



## 5-axis Iso-scallop Tool Paths along Parallel Planes

Christophe Tournier<sup>1</sup> and Claire Lartigue<sup>2</sup>

<sup>1</sup>ENS Cachan, [tournier@lurpa.ens-cachan.fr](mailto:tournier@lurpa.ens-cachan.fr)

<sup>2</sup>Paris Sud 11 University, [lartigue@lurpa.ens-cachan.fr](mailto:lartigue@lurpa.ens-cachan.fr)

### ABSTRACT

This paper presents a new method of computing constant scallop height tool paths in 5-axis milling. Indeed, usual iso-scallop tool path computation methods in 3-axis or 5-axis milling generate tool path self-intersections, which prevent from reaching the required quality and productivity. The proposed approach consists in following parallel planes in 5-axis milling to avoid self-intersection of the tool path. The constant scallop height is guaranteed by modifying the machined strip width by means of the orientation angles of tool axis (yaw and tilt angles). Resulting scallop height and surface roughness are simulated using the N-Buffer method.

**Keywords:** 5-axis milling, free-form surfaces, filleted end mill, constant scallop height.

**DOI:** 10.3722/cadaps.2008.278-286

### 1. INTRODUCTION

The manufacturing process of sculptured surfaces is an important issue in aeronautic, mould and die industries. This process, more generally based on 3 or 5-axis end milling, consists in sweeping the surface to be machined by a tool. At final, the process must produce parts respecting geometrical specifications, in a minimum machining time. In this context, the machining strategy, which defines the sweeping directions, plays a major role.

To overcome the limits of common machining strategies (parallel planes or isoparametric curves), which generate over-quality of surface finish, iso-scallop strategy has been developed in 3-axis milling [6],[16],[18],[19] as well as in 5-axis milling [10-11],[13],[20]. With such strategies, the respect of the discretization parameters specified by the user is ensured in a uniform way over the whole surface. Geometrical deviations are thus uniformly distributed, which allows forward a homogeneous polishing of the surface. Moreover, for a given tool geometry, tool path length is minimum [18]. Therefore, iso-scallop height machining strategies are of a major interest. However, in the case of iso-scallop strategies and unlike common strategies for which the feed direction is imposed by the strategy itself, the tool path geometry cannot be predicted beforehand. Iso-scallop tool path generation is a deterministic calculation depending on four parameters, which are the initial path, the tool geometry, the surface to be machined and the specified scallop height. According to these parameters, the behavior and the distribution of the paths on the surface are different. It is important to notice that the initial tool path necessary to computation is not necessarily the initial one during machining.

In every published algorithm of the literature, iso-scallop tool path calculation relies on an initial trajectory  $T_{p_0}$ , arbitrarily selected on the surface. This trajectory is sampled in a set of tool postures  $T_{p_{0,j}}$ . The algorithm is thus applied to each tool posture  $T_{p_{i,j}}$  with the objective of finding the posture  $T_{p_{i+1,j}}$  of the adjacent tool path respecting given constraints. The calculation of the complete tool path can be carried out by first looping on  $j$ , then on  $i$ , (pass by pass), or either by first looping  $i$  then  $j$ . For instance, let us consider the 3-axis ball-end milling of a portion of sphere connected to a plane with tangency continuity. The surface to be machined thus consists of a plane surface, a toroidal

surface and a spherical surface. The tool path is calculated according to the algorithm proposed in [6],[19] based on the Machining Surface model.

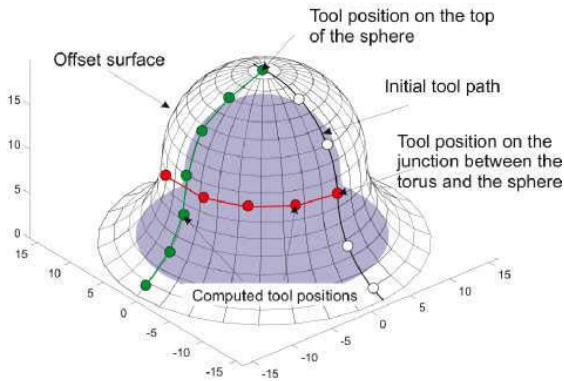


Fig. 1(a): loop in iso-scallop tool path.

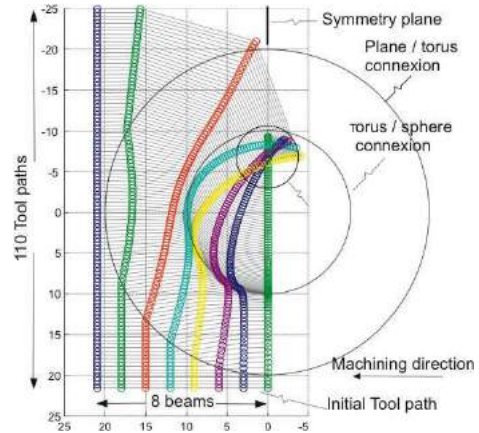


Fig. 1(b): loop in iso-scallop tool path.

