Mathematical Methods for Optimization of Cutting Operations of 5-Axis Milling Machines

Stanislav Makhanov
makhanov@siit.tu.ac.th
School of Information and Computer Technology
Sirindhorn International Institute of Technology Thammasat University
Rangsit, Pathum Thani 12000
Thailand

Abstract

The paper presents mathematical methods for tool path planning for five-axis machining based on minimization of the kinematics error. The error is derived explicitly from the inverse kinematics equations corresponding to a particular five-axis machine. A method for an automatic symbolic calculation the kinematics error for arbitrary machine kinematics is implemented and verified. The numerical experiments demonstrate that the proposed procedure is superior with the reference to conventional engineering techniques.

1 Introduction

Milling machines are programmable mechanisms for cutting complex industrial parts. The machines are designed in such a way that the cutting device (the tool) is capable of approaching the desired surface at a given point with a required orientation. The machine consists of several moving parts designed to establish the required coordinates and orientations of the tool during the cutting process. The movements of the machine parts are guided by a controller which is fed with the so-called NC program or G-code comprising commands carrying three spatial coordinates of the tool-tip and a pair of rotation angles needed to rotate the machine parts to establish the orientation of the tool. The tool path $\Pi = \{\Pi_0, \Pi_1, \dots, \Pi_m\}$ is a sequence of positions $\Pi_p = \{M_p, I_p\}$, where $M_p = \{x_p, y_p, x_p\}$ are the Cartesian coordinates of the tip of the tool in the machine coordinate system (cutter contact points) and $I_p = \{I_{x,p}, I_{y,p}, I_{z,p}\}$ the orientation vector. The rotation angles $\mathfrak{R}_p = \{a_p, b_p\}$ are calculated from the components of I_p given the configuration of the machine. The configuration of the five-axis machine is characterized by (see Figure 1)

- Two rotation matrices A, B corresponding to the two rotary axes.
- Two translations associated with the position of the workpiece and the design of the five-axis machine T_{23}, T_{34}.

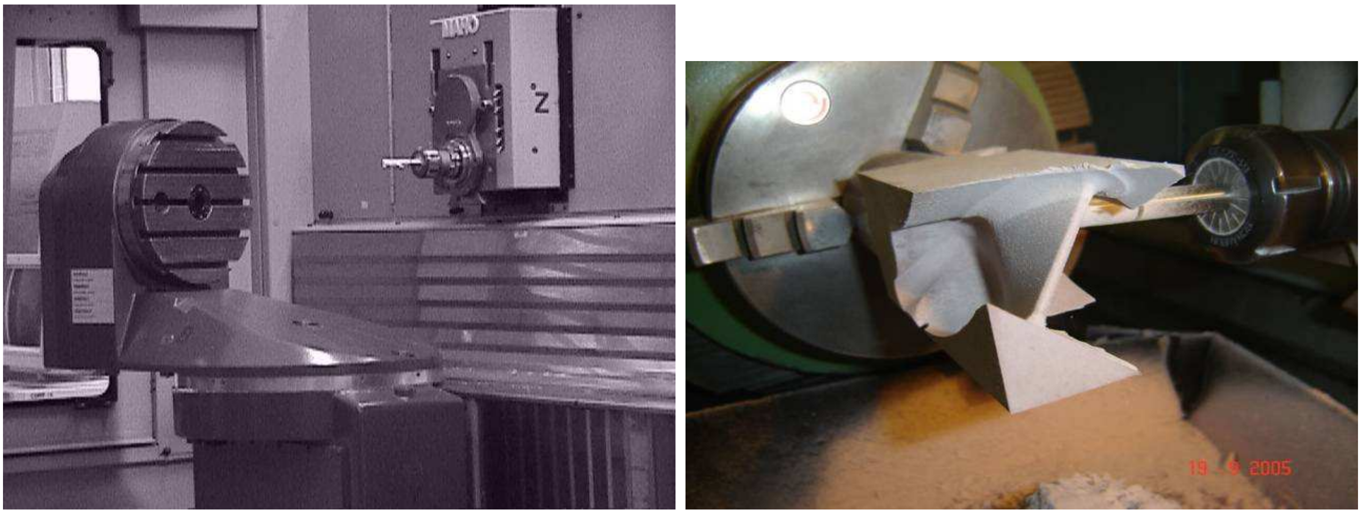

Figure 1: left: MAHO600E; right: MAHOO600E during cutting operations

- The length of the tool L. Since we consider the tool aligned with the Z_4-axis (Figure 1), L is treated as an additional translation T_4 (T_4 is either $(0, 0, L)$ or $(0, 0, -L)$ depending on the direction of the tool tip in the spindle coordinate system).

