Subhajit Sarkar
Partha Pratim Dey

ISSN 1333-1124
eISSN 1849-1391

TOOL PATH PLANNING FOR MACHINING FREE-FORM SURFACES

Summary

This paper is about new iso-parametric tool path planning for machining trimmed free-form surfaces. The trimmed surface has been re-parameterized by two different parameterization techniques, namely, the partial differential equation method and the newly developed boundary interpolation method. The efficiency of the scheme has been measured in terms of path length and computational time needed for machining some typical surfaces. Conventionally, the forward-step is calculated by approximating the cutting curve with the osculating circle. The actual tolerance of the forward-step may go beyond the prescribed limit due to the circular arc approximation. In this study, the actual cutting curve has been considered to keep the tolerance in the forward-step below the prescribed value. The new algorithm has been tested on some typical surfaces and the results show a significant improvement in the surface profile in terms of tolerance of the forward-step.

Key words: *iso-parametric tool path, CNC machining, CAM.*

1. Introduction

Sculptured or free-form surfaces are widely used in designing surfaces of various objects, such as turbine blades, automobile bodies, etc. In the industries equipped with CAD/CAM facilities, free-form surfaces are machined by using 3/5 axis CNC machines in which the tool follows the path defined by a set of coded instructions.

Due to the inherent nature of the motion of the tool, tool paths are always a series of straight lines/arcs whereas the actual sculptured surface is a surface of varying slope and curvature. Thus, the free-form surface is approximated by a series of straight lines and arcs for machining, and the goal of machining is to get as close as possible to the design surface.

A number of tool path generation techniques have been developed since the introduction of CAD/CAM [1-9]; they include the iso-parametric, iso-planer, iso-scallop, iso-phote technique, etc. The iso-parametric method is widely used due to its efficiency in calculation, and better surface quality and aerodynamic effect (required in blades/vanes of turbines and pumps). The present study is aimed at generating iso-parametric tool paths on a trimmed free-form surface using 3-axis CNC milling by keeping the actual tolerances within the given limit while calculating forward-steps.

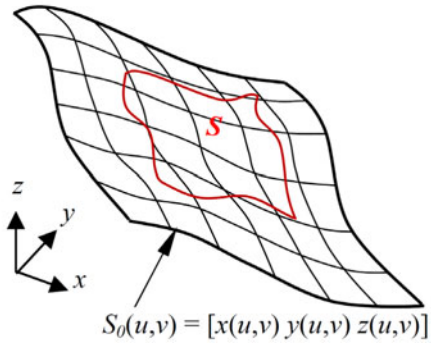


Fig. 1 Surface "S" is trimmed from the original surface "S_0"

For a trimmed surface (Fig. 1), the original parameterization does not conform to the new boundary conditions and hence the trimmed surface needs to be parameterized anew, which is named re-parameterization. For the surfaces with complex boundaries the algebraic parameterization cannot be applied due to some inherent problems [10]. In such conditions, re-parameterization can be done by two types of discrete parameterization techniques viz. partial differential equation (PDE) method [11-15] and boundary interpolation method [10]. In this paper, these two parameterization methods have been used for re-parameterization of the trimmed surface. Then, tool paths were generated from the interpolated curves yielded by re-parameterizations.

The tool paths formulation depends on side-steps and forward-steps. In section 3.2, the side-step is computed on the physical domain considering the flat, convex and concave profile of the surface [16]. Then, in section 3.3, a fresh concept is given for the conversion of the side-step from the physical domain into the computational or parametric domain. Thereafter, a novel method for the forward-step calculation is introduced (section 3.4) taking the true profile of the surface into consideration, unlike the circular arc approximation given in literature [16]. This forward-step algorithm is capable of maintaining the tolerance very close to the given value. The methods have been examined with some typical cases along with an analysis of the actual tolerances produced to show the accuracy of the present method.

2. Methods of re-parameterization

Here, the re-parameterization techniques viz. the Laplace PDE method and the boundary interpolation method are presented very briefly. For details, readers may refer to some of the relevant literature in which these methods are discussed in detail [10, 13-17].

2.1 PDE method

For two-dimensional surface, two elliptic partial differential equations (Eq. 2.1 and 2.2) are solved [13] with the independent variables (u,v) in the physical plane and the dependent variables (ξ,η) in the computational plane. The Laplace form of the elliptic PDEs are:

$$\frac{\partial^2 \xi}{\partial u^2} + \frac{\partial^2 \xi}{\partial v^2} = \xi_{uu} + \xi_{vv} = 0 \tag{2.1}$$

$$\frac{\partial^2 \eta}{\partial u^2} + \frac{\partial^2 \eta}{\partial v^2} = \eta_{uu} + \eta_{vv} = 0 \tag{2.2}$$

After the variable transformation [10, 18], Eqs. 2.1 and 2.2 take the following elliptic form:

$$au_{\xi\xi} - 2bu_{\xi\eta} + cu_{\eta\eta} = 0 \tag{2.3}$$

$$av_{\xi\xi} - 2bv_{\xi\eta} + cv_{\eta\eta} = 0 \tag{2.4}$$

where

$$a = (u_\eta^2 + v_\eta^2)\ ;\ b = (u_\xi u_\eta + v_\xi v_\eta)\ ;\ c = (u_\xi^2 + v_\xi^2)\ . \tag{2.5}$$

