Obtaining a spiral path for machining STL surfaces using non-deterministic techniques and spherical tool


Department of Design and Manufacturing Engineering, University of Zaragoza - Campus Río Ebro, C/ María de Luna, 3 - 50018 - Zaragoza, Spain

HIGHLIGHTS

• Machining STL surfaces, without a volume inside, is useful and it can be optimized.
• To optimize this process we use the piece contour, defining 2D spiral paths.
• With non-deterministic techniques, the third coordinate of the tool-path is obtained.
• This method was simulated and applied to a workpiece, with positive results.
• Its precision was tested using confocal microscopy and coordinate-measuring machine.

GRAPHICAL ABSTRACT

ABSTRACT

For milling an arbitrary surface, several models exist in which the tool must be moved close and away from the workpiece with its consequential problems caused by the contact between the tool and the mechanized material. One of these problems is the undesirable mechanizing marks on the surface, which can be avoided with the use of tangential tool-paths, causing another disadvantage: the additional time-consumption. Spiral tool-paths eliminate these problems and allow for continuous machining of the part without approach or withdrawal.

Deterministic models for obtaining the tool positions referred to the part need to apply complex mathematical operations. For this reason, the idea of using non-deterministic techniques, with a controlled tolerance margin, allows to avoid those mathematical and geometrical operations. It also makes possible to reduce the processing times with no loss of dimensional quality and a good superficial finishing, which was confirmed with a coordinate measuring machine and with confocal microscopy, respectively.

Finally, the effectiveness of the described method was assessed comparing its results with the performance of three of the most relevant CAM software commercially available. The obtained results are shown at the end of this paper.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

Preparation of tool-paths for sculptured surfaces is still being an interesting research aspect, since obtaining the interval between consecutive milling passes attending to the surface curvature [1].
which can be defined in several ways, such as segmenting using polygons [2], being triangles the most commonly used. This is how differential geometry is replaced by analytic geometry of line and plane.

Deciding the optimal path for milling is still a problem in which research keeps being useful [3]. Previous researchers have dealt with the more convenient cutting direction for sculptured parts [4], paying attention to the surface roughness and required milling time. Nevertheless, among all the possible milling strategies (zigzag, z-level, parallel plan, iso-planar, etc.), spiral paths seem to be the most commonly used ones [5]. The reason for this is, basically, that this kind of paths leaves no machining marks on the surface.

The definition of milling tool-paths for finishing free form surfaces can be also decided according to several curve models, e.g. B-splines. This curve model can be applied in fields like the shoe manufacturing, for example using B-splines to compensate radiuses in the machining of shoe lasts [6], or to make the shoe sole attending to 3D CAD profile curves, scaling for different sizes [7].

Therefore, linear interpolation trajectories (G01) should not be forgotten, as they keep providing good results, combined with smoothing features included in numeric control devices [8] and a correct subdivision of the curves used to define the tool-path. Although it is possible to generate a 2D spiral trajectory with biarcs and splines [9], in this case G01 has been applied because it fits properly to 3D tool-paths and smoothing functions in numeric control devices, thus obtaining continuous trajectories.

Productivity can also be enhanced when continuous arcs are applied to define the tool-path as described in the previous research [10], being necessary to define parallel sections to interpolate those arcs, while in 3D it is necessary to indicate the interpolation plane, making this method slower.

For studying the error in the milling results with a spherical tool tip [11], as a result of roughness and dimensional tolerance, the surface representation must be taken into account. This representation can be considered using NURBS [12] or STL [13]. In this research article, STL surfaces were applied.

After deciding the surface representation, the milling strategy is chosen according to quality and time parameters. For example, Lu et al. [14] obtained spiral tool-paths from conventional trajectories, although they applied a process of incremental sheet forming (ISF) instead of milling. Other researchers [15] have optimized tool-paths, using response surface method for single point ISF. Similarly, spiral tool-paths for machining sculptured surfaces have also been obtained by conformal map approach [16].

Considering a maximum $\Delta z$ is equivalent to use the z coordinate as a fuzzy parameter [17]. In this paper, $\Delta z$ is treated as a variable to reduce processing times.

