Iterative Tool Path Optimization for Five-Axis Machines with Optimal Point Insertion

M. Munlin[1] and S.S. Makhanov[2]

[1]Mahanakorn University of Technology, Thailand, mmunlin@gmail.com
[2]Sirindhorn International Institute of Technology, Thammasat University, Thailand, mahhanov@siit.tu.ac.th

ABSTRACT

We present a new algorithm to iteratively switch the rotation angles of five-axis milling machine when additional cutter location points are inserted. The efficiency of the algorithm has been verified by a virtual machining as well as by real cutting on five-axis machine MAHO600E. The second part of the paper deals with a closed form symbolic representation of the kinematics errors. It is derived as a function of locations of the additional cutter contact points. The closed form solution is obtained from the inverse kinematics associated with a particular five-axis machine through symbolic transformations. Simulation and cutting experiments demonstrate that the proposed procedure outperforms techniques based on the equi-spaced grids.

Keywords: inverse kinematics, 5-axis machining, tool path optimization, shortest path.
DOI: 10.3722/cadaps.2012.33-41

1 INTRODUCTION

Milling machines are programmable mechanisms for cutting industrial parts. The machine consists of several moving parts designed to establish the required coordinates and orientations of the tool during the cutting process. The axes of the machine define the number of the degrees of freedom of the cutting device. The movements of the machine parts are guided by a controller which is fed with a so-called NC program comprising commands carrying spatial coordinates of the tool-tip and angles needed to rotate the machine parts to establish the orientation of the tool [2],[4-5]. The rotations invoke large kinematics errors. Besides the machines with rotation axes on the table often have to turn around heavy work pieces. As a result, the machines may have low capacities for acceleration which significantly increases machining time, and is amplified in HSM when the rotation axes reach greater speeds. In [3] the authors analyze the sequence of rotations to minimize the number of the phase reverse steps at discontinuities of the first derivative of the surface (corners etc). A method of avoiding singularities has been presented in [1]. The method certainly has its merits since it allows inserting additional points without any modifications. However, the computation is complex, computationally expensive and does not preserve the original CC points. In [7] the authors proposed an angle switching algorithm to optimize the sequence of the rotation angles without increasing the number of tool positions or changing the tool orientation. The main idea is to minimize the distance traveled by the tool in the angular space at the expense of using multiple solutions of the inverse

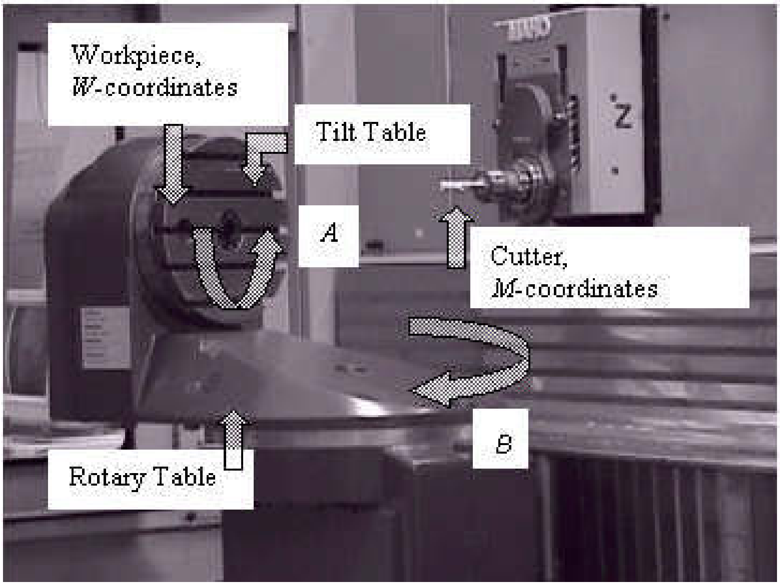


Fig. 1: Five-axis milling machine MAHO600E.

kinematics equations that is switching the rotation angles at certain position. Considering the entire set of angles requires the shortest path techniques to minimize the total angular distance. In [6] the algorithm was extended to the case when the cost function differentiates between more damaging undercuts and repairable over cuts.

However, inserting new points in the resulting path may not be always possible because some optimization is based on replacing large angle variations by smaller ones using different combinations of the angles [6]. Inserting the additional points decreases these variations so that the replacement may not be acceptable. In turn, a simple removal of an angle switch is not suitable either since it may affect the entire tool path. Therefore, we propose a new algorithm to iteratively switch the angles when the additional points are inserted. The efficiency of the algorithm has been verified by a virtual machining as well as by real cutting on five-axis machine MAHO600E. In some cases the proposed method requires 47% less additional points than conventional schemes performed near the stationary points.

