



HAL
open science

5-Axis Flank Milling Tool Path Smoothing Based on Kinematical Behaviour Analysis

Xavier Beudaert, Pierre-Yves Pechard, Christophe Tournier

► **To cite this version:**

Xavier Beudaert, Pierre-Yves Pechard, Christophe Tournier. 5-Axis Flank Milling Tool Path Smoothing Based on Kinematical Behaviour Analysis. 13th CIRP Conference on Modeling of Machining Operations, May 2011, Sintra, Portugal. pp.691-700, 10.4028/www.scientific.net/AMR.223.691 . hal-00626189

HAL Id: hal-00626189

<https://hal.science/hal-00626189>

Submitted on 24 Sep 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

5-axis Flank Milling Tool Path Smoothing Based on Kinematical Behaviour Analysis

BEUDAERT Xavier^{1,a}, PECHARD Pierre-Yves^{2,b} and TOURNIER Christophe^{1,c}

¹LURPA, ENS Cachan / Université Paris Sud 11, 61 av du Pdt Wilson 94230 Cachan (France)

²Missler Software, 7 rue du Bois Sauvage, 91055 Evry (France)

^axavier.beudaert@lurpa.ens-cachan.fr, ^bpy.pechard@topsolid.com,

^cchristophe.tournier@lurpa.ens-cachan.fr

Keywords: Tool path smoothness, 5 axis, flank milling, drive constraints

Abstract. In the context of 5-axis flank milling, the machining of non-developable ruled surfaces may lead to complex tool paths to minimize undercut and overcut. The curvature characteristics of these tool paths generate slowdowns affecting the machining time and the quality of the machined surface. The tool path has to be as smooth as possible while respecting the maximum allowed tolerance.

In this paper, an iterative approach is proposed to smooth an initial tool path. An indicator of the maximum feedrate is computed using the kinematical constraints of the considered machine tool, especially the maximum velocity, acceleration and jerk. Then, joint coordinates of the tool path are locally smoothed in order to raise the effective feedrate in the area of interest. Machining simulation based on a N-buffer algorithm is used to control undercut and overcut. This method has been tested in flank milling of an impeller and can be applied in 3 to 5-axis machining.

Introduction

Within the context of High-Speed Machining, the tool path and machine motions have to be smooth to achieve the required surface quality. Mathematically, the smoothness is usually defined by a continuous second derivative. But it is important to make a clear distinction between a smooth tool path and a smooth motion. A smooth tool path considers only the geometry which means a second derivative with respect to a geometrical parameter (the displacement for example) whereas a smooth motion deals with the temporal movement (i.e. the second derivative with respect to the time). Indeed, you can have a jerky motion even along a straight line or a really quiet travel along a curvy path.

In the literature, several articles deal with the smoothness of the motion. First works were carried out by robotics researchers [1-2]. Nowadays, it is important to limit the jerk in the trajectory planning to produce a soft motion [3-7]. The aim of these articles is to find a velocity profile which respects all the kinematical constraints of the drives and of the machine tool structure for a given tool path. Industrial CNC also offers the possibility to have a jerk limited motion along the tool path [8].

On the other hand, few works had been carried out about geometrical smoothing. A corner optimization is proposed in [9-10] but it is applied only in 3-axis machining. In 5-axis, different methods were proposed to smooth the rotary drives of a 5-axis milling machine [11-13]. The fundamental idea is that the slowdowns on the feedrate come from the rotary drives, which appears to be too restrictive. The method proposed in [14] increases the smoothness by minimizing the energy of deformation of the tool path in the context of 5-axis flank milling. This allows a global optimization of the tool path but as it is realized in the part coordinate system, machine tool constraints are not taken into account. Industrial CNC also provide solutions to smooth

Nomenclature:

s path displacement [mm],
 $s \in [0, L]$ with L the length of the tool path.
 \dot{s} feedrate [mm/s]
 \ddot{s} tangential acceleration [mm/s^2]
 \dddot{s} tangential jerk [mm/s^3]