To calculate any coordinate $(x, y$ or $z)$ of the tool-path position in the space, geometric-analytical methods are usually applied, using fuzzy techniques only for the analysis of cutting parameters related to the part finishing, such as roughness [18]. Furthermore, Lo [19] studied the effect of feed rate, cutting speed and depth as roughness causes. Non-deterministic techniques have also been successfully applied to change cutting conditions [20] during the machining process or, on the contrary, to adapt this process to its specific conditions, predicting its final features [21]. In a similar way, these techniques have been combined with the Taguchi method and genetic algorithms [22].

The presented method makes possible to develop spiral trajectories, where coordinates $x$ and $y$ are obtained from the part boundary. This part's surface is given in STL format, making possible to export and import from virtually any commercial 3D CAD. To obtain the third coordinate, $z$, an approximation based on non-deterministic techniques is used with a margin of tolerance.

2. Method

With STL surfaces, which do not enclose a volume and whose whole exterior faces have director vectors with a positive z component, Fig. 1, their load is carried out from a binary format file [13]. In order to optimize the process time and the amount of information in memory, a method based on vectors is used [23] to save the data in memory.

In order to optimize the machining phase, a series of operations have to be carried out:

1. obtaining the contour;
2. obtaining the 2D tool-path and its spiral transformation;
3. application of algorithms to smooth the profile;
4. projection of 2D tool-path points on the surface;
5. calculation of the final $x, y, z$ path with non-deterministic techniques.

2.1. Obtaining the surface contour

This process is of extreme importance, as it contains the basis of later development such as the calculation of the tool-paths and the final cut, if necessary, to free the machined part from the initial billet.

Mainly, although other variants exist, this is based on the fact that one side of the edge belongs only to one triangle, while the interior sides belong to two or more triangles, Fig. 2. To obtain this, an algorithm has been implemented to simplify the final polyline.

Finally, as a starting point of the obtained polyline, we choose the most significant point of the ends of the segments that form the polyline, which can be:

- the point of the coordinate $x$ furthest to the left;
- the closest point to a determined direction, given by a unitary vector $(\cos \alpha, \cos \beta, \cos \gamma)$;
- any criteria that allow obtaining a single point.

In our particular case we have chosen the point with the component $x$ or abscissa of least value (the furthest to the left when facing a milling machine).

The final part of this phase is the management, from the chosen initial point, of all the points according to the chosen tool-path direction.

2.2. Obtaining the spiral tool-path

The algorithm for carrying out the contour runs for milling, starts from the exterior profile or contour of the surface which is considered to be the first profile. The object is to calculate the following profiles for the material to be removed from the exterior
Initially, the machining runs are “parallel” profiles which are obtained as shown in Fig. 3, displacing each point a certain amount towards the interior of the profile in the direction of axis $OX$ and another in the direction of axis $OY$, the same as if a scale was carried out but variable in each direction. The entry datum is the distance $d$ between homologous points but measured in a certain direction which depends on the position of the point.

The position of the point is classified by quadrants originated by a system of coordinates. This system of coordinates can be chosen in function of various parameters. The origin has been taken as the gravity centre of the series of initial points.

$$x_{cg} = \frac{\sum x_i l_i}{\sum l_i} \quad y_{cg} = \frac{\sum y_i l_i}{\sum l_i}.$$  \hfill (1)

This system of coordinates classifies the plane in four quadrants (I, II, III and IV). To classify each point, its difference is simply carried out with respect to the gravity centre. At each point a displacement is applied (the coordinates are actually multiplied by a scale). The distances of this point to each coordinated axis are taken into account. For example:

- if the point is in the first quadrant, the scales in the direction $OX$ and $OY$, in function of some certain runs, $p_x$ and $p_y$ in each direction, which will be looked at later, are:

$$e_x = \left(1 - \frac{p_x}{x_{max} - x_{cg}} \right), \quad e_y = \left(1 - \frac{p_y}{y_{max} - y_{cg}} \right).$$  \hfill (2)

With this, it is possible to calculate the coordinates of the transformed point in the chosen system of universal coordinates.