The second part of the paper deals with a method of inserting additional points. A one dimensional grid of the CL or CC points is generated using a direct minimization of the kinematics error. A closed form representation of the kinematics error as a function of locations of the cutter contact points is derived from the inverse kinematics associated with a particular five-axis machine and obtained through automatic symbolic calculations. The grid of cutter contact points is generated so that it minimizes the kinematics error. Numerical and cutting experiments demonstrate that the proposed procedure outperforms distribution of points based on the equi-spaced grid.

2 THE SHORTEST PATH ALGORITHM FOR ANGLE SEQUENCING

In this section we present a concise version of the angle correction algorithm presented in [7]. Consider a typical five-axis milling machine in Fig. 1.

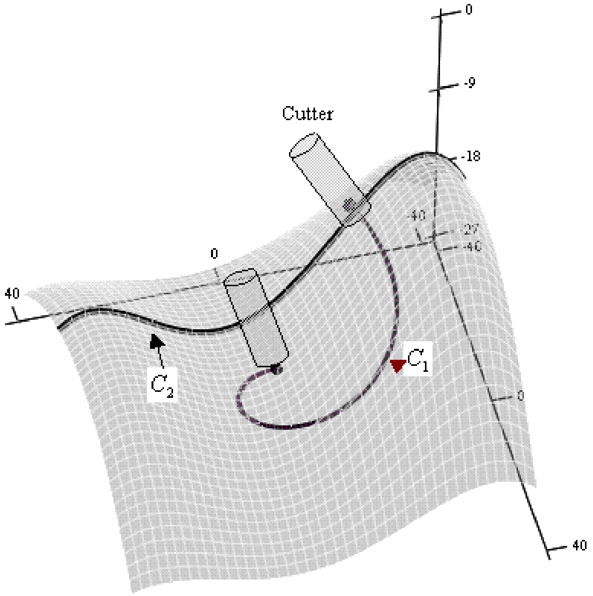


Fig. 2: Non-linearity of the tool-path in the workpiece coordinates C_1

The kinematics of the machine depend on matrix-functions $A(a)$, $B(b)$ associated with the rotations a and b around the primary (the rotary table) and the secondary (the tilt table) axes shown in Fig. 1. A simple analysis of the inverse kinematics equations reveals that a linear trajectory of the tool tip in the machine coordinates may produce a non-linear trajectory in the work piece coordinates (see Fig. 2 generated by our simulation software). Approaching the stationary point (minimum, maximum of saddle) involve sharp variations of the rotation angles even when the spatial steps are small.

Note that a fine cut of a smooth surface employing small spatial and angular steps may not demonstrate the detrimental effects near the singularity points. However, a rough cut characterized by large gradients could produce considerable errors because of the sharp angular jumps that the machine produces the loop-like trajectories of the tool. Moving along such trajectories may destroy the work piece and even lead to a collision with the machine parts. If the tool is aligned along the surface normal, then the rotation angles are evaluated by

$$a_{base} = \begin{cases} \arctan\left(\dfrac{I_y}{I_x}\right) & \text{if } I_x > 0 \text{ and } I_y > 0, \\ \arctan\left(\dfrac{I_y}{I_x}\right) + \pi & \text{if } I_x < 0, \\ \arctan\left(\dfrac{I_y}{I_x}\right) + 2\pi & \text{otherwise,} \end{cases} \qquad b_{base} = -\arcsin I_z, \tag{1}$$

where $I = (I_x, I_y, I_z)$ is the tool orientation vector. It is not hard to demonstrate that the inverse kinematics admits four solutions given by:

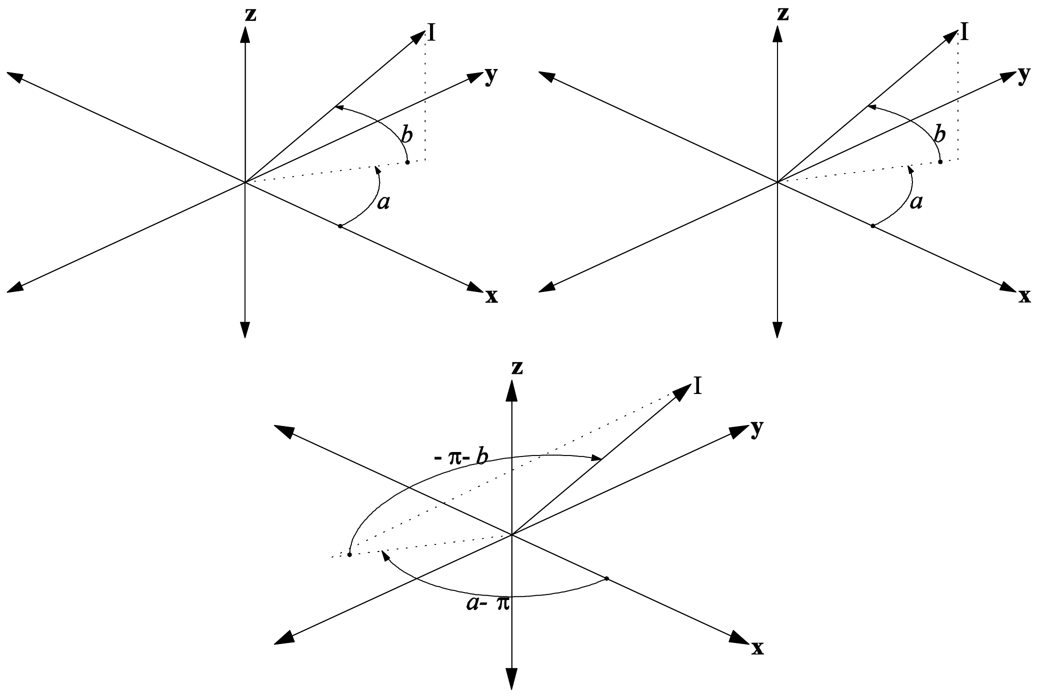


Fig. 3: Possible combinations of the rotation angles.

$$\Lambda = \begin{cases} a_{base}, b_{base} \\ a_{base} - 2\pi, b_{base} \\ a_{base} - \pi, -b_{base} - \pi \\ a_{base} + \pi, -b_{base} - \pi. \end{cases} \tag{2}$$

Fig. 3 illustrates formula (2).

Note that in many cases minimizing the kinematics errors can be replaced by minimization of the total angle variation as follows

$$\underset{\Lambda}{\text{minimize}}\,(w), \tag{3}$$

where $w = \frac{1}{N-1}\sum_{p=1}^{N-1}(a_p - a_{p+1})^2 + (b_p - b_{p+1})^2$ and N is the total number of the CL points.

In principle, angles a_p and b_p can be constrained depending on the configuration of the machine. For instance, the range of a_p and b_p for configuration in Fig. 1 is $-105^0 \le b_p \le 105^0, -\infty < a_p < +\infty$. However, problem (2)-(3) deals with the angles which have been already evaluated by a correct angle generation algorithm which includes constraints (if any). For each step we only select one of the 4 possible options given by (2). Therefore, (2)-(3) can be solved by a standard shortest path algorithm. The shortest path techniques performed without changing the

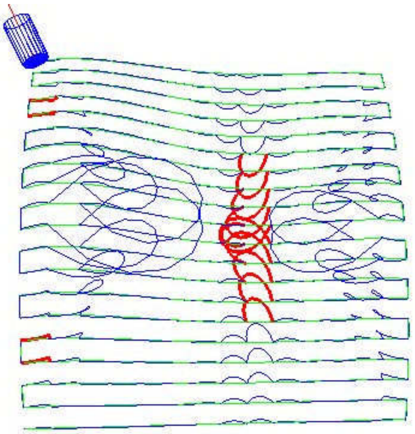


Fig. 4: The original trajectories.

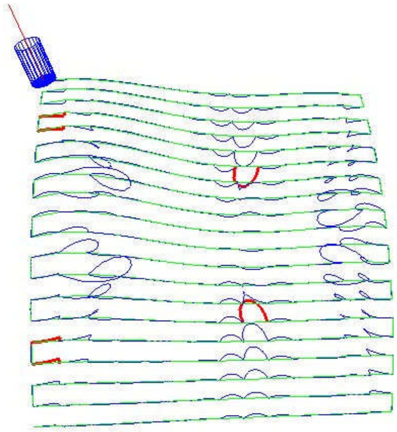


Fig. 5: The trajectories repaired using the shortest path algorithm.

spatial positions of the tool often leads to impressive results. The "before and after" trajectories are depicted in Fig. 4 and Fig. 5 for a rough cut of a test surface.