Furthermore, let $S(u, v)$ be the required surface. The tool path optimization problem is formulated as follows

$$\min_{\Pi} C, \tag{1}$$

where the criteria vector C includes one or several estimates of the difference between the required and the actual surface. The vector may include the length of the path, the negative of the machining strip (strip maximization), the machining time, etc (see, for instance, [2, 4, 6]). Besides, the optimization could be subjected to constraints the most important of which are the scallop height constraints, the local and the global accessibility constraints [3, 5, 7]. In any case, the optimization requires to reduce the scallop heights between the successive tracks and the kinematics error by distributing the cutter contact points along the tracks.

We consider a tool path represented by a space-filling curve such as the standard zigzag or an adaptive space-filling curve [1]. However, the proposed error minimization applies to an arbitrary set of (possibly disjointed) curves representing the tool path. The minimization procedure is similar to grid generation. However, as opposed to the majority of CFD problems, when only a numerical estimate of the error is possible, the kinematics of the five-axis machine allows to evaluate the error and its derivatives explicitly. Unfortunately, the resulting equations are long and inconvenient to deal with. For example, the first derivative of the error could occupy more than 100 text pages. Therefore, we propose a symbolic evaluation of the errors followed by generation of the corresponding C (MatLab) functions using Maple 11. The procedures for symbolic evaluation are general and can be used for any configuration of the five-axis machine given a correct set of the kinematics equations.

The functions generated by the Maple code are used for direct constraint minimization of the error, thus, generating an optimal grid of the cutter contact points.

2 Kinematics equations and kinematics error of the five-axis milling machine

Let $\mathfrak{K} \equiv \mathfrak{K}\{\text{parameters}\}[\text{arguments}] = \mathfrak{K}\{\mathfrak{R}\}[M]$ be a kinematics transformation from the machine coordinates to the workpiece coordinates. For simplicity we will denote the transformations by $\mathfrak{K}[M]$ (when possible) keeping in mind the dependence on $\mathfrak{R}$.

Let $\mathfrak{K}^{-1}[W]$ be the inverse transformation such that $\forall W, M, \mathfrak{R}$, $\mathfrak{K}^{-1}[\mathfrak{K}[M]] = M$ and $\mathfrak{K}[\mathfrak{K}^{-1}[W]] = W$. Let $\Pi_p \equiv (W_p, \mathfrak{R}_p)$, $\Pi_{p+1} \equiv (W_{p+1}, \mathfrak{R}_{p+1})$ be two successive coordinates of the tool path in $\mathbb{R}^5$. W_p and W_{p+1} denote two successive spatial positions of the tool path and $\mathfrak{R}_p$, $\mathfrak{R}_{p+1}$ the corresponding rotation angles. In order to calculate the tool trajectory between W_p and W_{p+1}, we first invoke the inverse kinematics to transform the part-surface coordinates into the machine coordinates $M_p \equiv (x_p, y_p, z_p)$ and $M_{p+1} \equiv (x_{p+1}, y_{p+1}, z_{p+1})$. Namely, $M_p \equiv \mathfrak{K}^{-1}\{\mathfrak{R}\}[W_p]$. Second, the rotation angles $\mathfrak{R} \equiv \mathfrak{R}(t) = (a(t), b(t))$ and the machine coordinates $M \equiv M(t) \equiv (x(t), y(t), z(t))$ are assumed to change linearly between the prescribed points, namely, $M(t) = tM_{p+1} + (1-t)M_p$, $\mathfrak{R}(t) = t\mathfrak{R}_{p+1} + (1-t)\mathfrak{R}_p$ where t is the fictitious time coordinate ($0 \le t \le 1$). Finally, transforming M back to W for every t yields a trajectory of the tool tip in the workpiece coordinates given by

$$W_{p,p+1}(t) = \mathfrak{K}\{\mathfrak{R}(t)\}[M(t)] = \mathfrak{K}\{\underline{t\mathfrak{R}_{p+1} + (1-t)\mathfrak{R}_p}\}[\underline{tM_{p+1} + (1-t)M_p}]. \tag{2}$$