$$x_p = \left(x_i - x_{cg} \right) \cdot e_x + x_{cg}, \quad y_p = \left(y_i - y_{cg} \right) \cdot e_y + y_{cg}.$$  \hfill (3)
On this occasion, the runs in x and y, $p_x$ and $p_y$, take the same value, given by the expression obtained in Fig. 4 in function of the radius of the spherical tool and the roughness crest $f$ if, supposedly, milling was in the horizontal plane. Other cases (concave and convex surfaces) can be consulted [24].

$$r^2 = (r - f)^2 + \left(\frac{P}{2}\right)^2 \Rightarrow p = 2\sqrt{r^2 - (r - f)^2}. \quad (4)$$

The process is repeated for each new profile, but, taking the maximum coordinates that correspond to the previous profile as maximum values to enter into (2).

Finally, worth noting is that the points which change quadrant with respect to the first of the line of points of the initial contour, cannot change due to the previous transformation. Therefore, a filter is used if the point changes to another quadrant; this filter reduces to zero the coordinate that causes this change.

In order to transform the previous points into a spiral, the process consists of working with the equivalent points of a profile of those obtained previously and the following profile. This is done considering that initially all the profiles have the same number of points. A formula is used for this purpose which brings the points of a profile closer to those of the following profile: the first point is the next to profile $p - 1$, and the last point next to profile $p$.

$$x_i = x_{i,p-1} + \left(x_{i,p} - x_{i,p-1}\right) \cdot \frac{i}{n}$$
$$y_i = y_{i,p-1} + \left(y_{i,p} - y_{i,p-1}\right) \cdot \frac{i}{n} \quad (5)$$

where, $i$ is the current point of profile, $p$ is the current profile, and $n$ is the number of profile points.

Fig. 5 shows the results obtained. In this figure, a high value has been taken in expression (4) for the height of the roughness crest, $f$, to obtain few profiles and so be able to illustrate the process. The contour forms part of the tool path to calculate the first or last cut, according to whether start-up is made from the outside inwards or vice versa.

### 2.3. Algorithms to smooth the profile

The use of processes of smoothing is not new and has been used in different fields [25]. These algorithms have a dual purpose. On the one hand, should the profile be too abrupt, it is smoothed over. The criteria can be very varied: to carry out the average of next points or to optimize trajectories to a minimum number of points [26], among other possibilities.

Once smoothing has been carried out according to the precision of the STL lattice, the addition of more points may be required. This is done depending on a step or interpolation precision (distance between two consecutive points), in the direction of the spiral orbit from one point to the next. Therefore, one or more intermediate points can be added between two points, Fig. 7. Some authors described the application of adaptive tool-paths [27] with excellent results in the surface finish; but in our case the tool-path follows a more uniform direction, so less acceleration milling is required to obtain precise results.

Again, the algorithms for smoothing can be applied to this new spiral path with added points.

### 2.4. Projection of the spiral path points onto the surface

The previously obtained spiral path 2D points are transformed into the 3D path by projecting the 2D points onto the STL surface, but also taking tool size into consideration.

Tool size is extremely important. Fewer runs can be made using large diameter tools, see expression (4). But the inconvenience is that the smallest curvature radius possible on the surface is that of the tool tip radius, as unmachined hollows can remain, Fig. 8.

For this choice between the tool radius and that of the surface curvature, the coordinate $z$ of the tool contact point with the surface must be determined. This point is not necessarily the projection of the coordinates $(x, y)$ on the surface. This is due to the fact that the surface curvature can cause the tool to make contact with another upper $z$ coordinate, Fig. 8 right.
The projection on the STL surface can be made once the triangle containing the coordinates \((x, y)\) is known. In order to locate the triangle where the tool is situated, classification is carried out by zones, Fig. 9, to speed up the result of the algorithm as, once the point \((x, y)\) is assigned to a zone, it is only necessary to work with the triangles of that zone and find the triangle containing the projection of the point \((x, y)\).