Fig. 4 shows trajectories of the tool produced with the original set of angles evaluated by (2). The trajectories include many unwanted loops and undercuts produced by kinematics of the particular machine. Such trajectories may destroy the surface or even lead to collisions with the machine parts. The commercial toolpath generation software usually linearizes the trajectories and does not take into account the kinematics of the machine. However, the real trajectories on the machine will be the ones on Fig. 4. This has been proven experimentally in [6] and [7]. Of course, a fine cut employing very small spatial and angular steps may not need the proposed optimization. However, a rough cut characterized by large steps and high surface curvatures could produce considerable errors. Finally, the tool paths in Fig. 4 and Fig. 5 have the same tool positions and orientations. The difference is only in the way the prescribed orientations are being achieved.

3 ITERATIVE SHORTEST PATH ANGLE SEQUENCING

Unfortunately, even a single additional point may destroy the integrity of a particular shortest path. It means that every time an additional point has been inserted, the shortest path must be recomputed. On the other hand, if we return back to the original (non-optimized) path the kinematics error may be very different. In order to handle this problem, we propose the following point insertion algorithm.

1. Run the shortest path (SP) algorithm [7].
2. If the kinematics error is within the prescribed tolerance, quit, otherwise, find the trajectories where the kinematics error exceeds the prescribed tolerance.
3. Mark these trajectories.
4. Return to the original tool path (ORG) and using the angle insertion (AI) algorithm [6] to insert a point inside the selected trajectory (even though in the original path they do not produce large kinematics errors).
5. Go to step 1.

We consider the equi spaced point insertion (PI) and the equi angular point insertion (AI). We will compare four methods: 1) a conventional PI without the shortest path improvement, 2) AI based on bisection [6], 3) the shortest path (SP) with PI, 4) the shortest path (SP) with AI.

Our test surface is now subjected to the above mentioned four methods after the maximum error has been reduced to a certain prescribed value. It has been proven experimentally that the proposed method requires 47.3 % less additional points than conventional schemes performed near the stationary points. The efficiency of the algorithms has been verified by the virtual milling machine simulator.

Tab. 1 compares the proposed iterative shortest path algorithm (SP-IT) with the shortest path scheme proposed in [7] combined with different methods for inserting additional CL points. PI stands for inserting equi-spaced points along the tool trajectory in the physical space (x,y,z), whereas AI

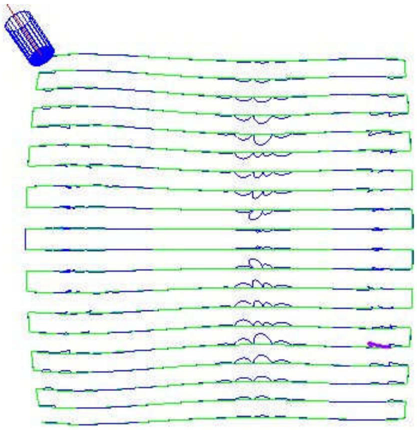


Fig. 6: Test surface and trajectories produced by SP-IT/AI.

Method	SP	PI	AI	SP-IT/PI	SP-IT/AI
# Points	0	74	48	56	39
Max Error	7.451	1.963	1.963	1.963	1.963

Tab. 1: Comparison of the inserted points with several methods on the test surface.

stands for points equi distributed in the angular space *(a,b)* along the angular arc. The maximum allowable error is 2 mm. Clearly the iterative shortest path algorithm combined with angular based inserting uses the smallest number of points. This is because the method presents an appropriate combination of angle switching and inserting, whereas SP and AI use only a single technique. The trajectories produced by SP-IT/AI are shown in Figure 6.

However, the procedure above is not optimized with regard to the position of the inserted point or a group of points. The next section deals with this problem.