In some cases the algorithm fails because of the geometry of the generated tool paths. A particular but foreseeable situation is observed when the initial tool path belongs to the vertical plane passing through the top of the sphere (Fig. 1(a)). The tool posture, which exactly belongs to the transition curve between the torus and the sphere, follows this curve and turns around indefinitely (figure 1a). The tool path thus becomes degenerated and it is not possible to control the direction of the tool path, which depends on the orientation of the normal to the surface to be machined at the considered point. Particularly, it depends on the projection of the surface normal in the machining direction [6,19].

Another problem occurs when the curvature radius of the tool path becomes smaller than the tool radius. In this case, the envelope surface of the tool movement may present self- intersections (figure 1b) and the resulting tool path thus contains loops. Hence, iso-scallop machining strategy loses all its interest as the tool sweeps several times the same area, decreasing accordingly machining efficiency [16,18]. A solution is to remove loops due to self-intersections. Such an approach is, for instance, used for pocket machining with a helical machining strategy [17]. All the techniques allowing this operation are referenced in [14]. However, they generally involve tangency discontinuities on the trajectory which are not acceptable within the context of High Speed Machining, [5,15].

To prevent previous problems, we propose a 5 axis milling iso-scallop strategy based on parallel planes. Rotation axes can be considered as degrees of freedom allowing a constant scallop height between predefined parallel tool paths. The interest of such a strategy is to avoid self-intersections by imposing beforehand tool paths' geometry. However, tool path length, which is roughly proportional to machining time, is no more optimal for machined strip width is not maximal for each cutter location. Nevertheless, such a strategy may improve kinematical performances during machining. Actually, classical iso-scallop algorithms provide tool paths the execution of which may be difficult at high feedrates, as all axes of the machine tool are used. In the proposed approach, as parallel planes can be chosen parallel to a specific plane of the machine tool, only 3 axes (two translations and one rotation) are used.

The paper is organized as follows. We first present the bi-parameterized surface model of tool paths we have developed for sculptured surface machining. The algorithm for 5-axis iso-scallop tool path calculation along parallel planes is next detailed. The efficiency of the algorithm is illustrated through an example for which classical iso-scallop machining strategies failed.

**2. BI-PARAMETRIZED SURFACE MODEL OF TOOL PATHS**

The proposed algorithm is based on a surface representation of the machining tool path [4]. Concerning the general case of 5-axis end milling using a filleted end cutter tool, a generic definition was proposed in [9].

In 5-axis machining, the tool axis orientation is specified by the user through two angles  $(\theta_t, \mathbf{t})$  and  $(\theta_n, \mathbf{n})$  in the local coordinate frame  $(C_c, \mathbf{f}, \mathbf{n}, \mathbf{t})$ , where  $\mathbf{f}$  is the tool feed vector,  $\mathbf{n}$  the normal vector to the surface and  $\mathbf{t}$  the tangent

vector to the surface defined by  $\mathbf{t} = \mathbf{f} \wedge \mathbf{n}$  (Fig. 2). For each cutter location, a point locating the tool and a vector orienting the tool axis must be specified. Two surfaces are thus necessary to calculate the tool trajectory.

Let us consider  $r_c$  and  $R$ , respectively, the corner radius and the tool radius. If  $C_c$  is the contact point between the tool and the surface to be machined,  $K$  is defined as the offset point of  $C_c$  by a distance value equal to  $r_c$ . When the tool is set in position, the point  $K$  remains fixed. Therefore, points  $K$  and  $C_L$  and the normal  $\mathbf{n}$  allow positioning the tool and define two surfaces:  $S_G$ , the guiding surface and  $S_O$ , the orientation surface (Fig. 2). The guiding surface  $S_G$  is the locus of the  $K$  point, with  $K = C_c + r_c \cdot \mathbf{n}$ . It is thus the offset surface of the nominal surface by a distance equal to the corner radius  $r_c$ . Therefore, the guiding surface does not depend on the machining strategy. The orientation surface  $S_O$  is the locus of the  $C_L$  point with  $C_L = C_c + r_c \cdot \mathbf{n} + R \cdot \mathbf{v}$ . Hence, the orientation surface gives the orientation of the tool axis according to the considered machining strategy.