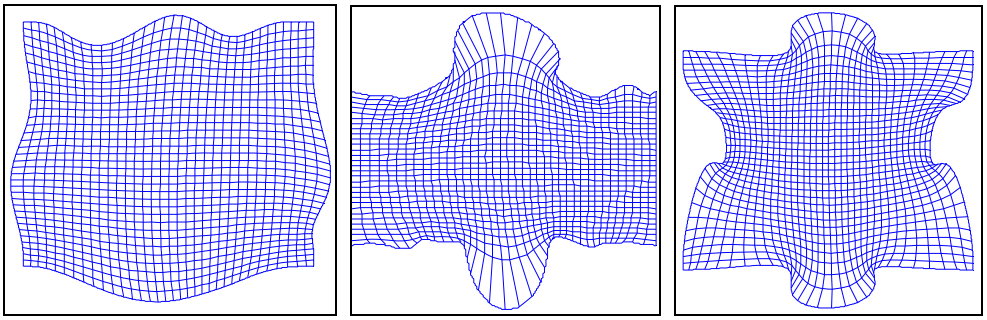


Fig. 2 Laplace PDE method

The solution to Eqs. 2.3, 2.4 and 2.5 produces a boundary fitted parameterized 2D surface. Figure 2 shows the result of the PDE method applied on the surfaces with various boundary conditions (simple and irregular). For irregular boundaries the distribution of interpolated curves is very uneven and causes difficulties in the tool path generation.

Although the Poisson equation can provide solution to this parameterization problem by means of forcing functions, it needs to be solved through the trial and error process [13, 14, 18]. This limits its usage for parameterization purpose in CAD/CAE applications where automatic processes are essential.

2.2 Boundary interpolation method

Boundary interpolation method [10] can resolve the anomalies yielded by the PDE method. In this method, the intermediate curves are generated by combining the linear and geometric vectors [10] at all the constituent data points of the two opposite boundary (or generating) curves. A technique of simultaneous displacement of the interpolated curves from the opposite boundaries has been adopted. The direction of displacement is from one of the curved boundaries to the other.

The interpolated curves are represented as functions of six geometric parameters,

$$C_j = f(C_0(\xi), \overline{C}_0(\xi), \alpha_j(\xi), \overline{\alpha}_j(\xi), d_j(\xi), \overline{d}_j(\xi)) \tag{2.6}$$

Here, $C_0(\xi)$ and $\overline{C}_0(\xi)$ is a pair of opposite boundary curves or generating curves or parent curves, $\alpha_j(\xi)$ and $\overline{\alpha}_j(\xi)$, $d_j(\xi)$ and $\overline{d}_j(\xi)$ are the displacement correction functions and the displacement vectors of the curves $C_0(\xi)$ and $\overline{C}_0(\xi)$, respectively. The new sets of curves are generated by a simultaneous interpolation from the two opposite boundary curves. This simultaneous displacement process continues till the entire surface is constructed. So the entire surface is treated as if it is divided into two halves.

Therefore, the basic algorithm is denoted as

$$C_j(\xi) = C_{j-1}(\xi) + d_j(\xi) \tag{2.7}$$

where

$$d_j(\xi) = D_j(\xi)\, R_j(\xi) \tag{2.8}$$

Here $D_j(\xi)$ and $R_j(\xi)$ represent the displacement magnitude and the unit displacement direction of $d_j(\xi)$, respectively.

Let m (which is user defined) be the total number of curves including the boundaries, to be generated on the entire surface.

When m is odd,

$$D_j(\xi) = \vec{\alpha}_j(\xi)\frac{Y_{j-1}(\xi)}{2(q-j)+2} \quad \text{and} \quad q = ((m+1)/2)-1 \tag{2.9}$$

When m is even,

$$D_j(\xi) = \vec{\alpha}_j(\xi)\frac{Y_{j-1}(\xi)}{2(q-j)+3} \quad \text{and} \quad q = (m/2)-1 \tag{2.10}$$

$$\vec{\alpha}_j(\xi) = \alpha_j\beta_j + \bar{\alpha}_j(1-\beta_j) \tag{2.11}$$

where

$$\beta_j = (1-j/2q)\ ;\ j\in[1,q] \tag{2.12}$$

β_j is a blending function. $\alpha_j(\xi)$ & $\bar{\alpha}_j(\xi)$ are the displacement correction functions of $C_j(\xi)$ & $\bar{C}_j(\xi)$, respectively.

$$\alpha_j(\xi)=[-\tanh(\theta_j^3(\xi)/12\pi)+((\theta_j(\xi)-\pi)/\pi)(0.4-\tanh(\pi^2/12))+1+\tanh(\pi^2/12)]^{\gamma} \tag{2.13}$$

Here, $\theta\in[0,2\pi]$ is the included angle formed by a set of three consecutive data points of a curve [10]. The user defined parameter γ (a real number) controls the characteristics of the displacement correction function $\alpha_j(\xi)$.

$$R_j(\xi) = \frac{(1-f)G_j(\xi) + fL_j(\xi)}{\left|(1-f)G_j(\xi) + fL_j(\xi)\right|} \tag{2.14}$$

where the weighted function

$$f = \left(\frac{j}{q}\right)^{\left(\left(1-\left(\frac{j}{q}\right)\right)/j\right)^{(p/100)}} \tag{2.15}$$

$$L_j(\xi) = \frac{\bar{C}_{j-1}(\xi) - C_{j-1}(\xi)}{\left\|\bar{C}_{j-1}(\xi) - C_{j-1}(\xi)\right\|} \tag{2.16}$$

In Eq. 2.14, $L_j(\xi)$ and $G_j(\xi)$ represent the unit linear displacement vector and the unit geometric vector, respectively [10].

Referring to Fig. 3, it may be concluded that this new method is capable of resolving the problem of uneven parametric distribution produced by the PDE method.

3. Tool path planning

After re-parameterization the iso-parametric tool paths are determined depending on the iso-parametric curves. In the following sub-sections the tool path generation scheme will be presented step by step.