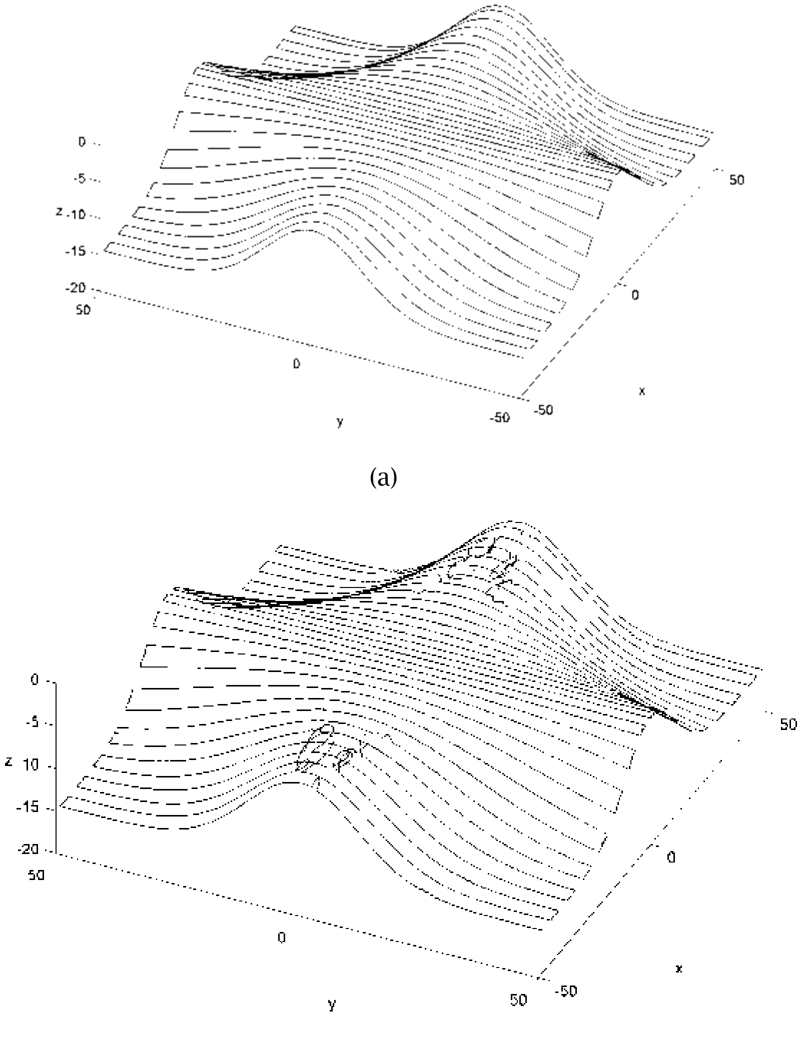
4 SYMBOLIC OPTIMIZATION

How to insert the additional points? Consider minimization of the kinematics error given by

$$\underset{\Pi}{\text{minimize}}(\varepsilon)\text{, subject to } \Pi \in \Pi'$$

where Π' is a set of curves belonging to the surface and Π is a set of cutter location points. The curves pass through a certain number of fixed points between which additional points are required.

The optimization procedure inserts a certain number of CL points between each pair of the control points in such a way that the kinematics error is minimized. In case of a non-differentiable objective function such as the one based on the Hausdorff distance (Dist_H), a non-gradient minimization procedure must be used. However, when the number of inserted points is large it may significantly increase the computational time. In order to combine the advantages of the gradient methods with the theoretical rigor of the parameterization-invariant distance the following heuristic algorithm can be used for the Hausdorff distance. First, a gradient based minimization is performed with regard to a differentiable norm such as L_2. After a certain numbers of steps the procedure switches to the generic Hausdorff distance. A further simplification is based on the following approximation. A pair of points $t_{k,1}$ and $t_{k,2}$ corresponding to the Hausdorff max-min between the discretized curve $W^D(s_k, s_{k+1}, t)$



(b)
Fig. 7: (a): Tool trajectory (optimal algorithm); (b): Tool trajectory (equal distribution).

and $W(s_k, s_{k+1}, t)$ is fixed for several iterations. This generates an auxiliary objective function given by: $\sum_k |W^D(s_k, s_{k+1}, t_{k,1}) - W(s_k, s_{k+1}, t_{k,2})|$.

After several steps the procedure switches back to the generic Hausdorff distance and a new set of the Hausdorff pairs is selected. The proposed technique is applied to a zigzag tool path designed for the surface in Fig. 7 manufactured with a 10 mm flat-end mill. The tool trajectories for the optimal and the equally arc length spaced insertion methods are shown in Fig. 7(a). Clearly, the loop-like trajectories on the top of the surface in Fig. 7(b) lead to large cutting errors and consequently to poor surface quality. The numerical comparison of the errors is shown in Tab. 2(a)-2(c).

Note that the advantage of the proposed optimization increases reaching 89% for 0.1 mm tolerance. This is because the flat-end mill allows selecting the best inclination angle providing an additional optimization variable for grid generation. Finally, the error associated with the optimal, equal arc length and the bisection techniques basically behaves similarly to the ball nose.

