Interpolation of Toolpath by a Postprocessor for Increased Accuracy in Multi-Axis Machining

Petr Vavruska*

Czech Technical University in Prague, Faculty of Mechanical Engineering, Department of Production Machines and Equipment, Horská 3, Prague 128 00, Czech Republic

* Corresponding author. Tel.: +420-221-990-928; fax: +420-224-359-348. E-mail address: p.vavruska@rcmt.cvut.cz

Abstract

The article focuses on the issue of generating points of the toolpath for multi-axis machining. In multi-axis machining, it is possible to control the toolpath using the coordinate transformations of tool center point (TCP) in the control system, or these transformations in the control system are not available and it is necessary to use the transformations in the postprocessor. However, without the use of TCP, the required toolpath tolerance is not respected. Therefore, an algorithm has been proposed in the postprocessor that dynamically generates new toolpath points so as to maintain the required tolerance and to ensure manufacturing accuracy. This algorithm has been verified by implementation in the postprocessor and generation of NC programs for machining of impeller blades. By the postprocessor recalculated tool path meets the required tolerance.

Keywords: Machining; Tool path; Accuracy; Multi-axis; Postprocessor

1. Introduction

Multi-axis machining of complex shape parts (e.g. an impeller in Fig. 1) is one of the most complex technologies. Many errors entering the machining process are involved in the resultant error of machined parts. These errors can be caused by specific accuracy of the machine tool, tools, the effects of stiffness of the machine tool – tool - workpiece system as well as thermal behavior. However, there are also errors resulting from data processing, whether on the level of the control system or CAD / CAM system. The issue of proper preparation of data for machining complex parts has been dealt with by many authors.

One method for compensating geometrical errors in the five-axis machining process is described for example in lit. [1]. The method first deals with compensation of geometric errors caused by rotational axes of the machine tool and then with compensation of errors caused by linear axes. The issue of minimizing deviations arising after machining is dealt by authors of lit. [7]. In this paper, the optimization of tool path in multi-axis machining is discussed with respect to two different approaches. One optimization algorithm is designed with respect to the achievement of minimum strain energy in the process of machining and the criterion is surface smoothness. The second optimization algorithm is based on minimizing deviations arising after machining of the surface. Extensive research on this issue was conducted by the authors of lit. [2] and [5].

Fig. 1. Typical complex shape part: impeller
These papers dealt with the proposal of a computational model for the prediction of cutting forces during machining. This issue is handled by authors of [10] or [3] too. Their papers also contain proposals of computational models for the prediction of cutting forces during machining. However, the model in lit. [10] is based on the knowledge of CL-data which are received from the CAM system. Using the proposed software, cutting forces can be predicted.

Authors of lit. [4] reported, that the use of spline interpolation has the effect of increasing the surface quality and also saving machining time. However, this function can be used only when using CAM CATIA and control system Sinumerik.

The toolpath for multi-axis continuous machining can be prepared in two ways. The first can be based on using transformations called Tool Center Point (TCP), which are included in the control system (such as TRAORI, or TCPM, etc.). In the second case the toolpath can be computed with the transformations in the postprocessor. The transformations TCP in the control system cannot be used in some cases. The main reason why it is not possible to use TCP is that the kinematical configuration of machine tool axes is not supported in the control system. The second reason is that the technologist does not want to use TCP because the toolpath can be very difficult to modify by the machine tool user.

According to the authors of lit. [6], geometric errors are also caused by the movements of the rotary axes near the so-called stationary points. These are the points at which the tool reaches parallelism with the rotational axis of the machine tool. At these points the machine tool performs additional movements of rotary axes so that the next point in the NC program can be achieved by linear interpolation. However, it will cause errors on the part surface. The error is dependent on the actual radius of rotation of the reference point of the tool towards the axis of rotation. A method is presented in the above-mentioned paper to minimize the occurrence of these errors. The sections of the tool path are replaced by a different tool path that avoids the stationary points. The algorithm is suitable only for point milling, not for flank milling, where it is necessary to accept the toolpath as calculated by the CAM system. Another solution is offered by the author of lit. [8], which deals with the same issue by additional interpolation of tool path points. Thanks to this, the orientation of the tool axis to the workpiece surface is maintained along the tool path so as to avoid undercutting of the workpiece surfaces. However, the algorithm is based only on the assumption of a pre-established maximum possible change in angular coordinates in two consecutive blocks of the NC program. This change is then constant for all cases of computation of the relative positions of the reference point of the tool and the current axis of rotation.