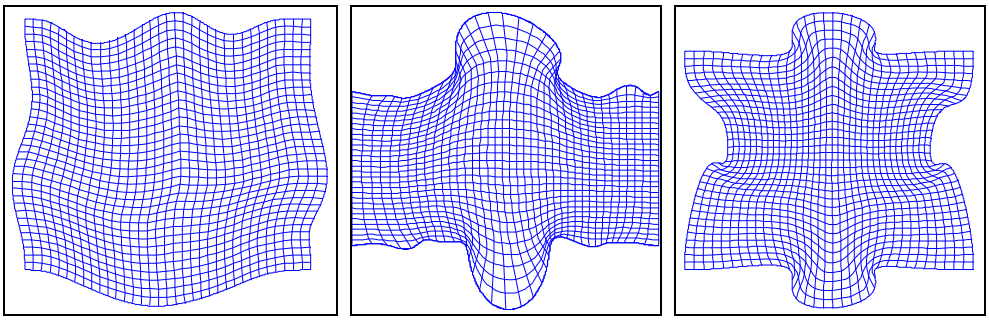


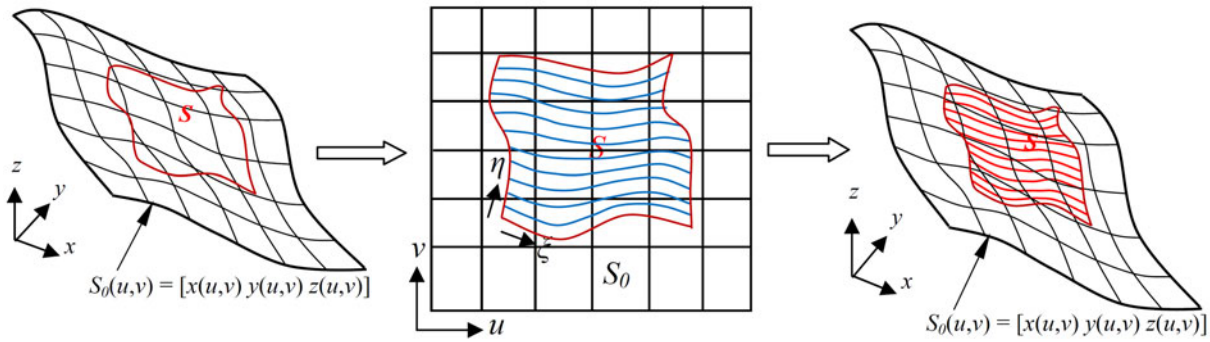
Fig. 3 Boundary interpolation method

3.1 Basic mathematics

A point on a free-form surface is described as a function of two independent variables or parameters. Therefore, a surface 'S_0' (Fig. 4) is represented in terms of two parameters (u, v) as:

$$S_0(u,v) = [x(u,v)\ \ y(u,v)\ \ z(u,v)]\ ;\ \ u,v \in [0,1] \subset R^2 \tag{3.1}$$

But as mentioned earlier, for a trimmed surface 'S' the original parameterization (on u,v) becomes invalid and so 'S' is re-parameterized onto a new unit square (computational domain) of two parameters $\xi,\eta \in [0,1]$.



(a) Original surface with trimming

(b) Mapped on (u,v) domain and re-parameterised on (ξ,η)

(c) Tool path generated on trimmed surface

Fig. 4 Scheme for iso-parametric tool path generation

Let $P(u,v)$ be a general point on the iso-η curve $B(u(\xi),v(\xi))$ embedded on the surface 'S'. The first fundamental form [18-20] at P, in the orthogonal direction of the curve B, on 'S' is defined as square of the differential arc length (quadratic form) and can be expressed as follows.

$$I = \frac{dP}{d\eta}\cdot\frac{dP}{d\eta} = E\,\frac{\partial u}{\partial \eta}\,\frac{\partial u}{\partial \eta} + 2F\,\frac{\partial u}{\partial \eta}\,\frac{\partial v}{\partial \eta} + G\,\frac{\partial v}{\partial \eta}\,\frac{\partial v}{\partial \eta} \tag{3.2}$$

where $E=P_u\cdot P_u$; $F=P_u\cdot P_v$; $G=P_v\cdot P_v$

and $P_u = \dfrac{\partial P}{\partial u}$; $P_v = \dfrac{\partial P}{\partial v}$.

The second fundamental form [18-20] is given as the quadratic form

$$II = L\,\frac{\partial u}{\partial \eta}\,\frac{\partial u}{\partial \eta} + 2M\,\frac{\partial u}{\partial \eta}\,\frac{\partial v}{\partial \eta} + N\,\frac{\partial v}{\partial \eta}\,\frac{\partial v}{\partial \eta} \tag{3.3}$$

where $L=P_{uu}\cdot$n ; $M=P_{uv}\cdot$n ; $N=P_{vv}\cdot$n

and P_{uu} ; P_{uv} ; P_{vv} are the second order partial derivatives of $P(u,v)$ w.r.t. parameters u and v.

