

An Optimization Post-Processing Module For Complex Tool-Tip Milling Operations

Luc Baron

Department of mechanical engineering, École Polytechnique de Montréal

P.O. box 6079, succ. CV, Montréal, Qc, Canada H3C 3A7

Email: luc.baron@polymtl.ca

Luc Rolland

Department of mechanical engineering, Memorial University

St-John's, NL, Canada

Email: lrolland@mun.ca

Abstract: This paper presents an optimization post-processing module allowing to improve and to optimize tool-tip milling operations. Moreover, this module verifies and insures milling feasibility of a complex tri-dimensional shape. During such situations, the tool is in contact with the workpiece through a point located on its ball-end mill. Knowing that a relatively large variation of the tool orientation does not necessarily change the resulting workpiece geometry, then the post-processing module can be formulated as an optimization problem. The tool orientation is computed to be as close as possible to our preferred orientation together with its joints as far as possible to the joint limits, and obviously, subject that the tool-tip is following the required trajectory. The module has been implemented for the Huron KX8-Five milling centre. Several surface shapes have been post-processed and verified in simulations with VERICUT, among which four parts have effectively been machined. Our optimal module produced a CAM program which could then be transferred into a CNC milling machine and this program succeeded in machining a complex workpiece which would normally prove unfeasible with a classical approach which usually maintains the tool normal to the surface without exceeding the joint limits.

Keywords: *Post-processor, five-axes milling, joint-limits avoidance, inverse kinematics, redundancy resolution.*

1. INTRODUCTION

The interface module that connects the CAM systems and NC machines is called the postprocessor and it converts cutter-location data (CL data) to machine code (G codes).

Sculptured surfaces (or free-form surfaces) have found extensive applications in many manufacturing sectors. Five-axis machine-tools can accurately machine such complicated surfaces if an appropriate computer-aided manufacturing (CAM) system is used to produce the numerical control (NC) program to be transferred to the machine-tool controller (CNC). In general, CAM systems proceed in three phases. During the first phase, generate cutter contact points (CC points) are determined, the second phase is dedicated to the generation of CL-data and during the third phase, the CL-data is converted into machine-tools executable G-codes. These G-code lines constitute then the so-called NC program. Moreover, during the last phase, a link is established between the remote CAM system and actual machine-tools and the program system is known as the post-processor [Lee 97]. The first two phases are concerned by the motion of the tool in the workpiece reference frame without considering the kinematics of the machine-tools onto which this motion is implemented. In the third phase, the CL-data transformation into machine-tool joint position requires further knowledge of the manipulator architecture in order to solve its inverse kinematics, i.e., transferring the tool positions and orientations (operational space) into machine joint positions (joint space).

The inverse kinematics for five-axis machine-tools can be found either as a closed-form solution (analytical solution) or as an iterative numerical solution, [Nenchev 89]. In general, analytical methods are faster than numerical ones provided closed-form solutions can be implemented. It is also easier to proceed with the solution choice process where one solution has to be selected among several possible ones or one continuous set determined by a function, but finding the optimal one becomes a complicated problem. Iterative numerical methods involve a large number of numerical operations, [Sorby 07], even though they are powerful in finding the optimal one in those problems which involve manipulator kinematics redundancy.

In traditional machine-tools where the three translational axes are consecutive and orthogonal, the kinematics of the machine-tool is fully decoupled, and hence, the orientation part of the problem can be solved independent of the position part, [Baron 95]. The latter is the most common implementation of inverse kinematics module in commercial post-processors. However, more general architectures of machine-tool exist and cannot be handled with such a simple model thereby requiring specific kinematics modeling, [Lee 97], [Baron 99].

The redundancy of machining operations may arise in many different situations, where the dimension of the task space (included in operational space) is smaller than the dimension of the

joint space. It means that the degree of freedoms (DOF)s required by the tool to perform the task is less than the number of machines axes. For example, a tool-tip surfacing task with a ball-end mill can be considered as a redundant operation because this task requires only 3-DOFs for positioning, and hence, can just be realized with a simple 3-axis machine-tool. However, if the task is realized with a 5-axes machine-tool, there are 2-DOFs of redundancy, and hence, additional specifications are allowable and desirable to improve operations such as to avoid obstructions and joint limits [Baron 99]. Obstacle avoidance can also lead to automatic recalculation of the paths, [Kruth 98], [Ding 00]. The subject of redundancy-resolution has been studied at both displacement level, [Angeles 87], and at velocity level, [Siciliano 92]. It is noteworthy that most of the inverse kinematics solution procedures implemented in post-processor modules are based on a displacement formulation. Here, we adopt the velocity formulation because the problem of redundancy-resolution is linear at that level. Moreover, we introduce a dual strategy in order to set our preferred tool orientation together with a joint-limits avoidance strategy.

The issue of post-processing has been investigated with various concerns in mind, [Baron 95]. During NC-postprocessing, some simulation packages check tool position for collision, [Kruth 98]. Others study the improved conversion of machine-tool task commands to machine joint commands, [Liangji 09]. In one paper, a general procedure is proposed for developing a post processor in which identified error sources could be corrected by modifying the kinematic model.

The first section will then be dedicated to kinematics analysis leading to the velocity analysis necessary for the forthcoming activities. Then, the second section investigates problem formulation of the optimization problem which will deal with redundancy for performance improvement. The third section continues with algorithm descriptions involved in the design of the post-processing module. Then, this paper terminates with experimental validation on a practical case performed on a real milling machine.

2. KINEMATIC MODEL

The kinematics of machine-tools can be modeled by various mathematical tools such as geometric reasoning or homogenous transformation matrices. With the later, we implement the Denavit-Hartenberg (DH) convention, [Angeles 02], to describe the geometry of the milling

machine based on a serial topology, since it allows a straightforward methodology to obtain the motion of the tool with respect to the workpiece, at both displacement and velocity levels.

