

# Generation of Collision-Free Cutter Location Data in Five-Axis Milling Using the Potential Energy Method

Inhaeng Cho, Kunwoo Lee and Jongwon Kim

Department of Mechanical Design and Production Engineering, Seoul National University, Seoul, Korea

*In five-axis milling, optimal cutter location data (CL-data) should be generated to have advantages over three-axis milling in terms of accuracy and efficiency. This paper presents an algorithm for generating collision-free CL-data for five-axis milling using the potential energy method. By virtually charging the cutter and part surfaces with static electricity, global collision as well as local interference is eliminated. Moreover, machining efficiency is simultaneously improved by minimising the curvature difference between the part surface and tool swept surface at a cutter contact point (CC-point).*

**Keywords:** CL-data; Collision; Five-axis milling; Interference; Potential energy

## 1. Introduction

Five-axis milling was developed in order to machine complicated sculptured surfaces such as turbine blades, impellers, and marine propellers, which cannot be machined by three-axis milling. Five-axis milling allows the whole region of the workpiece to be machined in a small number of set-ups and provides improved surface quality and high productivity; which is facilitated by using a flat-end cutter or filleted-end cutter with variable cutter orientation [1].

As shown in Fig. 1, the numerically controlled (NC) machining cycle can be classified into three stages. First, the part surfaces are designed using a CAD system. The designed surfaces are passed to the NC code generation system and

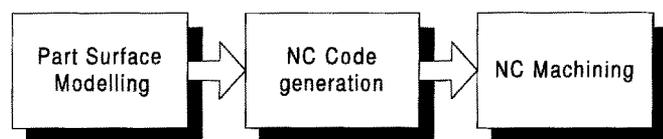


Fig. 1. NC machining process.

Correspondence and offprint requests to: Dr K. Lee, Department of Mechanical Design and Production Engineering, Seoul National University, San 56-1, Shinlim-Dong, Kwanak-Gu, 151-742 Seoul, Korea.

then machining is performed on a CNC machining centre. It is more difficult to develop an NC code generation system in five-axis milling than in three-axis milling, because the cutter orientations also have to be determined.

The basic procedure for five-axis NC code generation is as follows:

*Determination of set-up orientation.* The number of set-ups significantly affects machining productivity and accuracy. Tang et al. [2] formulated a method for minimising the number of set-ups in four- and five-axis NC machining. Elber and Cohen [3] presented a symbolic based method to compute the Gaussian map of a freeform surface that can be used to minimise the number of set-ups.

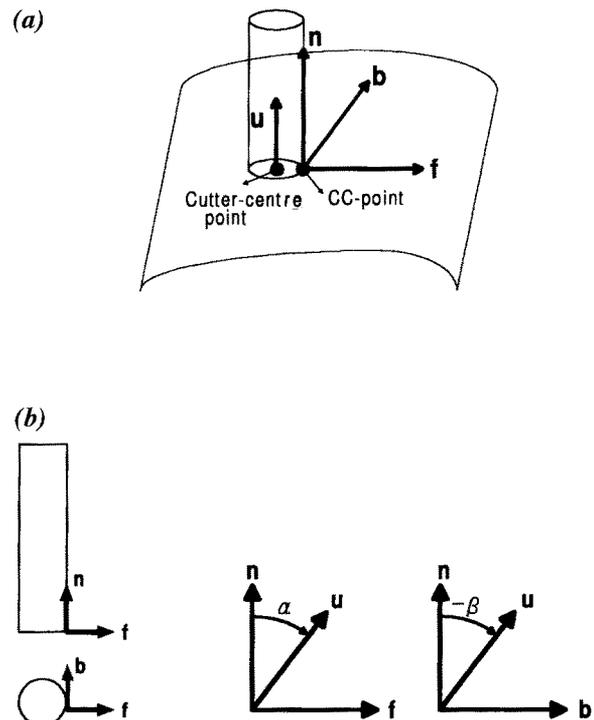


Fig. 2. (a) Local coordinate system ( $f, b, n$ ). (b) cutter in the local coordinate system and the definition of  $\alpha$  and  $\beta$ .

*Selection of cutting strategy.* Determination of the cutter path planning, cutter sizes and shapes is required before machining. Lee and Chang [4] suggested a systematic methodology to generate operation plans for five-axis machining.

*Generation of cutter path and CL-data.* With the determined set-up orientation and cutting strategy, NC codes are generated. In five-axis milling, NC code consists of cutter contact points and CL-data which are evaluated from the determined cutter orientation.