In order to represent the tool path in terms of the workpiece coordinates, we eliminate M_p, M_{p+1} by using the inverse transformation $M_p = \mathfrak{K}^{-1}\{\mathfrak{R}\}[W_p]$. Substituting, M_p, M_{p+1} yields

$$W_{p,p+1}(t) = \mathfrak{K}\{\underline{t\mathfrak{R}_{p+1} + (1-t)\mathfrak{R}_p}\}[\underline{t\mathfrak{K}^{-1}\{\mathfrak{R}_{p+1}\}[W_{p+1}] + (1-t)\mathfrak{K}^{-1}\{\mathfrak{R}_p\}[W_p]}]. \tag{3}$$

In order to classify the machine kinematics, we, introduce the following coordinate systems: the workpiece coordinate system O_1, a coordinate system of the first rotary part O_2, a coordinate system of the second rotary part O_3 and a coordinate system of the spindle O_4. We shall call the first rotary axis the A-axis and the second rotary axis the B-axis. Next, consider the three most important machine kinematics categorized by the positions of the rotational joints in the kinematics chain.

The 2-0 machine. Two rotary axes on the table (see Figure 2). In this case

$$\begin{aligned}
M &\equiv \mathfrak{K}^{-1} \equiv \mathfrak{K}^{-1}\{\mathfrak{R}\}[W] = GB[b]\left(A[a]\left(W + T_{12}\right) + T_{23}\right) + T_{34} - T_4,\\
a &= \begin{cases} \tan^{-1}\left(\dfrac{I_y}{I_x}\right) & \text{if } I_x > 0 \text{ and } I_y > 0,\\ \tan^{-1}\left(\dfrac{I_y}{I_x}\right) + \pi & \text{if } I_x < 0,\\ \tan^{-1}\left(\dfrac{I_y}{I_x}\right) + 2\pi & \text{otherwise,} \end{cases}\\
b &= -\sin^{-1} I_z,
\end{aligned} \tag{4}$$

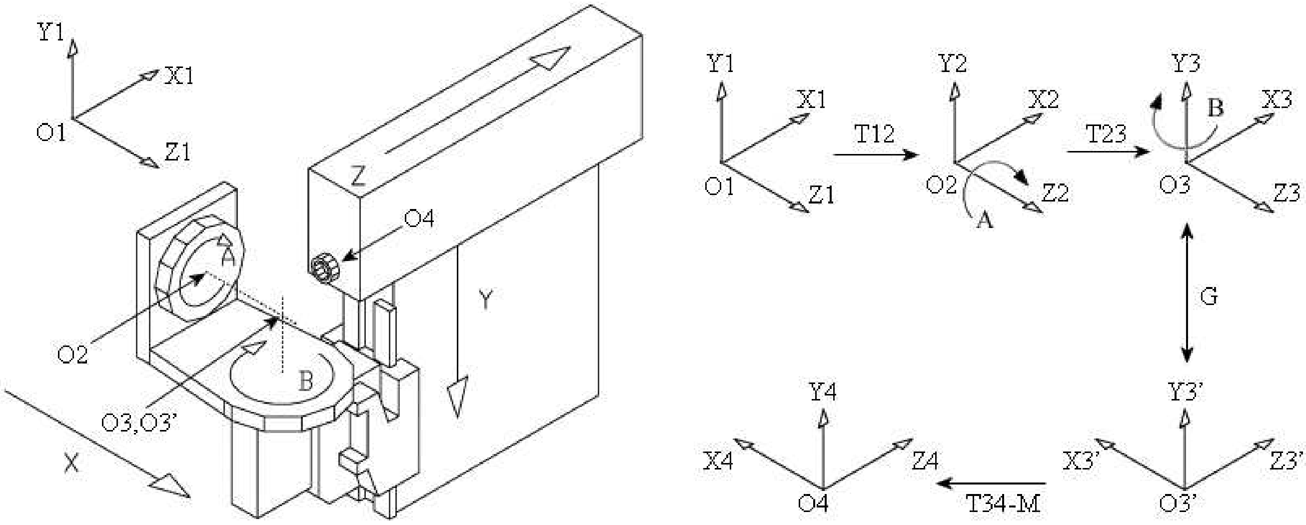

Figure 2: 2-0 machine and the reference coordinate systems

where

$$
\begin{aligned}
G &= \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \\
T_4 &= (0, 0, -L).
\end{aligned}
$$