n is the unit surface normal at $P(u,v)$ and defined as

$$\mathrm{n} = \frac{P_u \times P_v}{\left|P_u \times P_v\right|} \tag{3.4}$$

By using the above expressions the normal curvature k_n is given [20] as

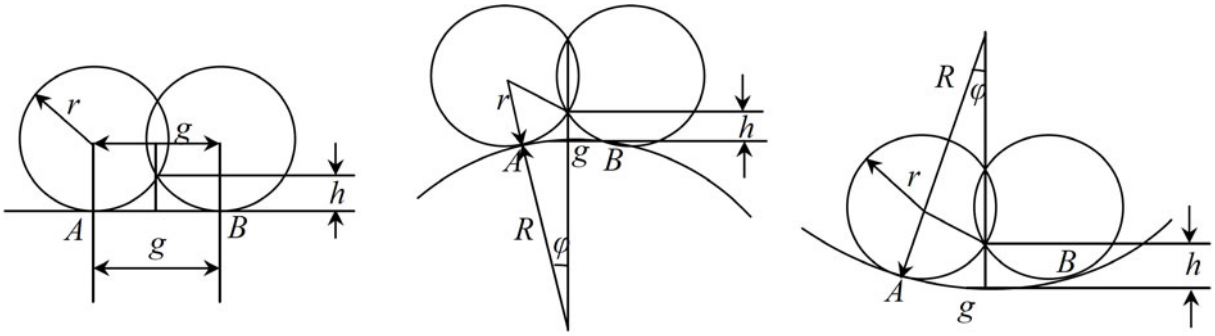
$$k_n = \frac{II}{I} \tag{3.5}$$

and therefore the radius of normal curvature is expressed as

$$R_n = \left|\frac{I}{II}\right| \tag{3.6}$$

3.2 Side-step

The maximum permissible distance between two adjacent tool paths along the direction orthogonal to the current tool path is defined as path interval or side-step (g) and is a function of the scallop height h, the tool radius r of the ball-end mill and the radius of the normal curvature R_n. g is calculated in the physical space depending on the given limiting value of h for a surface and the values of r & R_n [16].



(a) Flat profile **(b)** Convex profile **(c)** Concave profile

Fig. 5 Side-step

Referring to Fig. 5, the side-step g (AB) is calculated for the following three different profiles of surface [9].

i) For the flat profile (Fig. 5a), II=0 :

$$h = r - (r^2 - g^2/4)^{1/2} \Rightarrow g = 2(\mathrm{r}^2 - (r-h)^2)^{1/2} \tag{3.7}$$

ii) For the convex profile (Fig. 5b), II<0 :

Let φ be the half-angle subtended by R_n to cover the arc length AB which is the side-step g.

Therefore,

$$\cos(\varphi) = \frac{(R_n+h)^2+(R_n+r)^2 - r^2}{2(R_n+h)(R_n+r)} \Rightarrow \varphi = \cos^{-1}\frac{(R_n+h)^2+(R_n+r)^2 - r^2}{2(R_n+h)(R_n+r)} \tag{3.8}$$

From Fig. 5b and using Eq. 3.6, the side-step

$$g = 2R_n\varphi \tag{3.9}$$

iii) For the concave profile (Fig. 5c), $II>0$:

Therefore,

$$\cos(\varphi) = \frac{(R_n-h)^2+(R_n-r)^2-r^2}{2(R_n-h)(R_n-r)} \Rightarrow \varphi = \cos^{-1}\frac{(R_n-h)^2+(R_n-r)^2-r^2}{2(R_n-h)(R_n-r)} \tag{3.10}$$

From Fig. 5c and using Eq. 3.6, the side-step

$$g = 2R_n\varphi \tag{3.11}$$

3.3 Deriving tool paths based on side-steps

The side-step calculated above is in the physical domain instead of the computational or parametric domain. Therefore, the side-steps must be converted into the equivalent parametric interval in order to find the next tool path.

Let the current tool path be P_k (corresponding to η_k) in the direction ξ and the next tool path to be determined is P_{k+1} (corresponding to η_{k+1}). ψ is the angle between the tangent vector (T) of the tool path P_k and the tangent to a parametric curve $C(\eta)$ in the direction of the side-step.

Therefore, $$\psi = \cos^{-1}\left(\frac{\left(\frac{\partial C}{\partial u}\frac{\partial u}{\partial \eta}+\frac{\partial C}{\partial v}\frac{\partial v}{\partial \eta}\right)\cdot T}{\left|\left(\frac{\partial C}{\partial u}\frac{\partial u}{\partial \eta}+\frac{\partial C}{\partial v}\frac{\partial v}{\partial \eta}\right)\right|\cdot|T|}\right) \tag{3.12}$$

and $$g_p = g/\sin(\psi) \tag{3.13}$$

where g_p is the parametric side-step equivalent to g.

Now, expanding $C(\eta)$ with a Taylor series,

$$C(\eta) = C(\eta_k)+C'(\eta_k)\Delta\eta+(1/2)C''(\eta_k)\Delta\eta^2+(1/3!)C'''(\eta_k)\Delta\eta^3 + \ldots\ldots\ldots\ldots \tag{3.14}$$

where $\Delta\eta = \eta_{k+1} - \eta_k$

$\Delta\eta$ is the parametric increment of the side-step. Now, neglecting the higher order terms in Eqn. (3.14), the following expression for the side-step is obtained.

$$C(\eta_{k+1}) - C(\eta_k) = |C'(\eta_k)\Delta\eta+(1/2)C''(\eta_k\Delta\eta^2| = g_p \tag{3.15}$$

$$(g_p)^2 = C'(\eta_k)^2\Delta\eta^2+C'(\eta_k)C''(\eta_k)\Delta\eta^3+(1/4)C''(\eta_k)^2\Delta\eta^4 \tag{3.16}$$

Eqn. (3.16) can be solved by tedious iterative methods to find the $\Delta\eta$. Alternatively, the error-compensation method [21] can be used, which is a computationally efficient method.

The first order approximation of Eqn. (3.16)

$$\Delta\eta_a = g_p \Big/ \sqrt{C'(\eta_k)^2} \tag{3.17}$$

and the error term χ for the last two terms of Eqn. (3.16) is defined as follows:

$$\chi = C'(\eta_k)C''(\eta_k)\Delta\eta^3+(1/4)C''(\eta_k)^2\Delta\eta^4 \tag{3.18}$$

The error χ is calculated by using $\Delta\eta = \Delta\eta_a$ from Eqn. (3.17). Therefore, the parametric increment is obtained by the following equation.

$$\Delta\eta = \sqrt{(g_p - \chi)} \Big/ \sqrt{C'(\eta_k)^2} \tag{3.19}$$

To determine the next iso-parametric tool path P_{k+1} the minimum of all the $\Delta\eta$ along the current tool path P_k is taken into consideration to maintain the scallop height within the given limit and then a corresponding η_{k+1} is calculated.

