Compensation of machining errors of Bspline and Cspline

El Bechir Msaddek [1] · Maher Baili [2] · Zoubeir Bouaziz [1] · Gilles Dessein [2]

Abstract
The evolution of the interpolation methods towards a very high technicality requires a good choice of used type in the operation of high-speed milling (HSM). The “Bspline” and “Cspline” interpolations present good solutions to guarantee the tool’s contin-uous movement during machining. However, in a previous article, we have shown by a simulation tool that they generate significant dimensional errors that decrease the precision of the machined part. In this article, a method of compensating for these errors based on the insertion of the nodes, while respecting the predefined tolerance, has been developed. To do this, we have modeled and simulated machining errors before and after compensation for each type of interpolation. To validate our results, we have machined a test piece with the compensated and uncompensated Bspline and Cspline interpolations on the Huron KX10 machine and we have measured the corresponding machining errors. The results have shown that the method of com-pensation by the insertion of the nodes causes a significant reduction of the machining errors.

Keywords Bspline · Cspline · Modeling · Simulation · Compensation · Insertion · Nodes · Errors · HSM

1 Introduction

Nowadays, a lot of studies are focused on the types of interpolation that increase the smoothing of the toolpath when machining the free-form parts. Among these interpolations object of the study, we find the polynomial or spline functions (NURBS, Bspline). Our previous research [1, 2] has shown the influence of the interpolation type, such as the functions: Bspline and Cspline, on the machining errors in HSM of a free form (Fig. 1).

As results, we have noticed that the Bspline interpolation generates fewer errors on the convex and concave shapes of the warped shape, but it generates large errors when crossing the discontinuities in tangency. In contrast for the interpolation Cspline, the passage of the trajectory by the reference points increases the precision in changes of direction, but it causes the deceleration of the machine. Indeed, it is necessary to develop a compensation method for these errors to ensure trajectory smoothing while respecting the predefined (CAM) tolerance. The goal is to obtain a relevant precision result about high-speed machining of complex-shaped forms by polynomial interpolations: Bspline and Cspline.

Recent studies have been interested in the compensation of complex shape machining errors. Zuo et al. [3] and Zhu et al. [4] have developed methods of compensation of geometrical errors of machining system NC by the correction of the NC codes. Poniatowska [5] has proposed a method for compensating for systematic errors of free-form surfaces. The method is based on MPM modeling by NURBS interpolation. Zhong et al. [6] have developed a model of identification and compensation of geometric errors of position (5-axis machine) based on servo loops and recursive correction. Raksiri et al. [7] have modeled and compensated machining errors in time masked by “neural network” taking into account cutting forces and geometrical defects. Lei et al. [8] and Hsu et al. [9] have corrected geometric errors in real time or not by compensation algorithms for 5-axis CNC machine tools. Very little research has addressed the problem of errors

✉ El Bechir Msaddek
elbechir.msaddek@gmail.com

Maher Baili
Maher.Baili@enit.fr

Zoubeir Bouaziz
Zoubeir.Bouaziz@enis.rnu.tn

Gilles Dessein
gilles.dessein@enit.fr

[1] ENIS, Unit of Applied Fluids Mechanics of Process Engineering and Environment, University of Sfax, Sfax, Tunisia

[2] ENIT-INPT, Laboratoire Génie de Production, University of Toulouse, Tarbes, France

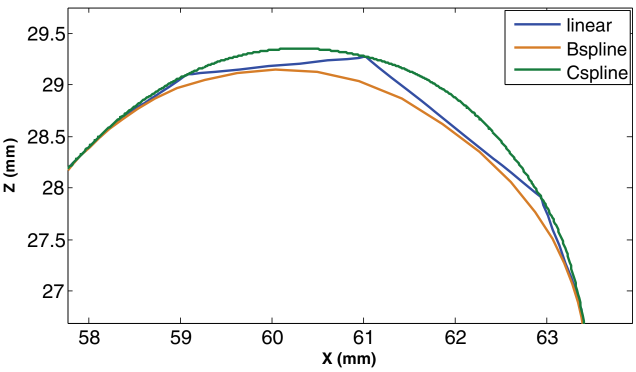


Fig. 1 Interpolations: linear, Bspline, Cspline, and generated errors

generated by interpolations provided by CAM software. Polynomial interpolations as Bspline and Cspline assist in smoothing tool paths of machining free forms, which accelerates the machining operation. So, it is useful to quantify and compensate these errors.