The 1-1 machine. One rotary axis on the table and one on the tool (see Figure 3). In this case

$$
\begin{aligned}
M &= GA[a]\left(W + T_{12}\right) + T_{23} + B^{-1}[b]\left(T_{34} - T_4\right), \\
a &= \begin{cases} -\tan^{-1}\left(\dfrac{I_y}{I_x}\right) & \text{if } I_x > 0 \text{ and } I_y \leq 0, \\ -\tan^{-1}\left(\dfrac{I_y}{I_x}\right) + \pi & \text{if } I_x < 0, \\ -\tan^{-1}\left(\dfrac{I_y}{I_x}\right) + 2\pi & \text{otherwise,} \end{cases} \\
b &= \cos^{-1} I_z,
\end{aligned} \tag{5}
$$

where

$$
\begin{aligned}
G &= \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \\
T_4 &= (0, 0, L).
\end{aligned}
$$

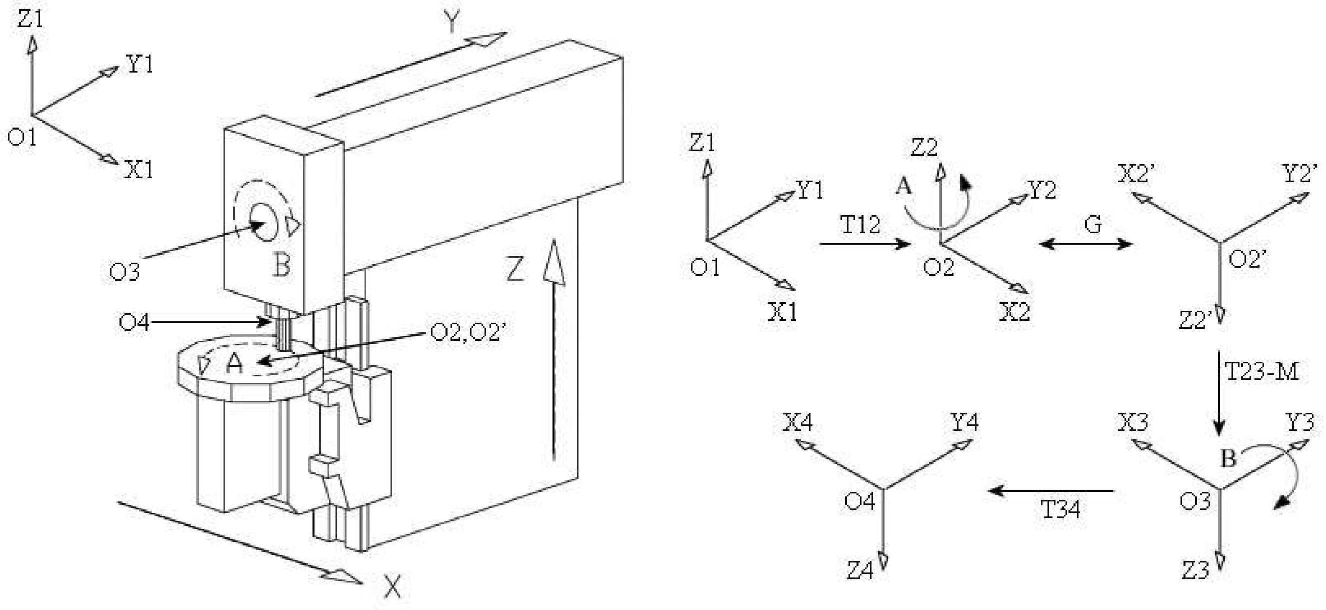

Figure 3: 1-1 machine and the reference coordinate systems

The 0-2 machine. Two rotary axes on the tool (see Figure 4). In this case

$$
\begin{aligned}
M &= GW + T_{12} + A^{-1}[a]\left(T_{23} + B^{-1}[b]\left(T_{34} - T_4\right)\right), \\
a &= \begin{cases} -\tan^{-1}\left(\dfrac{I_y}{I_z}\right) & \text{if } I_y \le 0 \text{ and } I_z > 0, \\ -\tan^{-1}\left(\dfrac{I_y}{I_z}\right) + \pi & \text{if } I_z < 0, \\ -\tan^{-1}\left(\dfrac{I_y}{I_z}\right) + 2\pi & \text{otherwise,} \end{cases} \\
b &= -\cos^{-1} I_x,
\end{aligned}
\tag{6}
$$

where

$$
\begin{aligned}
G &= \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{bmatrix}, \\
T_4 &= (0, 0, L).
\end{aligned}
$$