*Cutting simulation and correction.* Finally, the generated NC code is verified and corrections are performed if errors occur in it.

In this work, an algorithm for determining the cutter orientation at each CC-point is proposed. CC-points are assumed to be given. The cutter orientation should guarantee no collision between the cutter and part surfaces with good machining efficiency, i.e. the machined surface should have no overcut and minimum undercut in a short machining time. In order to generate an NC code meeting those criteria, an efficiency algorithm for avoiding collisions, as well as for calculating the amount of undercut, is required.

Both the avoidance of collisions and the machining efficiency are simultaneously considered in this work. In order to generate collision-free cutter orientation, an analytic equation is derived from the potential energy method that indicates whether collision occurs. This method represents Cartesian space by an artificial energy field. When the cutter and part surfaces are virtually charged with static electricity, a potential energy field is formed. As the cutter approaches the part surface, the stored potential energy and the repulsive force increase. In the case of a collision, the repulsive force makes the cutter move away from the part surface. At the same time, similarity of shapes between the tool swept surface and the part surface is required to be maximised to increase the machining efficiency. The curvatures characterise the shapes of the two surfaces at a CC-point. In this work, using an approximated curvature expression which was derived by Szende [5], the curvature difference between a tool swept surface and the part surface at a CC-point is minimised, to maximise the material removal rate. In the course of minimisation, the collision-avoidance equation derived from the potential energy method provides a constraint while adjusting the cutter orientation. Finally, the determined cutter orientations are converted into CL-data to generate the NC code.

## 2. Related Work

So far, many collision detection and correction algorithms have been proposed. Takeuchi and Idemura [6] approximated the cutter shape as a set of check-points. The collision check is replaced by a check on whether or not the check-points are inside the workpiece which is defined as a solid model. The collision-avoidance direction is determined by check-points which are inside the solid model. Saito and Takahashi [7] used the G-buffer method to detect collision for three-axis milling. The G-buffer contains the geometric property for all pixels such as depth, surface normal vector, etc. Li and Jerard [8]

generated cutter paths and performed collision checking using a triangular approximation of the part surface. Chang and Goodman [9] also used a discretised surface approximation to verify the generated CL-data. In earlier work on collision-avoidance, the cusp heights are not calculated accurately. That is, they approximated the shape of cusps to circular arcs or ignored the effect of cutter orientation on cusp heights.

Liu [10] generated a collision-free cutter path for side milling. Yu [11] developed a tool correction method that can generate CL-data for many milling cutter types.

In five-axis milling, the cutter orientation and stepover distance controls the cusp height between cutter paths. Many authors approximated the cusp height to improve the machining accuracy. Choi et al. [12] derived an analytic expression for approximating cusp height to formulate the CL-data optimisation problem as a 2D constrained minimisation problem. Szende [5] derived an analytical expression for the curvature of a tool-swept surface at a CC-point to determine the cutter orientation and stepover distance that controls the cusp height. Mullins et al. [13] proposed a curvature matching approach to provide a closer match to the designed surface. Kruth and Klewais [14] also used the local surface curvature at a CC-point to determine optimal cutter orientation. Though they determined the cutter orientation considering the local collision and cusp heights, most algorithms did not consider global collision which occurs between the tool system and the work-piece.

This work uses the artificial potential function method to avoid collisions. In the mobile robot planning field, this method is used in obstacle avoidance or path planning [15–18]. Bae [19] charged the parts of an assembly with artificial static electricity in order to generate the part's assembly trajectory.

## 3. Decision Variables

CL-data are calculated from the cutter orientations. To determine the cutter orientation, a surface local coordinate system is introduced, as shown in Fig. 2(a), where  $\mathbf{n}$  is a surface normal vector,  $\mathbf{f}$  is a surface tangent vector along the cutting direction, and  $\mathbf{b}$  is a cross-product of  $\mathbf{n}$  and  $\mathbf{f}$ . Figure 2(b) shows a cutter in the local coordinate system and the definition of dive angle  $\alpha$  and pivoting angle  $\beta$ . The cutter rotates about the  $\mathbf{b}$ -axis by the dive angle  $\alpha$ , and about the  $\mathbf{f}$ -axis by the pivoting angle  $\beta$ . In practice,  $\alpha$  is positive or zero, and  $\beta$  may be either positive or negative. When both  $\alpha$  and  $\beta$  equal zero, the cutter orientation aligns with the normal vector of the part surface.