The chosen compensation method is the knot insertion method. After trajectories modeling, we develop a postprocessor for the NC file generation of simulation and errors compensation. First, after the machining simulation of the test piece on the Mastercam© software, we generate the NCI file containing the point coordinates of the toolpath in linear interpolation. Since the Bspline (Bézier spline) interpolation assimilates these points as control points, it is necessary to determine the points of the Bspline trajectory as they are calculated by the controller and simulate them under Matlab©. Modeling is essential to insert other nodes around the control points, so that the final Bspline trajectory supports the CAM theoretical trajectory. On the other hand, the Cspline trajectory calculated by the controller passes through the points generated in CAM but with sometimes undesirable undulations due to a low polynomial degree. The compensation of the ripple errors of this interpolation also involves the modeling of a method of node insertion, forming segments of constant lengths in order to keep the trajectory in the desired tolerance.

In this paper, we show that our compensation method minimizes machining errors and limits large defects from the left-form test piece by using interpolations, Bspline and Cspline.

2 Compensation methodology

2.1 Principle

After the simulation of machining errors of the interpolations, Bspline and Cspline [2], our choice was oriented towards the method of the node insertion, in order to compensate the machining errors. It consists of adding nodes in the toolpath as a means to respect the proposed tolerance. The advantages of this method have been appreciated with regard to the accuracy of the recorded values.

The method of node insertion for Bspline interpolation is different from that applied for Cspline interpolation. The origin of this difference is the mathematical formula of each interpolation.

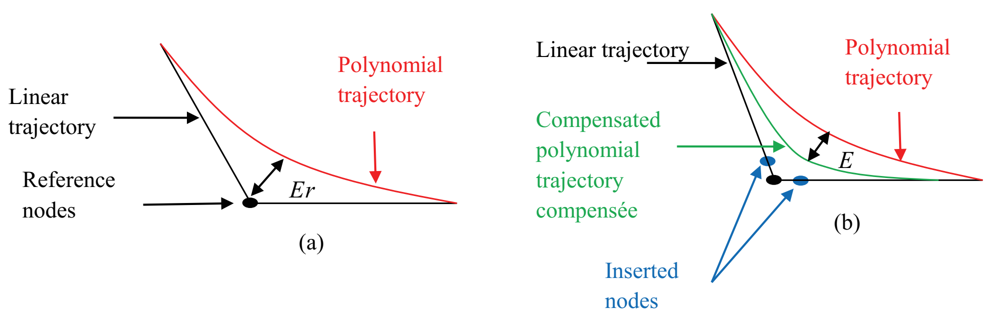
2.2 Definition of errors and inserted nodes

The controller simulator interpolates the points programmed in CAM according to the polynomial interpolation algorithm. In general, the interpolation generated by the CAM system is the linear interpolation. The points of this interpolation are assimilated as control points for the Bspline interpolation, from which we deduce the notion of interpolation error "*Er*." This error defines the difference between the linear trajectory and the polynomial trajectory (Fig. 2a). As a consequence, the compensated error "*E*" defines the difference between the initial polynomial trajectory and that of the compensated (Fig. 2b).

2.3 Model of node insertion in Bspline

The compensation approach of the generated errors using Bspline interpolation adopted inspires from the method of Zhao et al. [10]. As shown in Fig. 3, we have a cubic Bspline curve constructed as a transition curve between the lines $\overline{p_0p_1}$ and $\overline{p_1p_2}$. Zhao's method is based on the principle of inserting nodes so that the crossing error respects the predefined tolerance. Our method inspired by Zhao's idea allows to insert two control points around the initial control

Fig. 2 **a** Interpolation error in Bspline "*Er*." **b** Compensated error "*E*" and inserted nodes



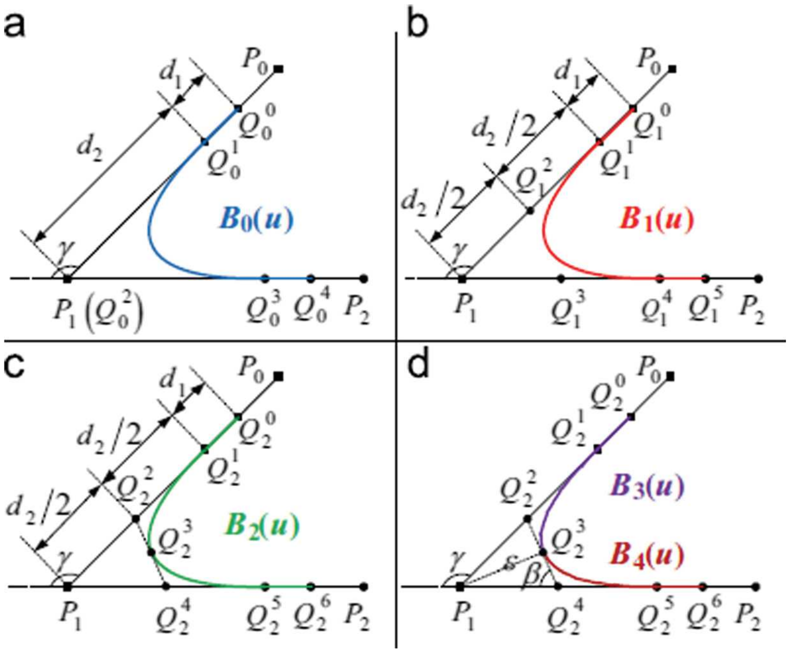


Fig. 3 Method steps of Zhao. **a** Transition curve B_0 (u). **b** Transition curve B_1 (u) obtained by insertion of node. **c** Transition curve B_2 (u) obtained by second insertion of nodes. **d** Two symmetrical Bézier curves B_3 (u) and B_4 (u) obtained by the subdivision of B_2 (u) [10]