Furthermore, we define the kinematics error as follows. Let $W^D_{p,p+1}(t) \equiv (x^D_{p,p+1}(t), y^D_{p,p+1}(t), z^D_{p,p+1}(t)) \in S(u,v)$ be a curve between two tool positions W_p and W_{p+1} extracted from the machined surface $S(u,v)$, where t is the parametric coordinate along the curve. The curve is extracted in such a way that it represents the desired tool trajectory. The kinematics error is defined as a norm of the difference between $W^D_{p,p+1}(t)$ and the actual trajectory $W_{p,p+1}(t) \equiv$

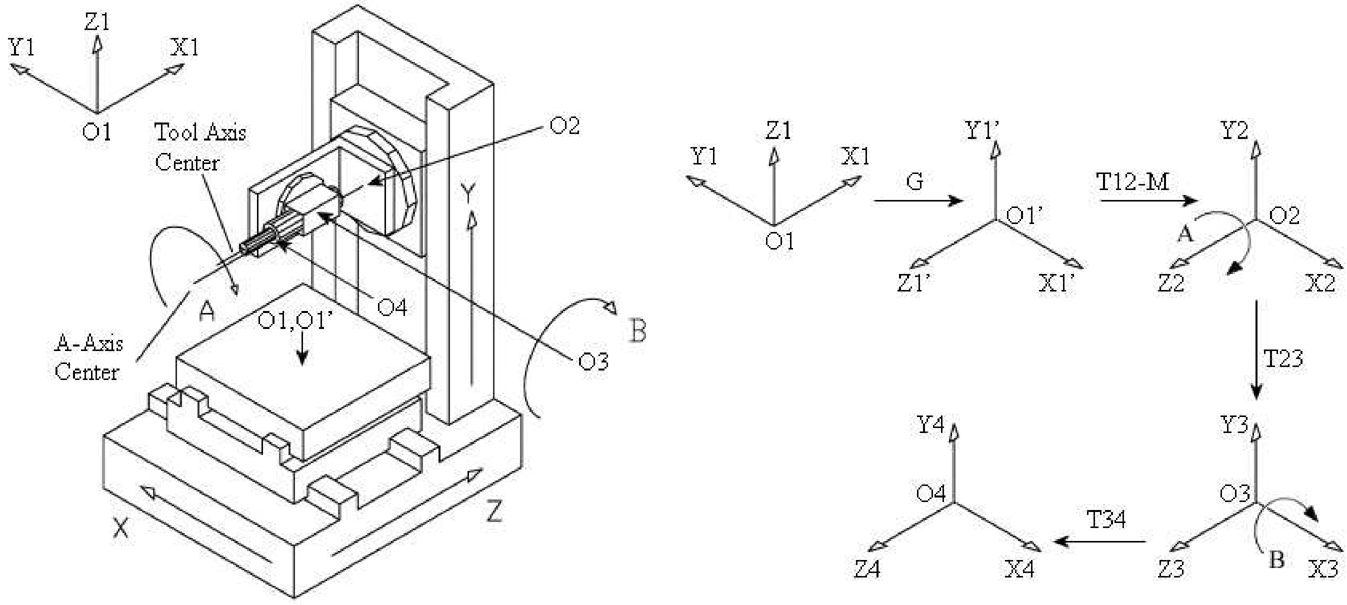

Figure 4: 0-2 machine and the reference coordinate systems

$(x_{p,p+1}(t), y_{p,p+1}(t), z_{p,p+1}(t))$, namely,

$$\begin{aligned} \epsilon &= \sum_p \int_0^1 \left| W^D_{p,p+1}(t) - W_{p,p+1}(t) \right|^2 dt, \\ &= \sum_p \int_0^1 \Big[\left(x^D_{p,p+1}(t) - x_{p,p+1}(t) \right)^2 + \left(y^D_{p,p+1}(t) - y_{p,p+1}(t) \right)^2 \\ &\qquad + \left(z^D_{p,p+1}(t) - z_{p,p+1}(t) \right)^2 \Big] \, dt, \end{aligned} \tag{7}$$

The actual trajectory $W_{p,p+1}(t)$ is generated using the inverse kinematics equations derived for the specific machine configuration.