Therefore,

$$\eta_{k+1} = \eta_k + \Delta\eta_{\min} \tag{3.20}$$

$$P_{k+1} = P(u(\xi, \eta_{k+1}), v(\xi, \eta_{k+1})) \tag{3.21}$$

In the derivation of the side-step, $\partial u/\partial\eta$ and $\partial v/\partial\eta$ are determined by linear interpolations as follows.

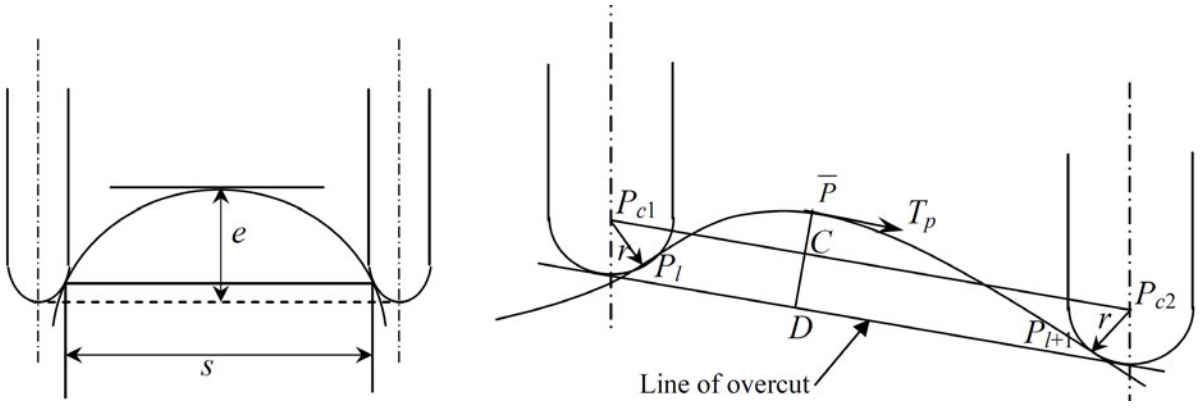
Let j-th iso-η curve be the immediate next to k-th tool path so that $\eta_k < \eta_j$. Therefore, at k-th tool path the derivatives are expressed as,

$$(\partial u/\partial\eta) = \Delta u/\Delta\eta = [u(\xi_i, \eta_j) - u(\xi_i, \eta_k)]/[\eta_j - \eta_k] \tag{3.22}$$

$$(\partial v/\partial\eta) = \Delta v/\Delta\eta = [v(\xi_i, \eta_j) - v(\xi_i, \eta_k)]/[\eta_j - \eta_k] \tag{3.23}$$

3.4 Forward-step

In order to optimize the length of tool paths the forward-step '*s*' must be maximized depending on the given tolerance '*e*' (Fig. 6a). Therefore, the forward-step is the maximum distance between two cutter contact (CC) points along a tool path, with a given tolerance. Figure 6a shows the conventional forward-step with a circular arc approximation of the tool path including overcut due to the tool nose.



(a) Conventional **(b)** Proposed

Fig. 6 Forward-step

In Fig. 6b, the actual profile of a tool path has been considered including overcut. Let the tool move from the current CC points P_l to the next CC point $P_{l+1,}$ which is required to be determined. On (k+1)-th tool path, $\bar{P}$ is a point located between P_l & P_{l+1} and T_p is the unit tangent vector in the direction of the tool path at $\bar{P}$. When T_p is parallel to $P_{c1}P_{c2,}$ the

deviational distance $\overline{P}C$ becomes maximum. So, for the optimum condition (when the forward-step length is maximum),

$$\overline{P}D = \overline{P}C\big|_{\max} + CD = \overline{P}C\big|_{\max} + r = e \;\Rightarrow\; \overline{P}C\big|_{\max} = e - r \tag{3.24}$$

The tool centre P_{c1} is given as,

$$P_{c1} = P_l + r\mathrm{n}_1 \tag{3.25}$$

where n_1 is the surface normal corresponding to P_l and r is the tool radius.

Now, to find P_{l+1}, first $\overline{P}$ is determined. Let us take a parametric point adjacent to P_l on the tool path $\overline{P}$. So, from Fig. 6b, $\overline{P}C$ is perpendicular to T_p and $P_{c1}C$ is parallel to T_p.

Therefore,

$$T_p \cdot \overline{P}C = 0 \tag{3.26}$$

$$T_p \cdot \frac{P_{c1}C}{|P_{c1}C|} = 1 \tag{3.27}$$

where $T_p = \dfrac{\dot{\overline{P}}}{\left|\dot{\overline{P}}\right|}$ (3.28)

and $\dot{\overline{P}}$ is the partial derivative of the embedded parametric tool path w.r.t. ξ at the point $\overline{P}$ (corresponding to ξ_i).