Threshold	Number of inserted Points		Reduction Percentage	Actual Dist_H (mm)	
$\text{Dist}_{\square}$	Bisection	Optimal Insertion		Bisection	Optimal Insertion
0.1 mm	2273	2238	1.54%	0.1000	0.0998
0.5 mm	971	925	4.74%	0.4998	0.4938
1.0 mm	687	634	7.71%	0.9987	0.9904

(a)

Threshold	Number of inserted Points	rms error (mm)				
$\text{Dist}_{\square}$		Optimal Insertion	Equi spaced	Reduction Percentage	Equi Arc length	Reduction Percentage
0.1 mm	2238	0.0467	0.4537	89.71%	0.4582	89.81%
0.5 mm	925	0.2201	0.6283	64.97%	0.5888	62.62%
1.0 mm	634	0.3978	0.9658	58.81%	0.9825	59.51%

(b)

Threshold	Number of inserted Points						
Dist_H	Bisection	Minimization					
		Rms	Reduction Percentage	Hausdorff	Reduction Percentage	rms+Hausdorff pair	Reduction Percentage
0.1 mm	2273	2238	1.54%	2236	1.63%	2221	2.29%
0.5 mm	971	925	4.74%	914	5.87%	913	5.97%
1.0 mm	687	634	7.71%	619	9.90%	584	14.99%

(c)

Tab. 2: (a): Optimal point insertion method vs. bisection; (b): Optimal point insertion vs. equally spaced points; (c): Optimal point insertion with different norms.

5 CONCLUSIONS

The proposed iterative sequencing of rotation angles based on the shortest path algorithm combined with the equidistant distribution of the cutter location points in the angular space improves the efficiency of 5 axis machining. Our preliminary results show a 20% improvement over the pure angular insertion scheme and about 50% improvement with the reference to spatially equi-distributed points. However, for fine cuts these advantages could be considerably reduced and the proposed scheme may even become unnecessary.

The second method of point insertion guarantees the optimality with regard to the kinematics error. The symbolic calculations generate an explicit representation of the error and its performed automatically and only once for a particular five-axis machine. The results can be converted into C-codes and used to optimize the tool path for any given surface. For particular cuts the method provides up to 89% decrease of the kinematics error with regard to iso-parametric and equi-arc length point method. On the other hand, the method provides less CL points for the same error tolerance, which is in particular important for high speed milling when an increase in number of the points leads to a substantial increase in the machining time. The advantage with regard to non-gradient techniques such as bisection exceeds 10%. For many industrial setups it amounts to hundreds of hours of machining.

REFERENCES

[1] Affouard, A; Duc, E; Lartigue, C; Langeron, J.-M.; Bourdet, P: Avoiding 5-axis singularities using tool path deformation, International Journal of Machine Tools and Manufacture, 44(4), 2004, 415-425. doi:10.1016/j.ijmachtools.2003.10.008

[2] Chiou, C-J; Lee, Y-S: A machining potential field approach to tool path generation for multi-axis sculpture surface machining, Computer Aided Design, 34(5), 2002, 357-371. doi:10.1016/S0010-4485(01)00102-6

[3] Jung, Y.H.; Lee, D.W.; Kim, J.S.; Mok, H.S.: NC post-processor for 5 axis milling machine of table-rotating/tilting type, Materials Processing Technology, 130-131, 2002, 641-646. doi:10.1016/S0924-0136(02)00725-2

[4] Kim, T; Sarma, S.E: Tool path generation along directions of maximum kinematics performance; a first cut at machine-optimum paths, Computer Aided Design, 34, 2002, 453-468. doi:10.1016/S0010-4485(01)00116-6

[5] Lo, C-C: Efficient cutter-path planning for five-axis surface machining with a flat-end cutter, Computer Aided Design, 31(9), 1999, 557-566. doi:10.1016/S0010-4485(99)00052-4

[6] Makhanov, S; Munlin, M: Optimal sequencing of rotation angles for five-axis machining, International Journal of Advanced Manufacturing Technology, 35, 2007, 41–54. doi:10.1007/s00170-006-0699-8

[7] Munlin, M; Makhanov, S; Bohez, E; Optimization of Rotations of a Five-Axis Milling Machine Near Stationary Points, Computer-Aided Design, 36, 2004, 1117-1128. doi:10.1016/S0010-4485(03)00164-7