In this section, we briefly recall this method in connection with the milling center Huron KX8-Five as shown in Fig. 1. Firstly, the Huron KX8-Five milling center is essentially a 5 DOF serial manipulator and it is actually divided into two separate serial mechanisms, the 2 DOF Cartesian manipulator with two prismatic axes, identified X and Z, and the 3 DOF table with one prismatic axis, identified Y, and two concentric rotary axes, identified A and C, comparable to wrist structure.

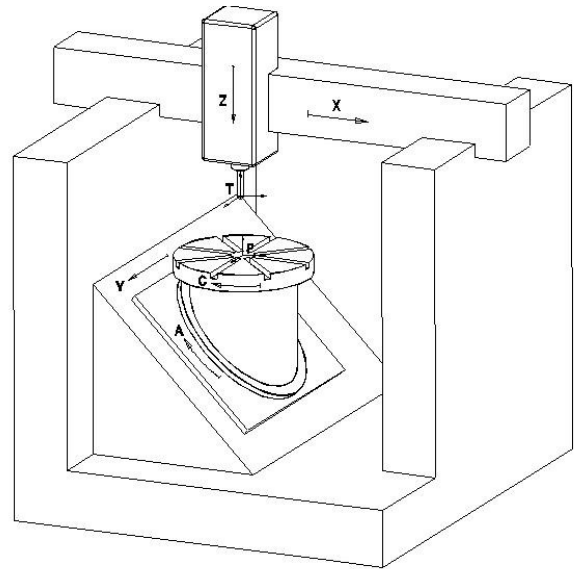


Fig. 1. Architecture of the Huron KX8-Five milling center

For the sake of kinematics modeling, the P coordinate frame is fixed on the work table surface with its origin located on the surface center and its z-axis is pointing perpendicularly upward or more precisely away from the table surface whatever the position of the C axis. Similarly, let us locate the T coordinate frame on the tool tip, with its origin at the center of the half-ball end-mill and its z-axis along the tool rotation axis, pointing towards or entering the spindle. The typical kinematic chain of the Huron KX8-five is shown on Fig. 2. To simplify calculations, the workshop frame is made to correspond to the machine-tool base one. Thus, it is possible to express coordinates from the P Cartesian frame to the T reference frame through the five controlled (actuated and measured) joints, i.e., C, A, Y, X and Z, with joint positive directions being where C and A are the workpiece rotations, Y is the workpiece translation, X and Z are the tool Cartesian axes as shown in Fig. 1. Note that the fixed base is located between the X and Y

prismatic axes and this is usually where the O reference is located and this frame is called the workshop frame.

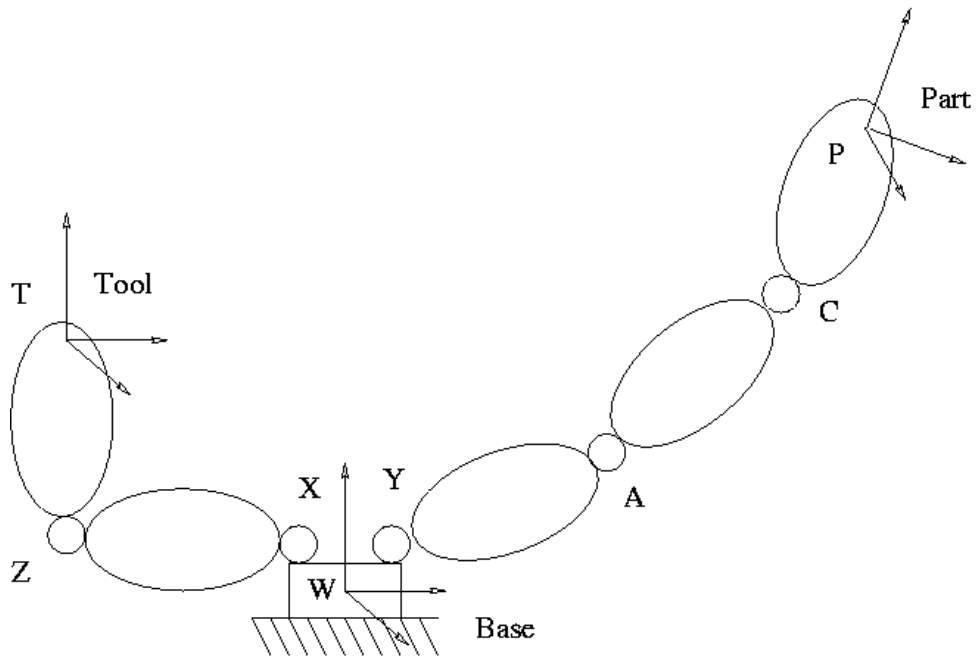


Fig. 2. The kinematic chain of the Huron KX8-Five

According to the DH convention, each set i of DH parameters is characterized by one rotation θ_i around the z -axis, a translation a_i along the x -axis, a translation b_i along the z -axis, and finally, a rotation α_i around the x -axis, [Angeles 02]. The five sets of DH parameters corresponding to the five joints are shown in Fig. 3 and listed in Table 1.