In determining the cutter orientation at each CC-point,  $\alpha$  and  $\beta$  are the design variables. From the determined  $\alpha$  and  $\beta$ , CL-data at each CC-point can be evaluated with respect to the surface local coordinate system.

## 4. Potential Energy

When a surface charged with static electricity is placed in Euclidean space, it generates a potential field. The potential field  $\phi$  at a position  $\vec{P}(x,y,z)$  is,

$$\phi(\bar{P}) = \int_{S^*} \frac{\rho}{k_i |\bar{R}|} dS \quad (1)$$

where,  $\bar{R}$  is the position vector from the charged surface  $dS$  to the position  $\bar{P}$ ,  $\rho$  is the charge density, and  $k$  is the proportional constant. When a particle  $\pi$  charged with  $q$  is imposed in this field at  $\bar{Q}$ , the potential energy stored in the particle is the same as the work required to move it from an infinite position to the current position as shown below,

$$U(\pi) = \int_{S^*} \frac{q\rho}{k_i |\bar{R}|} dS \quad (2)$$

where,  $\bar{R}$  is the position vector from the location of  $dS$  to  $\bar{Q}$ . Hence, when there are a set of charged surfaces  $\{S_1, S_2, \dots, S_n\}$ , the energy stored in the charged surface  $S^*$  can be defined by the superposition theorem as shown below,

$$U(S^*) = \sum_{i=1}^n \int_{S_i} \frac{\rho_i}{k_i} \int_{S^*} \frac{\rho^*}{|\bar{R}|} dS^* dS_i \quad (3)$$

where,  $\rho_i$  and  $k_i$  are the charge density, and the proportional constant of the  $i$ th surface, respectively, and  $\rho^*$  is the charge density on  $S^*$ . Surfaces charged with the same sign, repulse each other by a force governed by Coulomb's law. This force can be evaluated by the negative gradients of equation (3) as shown below,

$$\begin{aligned} \bar{F}(S^*) &= -\nabla U(S^*) \\ &= \sum_{i=1}^n \int_{S_i} \frac{\rho_i}{k_i} \int_{S^*} \frac{\rho^* \bar{R}}{|\bar{R}|^3} dS^* dS_i \end{aligned} \quad (4)$$

## 5. Collision Avoidance

When the cutter and part surfaces are virtually charged with static electricity, the potential energy shown in equation (3) will be stored. That is,  $S_i$  can be regarded as the part surface, and  $S^*$  as the cutter surface. Since the cutter moves continuously on the part surface, the stored potential energy also changes accordingly. When the cutter approaches the part surface, the stored potential energy will increase. In the case of collision, this energy increases enormously. Meanwhile, a relatively small amount of energy is stored for collision-free cases. This statement can be expressed as shown below,

$$U = \sum_{i=1}^m \int_{S_i} \sum_{j=1}^n \int_{S_j} \frac{\rho_i \rho_j}{k_i} \frac{1}{|\bar{R}|} dS_i dS_j < K \quad (5)$$

where,  $K$  is a collision-free threshold. In general,  $K$  cannot be small because a simple contact of the cutter and the part surface at each CC-point would result in a large amount of potential energy even when collision does not occur. This problem can be solved by excluding the CC-point in calculating the potential energy  $U$  in equation (5).

Since the geometric shape of the circumference is different at every CC-point, and the geometric relation between the cutter and the part surface always changes, the collision-free threshold  $K$  would vary at all times. Hence, threshold  $K$  should be set at each CC-point. In this research, the energy function

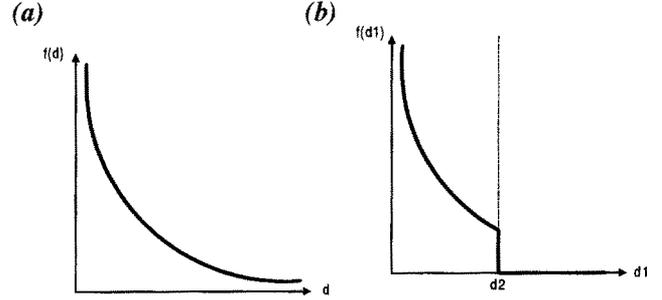


Fig. 3. (a) Original energy function. (b) Modified energy function.