According to the directions of the axes \(x\) and \(y\), the number of grids is:

\[
\begin{align*}
n_x &= \text{roundness}_{\text{upper}} \left[ \frac{\text{maximum}_x - \text{minimum}_x}{\text{radius}_{\text{tool}}} \right] \\
n_y &= \text{roundness}_{\text{upper}} \left[ \frac{\text{maximum}_y - \text{minimum}_y}{\text{radius}_{\text{tool}}} \right].
\end{align*}
\]  

To locate the zone where the tool point \((x, y)\) is situated, we apply:

\[
\begin{align*}
\text{zone}_x &= \text{whole part} \left( \frac{x - \text{minimum}_x}{\text{radius}_{\text{tool}}} \right) \\
\text{zone}_y &= \text{whole part} \left( \frac{y - \text{minimum}_y}{\text{radius}_{\text{tool}}} \right).
\end{align*}
\]

Within the zone, a search is made to find the triangle that contains the tool tip:

A triangle consists of three points (vertices) which, when grouped into two, form three straight lines (sides). Each side of the triangle is viewed to see whether the point we want to check and the opposite vertex are on the same half-plane, this means sharing the same sign when we substitute the coordinates of the points in the normal line equation.

\[
\begin{align*}
\text{half}_\text{plane} &= \text{sign} \left( \frac{ax + by + c}{\sqrt{a^2 + b^2}} \right) \\
&= \text{sign} (ax + by + c).
\end{align*}
\]

Fig. 10 shows that both points \(P\) and \(Q\) comply with this condition for the first side \((P1P2)\) but, on checking the following side \((P1P3)\), \(P\) is outside the triangle.

To obtain the projection of a point onto a particular triangle, best results are obtained using the normal plane equation, as the plane director vector containing the triangle is known.

\[
x \cos \alpha + y \cos \beta + z \cos \gamma = q
\]

where, Fig. 11, \(\alpha, \beta\) and \(\gamma\) are the angles of the director vector of the plane of each coordinate axis and \(q\), the distance of the origin of the coordinates of the plane. The director cosines are data of the STL file.

The value of parameter \(q\) is obtained by substituting the coordinates of a vertex, for example, the first (according to the matrix used).

\[
q = x_0 \cos \alpha + y_0 \cos \beta + z_0 \cos \gamma.
\]

The coordinate \(z\) of the projection point can now be obtained, Fig. 12:

\[
z_p = \frac{x_0 \cos \alpha + y_0 \cos \beta + z_0 \cos \gamma - x_S \cos \alpha - y_S \cos \beta}{\cos \gamma}.
\]

It can be seen that, when the plane is vertical, \(\cos \gamma = 0\), there is no projection.

2.5. Calculation of the final tool-path \(x, y, z\) using non-deterministic techniques

To calculate the 3D tool-path, the 2D path of Fig. 5 is projected onto the surface, Fig. 13, but, taking into account the three conditioners, Fig. 14:

(a) close to the contour of the surface, position \(z\) lower than the tool, is the point where it is tangential to the surface (point of intersection with the surface);
(b) the initial position \(z\) of each pair of \(x, y\) coordinates is the upper part of the billet block;
(c) the final position \(z\) of each pair of \(x, y\) coordinates is that which holds the tool in tangential contact with the surface.
The previous projection is not carried out using any mathematical calculation. Only the condition “the tool touches the surface” is used, and which is analysed following the application of a tool descent function from position (b) in Fig. 14.

The descent can be carried out in several ways [14], but in this case a lineal function with a maximum of ten steps has been chosen. Once contact is made between the tool and the surface, the tool returns to the previous step and the process is repeated with a reduction in the step size.

This process ceases (Fig. 15) when the step size is less than a predetermined value of tolerance and the final value of the vertical tool position is the third coordinate \( z \).

Fig. 15. Successive approaches. (a) Tool descends making contact with the surface and returns to the previous step. (b) The tool descends again but with smaller step sizes. This process continues until the increase is less than the required value. The tool always remains in the previous position.

This method always starts from the same coordinate \( z \), so it is not necessary to use the Eq. (10). However it is also possible to start from the position given by Eq. (10) and raise the position following the same routine until the intersection with the triangles disappears, descending again with a smaller increment.