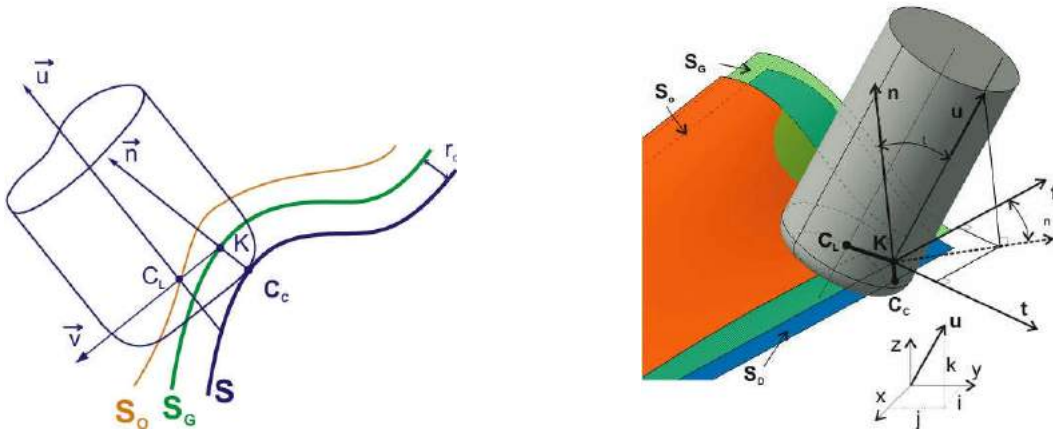


Fig. 2: The guiding and orientation surface in 5-axis milling.

The main advantage of the bi-parameterized surface model (also called the Machining Surface) in 5-axis milling is to know the whole positions of a fixed point of the tool, the point  $K$  for instance, which makes it possible to cover the entire surface without gouging. Advantages and difficulties to develop such a model in 5-axis milling are detailed in [12].

**3. “ISO-SCALLOP TOOL PATHS ALONG PARALLEL PLANES” ALGORITHM**

First, the direction of the guiding plane and the distance  $d$  between parallel planes must be set. The initial tool path as well as the orientation of the tool axis on this *initial* path is also given. Given a tool path  $i$ , the determination of the set of tool postures (tool positions  $K_{i,j}$  and tool axis orientations  $\mathbf{u}_{i,j}$ ) is divided into two stages (Fig. 3(a)):

- Computation of scallop points  $P_{i,j}$
- Computation of tool postures ( $K_{i,j}, \mathbf{u}_{i,j}$ ) of the next path.

Both stages are repeated from the initial to the last path.

**3.1 Computation of Scallop Points**

Let us consider a tool posture defined by the position of the tool center  $C_L$  and the tool axis orientation  $\mathbf{u}$ .  $S_h$  is the specified scallop height and  $S_n$  is the surface to be machined. The first stage consists in finding the scallop point  $P$  so that (Fig. 3(b)):

1.  $P$  belongs to the iso-scallop surface  $S_i(u, v)$ :

$$P = S_i(u, v) = S_n(u, v) + S_h \cdot \mathbf{n}(u, v) \tag{3.1}$$

2.  $P$  belongs to the cutting edge of the filleted end mill located at the given posture. The implicit representation  $F(x,y,z)$  of the tool geometry (apart from a coordinate system modification) can be set as:

$$\left( (P - C_L)_x^2 + (P - C_L)_y^2 + (P - C_L)_z^2 + R^2 - r^2 \right)^2 - 4R^2 \left( (P - C_L)_x^2 + (P - C_L)_y^2 \right) = 0 \tag{3.2}$$

3. P belongs to the generative profile of the tool which creates the scallop curve [1]:

$$\mathbf{V}_P \cdot \mathbf{n}_{tool,P} = 0 \tag{3.3}$$

Where  $\mathbf{V}_P$  is the velocity vector of P belonging to the tool:

$$\mathbf{V}_P = \mathbf{V}_K + \mathbf{PK} \dot{\mathbf{Y}} \mathbf{W}_{tool/R_0} \tag{3.4}$$

with K the point of the tool belonging to the guiding surface and  $\Omega_{tool/R_0}$  the instantaneous speed rotation vector of the tool when moving along the surface.