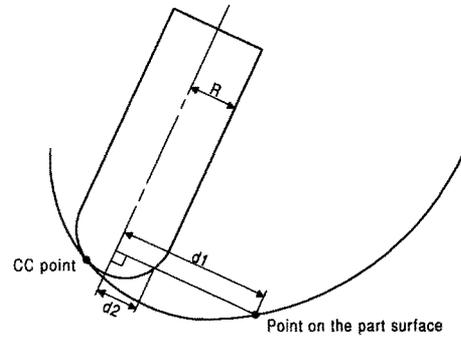


Fig. 4. Definition of  $d_1$  and  $d_2$ .

is modified to allow a constant threshold. Figure 3(a) shows the original energy function, where  $d$  is the distance between two surfaces, i.e. cutter surface and part surface. The energy generated by the charged surfaces is in inverse proportion to  $d$ . Figure 3(b) shows the modified energy function, where  $d_1$  and  $d_2$  are defined in Fig. 4. Note that  $d_1$  is the distance between the cutter axis and the sample point on the part surface that is used to calculate the numeric integration in equation (6). The distance between the point on the cutter axis, used in calculating  $d_1$ , and the cutter surface, is  $d_2$ .

In this energy function, the value of the energy function is zero when  $d_1$  is larger than  $d_2$ . Hence, if there is no collision, the stored energy would be zero. By calculating the stored energy  $U$  with this modified energy function, the collision-free condition can be modified as in equation (6), rather than as an inequality with threshold  $K$  as in equation (5). Equation (6) is used as a constraint to determine the collision-free cutter orientation later at each CC-point.

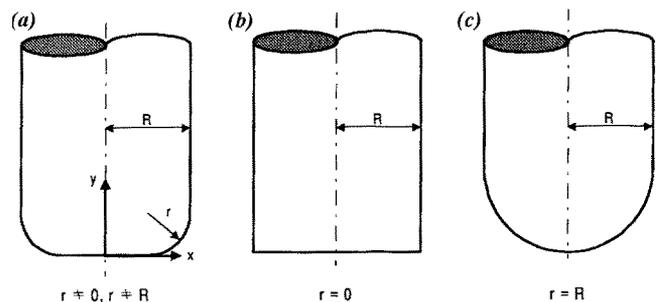


Fig. 5. Different shapes of cutter. (a) Filleted-end. (b) Flat-end. (c) Ball-end.

$$U = \sum_{i=1}^m \int_{S_i} \sum_{j=1}^n \int_{S_j} \frac{r_j f_j}{k_i} f(d_i) dS_i dS_j = 0$$

$$\text{where, } f(d_i) = \begin{cases} \frac{1}{d_1} & (d_1 \leq d_2) \\ 0 & (d_1 > d_2) \end{cases}$$

$$\text{where, } d_2 = \begin{cases} R - r + \sqrt{(2ry - y^2)} & (0 < y < r) \\ R & (y \geq r) \end{cases} \quad (6)$$

$d_1$ ,  $d_2$ ,  $R$ ,  $r$ , and  $y$  are defined in Figs 4 and 5. This expression can be adapted to a flat-end cutter by setting  $r=0$  and to the ball-end cutter by setting  $r=R$ .

In the course of finding the optimal angles, the tried angles may cause collisions. In this case, the feasible angle region, in which there is no collision, can be found quickly by the repulsive force in equation (4). After reaching the feasible angle region, optimisation that minimises the curvature difference is performed. Of course, if there are several feasible regions, local minima may occur. However, separated feasible regions do not exist, in general, when in determining the cutter orientation.

### 6. Minimisation of Curvature Difference

In order to increase the material removal rate and further reduce the polishing cost, achieving similarity of shape between the machined surface and the modelled part surface is most desirable. The shape of a surface at a point can be characterised by its curvature. In this work, the curvature difference between the tool swept surface and the part surface at each CC-point is minimised.