point of the tool trajectory generated in CAM while respecting the desired tolerance. The difference between the two methods is that the Zhao method is intended for the C^2 (linear-Bspline) transition trajectories between two blocks, but our method is developed to satisfy the machining of an entire left form by the Bspline interpolation.

$B_0\,(u)$ is symmetrical with respect to the angular bisector of the two lines $\overline{p_0p_1}$ and $\overline{p_1p_2}$. Thus, the maximum deviation between the Bspline and the two lines is the distance between the spline center and P0. The approximation of the error can be expressed as:

The estimated error *Er* can be expressed as:

$$Er = \varepsilon = \left|P_1Q_2^3\right| = (d_2 \sin\beta)/2 \qquad (1)$$

with

d_2 Distance between P_1 and Q_0^1

β Angle between $\overline{p_1p_2}$ and $\overline{Q_2^2Q_2^4}$

To confirm that the estimated error remains within the specified tolerance, the length d_2 shall be calculated as follows:

$$d_2(i) \le 2Er/\sin\beta(i), \quad i = 0, 1, \ldots N-2, \qquad (2)$$

So we must calculate the coordinates (X, Y) of the control nodes to be added for different cases of $\theta(i)$ and $\theta(i+1)$, in such a way that d_2 (i) respects this condition.

The algorithm used is the following:

$$if \quad \theta(i) \prec 0 \quad \& \quad \theta(i+1) \prec 0$$

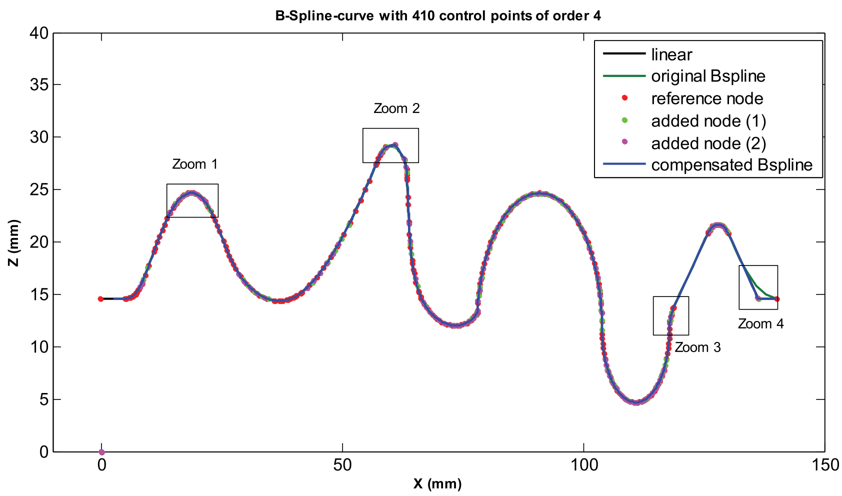


Fig. 4 Trajectories: linear, original, and compensated Bspline

Table 1 Zooms of the Bspline trajectory and simulated values of initial and compensated errors