$\mathbf{q} = [X(s) Y(s) Z(s) A(s) C(s)]^T$
 axes position [mm] and [$degree$].
 $\dot{\mathbf{q}} = [\dot{X}(s) \dot{Y}(s) \dot{Z}(s) \dot{A}(s) \dot{C}(s)]^T$
 axes velocity [mm/s] and [rpm].
 $\ddot{\mathbf{q}} = [\ddot{X}(s) \ddot{Y}(s) \ddot{Z}(s) \ddot{A}(s) \ddot{C}(s)]^T$
 axes acceleration [mm/s^2] and [rad/s^2].
 $\dddot{\mathbf{q}} = [\dddot{X}(s) \dddot{Y}(s) \dddot{Z}(s) \dddot{A}(s) \dddot{C}(s)]^T$
 axes jerk [mm/s^3] and [rad/s^3].

$\mathbf{q}_s = [X_s(s) Y_s(s) Z_s(s) A_s(s) C_s(s)]^T$
 $\mathbf{q}_{ss} = [X_{ss}(s) Y_{ss}(s) Z_{ss}(s) A_{ss}(s) C_{ss}(s)]^T$
 $\mathbf{q}_{sss} = [X_{sss}(s) Y_{sss}(s) Z_{sss}(s) A_{sss}(s) C_{sss}(s)]^T$
 first, second and third derivatives of
 the 5-axis positions with respect to the
 path displacement (s).

$\mathbf{V}_{max}^{axis} = [v_{max}^x v_{max}^y v_{max}^z v_{max}^a v_{max}^c]^T$
 axes velocity limits [m/s] and [rpm].
 $\mathbf{A}_{max}^{axis} = [a_{max}^x a_{max}^y a_{max}^z a_{max}^a a_{max}^c]^T$
 axes acceleration limits [m/s^2] and [rad/s^2].
 $\mathbf{J}_{max}^{axis} = [j_{max}^x j_{max}^y j_{max}^z j_{max}^a j_{max}^c]^T$
 axes jerk limits [m/s^3] and [rad/s^3].

the geometry of the tool path, such as corner rounding functions or tool path compressors [8]. These functions lead to a shorter machining time but the user cannot control the geometrical error generated on the part. Indeed the tolerance is handled axis by axis, which means that the resulting errors on the part in 5-axis milling cannot be controlled.

The main problem of these approaches is that the machine tool characteristics are not considered. Actually, the results of the algorithms will be the same whatever the desired feedrate and the kinematical capacity of the machine tool. However, it is clear that depending on the relative abilities of each drive the solution should change. For example in [15], kinematical constraints were included in the tool path generation.

The prediction of the velocity profile generated by the CNC was used in [16] to improve the machining time by changing the orientation of the tool. Although the complete motion planning is the best way to see where the feedrate is decreasing, it is time consuming and not necessary for the purpose of trajectory smoothing.

Another way to improve the machining time is to use a polynomial description of the tool path with a good parameterization as it is shown in [17-19]. This is used on top of the proposed algorithm to reduce the machining time even more.

In this paper, a decoupled approach is used as opposed to the coupled approach, which deals with the geometry and the motion at the same time [20]. This means that the tool path is first optimized and then the motion planning optimizes the velocity along this optimized path.

The proposed approach consists in computing the necessary reductions of the feedrate due to velocity, acceleration and jerk constraints of each drive. This information, taken from the field of smooth motion planning, is then used to optimize the geometry and smooth the tool path. This will finally lead to a reduction of the machining time.

As it is shown in Fig. 1, an evaluation of the kinematical constraints is first performed, and then a local geometrical smoothing is carried out on the selected axis. If needed, a N-buffer technique [21] is used to control the geometrical deviation on the part. So a compromise has to be made between the smoothness and the geometrical tolerance.

The rest of the paper is organized as follows: kinematical constraints are presented in section 2. These constraints give the axes and positions which have to be smoothed. Using this information the local geometrical smoothing is explained in section 3. A machining test is carried out in section 4. Results demonstrate the efficiency of the proposed algorithm by measuring the machining time. Finally the conclusions are summarized in section 5.