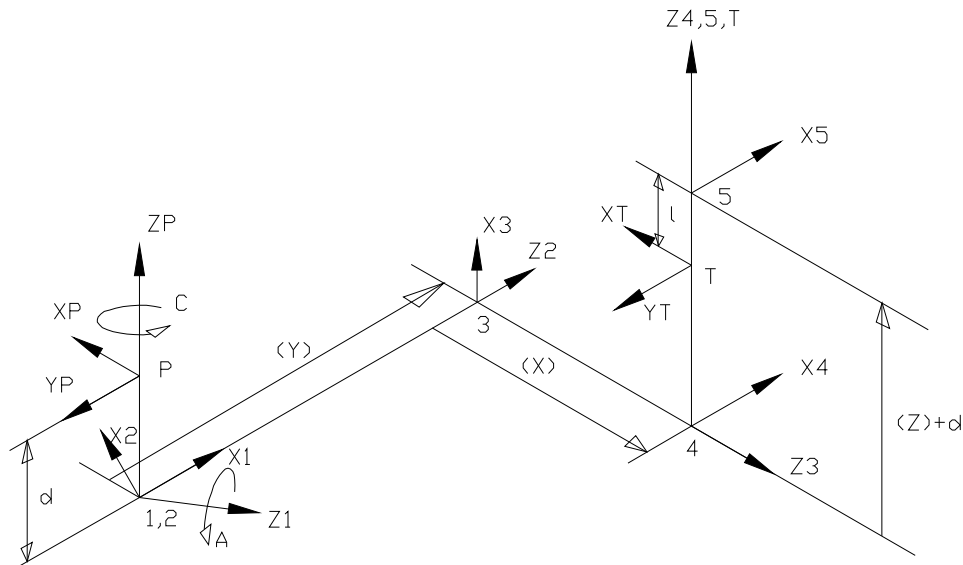


Fig. 3. The DH coordinate frames of the Huron KX8-Five from frame P to T through C A Y X Z.

Table 1. DH parameters of the Huron KX8-Five

i	θ_i	a_i	b_i	α_i	$i+1$
P	C-90	0	-d	45	1
1	A+90	0	0	90	2
2	45	0	Y	-90	3
3	-90	0	X	-90	4
4	0	0	Z+d	0	5
5	90	0	-I	0	T
frame	degree	mm	mm	degree	frame

2.1 Displacement Analysis

The motion of the tool tip with respect to the table of the milling centre can be modelled as the position and orientation of frame T relative to frame P, and computed as

$$A = A_1 A_2 A_3 A_4 A_5 = \begin{bmatrix} n & o & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix}_{4 \times 4} \quad (1)$$

where p is the position vector of the origin of T in P, and a the unit vector along the tool rotation axis and pointing toward the spindle.

The homogeneous transformation matrix corresponding to this sequence of motions is given as

$$A_i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \theta_i & \cos \theta_i & b_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Once substituting the DH parameters into eqs.(1) and (2), the tool tip position is given as

$$p = [x \quad y \quad z]^T \quad (3)$$

with

$$\begin{aligned} x &= YS_C C_A - \frac{1}{2} C_C (Z+d-l) - \frac{1}{2} X C_C + \frac{\sqrt{2}}{2} X S_C S_A - \frac{\sqrt{2}}{2} S_C S_A (Z+d-l) - \frac{1}{2} X C_C C_A + \frac{\sqrt{2}}{2} Y C_C S_A \\ y &= -Y C_C C_A - \frac{1}{2} S_C (Z+d-l) - \frac{1}{2} X S_C + \frac{\sqrt{2}}{2} Y S_C S_A + \frac{\sqrt{2}}{2} C_C S_A (Z+d-l) + \frac{1}{2} S_C C_A (Z+d-l) - \frac{\sqrt{2}}{2} X C_C S_A, \\ z &= -\frac{1}{2} X C_A + \frac{1}{2} X + \frac{\sqrt{2}}{2} Y S_A + \frac{1}{2} C_A (Z+d-l) + \frac{1}{2} (Z+d-l) \end{aligned} \quad (4)$$

while the tool orientation is given as

$$a = [i \quad j \quad k]^T \quad (5)$$

with

$$\begin{aligned} i &= -\frac{1}{2}C_C - \frac{\sqrt{2}}{2}S_C S_A + \frac{1}{2}C_C C_A \\ j &= -\frac{1}{2}S_C + \frac{\sqrt{2}}{2}S_C S_A + \frac{1}{2}S_C C_A, \\ k &= \frac{1}{2}C_A + \frac{1}{2} \end{aligned} \quad (6)$$

where S_A , C_A , S_C and C_C denote the $\sin(A)$, $\cos(A)$, $\sin(C)$ and $\cos(C)$, respectively. Moreover, l is the tool length, i.e., the distance between the gauge plane and the ball centre, while d is the distance between the surface of table and the intersection of the joint axes C and A . Moreover, from the configuration data, the joint limits of the Huron KX8-five are extracted and listed as follows:

$$\begin{aligned} -315 \text{ mm} &\leq X \leq 335 \text{ mm} \\ -350 \text{ mm} &\leq Y \leq 350 \text{ mm} \\ 0 \text{ mm} &\leq Z \leq 450 \text{ mm} \\ -180^\circ &\leq A \leq 45^\circ \\ -99\,999.999^\circ &\leq C \leq 99\,999.999^\circ \\ d &= 75 \text{ mm} \end{aligned} \quad (7)$$

2.2 Velocity Analysis

In the particular case of tool-tip milling, we have to consider the velocity of the tool-tip with respect to the joint velocities, but the tool tip orientation remains constant since it is determined by the Cartesian axes and therefore will not be \dot{p} retained in tool tip velocity. Hence, the required Jacobian matrix of the milling machine, $\dot{\theta}$ namely J , is used to \dot{p} map the joint-rates into the tool tip velocity as

$$J\dot{\theta} = \dot{p} \quad (8)$$

with

$$\dot{\theta} = [\dot{C} \quad \dot{A} \quad \dot{Y} \quad \dot{X} \quad \dot{Z}]^T; \quad \dot{p} = [\dot{x} \quad \dot{y} \quad \dot{z}]^T$$

where J is defined as

$$J \equiv \frac{\partial p}{\partial \theta} = \begin{bmatrix} \frac{\partial p}{\partial C} & \frac{\partial p}{\partial A} & \frac{\partial p}{\partial Y} & \frac{\partial p}{\partial X} & \frac{\partial p}{\partial Z} \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} & J_{15} \\ J_{21} & J_{22} & J_{23} & J_{24} & J_{25} \\ J_{31} & J_{32} & J_{33} & J_{34} & J_{35} \end{bmatrix}_{3 \times 5}, \quad (9)$$

with its components symbolically described in Appendix A. Note that in the case of this milling machine type, the Jacobian matrix is in a (3 X5) rectangular format.