In 5-axis milling, the tool is oriented in the local coordinate system  $R_1$  defined by  $(K, \mathbf{f}, \mathbf{n}, \mathbf{t})$  which evolves all path long. If  $R_0$  is the global coordinate system attached to the part, we can write:

$$\mathbf{W}_{tool/R_0} = \mathbf{W}_{tool/R_1} + \mathbf{W}_{R_1/R_0} \tag{3.5}$$

where  $\Omega_{tool/R_1}$  is defined by the tool orientation angles:

$$\mathbf{W}_{tool/R_1} = \frac{dq_t \cdot \mathbf{t} + q_n \cdot \mathbf{n}}{dt} \tag{3.6}$$

The instantaneous speed rotation vector  $\Omega_{R_1/R_0}$ , is calculated using the following equation (see appendix):

$$\mathbf{W}_{R_1/R_0} = \frac{1}{2} \cdot (\mathbf{f} \ddot{\mathbf{f}} \frac{d\mathbf{f}}{dt} + \mathbf{n} \ddot{\mathbf{n}} \frac{d\mathbf{n}}{dt} + \mathbf{t} \ddot{\mathbf{t}} \frac{d\mathbf{t}}{dt}) \tag{3.7}$$

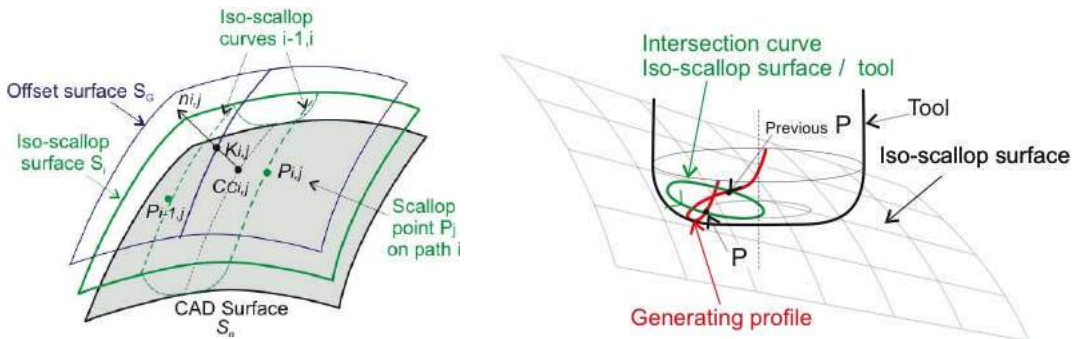


Fig. 3(a): Computation of scallop points and tool postures. Fig. 3(b): Intersection tool / iso-scallop surface.

In kinematics, the variable  $t$  represents the time but for this application,  $t$  is the parameter of the tool path curve followed by K. This leads to a non-linear system of three equations (3.1-3.3) with three unknowns ( $P_x, P_y, P_z$ ), coordinates of P in  $R_0$ . The resolution of this system using the Newton-Raphson method requires an initial solution very close to the solution in order to ensure rapid convergence. To find a good initial guess, we suggest to use the intersection curve between the tool and the iso-scallop surface (Fig. 3(b)), which means that we solve eq(3.1) and eq(3.2). More exactly, we follow the intersection curve from the previous scallop point (computed from the previous tool path) until we reach an area close to the solution that we search. The solution is found when eq(3.3) is verified. The non-linear function  $F$  from  $R^3$  into  $R$  is defined by:

$$F(P_x, P_y, P_z) = 0 \tag{3.8}$$

Since we are following the intersection curve between two surfaces, we know that the tangent vector  $\mathbf{T}_p$  to the intersection curve is equal to the cross product of the two normal vectors of the intersecting surfaces.

The resolution is done with a marching method starting from the initial solution  $X_0$  corresponding to the previous scallop point. Based on a known point  $X_p$ , we search for the following point  $X_{p+1}$  so that:

$$F(X_{p+1}) = 0 \quad (3.9)$$

$$((X_{p+1} - X_p) \cdot \frac{\mathbf{T}_p}{\|\mathbf{T}_p\|}) - \delta = 0 \quad (3.10)$$

$\delta$  is a numerical parameter ensuring convergence. Resolution is done in three steps: prediction, correction and progression. Prediction consists in moving a step forward along the tangent vector  $\mathbf{T}_p$  by a length equal to  $\delta$  to find the guess point  $X^*$ . In practice,  $\mathbf{T}_p$  is computed by numerical differentiation. Then, correction consists in applying Newton algorithm to find the solution  $X_{p+1}$  from the initial guess  $X^*$ . At final, progression is done by modifying parameter  $\delta$  regarding the curvature of the followed curve to minimize computing steps. Every times we find a point  $X_{p+1}$  on the intersection curve, equation (3.3) is evaluated. When its sign changes, the point  $(X_{p+1} + X_p)/2$  is used as the initial solution to solve the non-linear system of three equations (3.1-3.3) in order to find the scallop point P. This last operation is performed using Newton algorithm. Finally, it takes  $n_1$  iterations ( $\sim 6$ ) to find the guess solution and  $n_2$  iterations ( $\sim 4$ ) to find the solution. Experimentations have shown the robustness of the algorithm. Once the scallop point is determined, the tangent to the scallop curve  $\mathbf{T}_{sc}$  on each scallop point P is given by the cross product of the normals of the tool and the scallop surface.