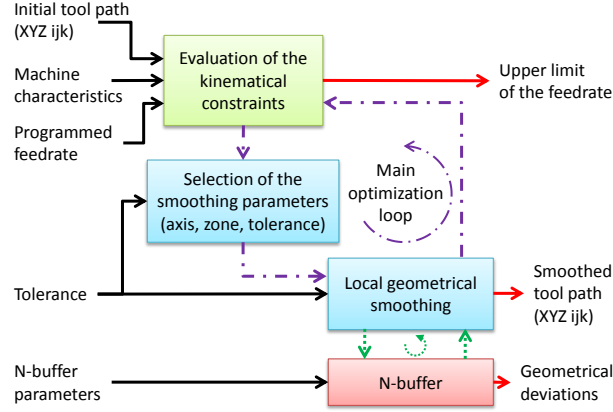


Fig. 1: Block diagram of the algorithm.

Kinematical Constraints

The aim of this section is to predict the slowdowns of the feedrate. A complete prediction of the velocity along the path would have been time consuming so an approximation of the maximum velocity profile is computed. This approximate maximum limit is given by the velocity, acceleration and jerk constraints of the 5 drives.

First, the mathematical formalism is introduced and then the constraints are exposed.

Mathematical Formalism Using the formula for the derivative of the composition of two functions, Eq. 1, it is possible to express the velocity of the drives $\dot{\mathbf{q}}$ as a function of the geometry \mathbf{q}_s multiplied by a function of the motion \dot{s} . Therefore, the motion is decoupled of the geometry which allows the further geometrical optimization. One can note that this formula is valid for linear and rotary axes; it is thus possible to compare the 5 drives of the machine tool at the same time which is an important advantage. The acceleration $\ddot{\mathbf{q}}$ and jerk $\dddot{\mathbf{q}}$ of the drives are obtained identically in Eq. 2 and 3.

$$\dot{\mathbf{q}}(s) = \frac{dq(s)}{dt} = \frac{dq(s)}{ds} \frac{ds}{dt} = \mathbf{q}_s(s) \dot{s} \quad (1)$$

$$\ddot{\mathbf{q}}(s) = \mathbf{q}_{ss}(s) \dot{s}^2 + \mathbf{q}_s(s) \ddot{s} \quad (2)$$

$$\dddot{\mathbf{q}}(s) = \mathbf{q}_{sss}(s) \dot{s}^3 + 3\mathbf{q}_{ss}(s) \dot{s} \ddot{s} + \mathbf{q}_s(s) \dddot{s} \quad (3)$$

\mathbf{q}_s , \mathbf{q}_{ss} , \mathbf{q}_{sss} are the geometrical derivatives with respect to displacement s along the tool path. They should be known as soon as the path is defined. However, the CNC has some options to round the sharp corners so the executed geometry is modified as well as the amplitude of these parameters. To overcome this problem two solutions are available: whether to have a model of the way the CNC is rounding the corners or to send a native B-Spline tool path to the machine. With the second solution, the tool path has no sharp corner so the geometry is not modified by the CNC.

Thanks to the Eq. 1-3 it is now possible to express the constraints of the drives.

Velocity, Acceleration and Jerk Constraints Because of the physical realization of the drives (motors, driving system, machine tool structure ...) the velocity, acceleration and jerk of each individual drive have to be limited. The jerk limitation is important to reduce the vibration due to the dominating vibratory mode of the axes.

Eq. 4 presents the velocity constraints.

$$- \begin{bmatrix} v_{max}^x \\ v_{max}^y \\ v_{max}^z \\ v_{max}^a \\ v_{max}^c \end{bmatrix} \leq \begin{bmatrix} X_s(s) \\ Y_s(s) \\ Z_s(s) \\ A_s(s) \\ C_s(s) \end{bmatrix} \dot{s} \leq \begin{bmatrix} v_{max}^x \\ v_{max}^y \\ v_{max}^z \\ v_{max}^a \\ v_{max}^c \end{bmatrix} \quad (4)$$

All the constraints are set to be symmetric as it is commonly used in the machine tool characteristics. Then we obtain the following set of inequations respectively for the velocity, acceleration and jerk constraints. The notation $||$ stands for the absolute value of each term.

$$|\dot{\mathbf{q}}| \leq \mathbf{V}_{max}^{axis} ; |\ddot{\mathbf{q}}| \leq \mathbf{A}_{max}^{axis} ; |\dddot{\mathbf{q}}| \leq \mathbf{J}_{max}^{axis} \quad (5)$$

The aim is now to find the maximum value of the feedrate \dot{s} allowed by these constraints. To obtain an exact solution, the Eq. 5 should be solved recursively because of the link between \dot{s} , \ddot{s} and $\ddot{\dot{s}}$. However a good approximation can be given in a closed form as it is explained below.