3. PROBLEM FORMULATION

The solution of eq.(8) is the well-know result:

$$\dot{\theta} = J^+ \dot{p} + (1 - J^+ J) h ; J^+ \equiv J^T (J J^T)^{-1} \quad (10)$$

where J^+ is the right-generalized inverse of J and h an arbitrary vector pointing toward the satisfaction of a secondary task. To express J^+ , the Moore–Penrose pseudo-inverse shall be implemented, [Moore 20], [Penrose 55], [Golub 96]. Let k be the rank of the ($m \times n$) matrix J and let J be decomposed as $J = B C$, where B is a ($m \times k$) matrix and C is a ($k \times n$) matrix. Then

$$J^+ = C^T (C C^T)^{-1} (B^T B)^{-1} B^T ;$$

(11)

If J has full row rank, so that $k = m$, then B can be chosen to be the identity matrix and the formula reduces to $J^+ = J^T (J J^T)^{-1}$. Similarly, if J has full column rank (that is, $k = n$), we have $J^+ = (J^+ J)^{-1} J^T$. In the case under study, the rows of matrix J are usually linearly independent; hence, the following formula shall be implemented:

$$J^+ \equiv J^T (J J^T)^{-1} ;$$

(12)

It is noteworthy that $(1 - J^+ J)$ is an orthogonal complement of J projecting h onto the nullspace of J . Clearly, the first part of eq.(10) is the minimum-norm solution of eq.(8) allowing the tool-tip to follow the required trajectory, while the second part of eq.(10) is an homogenous solution allowing to control the tool orientation. The projection of h onto the nullspace of J allows choosing a tool orientation without modifying the tool-tip position whatever the kinematic chain of the machine-tool we have. This is the so-called *self-motion* of the machine-tool, defined as the possible joint motions which keep the tool-tip to a constant position in the part frame. Consequently, vector h is determined as the normality condition of an optimization problem specifying our preferences.

In the context of tool-tip milling operations, it is possible to machine the required surface with different tool orientations. Three orientation approaches can be identified. Firstly, the easiest approach implements one constant tool orientation during the milling operation. This orientation is usually vertical, horizontal or equal to an average normal to the main workpiece surface. This greatly simplifies the post-processing operations, and, as we have observed, may result in different surface finishes. In the worst case, some complex shapes may not even be feasible, since the tooltip angle of attack on the workpiece becomes almost parallel to the surface and thus may not reach proper chip removal conditions. Secondly, the best finish approach always keeps the tool axis normal to the surface in order to ensure the best possible surface finish. For complex surface shapes, the variation of surface normal becomes large, and the post-processing may result in exceeding of the joint limits. Moreover, in the case of complex surfaces which feature no symmetries and which are represented by sets of points, the actual surface normal calculation is only leading to an approximation. Thirdly, in our proposed approach, it is possible to take advantage of the self-motion of the machine-tool in order to prepare an optimal joint-space path which insures that the tool-tip follows the required Cartesian trajectory, while satisfying an objective function that specifies our preferred of tool orientation, range of tool-tip orientations leading to satisfactory surface finish and joint-limit avoidance strategy. Hence, this optimization problem shall be handled by a novel optimal post-processing module taking into account the manipulator kinematics.

Let \bar{a} be our preferred tool orientation either constant (vertical, horizontal or an arbitrary orientation) or the surface normal at each given point of the tool-tip trajectory, while a is the actual tool orientation given as in eq.(6). Moreover, let $\bar{\theta}$ be the mid-joint position, while θ is the actual joint position defined as

$$\bar{\theta} = (\theta_{\max} + \theta_{\min})/2; \theta \equiv [C \ A \ Y \ X \ Z]^T. \quad (13)$$

To determine the objective function ζ , the following principles are thus implemented. The tool orientation vector a will be kept as close as possible to our preferred tool orientation vector \bar{a} and simultaneously the joint position vector θ will be selected as close as possible to the mid-joint position vector $\bar{\theta}$, i.e.,

$$\zeta = \frac{1}{2}(\theta - \bar{\theta})^T W^T W (\theta - \bar{\theta}) + w_a(1 - a^T \bar{a}) \rightarrow \min_{\theta} \quad (14)$$

where $W = \text{diag}(w_1, w_2, w_3, w_4, w_5)$ is a diagonal 5×5 positive definite matrix and w_a a positive scalar balancing the relative criteria weights. The arbitrary vector h in eq.(10) is thus chosen as minus the gradient of the objective function, i.e.,

$$h = -\nabla \zeta = -W(\theta - \bar{\theta}) + w_a \frac{d}{d\theta} (a^T \bar{a}) = [h_1 \ h_2 \ h_3 \ h_4 \ h_5]^T, \quad (15)$$

with its components described in Appendix B.

4. THE OPTIMAL POST-PROCESSING MODULE

When the cutter location data, namely CL-data, is generated by a computer-assisted manufacturing system (CAM), it is assumed that the tool path between two CL-data points is a straight line relative to the workpiece. However, due to the machine tool rotary axes, the tool path between two blocks in the numerical control (NC) program will not be linear relative to the workpiece, thereby reducing the accuracy of the tool path. Linearization of the tool path in the post-processing module can solve this problem. Moreover, the inverse kinematics of the machine-tool is performed using the velocity relationship of eq.(10) and the normality condition of eq.(15).

