Corner smoothing of 2D milling toolpath using b-spline curve by optimizing the contour error and the feedrate

Abdullah Özcan, Edouard Rivière-Lorphèvre, and François Ducobu

Corner Smoothing of 2D Milling Toolpath Using B-spline Curve by Optimizing the Contour error and the Feedrate

Abdullah Özcan[1, a)] Edouard Rivière-Lorphèvre[1(b)] François Ducobu[1(c)]

[1] *University of Mons, Machine Design and Production Engineering Lab, Place du Parc 20, 7000 Mons, Belgium*

[a)] Corresponding author: abdullah.ozcan@umons.ac.be
[b)]edouard.rivierelorphevre@umons.ac.be
[c)] francois.ducobu@umons.ac.be

Abstract. In part manufacturing, efficient process should minimize the cycle time needed to reach the prescribed quality on the part. In order to optimize it, the machining time needs to be as low as possible and the quality needs to meet some requirements. For a 2D milling toolpath defined by sharp corners, the programmed feedrate is different from the reachable feedrate due to kinematic limits of the motor drives. This phenomena leads to a loss of productivity. Smoothing the toolpath allows to reduce significantly the machining time but the dimensional accuracy should not be neglected. Therefore, a way to address the problem of optimizing a toolpath in part manufacturing is to take into account the manufacturing time and the part quality. On one hand, maximizing the feedrate will minimize the manufacturing time and, on the other hand, the maximum of the contour error needs to be set under a threshold to meet the quality requirements. This paper presents a method to optimize sharp corner smoothing using b-spline curves by adjusting the control points defining the curve. The objective function used in the optimization process is based on the contour error and the difference between the programmed feedrate and an estimation of the reachable feedrate. The estimation of the reachable feedrate is based on geometrical information. Some simulation results are presented in the paper and the machining times are compared in each cases.

INTRODUCTION

In part manufacturing, reducing the production costs and optimizing the manufacturing time is an important issue. However, the produced part quality should not be neglected. Thus, the cost quality ratio should always be as low as possible [1]. The efficiency of a 2D milling process is directly linked to the feedrate reached by the CNC during the process. For a toolpath defined by sharp corners, the programmed feedrate is different from the reachable feedrate due to kinematic limits of the motor drives. This phenomena leads to a loss of productivity.

Toolpaths with sharp corners need to be smoothen in order to obtain a $\mathcal{G}^2$ continuity curve. This is necessary for the temporal interpolation stage. Indeed, in a sharp corner, the velocity of the tool need to be null in order to satisfy the kinematical limits of the machine tool [2]. Smoothing the corner will avoid this problem.

In the literature, there are several toolpath smoothing methods to optimize the trajectory in term of productivity. A. Dugas [3] proposes a corner smoothing method with an arc, which is tangent to the segments. The maximum value of the contour error is located at the corner center. This method produces only a $\mathcal{G}^1$ continuity curve. Erkorkmaz [4] uses a fifth degree polynomial curve to specify all the connection conditions. Pateloup [5] has developed a method using b-spline curve with 8 control points and Zhao [6] uses a b-spline defined by 5 control points. Beudaert [2] also uses a b-spline defined by 5 control points.

The methods using b-spline curves are all locating the maximum of the error in the corner and the control points are located on the initial toolpath. In this work, the control points' positions are determined in order to minimize the contour error and to maximize the feedrate along the toolpath. The first three points and the last three points are necessarily located on the toolpath. Indeed, to ensure a $\mathcal{G}^2$ continuity between previous segment and the b-spline (also between the b-spline and the following segment), the first three (also the last three) of the control points must be

Proceedings of the 21st International ESAFORM Conference on Material Forming
AIP Conf. Proc. 1960, 070020-1–070020-6; https://doi.org/10.1063/1.5034916
Published by AIP Publishing. 978-0-7354-1663-5/$30.00

located on the toolpath. The optimization is done for the position of the remaining two points. Moreover, the symmetry of the toolpath imposes a link between the positions of these two control points. Thus, there are only two parameters to consider for the optimization process: the position of the 4th control point (in two direction).

This paper presents the optimization of the b-spline. The equation of the b-spline is given in the second section. Then, the contour error estimation and the indicator of the feedrate along the toolpath is presented in the third section. In section 4, the optimization of the contour error and the feedrate indicator, is described. Finally, some simulation results are presented and commented.