Approximation of the Maximal Feedrate The first inequation is easy to use as it gives immediately the highest feedrate allowed by the velocity of the axes.

$$\dot{s} \leq \min \left(\frac{\mathbf{V}_{max}^{axis \ i}}{|\mathbf{q}_s^i|} \right) \text{ with } i = 1..5 \quad (6)$$

But in the other inequations, there is the link $\ddot{s} = \frac{d}{dt}(\dot{s})$. As we are only looking for an approximate upper limit of the feedrate, it is possible to use the limit when $\ddot{s} = 0$. This will be exact at sharp corners as the feedrate is decreasing and then increasing. Furthermore these are the most interesting areas for us. Thus a limit given by the accelerations of the axes is obtained in a close form:

$$\dot{s} \leq \min \left(\sqrt{\frac{\mathbf{A}_{max}^{axis \ i}}{|\mathbf{q}_{ss}^i|}} \right) \text{ with } i = 1..5 \quad (7)$$

For the jerk, we face the same kind of problem. Taking $\ddot{s} = 0$ and $\ddot{\dot{s}} = 0$, a limit of the feedrate given by the jerks is obtained.

$$\dot{s} \leq \min \left(\sqrt[3]{\frac{\mathbf{J}_{max}^{axis \ i}}{|\mathbf{q}_{sss}^i|}} \right) \text{ with } i = 1..5 \quad (8)$$

Practically, these limits give really good indications about the real feedrate reached by the machine tool as it will be shown in the example below.

Of course, the feedrate is limited by the programmed feedrate F_d . Discretizing the tool path, it is possible to obtain an approximate limit of the real feedrate for each position along the path with the Eq. 9. It is important to point out that this evaluation is really fast as there is no iteration.

$$\dot{s} \leq \min \left(F_d, \frac{\mathbf{V}_{max}^{axis \ i}}{|\mathbf{q}_s^i|}, \sqrt{\frac{\mathbf{A}_{max}^{axis \ i}}{|\mathbf{q}_{ss}^i|}}, \sqrt[3]{\frac{\mathbf{J}_{max}^{axis \ i}}{|\mathbf{q}_{sss}^i|}} \right) \quad (9)$$

Finally, taking into account all the constraints of the drives, we are able to predict the areas where the real feedrate will decrease. Moreover, we know the cause of the slowdowns so we are able to smooth the corresponding axes in these areas.

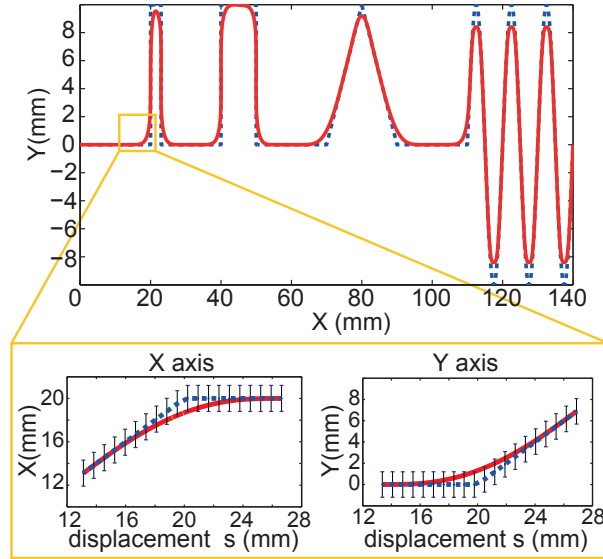


Fig. 2: 2D example tool path.