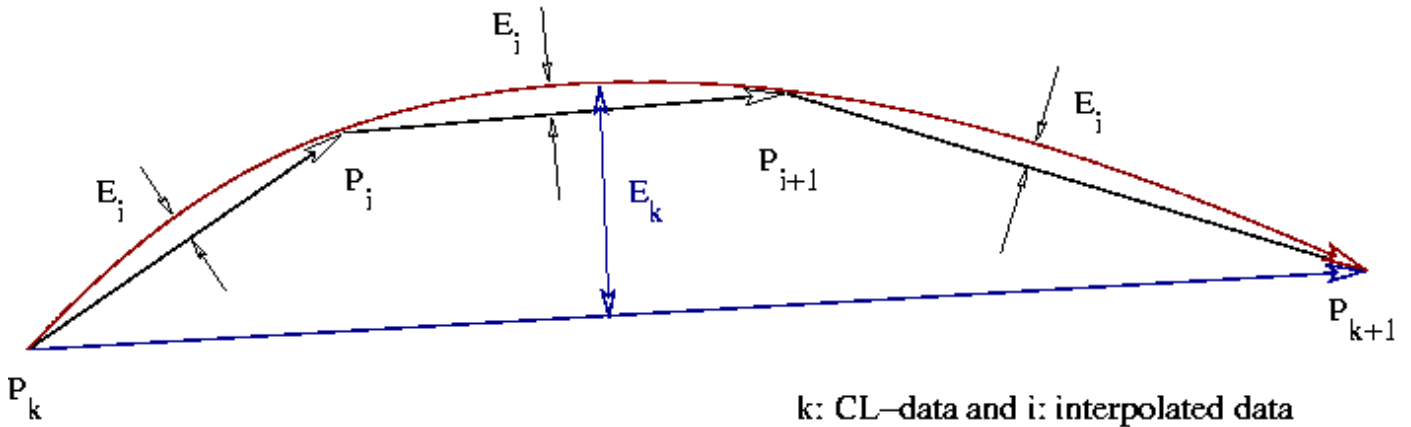


Fig. 4 – Interpolation of new points between CL-data

4.1 LINEARIZATION

Without linearization, each CL-data point is directly transformed into only one block of NC program, e.g., a G01 command. In this case, the number of lines of the CL-data is approximately equal to the number of blocks in the NC program. With the linearization of the tool path, the post-processor must interpolate new CL-data points along the ideal tool path, thereby adding new blocks into the NC program. One CL-data point may result in several blocks in the NC program. Linearization does not provide a perfect tool path, but the deviation from the required tool path can be reduced to an acceptable level. Linearization algorithms must implement calculations of

both forward and inverse kinematics to compute deviations from the ideal tool path and interpolate new CL-data. We have used the recursive linearization method proposed by Sorby [Sorby 07]. It takes into account deviations of both positions and orientations of the tool from the required tool path.

4.2 INVERSE KINEMATICS

An iterative numerical solution method is used to solve the inverse kinematics in a similar manner as the resolve-motion rate method introduced in [Whitney 69].

4.3 THE POST PROCESSING ALGORITHM

Based on the solution of eq.(10) and the secondary task expressed by vector h of eq.(15), we have implemented an optimal post-processing module as summarized in Table 2.

Once θ_i is set to the actual joint position, we read in the CL-data file the required tool-tip position \bar{p} and orientation \bar{a} . In step 3, we compute the corresponding tool-tip position p and orientation a from which we obtain the tool-tip deviation Δp (in fact, only half of it in order to avoid numerical instabilities). Once the arbitrary vector h is evaluated, the joint motion $\Delta\theta$ is computed with the generalized inverse J^+ . In step 8, the next joint position θ_{i+1} is computed with a damping factor of 0.8, for which we recompute the corresponding tool-tip deviation Δp . Finally, if this tool-tip deviation Δp is smaller than the maximum threshold ε and if we are still under the maximum number of iteration n , we go back to step 6.

Table 2. The optimal post-processing module algorithm

```

// Initialize  $\varepsilon = 0.000001$ ;  $n=500$ ; and the weights  $W, W_a$ ;
1.  $i=0$ ;  $\theta_i \leftarrow$  actual joint position;
2.  $\bar{p}, \bar{a} \leftarrow$  read one CL-data;
3.  $p, a, J \leftarrow$  forward kinematics( $\theta_i$ ); //eqs.(3), (5) and (9)
4.  $\Delta p = 0.5(\bar{p} - p)$ ;
5. do
6.  $h = -\nabla \zeta$ ; // eq.(13)
7.  $\Delta \theta = J^+ \Delta p + (1 - J^+ J)h$ ; // eq.(10)
8.  $\theta_{i+1} = \theta_i + 0.8 \Delta \theta$ ;  $i = i + 1$ ;
9.  $p, a, J \leftarrow$  forward kinematics( $\theta_i$ ); //eqs.(3), (5) and (9)
10.  $\Delta p = 0.5(\bar{p} - p)$ ;
11. while( $\|\Delta p\| \geq \varepsilon$  and  $i \leq n$ )

```

5. EXPERIMENTAL VALIDATION

5.1 THE MILLING PROCESS

The optimal post-processing module shown in Table 2 has been implemented in MatLab, especially written for the milling centre Huron KX8-Five. The CL-data was generated by the program CATIA V5 for the free-form surface shown in Fig. 5. The surface is inscribed within a 4-inch (101,6 mm) diameter cylinder. The surface shape was especially chosen to have the tool normal axis to exceed the limits of joint A of the KX8-five milling centre. A multi-axes “zigzag” strategy with a 0.5 inch (12,7 mm) ball-end mill tool is chosen as the finish milling operation.

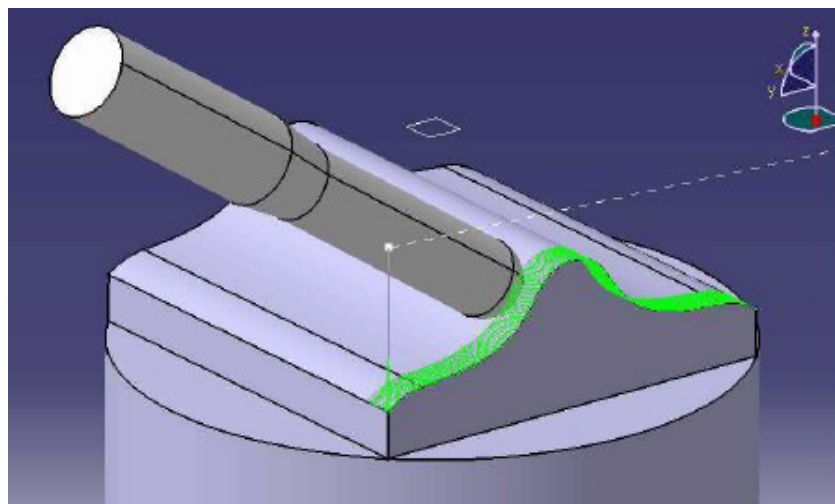


Fig. 5. Tool path generated by CATIA V5.