The number of newly interpolated points of the tool path is not controlled by the required tolerance of toolpath. If we assume that the change in angular coordinates will be identical in the next two blocks, then the same number of points is interpolated at greater distances between the reference point of the tool and the current axis of rotation as at shorter distances. This may have an impact on the characteristic of feed rate, because when too many points of toolpath are interpolated, it leads to very small increments in machine tool axes.

Consequently, it is possible that the control system is not able to achieve the required feed rate. It is necessary to emphasize that these errors may arise also in other areas of the toolpath. The main deficiency is that the algorithm is not controlled by the required tolerance of toolpath, specified by the user. Therefore, in the following part of this paper, a method for interpolation of the tool path in the postprocessor is presented, which is controlled by the given toolpath tolerance.

2. Interpolation of tool path points

The interpolation of the tool path is based on the fact that the real tool path as well as the required tool path can be calculated by the postprocessor. The Fig. 2 shows the points of the real path of the reference point of the tool, but also the points of the required tool path, with new points (red points) that are necessary to meet the required tolerance of the tool path. For the interpolation of these points transformation equations are used. These must be prepared for each of the kinematical configuration of the machine tool axes. The following equations are examples for the machine tool with rotary axes B and C on the machine table. For this machine tool the CL data (coordinates CL_X, CL_Y, CL_Z) from the CAM system have to be transformed to coordinates of the NC program (coordinates X, Y, Z) using the following matrix notation (1). The matrix T_BCS-MCS is the resulting transformation matrix and can be computed as product of the matrices T_BCS and T_EBC, which are the matrices for angular transformation, see (2) and (3).

\[
\begin{align*}
T_{EBC} &= T_{EBC-MCS} \cdot r_{EBC} = T_{EBC} \cdot r_{EBC} \cdot r_{EBC} \\
T_{EBC} &= \begin{bmatrix}
\cos C & -\sin C & 0 & 0 \\
\sin C & \cos C & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \\
r_{EBC} &= \begin{bmatrix}
X \\
Y \\
Z \\
1
\end{bmatrix} \\
T_{EBC} &= \begin{bmatrix}
\cos B & 0 & \sin B & 0 \\
0 & 1 & 0 & 0 \\
-\sin B & 0 & \cos B & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \\
r_{EBC} &= \begin{bmatrix}
CL_X \\
CL_Y \\
CL_Z \\
1
\end{bmatrix}
\end{align*}
\]
The maximum deviation of the real tool path from the required one in the given section.

The above equations are utilized for iterative calculation of partial coordinates in required steps of algorithm for interpolation of the tool path. The algorithm is designed so that the user can specify the desired tolerance \( \delta_T \) as the maximum deviation of the real tool path from the required one. Initially, the default number of segments \( d \) is chosen in the maximum deviation of the real tool path from the required one.

\[
\begin{align*}
X &= CL_X \cdot \cos C \cdot \cos B - CL_Y \cdot \sin C \cdot \cos B + CL_Z \cdot \sin B \quad (6) \\
Y &= CL_X \cdot \sin C + CL_Y \cdot \cos C \\
Z &= -CL_X \cdot \cos C \cdot B + CL_Y \cdot \sin C \cdot B + CL_Z \cdot \cos B \quad (8)
\end{align*}
\]

The above equations are utilized for iterative calculation of partial coordinates in required steps of algorithm for interpolation of the tool path. The algorithm is designed so that the user can specify the desired tolerance \( \delta_T \) as the maximum deviation of the real tool path from the required one. Initially, the default number of segments \( d \) is chosen in the maximum deviation of the real tool path from the required one.