3 Optimization problem

Consider problem (1) formulated with regard to kinematics error (7) as follows

$$\min_{\Pi} \ \epsilon, \tag{8}$$

subject to $\Pi \in \Pi'$, where Π' is a continuous curve which is supposed to pass through a fixed set of basic points such as turns. The optimization procedure inserts a certain number of the cutter location points between each pair of the basic points in such a way that the kinematics error is minimized.

Let (u_p, v_p) and (u_{p+1}, v_{p+1}) be the parametric coordinates corresponding to basic points (x_p, y_p, z_p) and $(x_{p+1}, y_{p+1}, z_{p+1})$. The desired trajectory is given by $W^D_{p,p+1}(t) = S((1-t)u_p + tu_{p+1}, (1-t)v_p + tv_{p+1})$.

Let us insert $m+1$ additional points $\{(u_{p,p+1}(s_k), v_{p,p+1}(s_k))\}$ ($s_0 = 0, s_m = 1$) along the line $(u_{p,p+1}(t), v_{p,p+1}(t)) = ((1-t)u_p + tu_{p+1}, (1-t)v_p + tv_{p+1})$ in the parametric space. We have

$$\epsilon(s_0, s_1, \ldots, s_m) = \sum_k \int_0^1 \left| W^D_{s_k,s_{k+1}}(t) - W_{s_k,s_{k+1}}(t) \right|^2 dt, \tag{9}$$

where $W^D_{s_k,s_{k+1}}(t)$ is a desired trajectory between two inserted (unknown) points $(u_{p,p+1}(s_k), v_{p,p+1}(s_k))$ and $(u_{p,p+1}(s_{k+1}), v_{p,p+1}(s_{k+1}))$. Note that $W^D_{s_k,s_{k+1}}(t)$ is easily evaluated from $W^D_{p,p+1}(t)$. $W_{s_k,s_{k+1}}(t)$ is obtained from (3) which invokes inverse kinematics (4)-(6). In order to obtain a closed form of $W_{s_k,s_{k+1}}(t)$ we developed a Maple code designed for symbolic calculation of the kinematics equations with the free boundary s_k, s_{k+1}. A closed form of $W_{s_k,s_{k+1}}(t)$ in terms of s_k and s_{k+1} is long and inconvenient to deal with. Therefore, we represent $W_{s_k,s_{k+1}}(t)$ in terms of auxiliary variables M_{s_k} and $M_{s_{k+1}}$ which are considered as functions of s_k and s_{k+1} respectively.

Substituting s_k and s_{k+1} into (2) yields

$$W_{s_k,s_{k+1}}(t) = \mathfrak{K}\{\underline{t\mathfrak{R}_{s_{k+1}} + (1-t)\mathfrak{R}_{s_k}}\}[\underline{tM_{s_{k+1}} + (1-t)M_{s_k}}]. \tag{10}$$

The mean squared error (MSE) is an approximation of (9) given by

$$\bar{\epsilon}(s_0, s_1, \ldots, s_m) = \frac{1}{N_{kt}} \sum_k \sum_t \left| W^D_{s_k,s_{k+1}}(t) - W_{s_k,s_{k+1}}(t) \right|^2. \tag{11}$$

The required derivative of (11) with respect to s_k is represented by

$$\begin{aligned} \frac{\partial \bar{\epsilon}}{\partial s_k} = \frac{2}{N_{kt}} \sum_t & \left[\left(W^D_{s_{k-1},s_k}(t) - W_{s_{k-1},s_k}(t) \right) \frac{\partial \left(W^D_{s_{k-1},s_k}(t) - W_{s_{k-1},s_k}(t) \right)^T}{\partial s_k} \right. \\ & \left. + \left(W^D_{s_k,s_{k+1}}(t) - W_{s_k,s_{k+1}}(t) \right) \frac{\partial \left(W^D_{s_k,s_{k+1}}(t) - W_{s_k,s_{k+1}}(t) \right)^T}{\partial s_k} \right]. \end{aligned} \tag{12}$$

Finally, the partial derivative of (10) in (12) is obtained using the chain rule, as follows:

$$\begin{aligned} \frac{\partial W_{s_k,s_{k+1}}(t)}{\partial s_k} = & \frac{\partial W_{s_k,s_{k+1}}(t)}{\partial M_{s_k,x}} \frac{\partial M_{s_k,x}}{\partial s_k} + \frac{\partial W_{s_k,s_{k+1}}(t)}{\partial M_{s_k,y}} \frac{\partial M_{s_k,y}}{\partial s_k} \\ & + \frac{\partial W_{s_k,s_{k+1}}(t)}{\partial M_{s_k,z}} \frac{\partial M_{s_k,z}}{\partial s_k}. \end{aligned} \tag{13}$$

The optimization is carried out using built-in procedures of MATLAB. The MSE is designed is such a way that it has five arguments $u_p, v_p, u_{p+1}, v_{p+1}$, and an array of optimization variables $\{s_k\}$. Each call of the MSE returns the error and its gradient.