5.2 THE POST-PROCESSING

After linearization of the CL-data, the result is post-processed with the aforementioned tool orientation strategies:

- 1) the tool is always kept vertical;
- 2) the tool is kept at a constant orientation but not vertical;
- 3) the tool is kept normal to the surface; and finally,
- 4) the tool is kept as close as possible to the normal and also the joints are kept as close as possible to the mid-joint position (see eq.(14)).

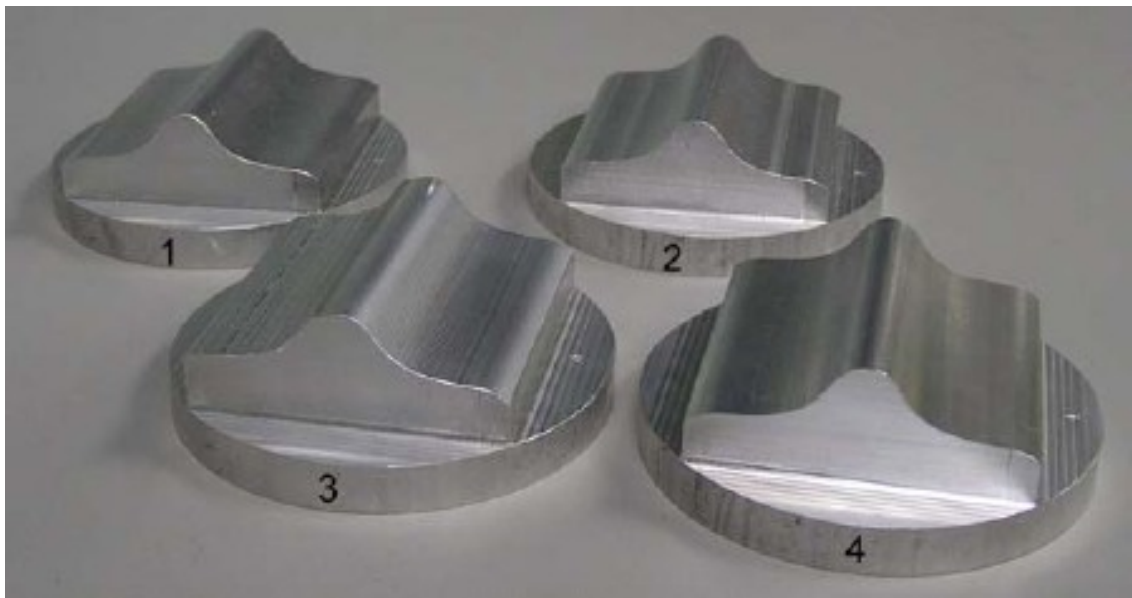


Fig. 6 The four machined parts.

5.3 MILLING RESULTS AND TOOL NORMAL

As shown in Fig. 6, tests #1 and #2 have succeeded in machining the required part. As can be seen on the image, test #3 was able to completely machine the required part, while keeping the tool normal to the surface, but failed to obtain the smooth surface finish. In fact, without the joint-limit avoidance strategy, to perform the milling task on part #3 did kinematically require joint A positions to range from -82.5° to $+82.5^\circ$ (see Fig. 7(a)), which is clearly out of its limits, i.e., $-180^\circ \leq A \leq +45^\circ$, therefore explaining the chatter. Alternatively, with the joint-limit avoidance strategy, test #4 has succeeded in machining the required part. In fact, part #4 requires positions of joint A from -78.5° to $+43.5^\circ$ (see Fig. 7(b)), which is clearly within the joint limits.

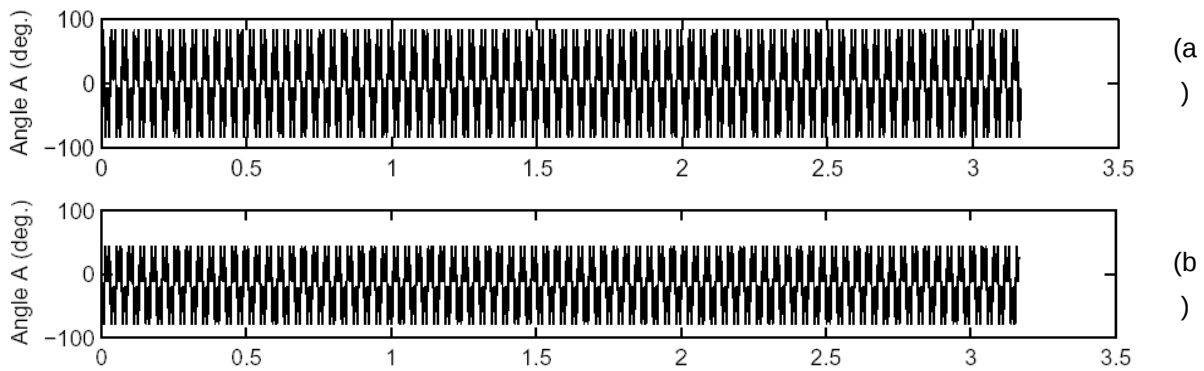


Fig. 7 Positions of joint A during (a) test #3 without the strategy; (b) test #4 with the strategy;