Each segment \( i \) is afterwards divided into a number of segments \( e \), where each segment is marked as \( j \), where \( j \leq 1 \); \( e > 0 \). This division is used for setting of calculation points (their number is \( e-1 \)) to calculate the deviation of the real tool path from the required one if the new points of the tool path are interpolated according to the number of segments \( d \). For each segment \( i \leq 1 \); \( d > 0 \) new points of the tool path at the end of the segment \( e \) (new) are calculated. The coordinates at the end of each segment \( (e \) new) and at the beginning of segments \( e \) (new) are calculated simultaneously \( (e \) new). Based on these coordinates the coordinates of points on the required tool path are calculated based on the relative distances of the coordinates are calculated based on the value of \( d \). If there have been newly interpolated points generated in the NC program (value of variable print is equal to one), the algorithm stops. This occurs when the value of the deviation \( \delta_2_{i,j} \) is not greater than the specified tolerance \( \delta_T \). A very shortened algorithm can be seen in Fig. 3. If the previous options are not fulfilled and variable in \( \delta_2_{i,j} \) is higher than the tolerance \( \delta_T \), then the number of segments \( d \) is increased by one and the operation of the algorithm is returned to the place where the relative coordinates are calculated based on the number of segments \( d \).

3. Multi-axis milling of an impeller blade

The function of the proposed algorithm for interpolation of the tool path has been tested by implementation to the postprocessor for the five-axis MAS MCV1000 machine tool with the Nikken rotary/tilting table (rotary axes \( B \) and \( C \)). The impeller in Fig. 1 has been used for testing. The blade is machined using a flank milling operation. It has been established that the tool path generated from the CAM system without the interpolation shows deviations in the order of hundredths to tenths of a millimeter, while the tolerance \( \pm 0.003 \text{ mm} \) has been set in the CAM system.
Using the interpolation algorithm with the tolerance $\delta_T = 0.01$ mm the tool path meets the requirement as can be seen in Fig. 4. The effect of four values of tolerance $\delta_T$ (0.03 mm and 0.01 mm, 0.005 mm, 0.001 mm) has been tested. The number of segments $e$ has been tested in the range of 5 to 15 segments. Tab. 1 contains the number of newly generated blocks of the NC program for machining the blade. It has been verified that the value of 10 segments $e$ is satisfactory.

The tool path for machining the blade is shown in Fig. 5. It can be clearly seen that the interpolated tool path (Fig. 5 right) consists of more points than the original tool path without interpolation (Fig. 5 left). Simulation of blade surface after machining is shown in Fig. 6. Some remaining material can be seen on the surface which has been machined using the original path without interpolation (Fig. 6 left). However, there is no remaining material on the surface which is machined using the newly interpolated tool path (Fig. 6 right) with the tolerance $\delta_T$ set on the value of 0.01 mm and the blade surface corresponds to the model of the blade.

### Tab. 1: Number of new interpolated point of tool path according to the specified tolerance $\delta_T$

<table>
<thead>
<tr>
<th>Tolerance $\delta_T$ [mm]</th>
<th>CAM tolerance: $\pm 0.003$ mm</th>
<th>NC program blocks before interpolation: 227</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Number of elements $e$ [-]</td>
<td>Number of new NC program blocks [-]</td>
</tr>
<tr>
<td>0.03</td>
<td>5</td>
<td>52</td>
</tr>
<tr>
<td>0.01</td>
<td>10</td>
<td>56</td>
</tr>
<tr>
<td>0.005</td>
<td>15</td>
<td>117</td>
</tr>
<tr>
<td>0.001</td>
<td>231</td>
<td>237</td>
</tr>
<tr>
<td></td>
<td></td>
<td>605</td>
</tr>
</tbody>
</table>

**4. Conclusion**

In this paper, an algorithm for interpolation of the tool path for multi-axis machining has been proposed. The interpolation of tool path points is needed to achieve the required tolerance of the tool path when the tool path is generated without usage of transformations in the control system (such as TRAORI or TCPM). The algorithm has been implemented in the postprocessor and has been verified by generation of a tool path for milling of an impeller blade. It has been found that the interpolated tool path meets the required tolerance.

### Acknowledgements

This paper has received funding from the Technology Agency of the Czech Republic.

### References