LOCAL CORNER ROUNDING USING B-SPLINE CURVE

As seen before, several rounding methods exist in the literature. In this article, a local rounding using a b-spline with eight control points is developed. A $\mathcal{G}^2$ continuous geometry is obtained using a cubic b-spline defined as follow:

$$\boldsymbol{C}(u) = \sum_{i=0}^{7} B_{i3}(u)\boldsymbol{P_i} \tag{1}$$

where $B_{i3}(u)$ are basis functions, P_i are the control points and $u = [0\ 0\ 0\ 0\ 1\ 2\ 3\ 4\ 5\ 5\ 5\ 5]$ is the knot sequence.

The basis function B_{ij} is defined using the Cox-de Boor recursion formula as:

$$B_{i0}(u) = \begin{cases} 1\ if\ u \in [u_i, u_{i-1}[\\ 0\ otherwise \end{cases} \tag{2}$$

$$B_{ij}(u) = \frac{u-u_i}{u_{i+j}-u_i} B_{i,j-1}(u) + \frac{u_{i+j+1}-u}{u_{i+j+1}-u_{i+1}} B_{i+1,j-1}(u) \tag{3}$$

With the chosen knot sequence and the number of control points, the basis function for a cubic b-spline is given in **TABLE 1**.

Once the b-spline is defined, the eight control points' positions need to be determined. As explained in the introduction, the first three points are on the first segment of the initial toolpath and the last three points are positioned on the second segment of the initial toolpath. This is necessary to ensure a $\mathcal{G}^2$ continuity between the previous segment and the b-spline and between the b-spline and the next segment. After that, there are two remaining points that need to be positioned (Fig. 1). In order to conserve the symmetry of the toolpath, the position of the 4th point fixes the position of the 5th point. Thus, we only need to determine the position of the 4th point.

The 4th point's position depends on the contour error and the estimation of the feedrate along the toolpath.

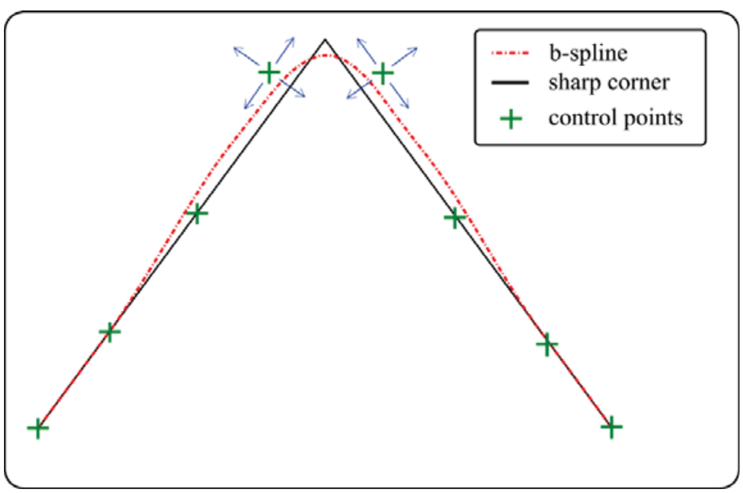

FIGURE 1. Cubic b-spline defined by 8 control points. The first three points and the last three points are on the initial toolpath. The remaining two points are positioned in order to minimize the contour error and maximize the feedrate along the toolpath.

CONTOUR ERROR AND FEEDRATE ESTIMATION

In this section, the contour error estimation and the feedrate estimation is detailed. The position of the 4th control point of the b-spline depends on these information. Indeed, the optimization process will position the 4th point in order to minimize the contour error and maximize the feedrate along the toolpath.

The b-spline is defined using the basis equations presented in the previous section and is expressed in terms of the parameter u. A change of variable is made to express the b-spline in terms of the curvilinear abscissa. If x and y represent the position of the axes, then the velocity, the acceleration and the jerk of each axis can be expressed as:

$$\dot{x} = \frac{dx}{dt} = \frac{dx}{ds}\frac{ds}{dt} = x_s\,\dot{s};\ \dot{y} = \frac{dy}{dt} = \frac{dy}{ds}\frac{ds}{dt} = y_s\,\dot{s} \tag{4}$$

$$\ddot{x} = x_{ss}\,\dot{s}^2 + x_s\,\ddot{s};\ \ddot{y} = y_{ss}\,\dot{s}^2 + y_s\,\ddot{s} \tag{5}$$

$$\dddot{x} = x_{sss}\,\dot{s}^3 + 3x_{ss}\,\dot{s}\,\ddot{s} + x_s\,\dddot{s};\ \dddot{y} = y_{sss}\,\dot{s}^3 + 3y_{ss}\,\dot{s}\,\ddot{s} + y_s\,\dddot{s} \tag{6}$$

where:

- x_s, x_{ss}, x_{sss}, y_s, y_{ss} and y_{sss} represent the geometrical derivative, geometrical second derivative and geometrical third derivative of axis position
- $\dot{s}$, $\ddot{s}$ and $\dddot{s}$ are respectively the feedrate, the acceleration and the jerk of the tool.

With these equations, the geometrical definition (x_s, x_{ss}, x_{sss}, y_s, y_{ss} and y_{sss}) of the toolpath is dissociated from the time planning ($\dot{s}$, $\ddot{s}$ and $\dddot{s}$).

The velocity, acceleration and jerk of each axis are constrained by kinematic limits of the machine:

$$|\dot{x}| \le V_{max}^{x}\ ;\ |\ddot{x}| \le A_{max}^{x}\ ;\ |\dddot{x}| \le J_{max}^{x} \tag{7}$$

$$|\dot{y}| \le V_{max}^{y}\ ;\ |\ddot{y}| \le A_{max}^{y}\ ;\ |\dddot{y}| \le J_{max}^{y} \tag{8}$$

If the equations (4) are combined to the equations (7) and (8), then it is possible to write [2]:

$$\dot{s} \le \min\left(\frac{V_{max}^{x}}{|x_s|}, \frac{V_{max}^{y}}{|y_s|}\right) \tag{9}$$

It is also possible to write a condition on the feedrate based on the acceleration and jerk limitations. But, as seen in equations (5) and (6), the axis acceleration depends on $\dot{s}$ and $\ddot{s}$ and the jerk depends on $\dot{s}, \ddot{s}$ and $\dddot{s}$. Thus, it is hard to express the feedrate $\dot{s}$ in terms of A_{max}, J_{max} and geometrical derivatives.

In order to achieve that, an approximation can be done. When the tool goes through a corner, the feedrate decreases and then increases. In this case, a minimum of feedrate appear and near this minimum value, the acceleration and the jerk can be neglected (when $\dot{s}$ is at the minimum value, $\ddot{s} = 0$ and $\dddot{s} = 0$) [2]. Therefore, form the equations (5), (6), (7) and (8), we have:

$$\dot{s} \le \min\left(\sqrt{\frac{A_{max}^{x}}{|x_{ss}|}}, \sqrt{\frac{A_{max}^{y}}{|y_{ss}|}}\right);\ \dot{s} \le \min\left(\sqrt[3]{\frac{J_{max}^{x}}{|x_{sss}|}}, \sqrt[3]{\frac{J_{max}^{y}}{|y_{sss}|}}\right) \tag{10}$$

As the aim of this work is to maximize the feedrate along the toolpath, the interesting region is where the feedrate reaches the minimum value. Hence, the equations (9) and (10) can be used to approximate the feedrate in the corner and to try to maximize this value in an optimization process.

The second parameter used in the optimization is the contour error. The contour error is the distance between the initial toolpath (segments forming the sharp corner) and the b-spline and is evaluated numerically. Once the whole toolpath has been processed, the maximum value of the contour error is conserved for the optimization stage.

OPTIMIZATION

The b-spline used to round the corner is defined by 8 control points. As explained in the previous section, only the position of the 4th control point need to be determined. In a 2-dimensional space, the position of a point is determined by 2 parameters. The aim of the optimization is to define the b-spline in order to minimize the contour error and optimize the feedrate along the toolpath. Therefore, the parameters that give the position the 4th control point need to be tuned in order to achieve that.

As there are two parameters to optimize (contour error and feedrate), an objective function which depends on these parameters need to be defined. Let F_{obj} be that function:

$$F_{obj} : \mathbb{R}^2 \to \mathbb{R} : (\varepsilon, \gamma) \to Fobj(\varepsilon, \gamma)$$

$$Fobj(\varepsilon, \gamma) = \begin{cases} k.\varepsilon & if\ \varepsilon > \varepsilon_{tol} \\ \frac{1}{\gamma} & if\ \varepsilon \leq \varepsilon_{tol} \end{cases} \tag{11}$$

where ε is the maximum value of the contour error, γ is the minimum value of the estimated feedrate along the toolpath, ε_{tol} is the tolerance on the contour error and k is a constant.