Local Geometrical Smoothing

Thanks to Eq. 9 of the previous section, the slowdowns of the feedrate are highlighted. So we know where each axis has to be smoothed in order to raise the real feedrate. Applying several times a local smoothing allows increasing the upper velocity limit by reducing the values of \mathbf{q}_s , \mathbf{q}_{ss} , \mathbf{q}_{sss} .

Selection of the Smoothing Parameters Three parameters have to be defined to apply the smoothing algorithm: the tolerance on the axis movement, the axis which has to be smoothed and the zone where it has to be smoothed. The tolerance on the axis movement is chosen with a heuristic guess (equal to the part tolerance for the linear as well as for the rotary drives) and a N-buffer technique is used to ensure that the tolerance on the part is respected. This is mandatory to smooth 5-axis flank milling tool paths.

So the first parameter is handled thanks to a second loop Fig. 1. Then section 2 gives the axis and the position where the axis has to be smoothed. The main difficulty comes from the selection of the width of the zone which is an adjustable parameter. It depends mainly on the shape of the axis movement. With a really small zone, there is not enough freedom whereas with a large zone, you can be stuck because the part tolerance is exceeded in a small area.

Local Geometrical Constraints As the axis is locally smoothed, the connection between the initial axis movement and the smoothed zone should be at least C2 in order to avoid the slowdowns due to the junction discontinuities. So the geometrical derivatives at the extremity of the smoothed zone have to be specified. The objective of the function used [22] is to obtain the smoothest curve, that is to say to minimize the second derivative with respect to the displacement. Finally the smoothing polynomial curve should respect a given geometrical tolerance. One should notice that models based on circle arcs are irrelevant here because they would create a discontinuity at the junctions.

Detailed Algorithm To explain precisely how this method is working, it is applied on a simple 2D path even if the algorithm is not designed to be applied in such an easy case. The tolerance used is really large for the purpose of this explanation Fig. 2. For each sharp corner,

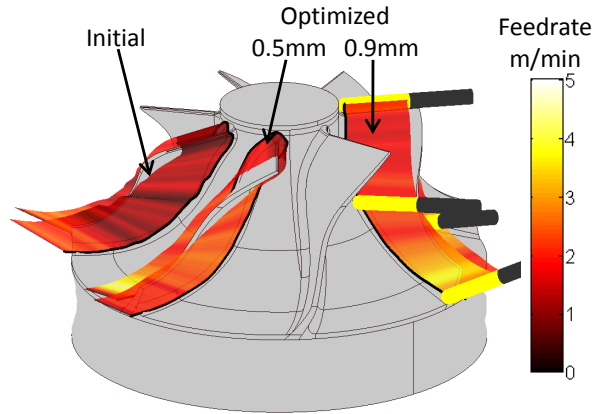


Fig. 3: Impeller with tool paths.

the second geometrical derivative is infinite $\mathbf{q}_{ss} \rightarrow \infty$ so according to Eq. 2 the feedrate has to be zero to respect the inequations 5. That means that each axis instruction has to be modified as it is shown in the bottom of Fig. 2. We see that the derivatives at the junctions are specified as well as the axis tolerance. Thanks to an optimization, it is possible to find the piecewise polynomial curve which minimizes its second derivative while respecting the constraints.

The following simplified algorithm explains the proposed method to smooth a given tool path.

```

While improvement is possible
  Find the lowest point of the upper limit
  Chose the smoothing parameters
  Smooth the corresponding axis
  Compute geometrical deviations (N-buffer)
  While part out of tolerance
    Reduce the axis tolerance
  end
  Evaluate the new upper limit
end

```

The smoothing algorithm can be applied several times in the same area. But as one can see, the part tolerance is controlled all along the optimization process. That means that when the part tolerance is completely used in an area, this area is set as fully optimized and cannot be modified anymore.

After few iterations of this algorithm, a really smoothed tool path can be obtained as one can see on the top of Fig. 2. We can notice that it is working identically on the sine wave and that the minimum curvature's areas are smoothed.