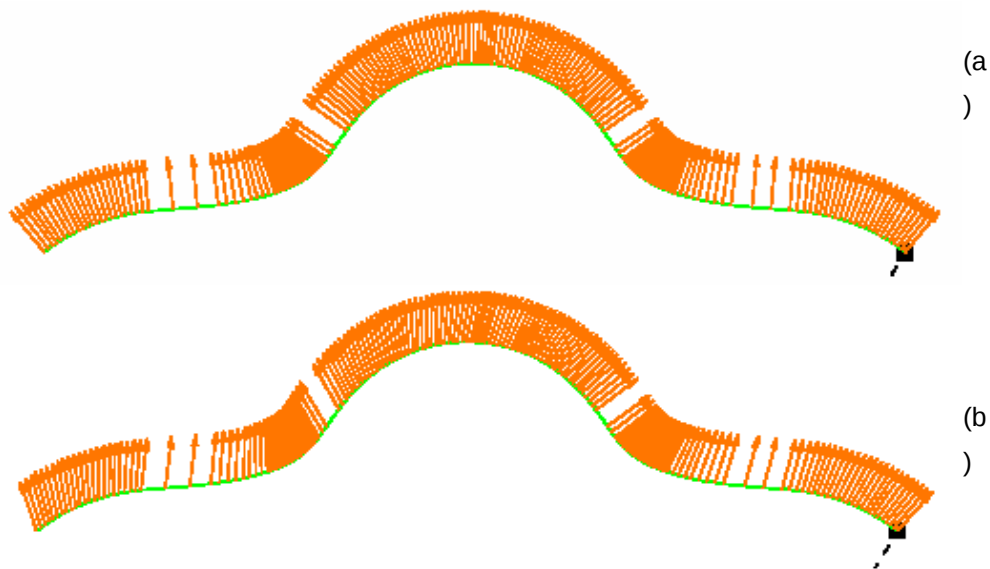


Fig. 8 Tool orientations for (a) test #3; (b) test #4

Figure 8(a) shows the resulting tool orientation of test #3, which is obviously normal to the surface. Alternatively, figure 8(b) shows the resulting tool orientation of test #4, which is not always normal to the surface. Apparently, the tool orientation is kept normal to the surface on the right side and far from the normal on the left side. On the right side, it is possible to bring the tool to the normal at -82.5° , because it is far away from the limit of -180° . On the left side, it is not possible to bring the tool to the normal at $+82.5^\circ$, which is beyond the limit of $+45^\circ$, and thus, the optimal post-processing module kept it to $+43.5^\circ$.

5.4 MILLING RESULTS AND SURFACE ROUGHNESS

As shown in Table 3, the roughness of the four machined surfaces has been measured by a Mitutoyo SurfTest measurement machine. For all tests, roughness results yield results below the micron. They seem very close together. However, the smallest roughness is being measured on part #4 obtained with our optimal post-processing module.

Table 3. Average roughness of surfaces

Part number	Average of roughness Ra (μ m)	Improvement (%)
1	1.54	0
2	1.45	5.84
3	1.5	2.60
4	1.34	13 (10.7)

The first strategy is set as the reference, since it would be the easiest milling approach. Hence, we can compare the roughness in terms of improvements related to that reference as shown on table 3. In parenthesis, the improvement percentage is given vs the third strategy. Our approach clearly improves surface finish by more than 10% which justifies the post-processor presence in the off-line milling process. Note that these tribology results are confirming visual analysis of the surface finish.

6. CONCLUSION

Based on the velocity relationship of the machine-tool, we have proposed an optimal post-processing module for tool-tip milling operations, where the tool-tip follows the required trajectory. During the milling process, the module tries to keep the tool axis as close as possible to the surface normal and simultaneously tries to keep the milling machine joint positions as far as possible to the joint limits. This dual goal was solved by an optimization problem solved by the post-processor which succeeded in machining parts that were not possible beforehand, i-e to be completely machined without exceeding the joint limits or excess chatter.

Finally, our optimal post-processing module has been tested on many different complex surfaces with the simulation software VERICUT and experimentally validated on a milling centre Huron KX8-five. The results show that the post-processor not only could produce good resulting surface quality but was able to significantly improve the surface finish. The results proved equally good for different tool orientation preferences.

ACKNOWLEDGMENT

The authors want to acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada under grants RGPIN-203618. The authors also would like to acknowledge the research results obtained by Mr. Hojjat Valipour and reported in his master's thesis.

REFERENCES

Angeles, J., Anderson, K. and Gosselin, C., "An Orthogonal-Decomposition Algorithm for Constrained Least-Square Optimization", In proceedings of the 13th ASME design and automation conference, *ASME Robotics, Mechanisms, and Machine Systems*, Design Eng. Division, Vol. 2, pp. 215-220, 1987.

Angeles, J., "Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms", Second Edition, Springer-Verlag, New York, 2002.

Baron, L., Mayer, R., Arghavani, J. and Fortin C., 1995, Literature Review of the Kinematic Theories Pertaining to NC Post-Processors", CDT Research Contract No. P1981, Ecole Polytechnique, 75 pages.

Baron, L., "An optimal surfacing post-processor module for 5-axes cnc milling machines". *Third International Conference on Industrial Automation*, Montreal, Canada, 1999, pp.23.17-23.23.

Ding, M Z, Ong, C J and Poo A N. Resolution of redundant manipulators via distance optimization

. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 2000 214:

1037

Golub, G. H.; Charles F. Van Loan (1996). "*Matrix computations*" (3rd edition ed.). Baltimore: Johns Hopkins. pp. 257–258.

Kruth J.P. et al. "NC-postprocessing and NC-simulation for five-axis milling operations with automatic collision avoidance". In *International Journal for Manufacturing Science and Technology*, Vol. 1, No. 1, 1998.

Lee R. -S. and She C. -H. "Developing a postprocessor for three types of five-axis machine tools". *The International Journal of Advanced Manufacturing Technology*. Springer London, Volume 13, Number 9, septembre 1997, pp 658-665.

Liangji C. "Kinematics Modeling and Post-Processing Method of Five-Axis CNC Machine". ETCS '09 Proceedings of the 2009 First International Workshop on Education Technology, vol. 1, 2009. Pages 300-303.

Moore, E. H. (1920). "On the reciprocal of the general algebraic matrix". *Bulletin of the American Mathematical Society* 26: 394–395.