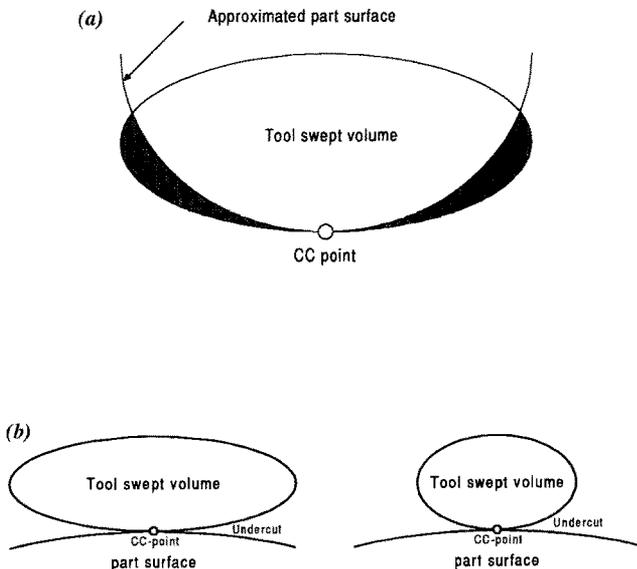


Fig. 6. (a) Overcut occurs when  $K$  is larger than  $\kappa$ . (b) Small curvature difference makes small undercut for convex part surface.

The curvature of the part surface at a CC-point can be evaluated by the Euler equation when the principal surface curvatures are known at the given point. Because the part surface is represented by NURBS in this work, the curvature can be evaluated [20]. The curvature of a tool swept surface at a CC-point should be approximated, for it is difficult to define it by an analytic surface equation. Szende [5] derived the curvature of a swept curve at a CC-point for a filleted-end cutter as shown below,

$$\kappa = \frac{(1 - (\cos\alpha\cos\beta)^2)^{3/2}}{\text{sign}(\sin\alpha)R\sin^2\alpha - r(1 - (\cos\alpha\cos\beta)^2)^{3/2}} \quad (7)$$

where,  $\alpha$  is the dive angle,  $\beta$  is the pivoting angle,  $R$  is the cutter radius, and  $r$  is the fillet radius. A swept curve is the silhouette of the bottom of a filleted-end cutter projected onto the  $(\mathbf{b}, \mathbf{n})$ -plane as shown in Fig. 2. This equation can also be applied to a flat-end cutter by setting  $r=0$  and to a ball-end cutter by setting  $r=R$  as shown in Fig. 5. With the evaluated curvatures, an optimisation process is performed to minimise the curvature difference  $\kappa - K$ . Here,  $\kappa$  is the curvature of the profile of tool swept surface and  $K$  is that of the part surface at the CC-point. Design variables of the objective function are  $\alpha$  and  $\beta$  as shown in Fig. 2. Constraints given in equation (6), which can be expressed in terms of  $\alpha$  and  $\beta$  are applied to prevent both global and local collision. In the course of optimisation, another constraint given in equation (8) is used for detecting the local interference. That is, the curvature of the part surface should be smaller than that of the tool swept

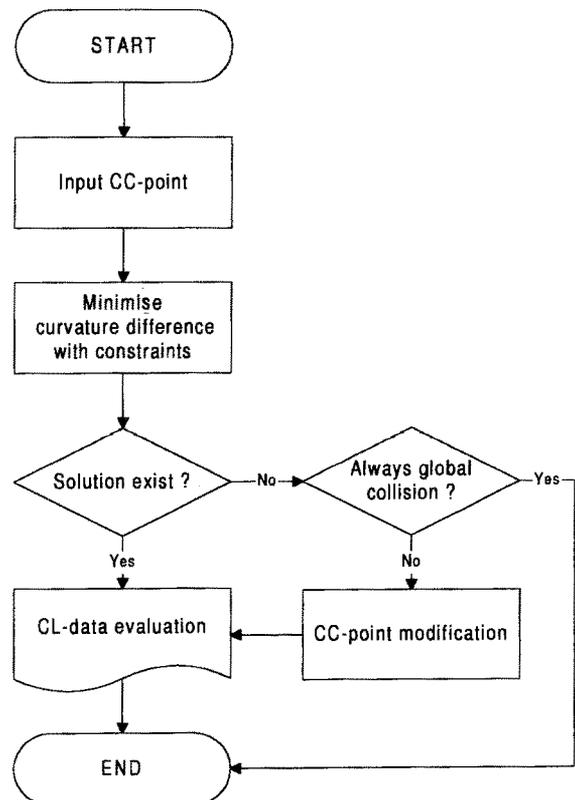


Fig. 7. Procedure of determining cutter orientation.

volume. Otherwise, interference may occur, as shown in Fig. 6(a).

$$\kappa - K \geq 0 \tag{8}$$

### 7. Algorithm

Figure 7 shows the procedure for determining the cutter orientation. For a given CC-point, the curvature difference is minimised with two constraints as mentioned above. For the convex part surface, minimisation is also performed because undercut is reduced with the small curvature difference as shown in Fig. 6(b). The optimisation package ADS [21] is used to solve the constrained 2D optimisation problem. The determined  $\alpha$  and  $\beta$  are converted into the corresponding CL-data.