Let call p_x and p_y the position along the x-axis and y-axis of the 4th control point. As the position of the 4th control point define the shape of the b-spline, once p_x and p_y are fixed, it is possible to determine ε and γ. Therefore, ε and γ depends on p_x and p_y. Then, let us define a function G such as:

$$G : \mathbb{R}^2 \to \mathbb{R}^2 : (p_x, p_y) \to G(p_x, p_y) = (\varepsilon, \gamma)$$

By composing the functions F_{obj} and G, the following result is obtained:

$$F_{obj} \circ G : \mathbb{R}^2 \to \mathbb{R} : (p_x, p_y) \to F_{obj}\left(G(p_x, p_y)\right)$$

The optimization process will minimize the value of F_{obj} by tuning the value of p_x and p_y. The value of k is set very high in order to penalize the case $\varepsilon > \varepsilon_{tol}$. Indeed, this case is not suitable because the contour error is out of the tolerance. In the case $\varepsilon \leq \varepsilon_{tol}$, the optimum value is obtained by minimizing $1/\gamma$, which means maximizing γ.

RESULTS

In this section, the optimization of sharp corner rounding is illustrated through simple examples. In each case, a contour error tolerance is fixed. In this paper, 2 cases are presented: $\varepsilon_{tol} = 0.1\ mm$ and $\varepsilon_{tol} = 0.05\ mm$. In each cases, the feedrate is equal to 1500 mm/min. For a given ε_{tol}, the machining time of 3 different strategies are compared: 1) the sharp corner (called "sharp" in the following); 2) the corner rounded by a b-spline with positioning the control points on the segments (called "usual" in the following); 3) the corner rounded by a b-spline obtained with the optimization process (called "optimum" in the following) (Fig. 2).

The machining time is estimated using the software VPOP, developed by LUPRA - ENS Cachan [2]. The kinematic limits of each axis for the simulation are given in **TABLE 1**.

TABLE 1. Kinematic limits of each axis

	X	**Y**	**Z**	**tangential**
Vmax (m/min)	30	30	30	/
Amax (m/s²)	2.5	3	2.1	10
Jmax (m/s³)	5	5	50	100

The following points define the sharp corner (Fig. 2):

- Initial point's coordinates: (0; 0)
- The corner's coordinates: (13; 13)
- Last point's coordinates: (26; 0)