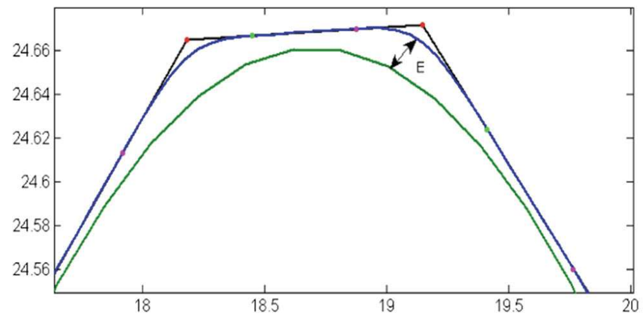
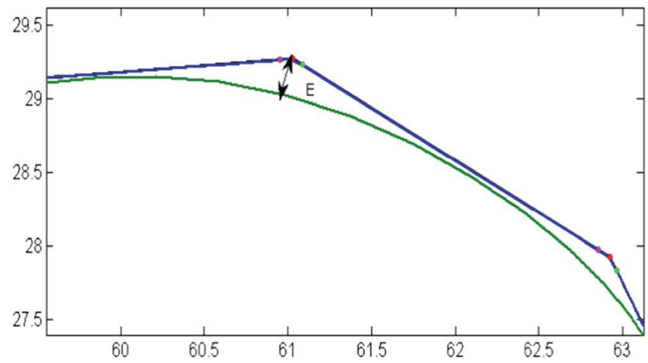
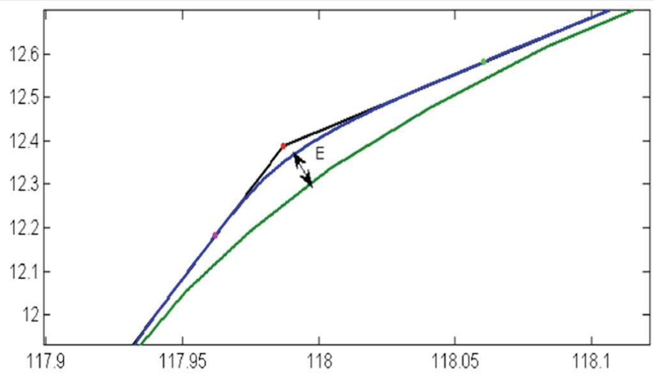
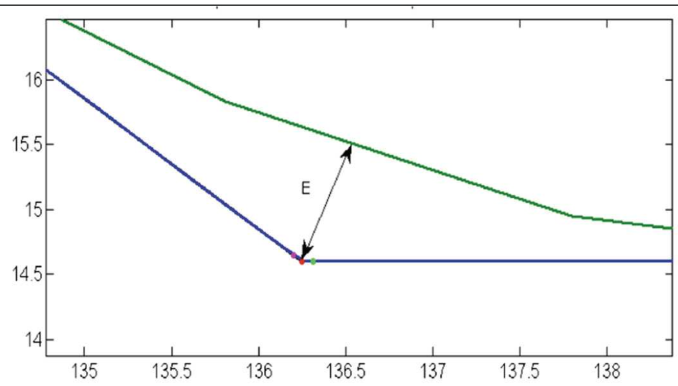
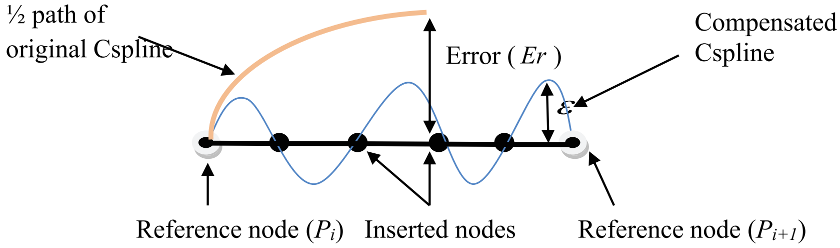
Zoom	Schema	Initial error « Er » (mm)	Compensated error « E » (mm)
1		0.030	0.020
2		0.260	0.250
3		0.160	0.100
4		1.622	1.577

Fig. 5 Original and compensated Cspline on one block



$$Q_1(i,1) = B(i,1) + \cos\theta(i) \times \frac{d_2(i)}{2} \tag{3}$$

$$Q_1(i,2) = B(i,2) - |\sin\theta(i)| \times \frac{d_2(i)}{2} \tag{4}$$

$$Q_2(i,1) = B(i,1) - \cos\theta(i+1) \times \frac{d_2(i)}{2} \tag{5}$$

$$Q_2(i,2) = B(i,2) + |\sin\theta(i+1)| \times \frac{d_2(i)}{2} \tag{6}$$

$$elseif \quad \theta(i) \succ 0 \quad \& \quad \theta(i+1) \succ 0$$

$\vdots$

with

$Q_1(i,1)$	Matrix of the coordinate X of the first inserted nodes.
$Q_1(i,2)$	Matrix of the coordinate Y of the first inserted nodes.
$Q_2(i,1)$	Matrix of the coordinate X of the second inserted nodes.
$Q_2(i,2)$	Matrix of the coordinate Y of the second inserted nodes.
$B(i,j)$	Matrix of the reference nodes.
$\theta(i)$	Angle between two linear blocks