The variation of the membership function for the descending stage (b) of Fig. 16 can be observed in the following figure (Fig. 17). The tolerance of this function decreases when there is a collision with any triangle in the influence area of the tool tip. The number of tolerance variations is not very high: three changes are typically enough, e.g. 1, 0.1, and 0.01 mm. If the acceptable tolerance increases these values could change slightly, e.g. 1, 0.1, and 0.02 mm, i.e. 0.01 mm more than in the previous case. In the first case the maximum theoretical distance between the nominal and the milling surfaces is 0.01 mm vs. the 0.02 mm of the second case, in which the required time decreases (obviously, mechanical tolerances should also be considered).
Fig. 16. Flowchart of the tool descending stage (b), where “t” is the tolerance, which can be up (b), centred (c) or down (d).

Fig. 17. Evolution of several membership functions with the tolerance.

Fig. 18. Relative positions of the tool tip sphere and the triangles that compose the STL surface.

To correct the tool position given in (10) several cases of tool interference with the surface [11] have been studied. Fig. 18 shows the case that can happen. As mentioned before, this can be solved either in a deterministic way [11] or by correcting the displacement of the tool by means of successive approaches within a certain tolerance.

Study case A, Fig. 19: if any one of the three triangle vertices is contained within the sphere, an intersection of the tool tip with the surface occurs. That is, in this situation, the distance from the vertex to the sphere centre is less than the tool tip radius. This distance is checked for each of the j vertices (j = 0, 1, 2) of each triangle -i- under study in the tool tip influence zone (see Fig. 9). If the distance then
\[ d = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2} < r \]
the tool tip position must be corrected.

Study case B, Fig. 20: if the projection of the sphere centre on the plane (less perpendicular distance) is within of the triangle, an intersection occurs. Should this point fall outside the triangle there may be an intersection, as in cases A and C. In fact, the only real case could be B under certain conditions.

Study case C, Fig. 21: if any of the sides of the triangle cross the sphere and its points of intersection, M and N, are contained on this side. It should then be checked to see if these two points are between the vertices of the side. If only one point is inside, we are faced with case A.

3. Results: comparison with three commercial cam packages

So et al. [28] compared their method with commercial CAD–CAM software in order to check the results of the proposed algorithm. In a similar way, three experts on different commercial CAM packages were contacted to apply the spiral tool-path using their software, so it was possible to compare their results with those obtained in this study, under the same machining conditions (roughness, feed rate and speed).

To avoid influencing the results, we did not participate in the process of obtaining the NC program with the commercial CAM
Table 1
Empty cells are filled to apply the addition (11) correctly.

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>-57.870</td>
<td>-46.796</td>
<td>5.074</td>
<td>-57.870</td>
<td>-46.796</td>
<td>5.074</td>
</tr>
<tr>
<td>-57.995</td>
<td>-46.796</td>
<td>5.304</td>
<td>-57.995</td>
<td>-46.796</td>
<td>5.304</td>
</tr>
<tr>
<td>-59.815</td>
<td>-47.546</td>
<td>5.598</td>
<td>-59.815</td>
<td>-47.546</td>
<td>5.598</td>
</tr>
</tbody>
</table>

Fig. 22. Machining with “WinUnisoft” simulator for the two crest heights. (Left: roughing with 10 mm tool tip. Right: finishing with 3 mm tool tip and 0.02 mm cress height.)

Table 2
Comparison of tool-path lengths and numbers of points.

<table>
<thead>
<tr>
<th>Tool-path length (mm)</th>
<th>Number of points</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAM A</td>
<td>45 675</td>
</tr>
<tr>
<td>CAM B</td>
<td>37 797</td>
</tr>
<tr>
<td>CAM C</td>
<td>68 728</td>
</tr>
<tr>
<td>New algorithm</td>
<td>31 524</td>
</tr>
</tbody>
</table>

Fig. 23. Roughing process (tool tip diameter = 10 mm) and machined piece (cycle saddle) before finishing (tool diameter = 3 mm).

packages. Thus, this task was performed by a technical expert on each CAM package, so their results could be considered optimal.

The conditions and materials given to each expert were exactly the same, as listed below:

(a) STL file with the geometry to be machined. This file was exactly the same that was used to test our method, as shown in this paper;
(b) maximum roughness: 20 μm;
(c) the followed tool-path is the same from the inside to the outside and vice versa;
(d) each expert had to obtain the NC program of just one complete path with a spherical tool tip with diameter = 3 mm;
(e) the machining conditions (advance and cut-speed) were also defined by us in order to be the same as we used, although they are not really relevant in this study;
(f) lineal interpolation was the only one to be used, so the list of coordinates \((x, y, z)\) was preceded by G01.