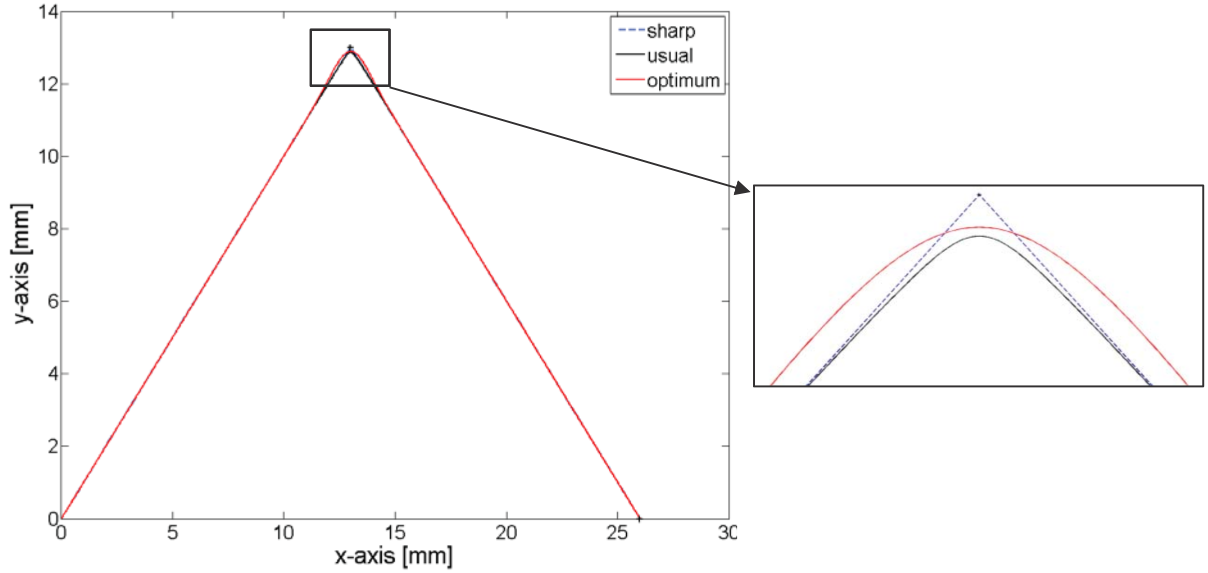

FIGURE 2. The initial toolpath in broken blue line (sharp), the b-spline rounding by placing the control points on the toolpath in continuous black line (usual) and the b-spline rounding using the optimization process in continuous red line (optimum). $\varepsilon_{tol} = 0.1\ mm$.

TABLE 2. The coordinates of the control points for $\varepsilon_{tol} = 0.1\ mm$

	1st	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**
Usual	(10;10)	(10.75;10.75)	(11.5;11.5)	(12.92;12.92)	(13.08;12.92)	(14.5;11.5)	(15.25;10.75)	(16;10)
Optimum	(10;10)	(10.75;10.75)	(11.5;11.5)	(12.67;12.95)	(13.33;12.95)	(14.5;11.5)	(15.25;10.75)	(16;10)

TABLE 3. The coordinates of the control points for $\varepsilon_{tol} = 0.05\ mm$

	1st	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**
Usual	(10;10)	(10.75;10.75)	(11.5;11.5)	(12.99;12.99)	(13.01;12.99)	(14.5;11.5)	(15.25;10.75)	(16;10)
Optimum	(10;10)	(10.75;10.75)	(11.5;11.5)	(12.86;12.99)	(13.14;12.99)	(14.5;11.5)	(15.25;10.75)	(16;10)

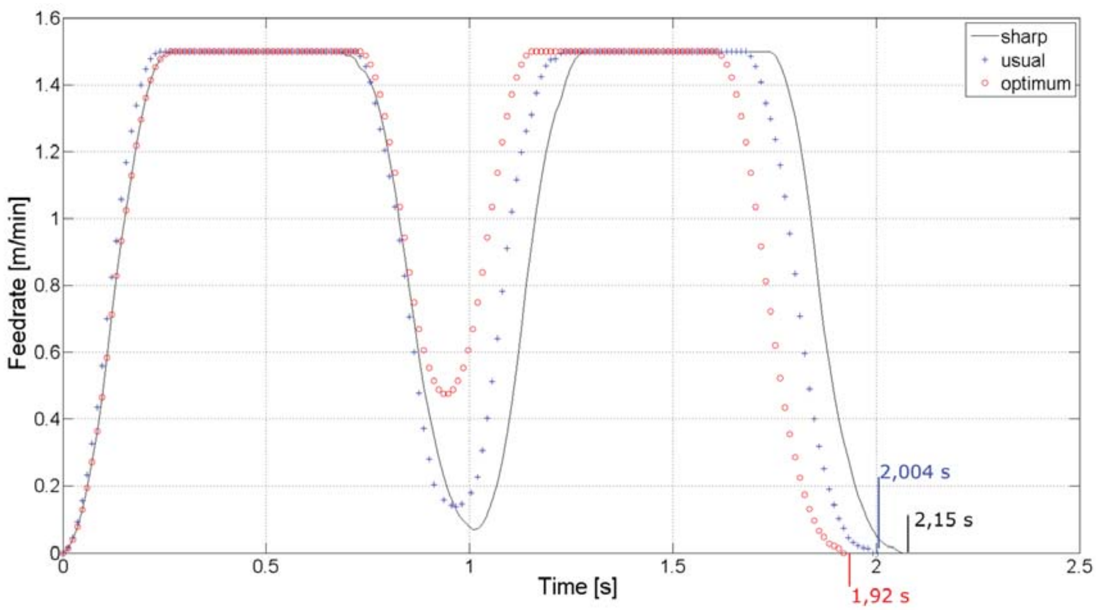

FIGURE 3. The feedrate evolution along the toolpath for $\varepsilon_{tol} = 0.1\ mm$

As seen in Fig. 2, the contour error of the "optimum" toolpath is not located in the corner. In the **TABLE 2** and **TABLE 3**, the coordinates of the 8 control points are given for, respectively, $\varepsilon_{tol} = 0.1\ mm$ and $\varepsilon_{tol} = 0.05\ mm$. The toolpath and the feedrate evolution for $\varepsilon_{tol} = 0.05\ mm$ are not given in this paper as they are very similar to the case $\varepsilon_{tol} = 0.1\ mm$.