$$\theta(i) = 180 - \gamma(i). \tag{7}$$

Fig. 6 Calculation method of distance d_2

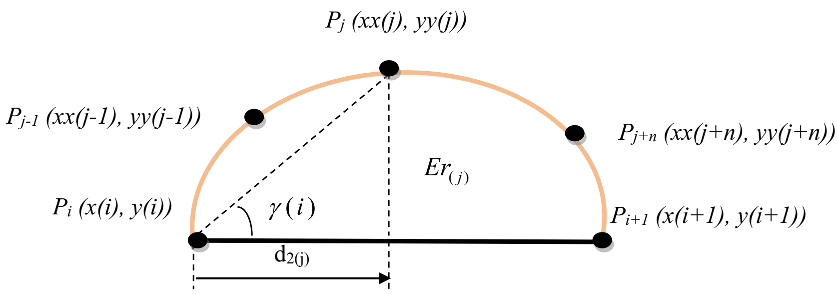


Fig. 7 Interpolations: linear, original, and compensated Cspline

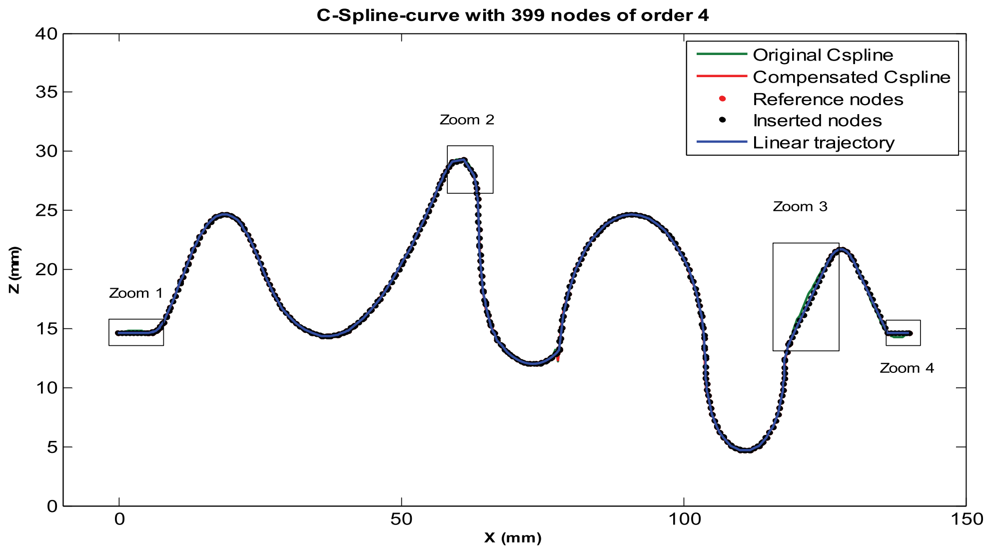


Table 2 Zooms of the Cspline trajectory and simulated values of initial and compensated errors

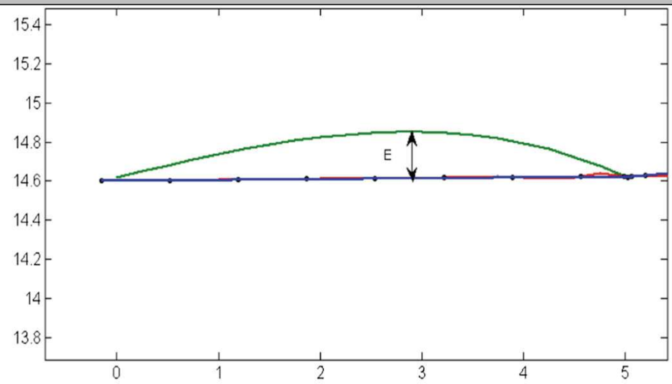
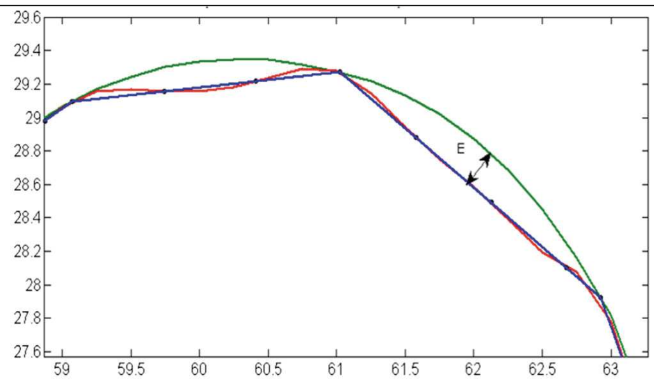
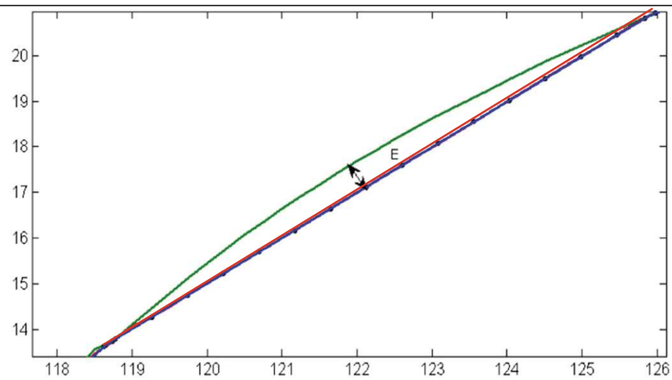
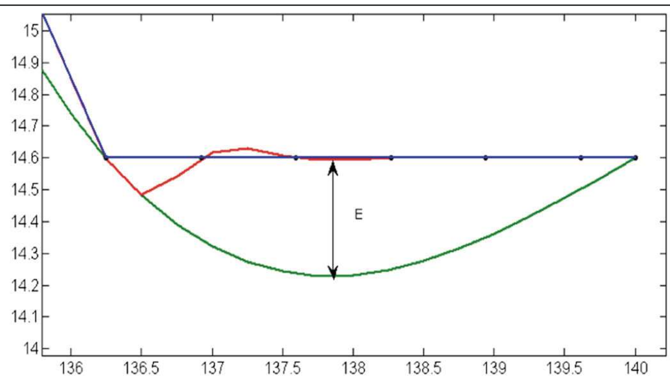
Zoom	Schema	Initial error « Er » (mm)	Compensated error « E » (mm)
1		0.230	0.230
2		0.313	0.310
3		0.600	0.600
4		0.376	0.370