Application

The proposed smoothing algorithm is applied on a complex industrial impeller [14] presented in Fig. 3. The experiments are carried out on a 5-axis MIKRON UCP 710 machining center; the kinematical characteristics are given in Table 1. The programmed feedrate and tangential acceleration are set to $F_d = 5000m/min$ and $A_{tan} = 10m/s^2$, respectively. The machine is controlled thanks to a SIEMENS 840D CNC which allows the measurement of the position and velocity of each axis during the movement.

Table 1: Machine tool drive limits.

	X	Y	Z	A	C
\mathbf{V}_{max}^{axis} [m/min - rpm]	30	30	30	15	20
\mathbf{A}_{max}^{axis} [m/s ² - r/s ²]	2.5	3	2.1	0.83	0.83
\mathbf{J}_{max}^{axis} [m/s ³ - r/s ³]	5	5	50	5	100

The blades are machined in flank milling and as they are non-developable ruled surfaces overcuts and undercuts are unavoidable. The initial tool path is obtained minimizing the geometrical deviations as explained in [14].

As we are in a flank milling operation, the orientation of the tool has to be controlled accurately. A N-buffer technique is introduced in the optimization loop to control the geometrical errors on the part Fig. 1.

Indeed during the local smoothing it is impossible to have a relation between the tolerance on each axis and the resulting effect on the part. To overcome this problem, the direct kinematical transformation is realized and the geometry is checked for each iteration.

Table 2 presents the results. The initial tool path has a maximum deviation of 0.4 mm and the aim was to smooth the tool path as much as possible with an allowed tolerance of 0.5 mm and 0.9 mm. For each tool path, two tests have been carried out with G1 and native B-Spline description of the path. First of all, one can see that the tolerance specified is respected. Then the machining time is drastically reduced whatever the format of the tool path. We see that for the tolerance of 0.9 mm, the geometrical deviation is increased a lot whereas the reduction of the machining time is small. Of course, using a native B-Spline the machining time is reduced because the tool path is smoother than with G1 discontinuities. The estimated machining time is computed assuming that the feedrate is always equal to the limit calculated in section 2. Doing this we under estimate the real machining time but that gives a good evaluation of its magnitude. The estimated machining time on the G1 tool path is obtained using the exact geometry measured on the machine tool as the model to go through G1 5-axis discontinuities is not valid yet.

Table 2: Results of the optimization.

Impeller	Undercut [mm]	Overcut [mm]	Estimated machining time [s]	Measured machining time [s]
Initial tool path	0.13	-0.39	G1:19.5 BS:7.3	G1:23.5 BS:9.3
Optimized tool path tol=0.5mm	0.40	-0.49	G1:12.1 BS:3.7	G1:13.7 BS:6.4
Optimized tool path tol=0.9mm	0.89	-0.66	G1:9.7 BS:3.3	G1:11.3 BS:6.0