### 3.2 Computation of Tool Postures

The second stage of our method consists in computing the tool posture based on the scallop point. The geometrical conditions associated to the second stage are the followings:

1. The driving point K belongs to the guiding surface  $S_G$ :

$$K = S_G(u, v) = S_n(u, v) + r_c^3 \mathbf{n}(u, v) \quad (3.11)$$

2. The driving point K belongs to the guiding plane defined by  $(M, \mathbf{N}_{plane})$ :

$$\mathbf{MK} \cdot \mathbf{N}_{plane} = 0 \quad (3.12)$$

3. The scallop point P belongs to the cutting edge of the filleted endmill

$$\left( (P - C_L)_x^2 + (P - C_L)_y^2 + (P - C_L)_z^2 + R^2 - r^2 \right)^2 - 4R^2 \left( (P - C_L)_x^2 + (P - C_L)_y^2 \right)^2 = 0 \quad (3.13)$$

4. The normal vector to the tool at the scallop point  $\mathbf{n}_{tool,P}$  is perpendicular to the tangent vector to the scallop curve  $\mathbf{T}_{sc}$ ,

$$\mathbf{T}_{sc} \cdot \mathbf{n}_{tool,P} = 0 \quad (3.14)$$

In order to simplify the problem, we set the yaw angle  $(\theta_n, \mathbf{n})$  equal to zero. This also yields to a non-linear system of 4 equations (3.11-3.14) with 4 unknowns  $(K_x, K_y, K_z, \theta_t)$ . For convenience, we transform this system into a non-linear system of 2 equations with 2 unknowns,  $\tau$  et  $\theta_t$ , with  $\tau$  the parameter of the intersection curve between the guiding surface and the guiding plane, and  $\theta_t$  the tilt angle. As we conserve equations (3.13) and (3.14), we have to solve the intersection between the guiding surface and the guiding plane. This yields to solve:

$$g(u, v) = (S_G(u, v) - \mathbf{OM}) \cdot \mathbf{N}_{plane} = 0 \quad (3.15)$$

The approach consists in computing a set of points belonging to the intersection curve and to fit them with a cubic B-spline curve according to the least squares approximation. The computation of the intersection points is done exactly the same way we used for the intersection curve between the tool and the iso-scallop surface (3.8). The problem is

now: finding parameters  $\tau$  and  $\theta_t$  in order to solve equations (3.13) and (3.14). The main difficulty is once again to find the guess of the initial solution for the Newton algorithm.

Let us consider the machining of a revolution cylinder along the circle arc cross-sections. This simple example represents an approximation of any surface to be machined, at the first order in the transversal direction, and the second order in the machining direction. This choice is coherent as regards the small distance between two successive tool paths.

Let us assume that the first stage of the method gives a set of scallop points based on an initial tool path. Fig. 4 is the representation of the zero contour boundaries of the two non-linear equations (3.13) and (3.14) in the  $(\tau, \theta_t)$  plane. Each equation divides the  $(\tau, \theta_t)$  plane into positive and negative regions bounded by zero curves. Solid curves refer to equation (3.13) (also identified by numbers) and express the belonging of the scallop point P to the tool. Dashed curves (also identified by letters) refer to equation (3.14), the condition according to which the tool normal  $n_{\text{tool},P}$  is perpendicular to the tangent to the scallop curve  $T_{sc}$ . The desired solutions are the intersections of the zero curves. The number of solutions is a priori equal to six, but only one ensures a tool posture without gouging.

Actually, the four solutions associated to the intersections between the couples of curves (1, b) and (2, c) correspond to cases for which the tool machines the scallop point with its backside. Curve 3 describes the situations for which the tool machines the scallop point with its front side. The reason why this zone does not extend up to values of the tilt angle close to  $\pi/2$  is that the effective tool radius is too small to reach the scallop point, regarding the selected distance between planes d. Moreover, the set of solutions is restricted because we must also ensure a tangent contact between the tool and the surface, i.e. an interference free tool path. Hence, the tilt angle is located in the  $[0, \pi/2]$  interval. Then, the only root that respects all the constraints is located at the intersection of the couple of curves (3,d).

