y.; Ma, X.; Chen, L.; Han, Z.: Realization methodology of a 5-axis spline interpolator in an open CNC system, *Chinese Journal of Aeronautics*, 20(4), 2007, 362–369. http://dx.doi.org/10.1016/S1000-9361(07)60056-9

[30] Yang, D. C. H.; Kong, T.: Parametric interpolator versus linear interpolator for precision CNC machining, *Computer Aided Design*, 26(3), 1994, 225–234. http://dx.doi.org/10.1016/0010-4485(94)90045-0

[31] Yuan, L.; Yong-Zhang, W.; Zhen-Yu, H.: 5-axis polynomial spline interpolation in an open CNC system, In: Proceedings of 2nd International Symposium on Systems and Control in Aerospace and Astronautics (ISSCAA 2008), Shenzhen, China, IEEE Society, 2008, 1–5. http://dx.doi.org/10.1109/ISSCAA.2008.4776329

[32] Zhao, S.; Zhao, D.; Fu, Y.: Algorithm of variable forward step size planning in freeform surface machining, In: Proceedings of IEEE International Conference on Information and Automation (ICIA 2009), Zhuhai, Macau, China, IEEE Computer Society, 2009, 618–622. http://dx.doi.org/10.1109/ICINFA.2009.5204996