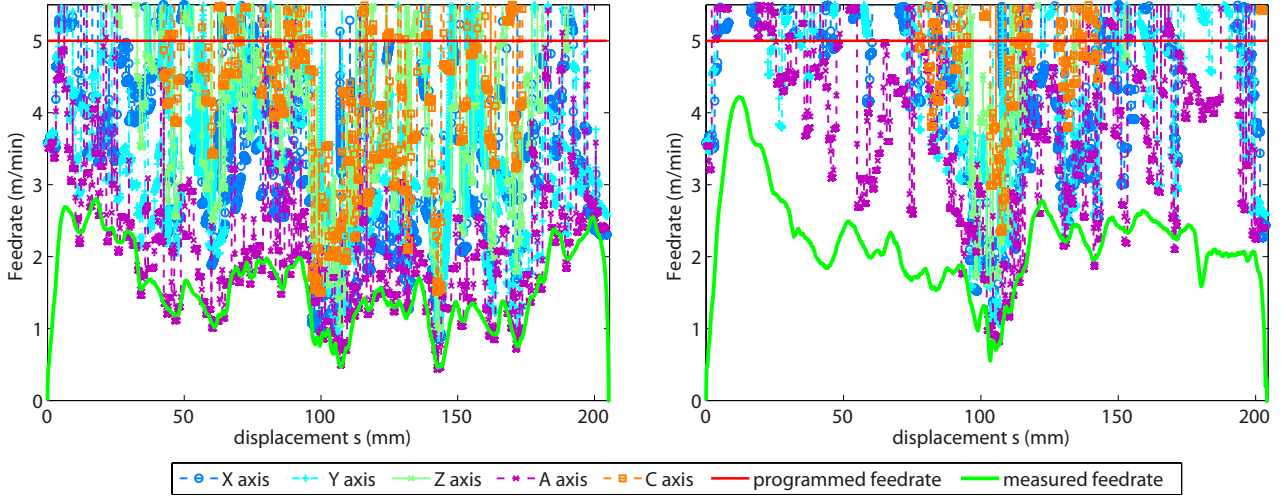


Fig. 4: Approximate upper limit of the feedrate Left: Initial B-Spline tool path, Right: Optimized B-Spline tool path $\text{tol}=0.5$ mm.

Fig. 4 shows the limits given by Eq. 9 presented in section 2. The graphics correspond respectively to the initial (left) and optimized with a tolerance of 0.5 mm (right) tool paths. The results shown here are obtained using the native B-Spline format in order to avoid the discontinuity problems. For these examples, the feedrate is limited by the jerk most of the time. We can notice that the correlation between the approximate upper limit and the measured feedrate is striking. Moreover, the X and Y axes limits are not far from the A axis limit so it shows that the linear axes have to be taken into account too to smooth 5-axis tool paths.

We see that the optimization smoothed the tool path and that the computed feedrate limits are increased. One can notice that the programmed feedrate is never reached. On the right plot, the feedrate between 0 and 100 mm could have been higher. The difference between the predicted upper limit and the measured feedrate comes from an unpredictable behaviour of the algorithm of the industrial CNC.

Fig. 5 presents the result obtained on the C axis. We clearly see that the oscillations are filtered; so the geometrical derivatives C_s, C_{ss}, C_{sss} are decreased and the tool path is smoother.

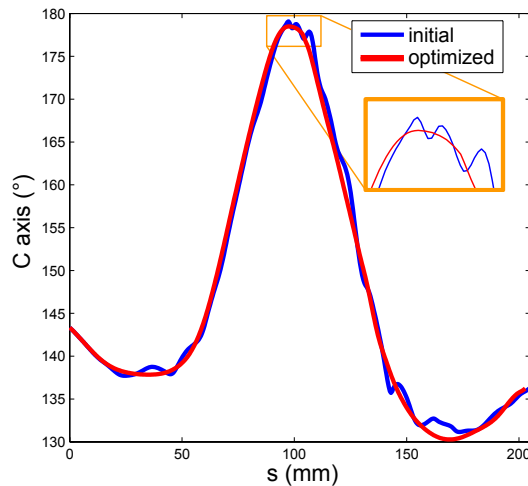


Fig. 5: C-axis initial and optimized ($\text{tol}=0.5$ mm) instructions.

Conclusion

In this paper, a new approach is proposed to address the problem of tool path smoothing. We first made a distinction between a smooth tool path and a smooth motion along this tool path. The kinematical behavior of the machine tool is taken into account to smooth the trajectory thanks to a simple evaluation of the upper limit of the velocity profile along the tool path. This evaluation is based on the maximum velocity, acceleration and jerk of each drive.

Once the critical areas are located, a local smoothing of the joint geometrical evolution is carried out to improve the smoothness at these critical points. The illustrated experimental results show a significant reduction of the machining time. Moreover, the possibility to handle a geometrical tolerance was demonstrated on the 5-axis flank milling of an impeller blade.