The compared parameters are the tool-path length and the exact number of interpolation points. The followed method is explained below. Other studies [29] use roughness as a parameter to compare results, but we considered that the tool-path length clearly indicated the required machining time and the number of interpolation points is related to the processing time of the NC [26].

The tool-path length has been obtained using the following Eq. (11), taking all the coordinates from an Excel spreadsheet.

\[
l = \sum_{i=2}^{n} \left( (x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2 \right)^{1/2}.
\] (11)

The interpolation points in which a coordinate has been omitted because it was equal to the previous one have been completed, so the previous summation (11) could be correctly applied, as shown in Table 1.

Table 2 has been obtained applying the previous addition. This table compares the tool-path lengths and the number of points obtained using each commercial CAM package and, in the last arrow, using our method.

As the distance between points is very small, the applied addition (11) gives a good approximation to the tool-path length and the machining time can be obtained dividing the total length by the advance speed. For that reason, the shorter tool-path will be described in less time.

The number of points influences the time needed by the NC machine to read all the information. If this number is too high in a tool-path, the buffer of the look-ahead function will be full and the processing speed will be lower [26]. This is why tool-paths with lower number of points are faster.

As said before, the followed method by each commercial CAM package and the routine presented in this paper has not been compared, because commercial CAM packages are oriented to general purposes. This means that the steps that must be followed, to achieve the same result, will be very similar to this algorithm steps; but in this case, those steps have been automated for the type of surfaces used in this paper. This is why, it is only needed to load the STL file. In the case of commercial CAM software, the developer can provide the customized macro, if required, to perform the same operation in just one step.
4. Conclusions

With the algorithms presented it is possible to mill STL surfaces, Figs. 22 and 23, providing that they comply with certain conditions, thereby obtaining regular profiles and with a quality of finish very close to that required.

The dimensional results (Fig. 24) are within the programmed margin of error for this piece (0.05 mm), providing enough quality for manufacturing different custom-made objects, such as the cycle saddle shown in this example or different types of ergonomic components, among other applications.

The measurement path was obtained from the points used in the milling process of the workpiece, but increasing their separation, i.e., decreasing the number of points (Fig. 25). The points were collected in a spreadsheet and, after that, exported to CAD to compare them with the nominal surface. In order to evaluate the results, the surface was cut into longitudinal and transversal sections.

The spherical cavities were used to position the horizontal plane of reference and to orientate the coordinate axes, as shown in Fig. 25. A longitudinal nominal profile is represented in Fig. 26, which also includes its measured profile, obtained by a linear interpolation of the measurement spiral points. The deviation in the measured points has always been smaller than the chosen tolerance value of 0.05 mm.

As shown in Table 2, the presented algorithm generates tool-paths with a length that makes possible to improve the machining time compared with commercial CAM packages.

Finally, the surface roughness was measured using confocal microscopy, obtaining the results shown in Fig. 27. These results displayed a maximum roughness below the programmed value, 20 μm in this case, in both the wide and the narrow areas of the piece.

Acknowledgements

The present work corresponds to a part of the innovation project titled “Knowledge as a development engine in mechanical manufacturing”, financed by the Education Ministry of Spain and the European Social Fund.

Surface measurements (Fig. 27) have been performed at the Surfaces and Coatings Characterization Service at CEQMA (CSIC-Universidad de Zaragoza).

Our thanks go to fellows of the Integrated College of Technical Training “Corona de Aragón” from Zaragoza (Spain), Francisco Valdivia Calvo and Juan José Garde Barace, for their assistance in the preparation of parts and field work.

We are grateful to the English teacher Richard J. Stephenson, for the English correction.

This paper could not be the same without the help of three anonymous experts of leader CAM companies. Thank you very much for making possible the comparison for the third part of this paper.
Fig. 27. Images of the piece surface obtained with confocal microscopy. (Left: narrow area. Right: wide area.)

References