Therefore,

$$\dot{\overline{P}} = \frac{\partial P}{\partial u}\frac{\partial u}{\partial \xi} + \frac{\partial P}{\partial v}\frac{\partial v}{\partial \xi}\bigg|_{\xi = \xi_i} \tag{3.29}$$

To find u and v corresponding to ξ_i a linear interpolation is applied between two discrete points at ξ_l & ξ_{l+1} on the current tool path (i.e. (η_{k+1})-th) so that $\xi_l \le \xi_i \le \xi_{l+1}$.

$$u(\xi_i) = u(\xi_l) + [u(\xi_{l+1}) - u(\xi_l)].[(\xi_i - \xi_l)/(\xi_{l+1} - \xi_l)] \tag{3.30}$$

$$v(\xi_i) = v(\xi_l) + [v(\xi_{l+1}) - v(\xi_l)].[(\xi_i - \xi_l)/(\xi_{l+1} - \xi_l)] \tag{3.31}$$

Now, using Eqns. (3.30) and (3.31), partial derivatives w.r.t. ξ in Eqn. (3.29) can be expressed as

$$\frac{\partial u}{\partial \xi}\bigg|_{\xi = \xi_i} = [u(\xi_{l+1}) - u(\xi_i)]/[(\xi_{l+1} - \xi_i)] \tag{3.32}$$

$$\frac{\partial v}{\partial \xi}\bigg|_{\xi = \xi_i} = [v(\xi_{l+1}) - v(\xi_i)]/[(\xi_{l+1} - \xi_i)] \tag{3.33}$$

By solving Eqns. (3.26) and (3.27), C can be determined. If $\overline{PC} = \overline{P}\,C\big|_{\max}$, then $\overline{P}$ is the point for the optimum condition i.e. the forward-step is maximum. If $\overline{PC}$ is greater than $\overline{P}\,C\big|_{\max}$ then the parametric interval between P_l & $\overline{P}$ is reduced and, on the other hand, if the deviational distance is less than $\overline{P}\,C\big|_{\max}$, then the parametric interval between P_l & $\overline{P}$ increases by a small amount. This method continues until $\overline{PC}$ converges to $\overline{P}\,C\big|_{\max}$.

Now, let us consider another parametric point on the tool path adjacent to $\overline{P}$ as the next CC point $P_{l+1,}$ so that,

$$P_{c2} = P_{l+1} + r\mathrm{n}_2 \tag{3.34}$$

where n_2 is the surface normal corresponding to P_{l+1}.

If P_{l+1} is the desired CC point for the optimum condition (i.e. when the forward-step is maximum) then P_{c1}, point C and P_{c2} will be collinear, i.e. the angle (ω) between vectors $P_{c1}C$ & CP_{c2} will be 0; i.e.,

$$\omega = \cos^{-1}\left(\frac{P_{c1}C \cdot CP_{c2}}{\left|P_{c1}C\right| \cdot \left|CP_{c2}\right|}\right) = 0 \tag{3.35}$$

If $\omega \neq 0$, then the next CC point P_{l+1} should be adjusted until the condition in Eqn. (3.35) is met.

So, after the determination of the side-steps and forward-steps as above, the CC points are now completely defined for the entire surface. The CC points (P_{cc}) thus calculated have to be converted into CL (cutter location) points P_{cl} for the input to the post-processor of the CNC machine. For the ball-end cutter the centre of the ball-end is taken as CL point. Hence, P_{cl} is given as,

$$P_{cl} = P_{cc} + r\mathrm{n} \tag{3.34}$$

where n is the surface normal.

4. Implementation of the algorithm in case studies

The tool path generation scheme presented in section 3 has been applied to two cases as shown in Fig. 7. The radius of the ball-end cutter is 5mm and both the scallop height and the specified tolerance are 0.5mm. For Case-I with simple boundaries, both methods have successfully produced tool paths, though the boundary interpolation method has more evenly distributed tool paths. In Case-II, the boundaries are irregular in shape. Due to this, the Laplace PDE produces unevenly distributed iso-parametric curves near the irregular boundaries. This causes tool paths to become crowded near the irregular boundaries. The problem of crowdedness of the tool paths is eased by applying the boundary interpolation technique on the same surfaces in Case-II. The evenness in the distribution of tool paths determines the efficiency of machining in terms of tool path length (PL) and computational time (CT). Table 1 shows a comparison of efficiencies of machining for the two different parameterization methods for Case-I and Case-II. *MATLAB7* was used for coding and plotting the results.

Table 1 Comparison between two methods in terms of path length (PL) and computational time (CT)

Methods	Criteria	Case-I	Case-II
Laplace PDE	PL (mm)	9329	9532
	CT (s)	566	617
Boundary Interpolation	PL (mm)	9234	9379
	CT (s)	412	461

To find the effect on the tolerance of the proposed method, Case-I was tested once again by using the boundary interpolation re-parameterization. In Figs. 8(a) and (b), which show the tool path generated by the conventional and by the proposed method, respectively, the fourth tool path was taken into consideration (marked with dots on the firm line) to calculate the tolerances. The values were plotted in Fig. 9, where it is seen that the proposed method is capable of keeping the tolerances very close to the given value of tolerance, i.e. 0.5mm, but in the case of the conventional method the tolerances deviate far from the given value. A very small amount of deviations (which may be ignored practically) in tolerance that were observed in the proposed method is a result of computational approximations.

5. Concluding remarks

In this study, the trimmed free-form surfaces have been re-parameterized by the Laplace PDE and boundary interpolation methods. Re-parameterization is necessary as the original parameterization becomes invalid for trimmed surfaces. By the process of re-parameterization a number of iso-parametric curves are produced to cover up the entire surface. Based on these iso-parametric curves side-steps are calculated considering the convex and the concave profile of the surface besides the flat region. Overcut due to the tool's nose and the actual geometric profile of each embedded tool path rather than the circular arc approximation have been taken into account to get a better computational accuracy in forward-steps. Unlike the conventional tool path generation method, this new method ensures that the tolerance of the forward-step is maintained very close to the given value of tolerance over the entire surface.

The method has been implemented in some cases. Case studies show that the method is capable of producing better tolerance than the conventional method. From the results of the examination of the cases it is also found that the Laplace PDE method is computationally very expensive and also produces longer tool paths. On the other hand, the boundary interpolation technique is capable of producing reasonably good parameterization without any anomaly and a resulting efficient tool path for machining trimmed free-form surfaces.