The Fig. 3 shows the evolution of the feedrate along the toolpath with $\varepsilon_{tol} = 0.1\ mm$, for each machining strategy (sharp, usual and optimum). The machining times are reported in the TABLE 4. The acceleration at the beginning and deceleration at the end of the toolpath are similar in the 3 cases. In fact, the toolpath is exactly the same in the beginning and in the end (Fig. 2). The difference is around the corner. As expected, the feedrate in the corner for the sharp toolpath is very low (less then 0.1 m/min) while the feedrate for the optimum toolpath is higher than 0,4 m/min. This results in a diminution of 10.7 % in machining time. The optimum toolpath is also better than the usual one in term of machining duration. A diminution of 6.79 % is observed between the usual toolpath and the optimum toolpath.
The same trend is observed in the case $\varepsilon_{tol} = 0.05\ mm$. The optimum toolpath gives a machining time smaller than the usual one and the machining time for the usual toolpath is smaller than the sharp one. But the differences are smaller than the case $\varepsilon_{tol} = 0.1\ mm$.

TABLE 4. Machining time for "sharp", "usual" and "optimum" machining strategy.

	Sharp [s]	**Usual [s]**	**Optimum [s]**	**Difference between sharp and usual [%]**	**Difference between sharp and optimum [%]**
$\varepsilon_{tol} = 0.1\ mm$	2.150 s	2.004 s	1.920 s	6.79 %	10.67 %
$\varepsilon_{tol} = 0.05\ mm$	2.150 s	2.101 s	2.028 s	2.32 %	5.67 %

CONCLUSION

In this paper, a method to round a sharp corner using an optimized bspline has been presented. The bspline used in this work is a cubic bspline with 8 control points. The optimization process try to find the best position of the 4th control point of the bspline in order to minimize the contour error and maximize the feedrate along the toolpath. The contour error is calculated numerically using a MatLab routine. The feedrate estimation is based on the geometrical information of the toolpath. Using the feedrate approximation and the contour error, an objective function is constructed. The aim of the optimization is to find the position of the 4th control point that minimizes the value of the objective function. Thus, the contour error is maximized and the feedrate along the toolpath is minimized.

In the last section, the optimization process is illustrated through some simple examples. The machining time of a sharp corner is estimated (using the software VPOP [2]). Then, for a given contour error tolerance, the bspline used to round the sharp corner is evaluated in two different ways:

- The "usual" method. This is the common method encountered in the literature. The control points of the bspline are positioned on the initial toolpath.
- The "optimum" method. This is the bspline obtained using the optimization process.

A significant difference of the machining time between the different methods is noticed (from 2 to 10 %). We also noticed a smaller difference in machining time for a smaller contour error tolerance.

As a perspective for further work, some experimental validations will be necessary to corroborate the simulation results.

REFERENCES

1. Y. Altintas, K. Erkorkmaz, "Feedrate Optimization for Spline Interpolation In High Speed Machine Tools", CIRP Annals, Vol. 52-1, pp. 297-302, 2003.
2. X. Beudaert, "Open CNC: Optimized interpolation for 5-axis high speed machining of complex surfaces", Ph.D. thesis, Ecole normale supérieur de Cachan – ENS Cachan, 2013.
3. A. Dugas, J.-J. Lee, J.-Y. Hascoët, "An Enhanced Machining Simulator with Tool Deflection Error Analysis", Journal of Manufacturing Systems, Vol. 21-6, pp. 451-463, 2002.
4. K. Erkorkmaz, C.-H. Yeung, Y. Altintas, "Virtual CNC system. Part II. High speed contouring application", International Journal of Machine Tools & Manufacture, Vol. 46, pp. 1124-1138, 2006.
5. V. Pateloup, E. Duc, P. Ray, « Bspline approximation of circle arc and straight line for pocket machining", Computer-Aided Design, Vol. 42-9, pp. 817-827, 2010.