However, there are cases where the collision cannot be avoided by changing angles only. In this case, collision happens for all the allowable cutter orientations and CC-point has to be modified. If there are global collisions for all the cutter orientations, as shown in Fig. 8(a), the model cannot be machined accurately in this set-up orientation. However, CC-points that have only local collision can be modified as shown in Fig. 8(b). In this case, the retracting direction and retracting distance should be determined to avoid overcut. The direction toward the least potential energy is chosen to be the retracting

direction. With this retracting direction, the smallest distance that does not store any potential energy is the retracting distance. If more accurate machining is required, a smaller cutter should be used.

### 8. Experimental Result

The free-form surfaces shown in Fig. 9(a) are used to test this algorithm. Figure 9(b) is the generated potential energy field in the  $\alpha, \beta$  domain. One point on the surface corresponds to one cutter orientation. Because potential energy is not generated by a collision-free cutter orientation, the bottom of the field is

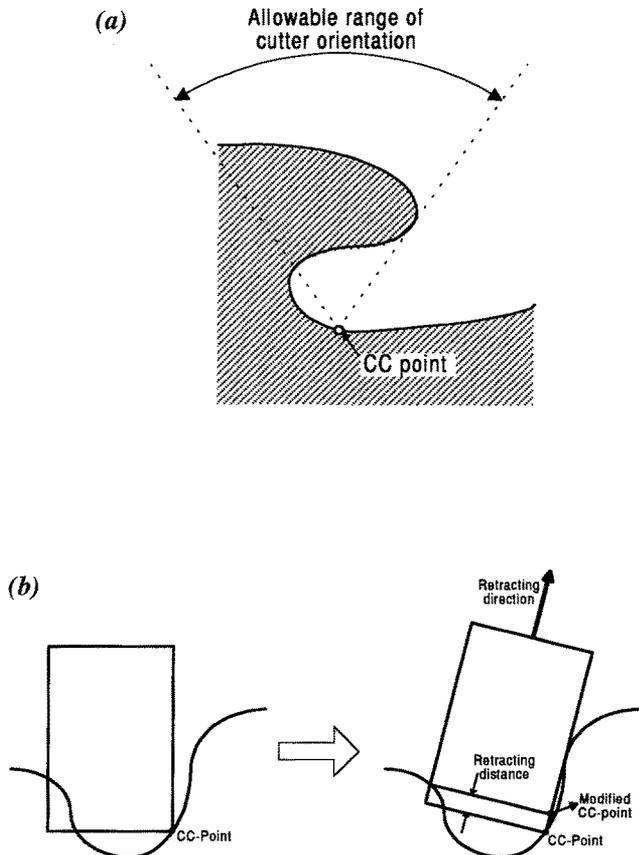


Fig. 8. (a) Non-machinable CC-point. (b) Example of the local collision and retraction.