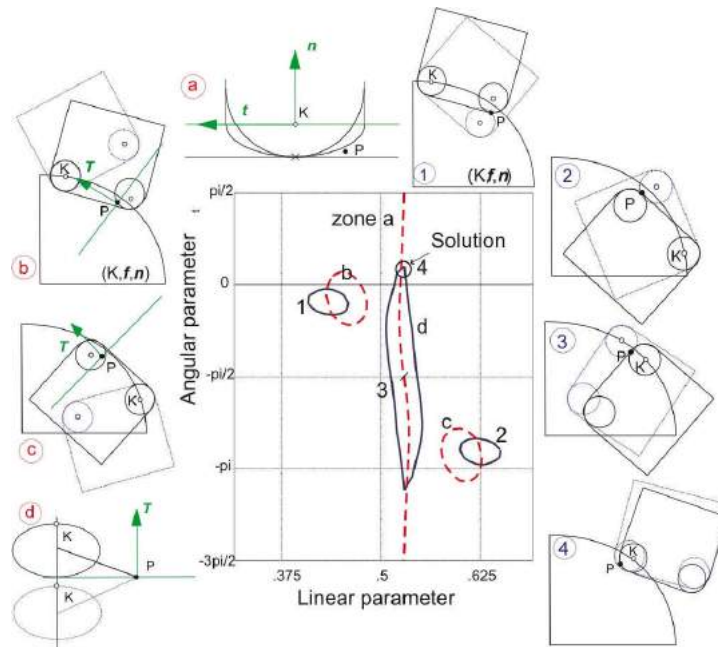


Fig. 4: Zero contour boundaries of the two non-linear equations (3.13 - 3.14).

It is possible that no satisfying root exists according to the value of the distance between guiding planes, the scallop height and the tool geometry parameters. Indeed, if we take the equation suggested in [10] concerning the effective tool radius in 5-axis milling with a filleted end-mill, we obtain:

$$R_{eff} = \frac{r_c (R + r_c^3 \sin q_t)}{r_c \sin q_t^3 \cos q_n^2 + (R + r_c^3 \sin q_t) \sin q_n^2} \tag{3.16}$$

and with  $\theta_n = 0$  :

$$R_{eff} = \frac{(R + r_c \sin^3 q_t)}{\sin q_t} \tag{3.17}$$

The value of the effective tool radius  $R_{eff}$  reaches its minimum for  $\theta t = \pi/2$ . If we consider a first order approximation of the surface, by neglecting the transversal curvature, the minimum distance between the guiding planes is given by the well-known formula:

$$d_{min} = 8^3 S_h^3 (R + r_c) \tag{3.18}$$

Finally, considering the previous discussion about  $d_{min}$ , the choice of the initial guess is facilitated because the zero curve “d” covers all the interval of variation of the tilt angle  $\theta t$ . Regardless the tilt angle, there is always one solution to equation (3.14) when the tool follows the considered tool path.

**4. APPLICATIONS**

We implemented this approach using Matlab for the machining of a surface presenting curvature variations so that, if using a classical iso-scallop strategy, some loops may appear (Fig. 5(a)). The scallop height  $S_h$  is set to  $10\mu m$ . The filleted endmill radii are  $R = 5mm$  and  $r_c = 1.5mm$ . We generate preliminary parallel tool paths on the guiding surface (i.e. the guiding point follows parallel planes) by intersecting a set of YZ parallel planes with the guiding surface using the algorithm previously described. The distance between guiding planes is fixed to  $d = 5mm$ . The tilt angle along the initial tool path, is set to:  $\theta t = 5^\circ$ . We use a set of 25 points of the initial tool path to compute the iso-scallop tool paths. Finally, we obtain for each tool path the tilt angle  $\theta t$  of the tool axis according to the tool path parameter  $\tau$  of the guided point K (Fig. 5(b)). Some of the curves  $\theta t(\tau)$  are represented along the tool path in Fig. 5(b). In order to simulate machining and to assess our algorithm, we developed numerical tools such as the N-buffer simulation proposed in [7]. Each tool path, described as a cubic B-spline, is sampled so that the chordal deviation does not influence the scallop height [8].

Figures 5c shows that the resulting scallop height matches the specification. It can be noticed (Fig. 5(d)) that the widths of cut are not any more constant as when using classical iso-scallop strategies.

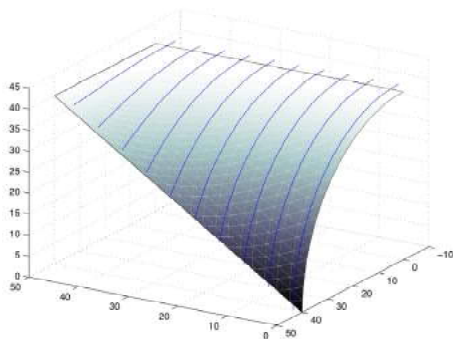


Fig. 5(a): NURBS test surface.