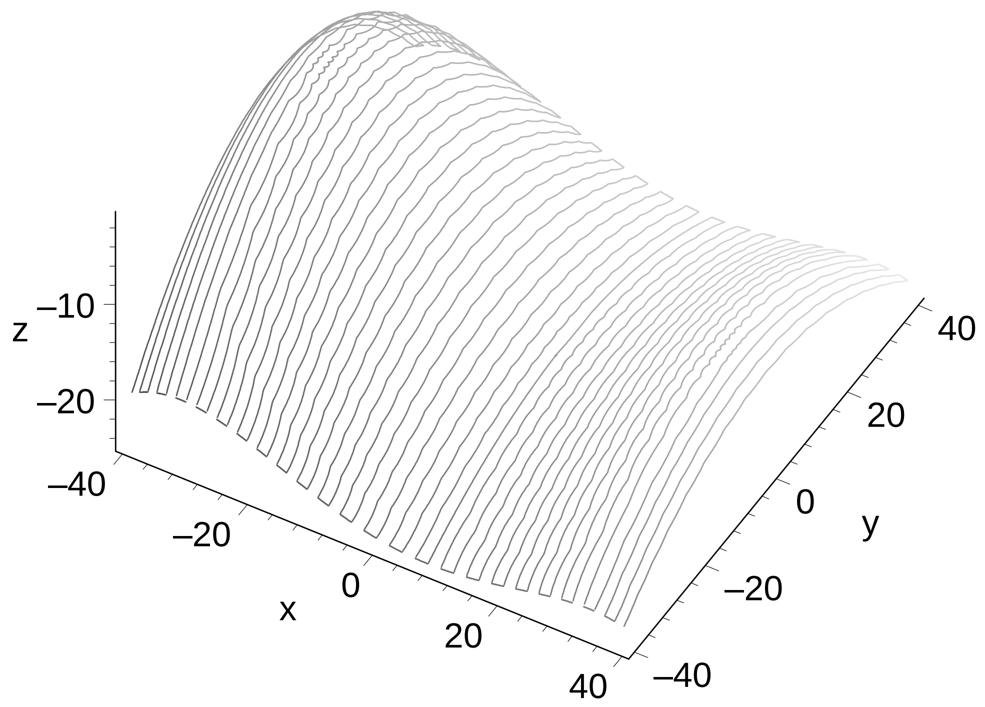

Figure 5: Tool trajectories. The optimal insertion method

3.1 Example

Consider a zigzag iso-parametric tool path on the surface described by the following equations

$$\begin{aligned} x &= -40.0 + 80.0u, \\ y &= -40.0 + 80.0v, \\ z &= -400[0.355 + 2.840u - 14.8((0.1 + 0.8u))^2 + 21.15((0.1 + 0.8u))^3 \\ &\quad - 9.9((0.1 + 0.8u))^4](0.1 + 0.8v)(-0.9 + 0.8v) - 28. \end{aligned} \tag{14}$$

The basic points are fixed at both ends of each track. Figure 5 shows tool trajectories after inserting additional points along each track using the proposed minimization technique. Figure 6 shows the tool trajectories after inserting equally spaced (in the parametric domain) points. Table 1 displays the MSE as well as the maximum error for each insertion scheme versus the total number of additional points. The optimal grid techniques requires 1,138 points to achieve a maximum error below 0.1 mm (the number of points inserted in each track varies). The optimal insertion is compared with a traditional technique when a point is inserted in the middle between two points if the error between them exceeds the specified tolerance. The traditional method requires 1,508 additional points which is 33% more than the number of points required by the proposed optimal insertion .