Nenchey, D. Redundancy Resolution through

Local Optimization: A Review

. In *Journal of Robotic Systems* Vol. 6, No. 6, pages 769-798, 1989.

Penrose, R. (1955). "A generalized inverse for matrices". *Proceedings of the Cambridge Philosophical Society* 51: 406–413.

Sørby K. "Inverse kinematics of five-axis machines near singular configurations". *International Journal of Machine Tools and Manufacture*. Volume 47, Issue 2, February 2007, pp 299-306.

Siciliano, B., "Solving Manipulator Redundancy with the Augmented Task Space Method Using the Constraint Jacobian Transpose", *IEEE Int. Conference on Robotics and Automation*, Tutorial M1, pp. 5-1-5-8, 1992.

Whitney, D.E., "Resolved motion rate control of manipulators and human prostheses", *IEEE Trans. Man-Machine Syst.*, Vol. 10, No. 2, pp. 47-53, 1969.

Appendix A – The components of the Jacobian matrix of the Huron KX8-five milling centre

$$J_{11} = \frac{\partial x}{\partial C} = -\frac{\sqrt{2}}{2} C_C S_A (Z+d-l) + \frac{1}{2} S_C (Z+d-l) + C_C C_A Y \\ - \frac{1}{2} S_C C_A (Z+d-l) + \frac{1}{2} S_C X + \frac{\sqrt{2}}{2} C_C S_A X - \frac{\sqrt{2}}{2} S_C S_A Y + \frac{1}{2} S_C C_A X;$$

$$J_{21} = \frac{\partial y}{\partial C} = -\frac{\sqrt{2}}{2} S_C S_A (Z+d-l) - \frac{1}{2} C_C (Z+d-l) + S_C C_A Y \\ + \frac{1}{2} C_C C_A (Z+d-l) - \frac{1}{2} C_C X + \frac{\sqrt{2}}{2} S_C S_A X + \frac{\sqrt{2}}{2} C_C S_A Y - \frac{1}{2} C_C C_A X;$$

$$J_{31} = \frac{\partial z}{\partial C} = 0;$$

$$J_{12} = \frac{\partial x}{\partial A} = -\frac{\sqrt{2}}{2} S_C C_A (Z+d-l) - S_C S_A Y - \frac{1}{2} C_C S_A (Z+d-l) \\ + \frac{\sqrt{2}}{2} S_C C_A X + \frac{\sqrt{2}}{2} C_C C_A Y + \frac{1}{2} C_C S_A X;$$

$$J_{22} = \frac{\partial y}{\partial A} = \frac{\sqrt{2}}{2} C_C C_A (Z+d-l) + C_C S_A Y - \frac{1}{2} S_C S_A (Z+d-l) \\ - \frac{\sqrt{2}}{2} C_C C_A X + \frac{\sqrt{2}}{2} S_C C_A Y + \frac{1}{2} S_C S_A X;$$

$$J_{32} = \frac{\partial z}{\partial A} = \frac{1}{2} S_A X - \frac{1}{2} S_A (Z+d-l) + \frac{\sqrt{2}}{2} C_A Y;$$

$$J_{13} = \frac{\partial x}{\partial Y} = \frac{\sqrt{2}}{2} S_C S_A - \frac{1}{2} C_C C_A - \frac{1}{2} C_C;$$

$$J_{23} = \frac{\partial y}{\partial Y} = -C_C C_A + \frac{\sqrt{2}}{2} S_C S_A;$$

$$J_{33} = \frac{\partial z}{\partial Y} = \frac{\sqrt{2}}{2} S_A;$$

$$J_{14} = \frac{\partial x}{\partial X} = \frac{\sqrt{2}}{2} S_C S_A - \frac{1}{2} C_C C_A - \frac{1}{2} C_C;$$

$$J_{24} = \frac{\partial y}{\partial X} = -\frac{\sqrt{2}}{2} C_C S_A - \frac{1}{2} S_C C_A - \frac{1}{2} S_C;$$

$$J_{34} = \frac{\partial z}{\partial X} = -\frac{1}{2} C_A + \frac{1}{2};$$

$$J_{15} = \frac{\partial x}{\partial Z} = -\frac{\sqrt{2}}{2} S_C S_A + \frac{1}{2} C_C C_A - \frac{1}{2} C_C;$$

$$J_{25} = \frac{\partial y}{\partial Z} = \frac{\sqrt{2}}{2} C_C S_A + \frac{1}{2} S_C C_A - \frac{1}{2} S_C;$$

$$J_{35} = \frac{\partial z}{\partial Z} = \frac{1}{2} C_A + \frac{1}{2};$$

Appendix B – The components of the normality condition h

$$h_1 = -\frac{\partial \zeta}{\partial C} = -w_1(C - \bar{C}) + w_a \left(\frac{1}{2} S_C \bar{a}_x - \frac{1}{2} S_C C_A \bar{a}_x - \frac{\sqrt{2}}{2} C_C S_A \bar{a}_x - \frac{1}{2} S_C C_A \bar{a}_y + \frac{1}{2} S_C \bar{a}_x - \frac{\sqrt{2}}{2} C_C S_A \bar{a}_y \right);$$

$$h_2 = -\frac{\partial \zeta}{\partial A} = -w_2(A - \bar{A}) + w_a \left(\frac{\sqrt{2}}{2} C_C C_A \bar{a}_y - \frac{1}{2} C_C S_A \bar{a}_x - \frac{\sqrt{2}}{2} S_C C_A \bar{a}_x - \frac{1}{2} S_C S_A \bar{a}_y - \frac{1}{2} S_A \bar{a}_z \right);$$

$$h_3 = -\frac{\partial \zeta}{\partial Y} = -w_3(Y - \bar{Y});$$

$$h_4 = -\frac{\partial \zeta}{\partial X} = -w_4(X - \bar{X});$$

$$h_5 = -\frac{\partial \zeta}{\partial Z} = -w_5(Z - \bar{Z});$$