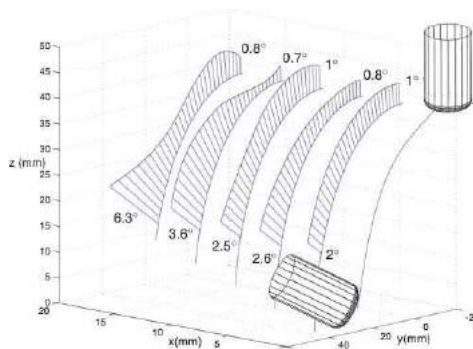


Fig. 5(b): Resulting tool orientations.

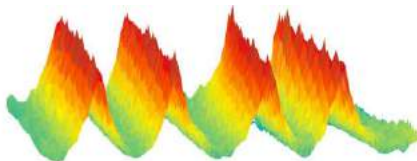


Fig. 5(c): Generated scallops.

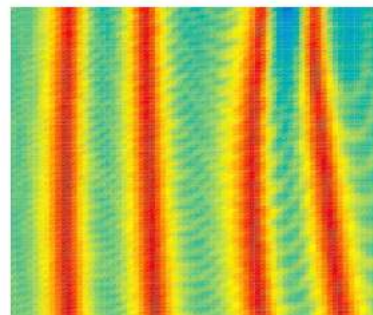


Fig. 5(d): Machining stripes.

**5. CONCLUSIONS AND FUTURE WORKS**

The proposed algorithm provides iso-scallop tool paths without self-intersections. This goal is achieved by guiding the tool along parallel planes and by using the tilt angle to modify the width of cut and to maintain a constant scallop height. In the application, we choose a constant distance between tool path and a null yaw angle. An optimization of this first approach would consist in maximizing the distance between planes in order to minimize the number of tool paths. Another possibility to improve the algorithm would be the use of the yaw angle to increase the number of tool axis orientation candidates to solve the problem. It also should be interesting to compare both classical and parallel planes iso-scallop strategies in terms of effective machining time. Indeed, the tool path geometry has great influence on the kinematical behavior of the machine tool. Our approach generates 2D tool paths with tool axis modifications along the tool path while standard approaches generate 3D tool paths with constant tool axis orientation.

**6. APPENDIX**

Let  $R_1 = \{O_1, x_1, y_1, z_1\}$  be a coordinate system attached to a solid S, moving in a fixed coordinate system  $R_0 = \{O, x_0, y_0, z_0\}$ . The basis vector of  $R_1$  are normalized and perpendicular. Let  $\vec{\Omega}_{R_1/R_0}$  be the rotation vector of S (or  $R_1$ ) compared to  $R_0$ . We can write:

$$\left(\frac{d\vec{x}_1}{dt}\right)_{R_0} = \vec{\Omega}_{R_1/R_0} \wedge \vec{x}_1 \quad , \quad \left(\frac{d\vec{y}_1}{dt}\right)_{R_0} = \vec{\Omega}_{R_1/R_0} \wedge \vec{y}_1 \quad , \quad \left(\frac{d\vec{z}_1}{dt}\right)_{R_0} = \vec{\Omega}_{R_1/R_0} \wedge \vec{z}_1$$

By multiplying each term with the considered basis vector, we get:

$$\vec{x}_1 \wedge \left(\frac{d\vec{x}_1}{dt}\right)_{R_0} + \vec{y}_1 \wedge \left(\frac{d\vec{y}_1}{dt}\right)_{R_0} + \vec{z}_1 \wedge \left(\frac{d\vec{z}_1}{dt}\right)_{R_0} = \vec{x}_1 \wedge (\vec{\Omega}_{R_1/R_0} \wedge \vec{x}_1) + \vec{y}_1 \wedge (\vec{\Omega}_{R_1/R_0} \wedge \vec{x}_1) + \vec{z}_1 \wedge (\vec{\Omega}_{R_1/R_0} \wedge \vec{x}_1)$$

We then use the double vector product simplification:

$$A \wedge (B \wedge C) = B.(A.C) - C.(A.B)$$

We get:

$$\vec{x}_1 \wedge \left(\frac{d\vec{x}_1}{dt}\right)_{R_0} + \vec{y}_1 \wedge \left(\frac{d\vec{y}_1}{dt}\right)_{R_0} + \vec{z}_1 \wedge \left(\frac{d\vec{z}_1}{dt}\right)_{R_0} = \vec{\Omega} - \vec{x}_1.(\vec{\Omega} . \vec{x}_1) + \vec{\Omega} - \vec{y}_1.(\vec{\Omega} . \vec{y}_1) + \vec{\Omega} - \vec{z}_1.(\vec{\Omega} . \vec{z}_1)$$

after gathering all the terms:

$$\vec{x}_1 \wedge \left(\frac{d\vec{x}_1}{dt}\right)_{R_0} + \vec{y}_1 \wedge \left(\frac{d\vec{y}_1}{dt}\right)_{R_0} + \vec{z}_1 \wedge \left(\frac{d\vec{z}_1}{dt}\right)_{R_0} = 3\vec{\Omega} - (\Omega_x . x_1 + \Omega_y . y_1 + \Omega_z . z_1) = 2\vec{\Omega}$$

finally:

$$\vec{\Omega}_{R_1/R_0} = \frac{1}{2} \cdot \left( \vec{x}_1 \wedge \left(\frac{d\vec{x}_1}{dt}\right) + \vec{y}_1 \wedge \left(\frac{d\vec{y}_1}{dt}\right) + \vec{z}_1 \wedge \left(\frac{d\vec{z}_1}{dt}\right) \right)$$

**7. REFERENCES**

[1] Chiou, C. J.; Lee, Y. S.: A shape-generating approach for multi-axis machining G-buffer models, Computer-Aided Design, 31, 1999, 761-776.



- [2] Choi, B. K.; Lee, C. S.; Hwang, J. S.; Jun, C. S.: Compound surface modeling and machining, *Computer-Aided Design*, 20 (3), 1988, 127-136.
- [3] Dragomatz, D.; Mann, S.: A classified bibliography of literature on NC milling path generation, *Computer-Aided Design*, 29(3), 1997, 239-247.
- [4] Duc, E.; Lartigue, C.; Tournier, C.; Bourdet, P.: A new concept for the design and the manufacturing of free-form surfaces: the machining surface, *Annals of the CIRP*, 48(1), 1999, 103-106.
- [5] Dugas, A.; Lee, J. J.; Hascoet, J-Y.: An Enhanced Machining Simulator With Error Analysis, *Int. Journal of Manufacturing System/SME*, 21(6), 2003, 451-564.
- [6] Feng, H. Y.; Li, H.: Constant scallop-height tool path generation for three-axis sculptured surface machining, *Computer-Aided Design*, 34(9), 2002, 647- 654.
- [7] Jerard, R.; Hussaini, S.; Drysdale, R.; Schaudt B.: Approximate methods for simulation and verification of numerically controlled machining programs, *The Visual Computer*, 5, 1989, 329-348.
- [8] Lartigue, C.; Duc, E.; Tournier, C.: Machining of free-form surfaces and geometrical specifications, *Journal of the Engineering Manufacture, Proc. of the Institution of Mechanical Engineers, Part B*, 213, 1999, 21-27.
- [9] Lartigue, C.; Tournier, C.; Ritou, M.; Dumur, D.: High-performance NC for High- Speed Machining by means of polynomial trajectories, *Annals of the CIRP*, 53(1), 2004, 317-320.
- [10] Lee, Y. S.: Non isoparametric tool path planning by machining strip evaluation for 5-axis sculptured surface machining, *Computer-Aided Design*, 30(7), 1998, 559-570.
- [11] Li, H.; Feng, H. Y.: Efficient five-axis machining of free-form surfaces with constant scallop height tool paths, *International Journal of Production Research*, 42(12), 2004, 2403 - 2417.
- [12] Li, S. X.; Jerard, R. B.: 5-axis machining of sculptured surfaces with a flat-end cutter, *Computer-Aided Design*, 26(3), 1994, 165-178.
- [13] Lo, C-C.: Efficient Cutter-path planning for five-axis surface machining wit flat-end cutter, *Computer-Aided Design*, 31(9), 1999, 557-566.
- [14] Maekawa, T.: An overview of offset curves and surfaces, *Computer-Aided Design*, 31(3), 1999, 165-173.
- [15] Pateloup, V.; Duc E.; Ray, P.: Corner optimization for pocket machining, *International Journal of Machine Tools and Manufacture*, 44(12-13), 2004, 1343-1353.
- [16] Sarma, R.; Dutta, D.: The Geometry and Generation of NC Tool Paths, *Journal of Mechanical Design*, 119, 1997, 253-258.
- [17] Suh, Y. S.; Lee, K.: NC milling tool path generation for arbitrary pockets defined by sculptured surfaces, *Computer-Aided Design*, 22(5), 1990, 273-284.
- [18] Suresh, K.; Yang, D. C. H.: Constant scallop-height machining of free-form surfaces, *Journal of Engineering for Industry*, 116, 1994.
- [19] Tournier, C.; Duc, E.: A surface based approach for constant scallop height tool path generation, *International Journal of Advanced Manufacturing Technology*, 19, 2002, 318-324.
- [20] Tournier, C.; Duc, E.: Iso-scallop tool path generation in 5-axis milling, *International Journal of Advanced Manufacturing Technology*, 25 (9-10), 2005, 867-875.