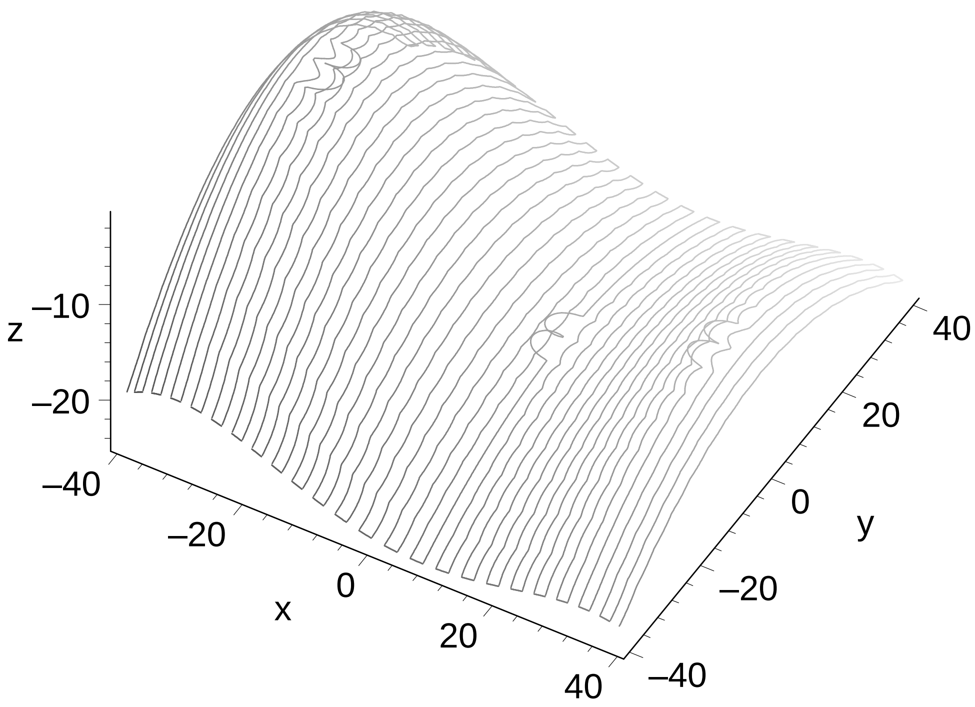

Figure 6: Tool trajectories. The equi-spaced points

Table 1: Performance of the optimal point insertion technique versus equi-spaced insertion

Total number of points inserted (points)	Optimal insertion MSE (mm^2)	Optimal insertion Max Error (mm)	Equi-spaced insertion MSE (mm^2)	Equi-spaced insertion Max Error (mm)
465	7.80	0.51	8.93	4.45
1,138	2.96	0.10	3.28	2.76

4 Conclusion

Mathematical foundations for generation of optimal cutter location points for five-axis machining have been presented and verified. The techniques are based on a direct evaluation of the kinematics error, the use of the symbolic engine of Maple-11 and build-in minimization procedures of MATLAB. The numerical experiments demonstrate that the proposed procedure is superior with the reference to conventional techniques.

5 Acknowledgement

We acknowledge a sponsorship of the Thailand Research Fund, grant number BRG50800012.

References

[1] Anotaipaiboon, W. and Makhanov, S. S. 2005. Tool path generation for five-axis NC machining using adaptive space-filling curves. *International Journal of Production Research*, 43(8):1643–1665.

[2] Lauwers, B., Dejonghe, P., and Kruth, J. P. 2003. Optimal and collision free tool posture in five-axis machining through the tight integration of tool path generation and machine simulation. *Computer-Aided Design*, 35(5):421–432.

[3] Lee, Y.-S. 1997. Admissible tool orientation control of gouging avoidance for 5-axis complex surface machining. *Computer-Aided Design*, 29(7):507–521.

[4] Lo, C. C. 1999a. Efficient cutter-path planning for five-axis surface machining with a flat-end cutter. *Computer-Aided Design*, 31(9):557–566.

[5] Makhanov, S. S., Anotaipaiboon, W. 2007. Advanced Numerical Methods to Optimize Cutting Operations of Five Axis Milling Machines *Springer*, 206p.

[6] Sarma, R. 2000. An assessment of geometric methods in trajectory synthesis for shape-creating manufacturing operations. *Journal of Manufacturing Systems*, 19(1):59–72.

[7] Yoon, J.-H., Pottmann, H., and Lee, Y.-S. 2003. Locally optimal cutting positions for 5-axis sculptured surface machining. *Computer-Aided Design*, 35(1):69–81.