References

- [1] K.G. Shin and N.D. McKay: Minimum-Time control of robotic manipulators with geometric path constraints, *IEEE Transactions on Automatic Control*, Vol. 30 (1985) p.531
- [2] J.E. Bobrow, S. Dubowsky and J.S. Gibson: Time-optimal control of robotic manipulators along specified paths, *International Journal of Robotics Research*, Vol. 4 (1985) p.3
- [3] K. Erkorkmaz and Y. Altintas: High speed CNC system design. part I: jerk limited trajectory generation and quintic spline interpolation, *International Journal of Machine Tools and Manufacture* Vol. 41 (2001) p.1323
- [4] B. Sencer, Y. Altintas and E. Croft: Feed optimization for five-axis CNC machine tools with drive constraints, *International Journal of Machine Tools and Manufacture*, Vol. 48 (2008) p.733
- [5] J-Y. Lai, K-Y. Lin, S-J. Tseng and W-D. Ueng: On the development of a parametric interpolator with confined chord error, feedrate, acceleration and jerk, *The International Journal of Advanced Manufacturing Technology*, Vol. 37 (2008) p.104
- [6] M. Heng and K. Erkorkmaz: Design of a NURBS interpolator with minimal feed fluctuation and continuous feed modulation capability, *International Journal of Machine Tools and Manufacture*, Vol. 50 (2010) p.281
- [7] A. Olabi, R. Béarée, O. Gibaru and M. Damak: Feedrate planning for machining with industrial six-axis robots, *Control Engineering Practice*, Vol. 18 (2010) p.471
- [8] SIEMENS, SINUMERIK, 2009, Manual, 5-axis machining.
- [9] V. Pateloup, E. Duc and P. Ray: Corner optimization for pocket machining, *International Journal of Machine Tools and Manufacture*, Vol. 44 (2004) p.1343
- [10] X. Pessoles, Y. Landon and W. Rubio: Kinematic modelling of a 3-axis NC machine tool in linear and circular interpolation, *International Journal of Advanced Manufacturing Technology*, Vol. 47 (2010) p.639
- [11] C.-S. Jun, K. Cha and Y.-S. Lee: Optimizing tool orientations for 5-axis machining by configuration-space search method, *Computer-Aided Design*, Vol. 35 (2003) p.549
- [12] M.-C. Ho, Y.-R. Hwang and C.-H. Hu: Five-axis tool orientation smoothing using quaternion interpolation algorithm, *International Journal of Machine Tools and Manufacture* Vol. 43 (2003) p.1259
- [13] C. Castagnetti, E. Duc and P. Ray: The domain of admissible orientation concept: A new method for five-axis tool path optimisation, *Computer-Aided Design* Vol. 40 (2008) p.838

- [14] P.-Y. Pechard, C. Tournier, C. Lartigue and J.-P. Lugarini: Geometrical deviations versus smoothness in 5-axis flank milling, *International Journal of Machine Tools and Manufacture* Vol. 49 (2009) p.454
- [15] T. Kim and S.E. Sarma: Toolpath generation along directions of maximum kinematic performance; a first cut at machine optimal paths, *Computer-Aided Design*, Vol. 34 (2002) p.453
- [16] S. Lavernhe, C. Tournier and C. Lartigue: Kinematical performance prediction in multi-axis machining for process planning optimization, *International Journal of Advanced Manufacturing Technology* Vol. 37 (2008) p.534
- [17] R. Fleisig and A. Spence: A constant feed and reduced angular acceleration interpolation algorithm for multi-axis machining, *Computer-Aided Design*, Vol. 33 (2001) p.1
- [18] C. Lartigue, C. Tournier, M. Ritou and D. Dumur: High-performance NC for HSM by means of polynomial trajectories, *CIRP Annals - Manufacturing Technology*, Vol. 53 (2004) p.317
- [19] J. Langeron, E. Duc, C. Lartigue and P. Bourdet: A new format for 5-axis tool path computation, using bspline curves. *International Journal of Machine Tools and Manufacture*, Vol. 36 (2004) p.1219
- [20] G. Pardo-Castellote and R.H. Cannon: Proximate time-optimal algorithm for on-line path parametrization and modification, *IEEE International Conference on Robotics and Automation*, Vol. 2 (1996) p.1539
- [21] R. Jerard, R. Drysdale, K. Hauck, B. Schaudt and J. Magewick: Methods for detecting errors in numerically controlled machining of sculptured surfaces. *IEEE Computer Graphics and Applications* Vol. 9 (1989) p.26
- [22] J. D'Errico, (2005). SLM Shape Language Modeling, MATLAB Central File Exchange. Retrieved November 04, 2010.