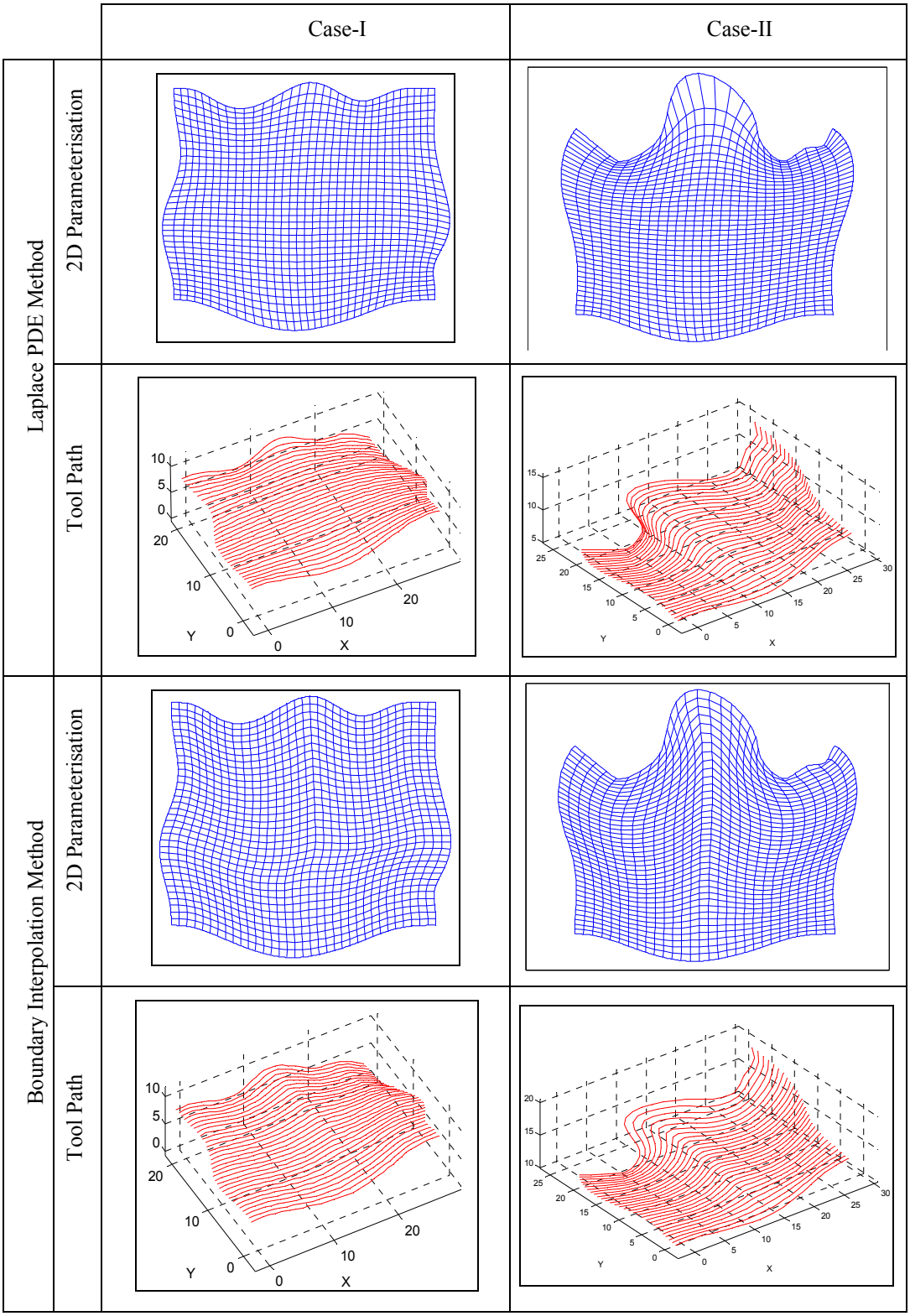
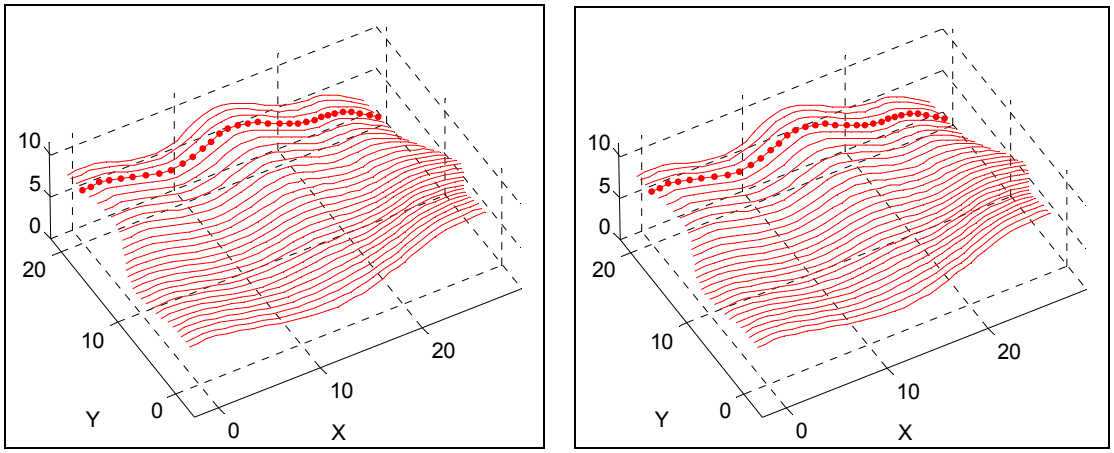


Fig. 7 Comparison between tool paths generated by different parameterization methods



(a) With conventional forward-steps **(b)** With improved method of forward-steps