Table 3 Cut parameters of finishing operation

Programmed feed rate	Spindle speed	Workpiece material	Tool	CAM tolerance
4.8 m/min	24,000 rpm	Aluminum	Ball end mill Ø6	25 µm

2.4 Simulation of compensated errors in Bspline

Using the previous model, the compensated Bspline trajectory was simulated and represented in Fig. 4.

Table 1 shows the different zooms of some arbitrarily selected critical areas and the simulated values of the corresponding compensated errors.

The compensation method developed for Bspline interpolation showed good efficiency. Simulated compensation errors can reach more than 1.5 mm in critical areas.

2.5 Model of node insertion in Cspline

The compensation principle of the errors generated by the Cspline interpolation is also based on the insertion of nodes to the reference points generated in CAM while respecting the predefined tolerance. Liang et al. [11] have proposed a real-time interpolator with constant length segments to improve machining of surfaces modeled by NURBS curves. Indeed, we have inspired from this method to add nodes forming segments of constant lengths in order to keep the trajectory in Cspline interpolation in the desired toleranceε.

Figure 5 shows the original and the compensated Cspline interpolation on one block of trajectory having two nodes P_i and P_{i+1}.

In order to calculate the distance between the added nodes, we take a block of two reference points P_i and P_{i+1} of coordinates respectively x(i), y(i) and x(i + 1), y(i + 1). The points P_j, P_{j+1},... P_{j+n} are the calculated nodes of the Cspline trajectory of coordinates respectively xx(j), yy(j), and xx(j + 1), yy(j + 1)..., and xx(j + n), yy(j + n) (Fig. 6).

The error Er(j) is calculated as follows:

For $j = 1,...,n$

$$xx(j) \leq x(i+1) \quad and \quad xx(j) \succ x(i) \tag{8}$$

$$Er(j) = abs\ (yy(j) - y(i)) \tag{9}$$

$$\text{If } Er(j) \leq\ \varepsilon \tag{10}$$

$$d_2(j) = \frac{xx(j) - x(i)}{\cos(\gamma(i))} \tag{11}$$

with $\gamma(i)$: angle between $\overline{p_i p_{i+1}}$ and $\overline{p_i p_j}$

So we find a distance matrix d_2 (i, j), we calculate the average distance of the distances which respect the tolerance ε. Therefore, d_{2avr} is the retained distance between the inserted nodes. Then, we calculate the coordinates of the added nodes by the same algorithm such as the Bspline curve with little difference.

2.6 Simulation of compensated errors in Cspline

Using the previous model, the compensated Cspline trajectory was simulated and represented in Fig. 7.

Table 2 shows the different zooms of some critical areas and values of the simulated initial and compensated errors.

The method of node insertion distributed on equidistant segments has shown good efficiency in the errors correction. Simulated compensation errors can reach more than 0.5 mm in critical areas.

3 Experimental validation

3.1 Machining of the test piece

The experimental validation consists of machining two bands of 10-mm width on the test piece by the Bspline- and Cspline-compensated interpolations, next to eight bands machined by Bspline, Cspline, and other interpolations on the HSM machine Huron KX10 [1].

The used parameters of the finishing operation of the test piece are shown in Table 3.

Figure 8 shows the test piece machined by different interpolations on 10 bands, the bands machined by the original and the compensated Bspline interpolations 1 and 1′, and the bands machined by the original and the compensated Cspline interpolations 2 and 2′.