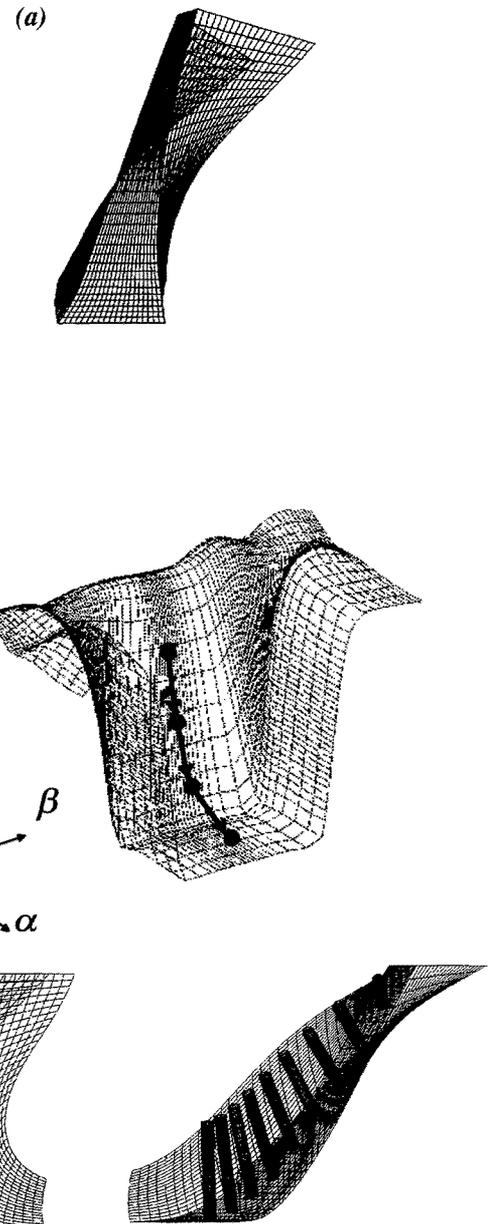


Fig. 9. (a) Impeller's two blades and part of rim bridging the blades. (b) Potential field in  $\alpha, \beta$  domain. (c) Determined cutter orientation.

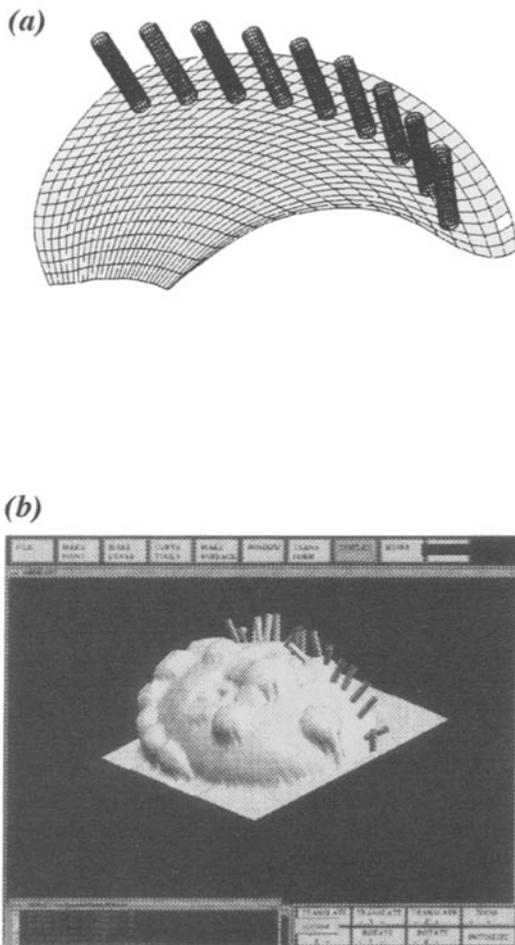
**Table 1.** Computing time for test surfaces.

Model surface	Number of CC-points	Computing time (s)	Computing time for each CC-point (s)
Impeller blade (Fig. 9)	40	49.6	1.24
Fan (Fig. 10(a))	30	25.5	0.85
Korean Mask (Fig. 10(b))	75	69.0	0.92

the feasible angle region where there is no collision. As shown in Fig. 9(b), in the case of collision, this algorithm traces to the bottom, using the gradient of potential energy. The determined cutter orientations for each of the CC-points are shown in Fig. 9(c). Figures 10(a) and 10(b) are the determined cutter orientations for the fan model and the Korean mask model along an iso-parametric tool path. Table 1 shows the computing time on a 20 MIPS engineering workstation for each model.

## 9. Further Work

Our algorithm has some limitations which we hope to overcome through further studies. The proposed algorithm handles only



**Fig. 10.** (a) Determined cutter orientation for “fan”. (b) Determined cutter orientation for “Korean Mask”.

the end milling operation. Proper modification would enable the proposed algorithm to be applied to any milling operation, e.g. side milling with a cylindrical cutter.

Cutter orientation derived by the proposed algorithm may change abruptly between each CC-point, because the cutter orientation at each of the pre-determined CC-points are calculated independently in the local coordinate system. Thus, it is necessary to generate smooth CL-data somehow.

The current approach assumed that the CC-points are given in advance, as in other methods. In general, because it is difficult to obtain the CC-points from CL-data, CC-points are generated in advance and CL-data are calculated from CC-points. Iso-parametric curves of the part surface, or intersection curves between parallel planes and the part surface, are usually used as the cutter path from which CC-points are derived. We can expect that better machining would result when the cutter orientations and CC-points are calculated simultaneously.

Finally, the current approach calculates the CL-data from only the geometries of the part surface and the cutter. It may be necessary to include the effect of cutting force in determining the CL-data, especially for machining hard material.