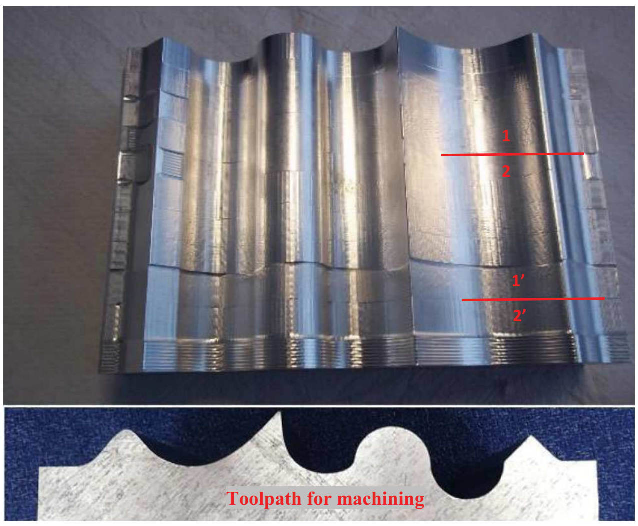
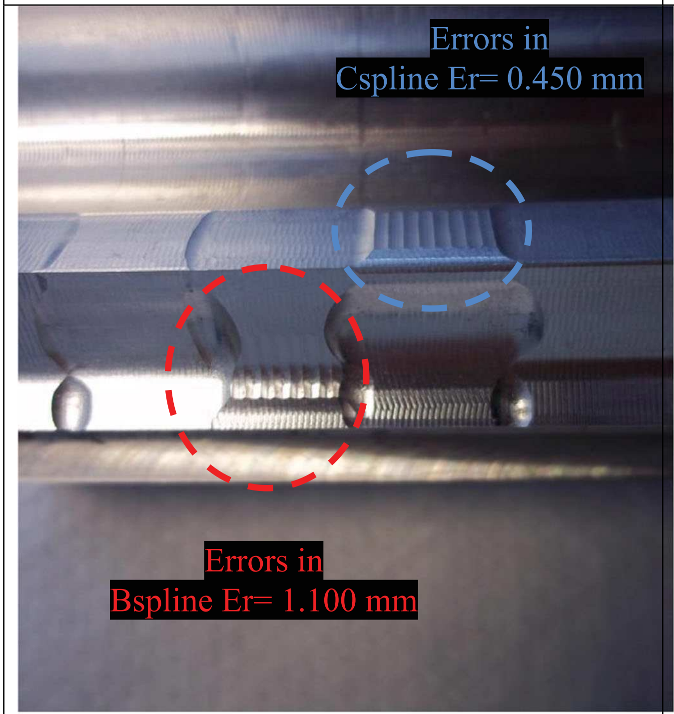


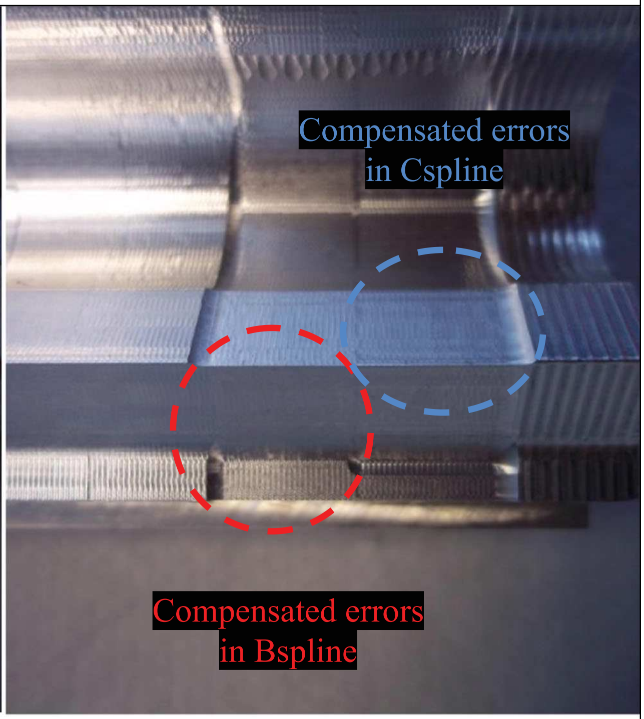
Fig. 8 Test piece machined by original and compensated interpolations: Bspline (1, 1′) and Cspline (2, 2′)

Table 4 Pre- and post-compensation errors for Bspline and Cspline interpolations

Before compensation (area 1)



After compensation (area 1)



Before compensation (area 2)

After compensation (area 2)