## 10. Conclusion

The potential energy method that was used to keep the robot away from obstacles is applied to generate collision-free cutter location data in 5-axis milling. In order to use the potential energy method in an approach involving contact between objects, the energy function has been modified. To improve the machining efficiency, the curvature difference between the tool swept surface and the part surface at each CC-point is minimised. Because a filleted-end cutter is used in curvature calculation, this approach can also be applied to a ball-end cutter and a flat-end cutter. In the minimisation process, the collision detection equation derived by an artificial potential energy function is used as a constraint equation. Using the repulsive force, the collision-avoidance calculation is enhanced.

### Acknowledgements

The authors are grateful to the Institute of Advanced Machinery and Design for partially supporting this research.

### References

1. G. W. Vickers and K. W. Quan, “Ball-mills vs. end-mills for curved surface machining”, *Transactions of the ASME*, **111**, pp. 22–26, February 1989.

2. K. Tang, T. Woo and J. Gan, "Maximum intersection of spherical polygons and workpiece orientation for 4- and 5-axis machining", *Journal of Mechanical Design*, **114**, pp. 477–485, September 1992.
3. G. Elber and E. Cohen, "Arbitrarily precise computation of Gauss maps and visibility sets for freeform surfaces", *Proceedings, Solid Modeling '95, Salt Lake City, USA*, pp. 271–279, 1995.
4. Y. S. Lee and T. C. Chang, "Automatic planning for 5-axis sculptured surface machining", *Computers in Engineering*, **1**, pp. 281–291, 1994.
5. A. Szende, "Mathematical foundations of free-form surface machining with flat-end cutters", Ph.D. thesis, Seoul National University, Korea, 1995.
6. Y. Takeuchi and T. Idemura, "5-axis control machining and grinding on solid model", *Annals of the CIRP*, **40**(1), pp. 455–458, 1991.
7. T. Saito and T. Takahashi, "NC machining with G-buffer method", *Computer Graphics*, **25**(4), pp. 207–216, July 1991.
8. S. X. Li and R. B. Jerard, "5-axis machining of sculptured surfaces with a flat-end cutter", *Computer Aided Design*, **26**(3), pp. 165–178, March 1994.
9. K. Y. Chang and E. D. Goodman, "A method for NC toolpath interference detection for a multi-axis milling system", *Control of Manufacturing Processes, ASME*, **28**, pp. 23–30, 1991.
10. X. W. Liu, "Five-axis NC cylindrical milling of sculptured surfaces", *Computer Aided Design*, **27**(12), pp. 887–894, 1995.
14. G. Yu, "General tool correction for five-axis milling", *International Journal of Advanced Manufacturing Technology*, **10**, pp. 374–378, 1995.
12. B. K. Choi, J. W. Park and C. S. Jun, "CL-data optimization in 5-axis surface machining", *Computer Aided Design*, **25**(6), pp. 377–386, 1993.
13. S. H. Mullins, C. G. Jensen and D. C. Anderson, "Scallop elimination based on precise 5-axis tool placement, orientation, and step-over calculations" *Advances in Design Automation, ASME*, **2**, pp. 535–544, 1993.
14. J. Kruth and P. Klewais, "Optimization and dynamic adaptation of the cutter inclination during five-axis milling of sculptured surfaces", *Annals of the CIRP*, **43**(1), pp. 443–448, January 1994.
15. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots", *International Journal of Robotics Research*, **5**(1), pp. 500–504, 1986.
16. R. Volpe and P. Khosla, "Manipulator control with superquartic artificial potential functions: theory and experiments", *IEEE Proceedings on Robotics and Automation*, April 1988, pp. 1778–1784.
17. Y. K. Hwang, "A potential field approach to path planning", *IEEE Transactions on Robotics and Automation*, **18**(1), pp. 23–32, February 1992.
18. C. W. Warren, "Global path planning using artificial potential fields", *IEEE Proceedings on Robotics and Automation*, pp. 316–321, 1989.
19. S. H. Bae, "Development of the automated assembly sequence generation system to support mechanical assembly planning", Ph.D. thesis, Seoul National University, Korea, 1995.
20. G. Farin, *Curves and Surfaces for Computer Aided Geometric Design, A Practical Guide*, 3rd edn, Acaemic Press, 1993.
21. G. H. Vanderplaats, *ADS, A FORTRAN Program for Automated Design Synthesis*, May 1985.
22. I. Cho and K. Lee, "Determination of tool orientations in 5-axis milling using potential energy method", *Proceedings of the first world congress on intelligent manufacturing processes and systems*, February 1995.