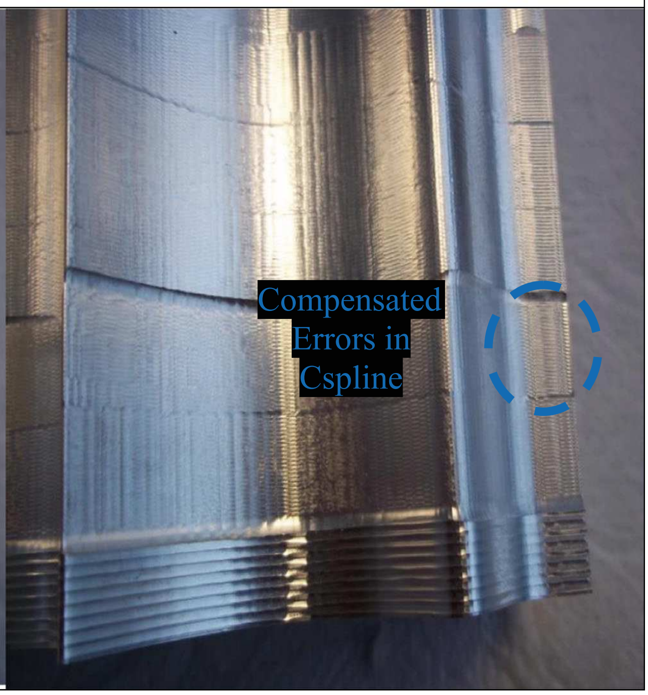


Fig. 9 Histogram of measured errors before and after compensation in the most critical areas

3.2 Measurement of compensated machining errors in Bspline and Cspline

Based on previous machining operations, we have measured the errors before and after the compensation for Bspline and Cspline interpolations using the three-dimensional measuring machine MMT.

Table 4 shows the machining errors before and after compensation in two critical areas of the test piece for the two interpolations: Bspline (encircled in red) and Cspline (encircled in blue).

Figure 9 presents the histogram of the measured errors before and after compensation in the most critical areas.

According to Table 4, the highest compensated error is of the order of 450 μm for Cspline and of the order of 1100 μm for Bspline. The measured errors after compensation decreases until almost zero in some areas (zoom 2). Besides, the experimental compensated errors are very close to theoretical ones, so our model is validated. Indeed, we have shown that our approach is a good solution for machining of warped shapes with high accuracy by smooth interpolations such as spline-based interpolations.

4 Conclusion

In order to compensate the errors caused by the high-speed milling of free forms when using the polynomial (spline) interpolations, we have studied functions such as "Bspline" and "Cspline." After the modeling of the trajectories, we have developed the method of node insertion so as to avoid the machining errors. Finally, we have validated our theoretical approach with experimental tests on the HSM machine Huron KX 10. As a result, we have succeeded to eliminate significant errors of the order of 1100 μm, which proves the effectiveness of this method in terms of accuracy, despite it does not completely eliminate errors. However, machining by these types of interpolations affects very important criteria for the complex shape machining such as cycle time and surface quality. A research work is required to know: is this compensation method favorable towards these criteria or not.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Msaddek EB, Bouaziz Z, Baili M, Dessein G (2014) Influence of interpolation type in high speed machining (HSM). Int J Adv Manuf Technol 72(1–4):289–302
2. Msaddek EB, Bouaziz Z, Baili M, Dessein G, Akrout M (2017) Simulation of machining errors of Bspline and Cspline. Int J Adv Manuf Technol 89(9):3323–3330
3. Xiaoyan Zuo, Beizhi Li, Jianguo Yang, Xiaohui Jiang, Integrated geometric error compensation of machining processes on CNC machine tool, 14th CIRP Conference on Modeling of Machining Operations CIRP CMMO Procedia CIRP 8 (2013) 135–140
4. Zhu S, Ding G, Qin S, Lei J, Zhuang L, Yan K (2012) Integrated geometric error modeling, identification and compensation of CNC machine tools. Int J Mach Tools Manuf 52:24–29
5. Poniatowska M (2015) Free-form surface machining error compensation applying 3D CAD machining pattern model. Comput Aided Des 62:227–235
6. Zhong G, Wang C, Yang S, Zheng E, Ge Y (2015) Position geometric error modeling, identification and compensation for large 5-axis machining center prototype. Int J Mach Tool Manu 89:142–150

7. Raksiri C, Parnichkun M (2004) Geometric and force errors compensation in a 3-axis CNC milling machine. Int J Mach Tool Manu 44: 1283–1291
8. Lei WT, Hsu YY (2003) Accuracy enhancement of five-axis CNC machines through real time error compensation. Int J Mach Tool Manu 43:871–877
9. Hsu YY, Wang SS (2007) A new compensation method for geometry errors of five-axis machine tools. Int J Mach Tool Manu 47:352–360
10. Zhao H, Zhu LM, Ding H (2013) A real-time look-ahead interpolation methodology with curvature-continuous B-spline transition scheme for CNC machining of short line segments. Int J Mach Tool Manu 65:88–98
11. Liang S, Zhao W, Xi X (2013) Design of a real-time NURBS interpolator with constant segment length for milling EDM. Int J Adv Manuf Technol 67:427–440