Fig. 8 Tool path generated based on two different forward-step conditions using boundary interpolation re-parameterization

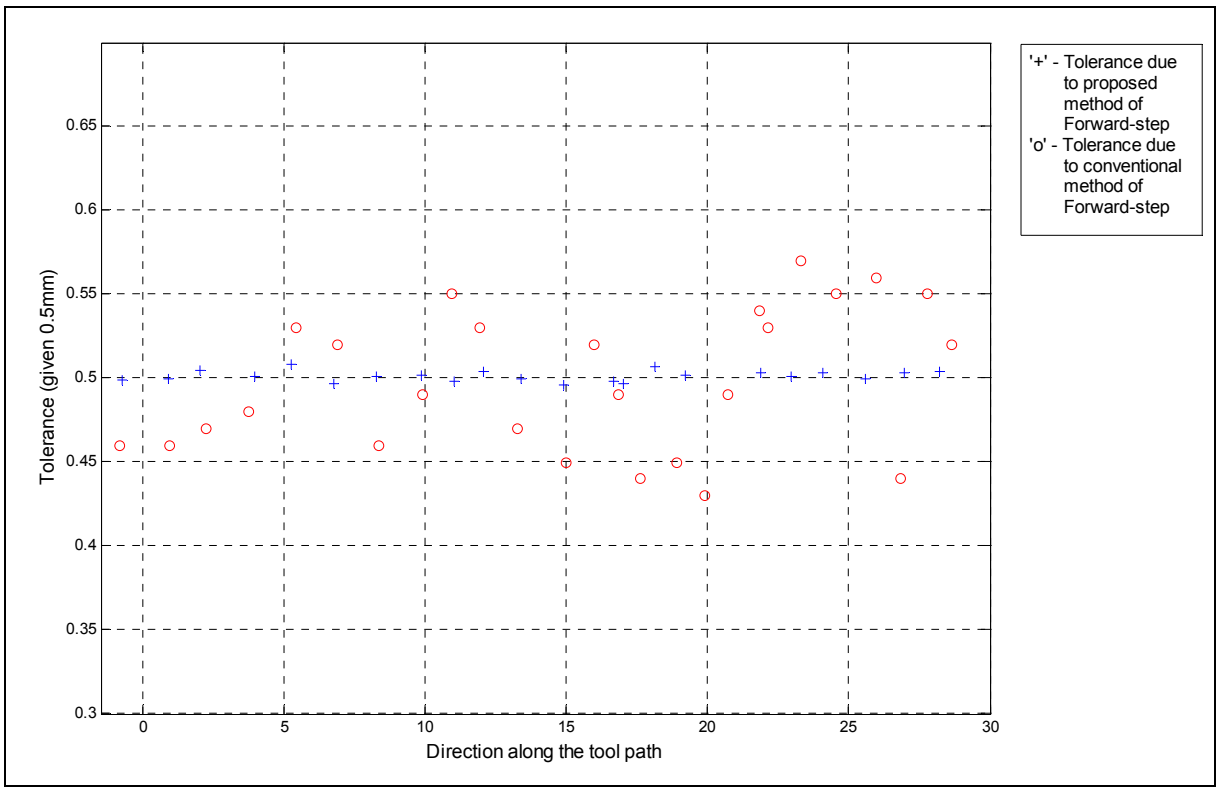


Fig. 9 Distribution of tolerances determined by two different forward-step conditions

REFERENCES

[1] Suresh K, Yang D: *Constant scallop-height machining of free-form Surfaces*. ASME J. Eng. Ind., 116(2), 253–259, 1994.

[2] Ding S, Yang D C H, Han Z: *Boundary-conformed machining of turbine blades*. Proc. IMechE, Part B: J. Engineering Manufacture, 219, 255-263, 2005.

[3] George K K, Babu N R: *On the effective tool path planning algorithms for sculptured surface manufacture*. Computers Ind. Engineering, 28(4), 823 – 836, 1995.

[4] Huang Y, Oliver J H: *Non-Constant Parameter NC Tool Path Generation of Sculptured Surfaces*. Int. J. Adv. Manuf. Technology 9, 281-290, 1994.

[5] Li C L: *A geometric approach to boundary-conformed tool path generation*. Computer-Aided Design, 39, 941-952, 2007.

[6] Li H, Feng H Y: *Constant scallop-height tool path generation for three axis sculpture surface machining*. Computer-Aided Design, 34, 647-654, 2002.

[7] Loney G C, Ozsoy T M: *NC Machining of Free Form Surfaces*. Computer Aided Design, 19(2), 85-90, 1987.

[8] Zhang L P, Nee A Y C, Fuh J Y H: *An efficient CC-curve tool path regeneration algorithm for sculptured surface machining*. Proc. of IMechE, Part B: J. Engineering Manufacture, 218, 389-402, 2004.

[9] Sarkar S, Dey P P: *Tool path generation for algebraically parameterized surface*. J Intell Manuf doi: 10.1007/s10845-013-0799-x, 2013.

[10] Sarkar S, Dey P P: *A new boundary interpolation technique for parameterisation of planar surfaces with four arbitrary boundary curves*. Proc. of the IMechE, Part C: J. Mechanical Engineering Science, 227, 1280-1290, 2013.

[11] Hsu K, Lee S L: *A numerical technique for two-dimensional grid generation with grid control at all of the boundaries*. J. Comput.Phys., 96(2), 451–469, 1992.

[12] Westgaard G, Nowacki H: *Construction of fair surfaces over irregular meshes*. ASME, J. Mech. Des., 1, 376-384, 2001.

[13] Thompson J F, Warsi Z U A, Mastin C W: *Numerical Grid Generation, Foundations and Applications.* Elsevier, North Holland, 1985.

[14] Hoffmann K A, Chiang S T: *Computational Fluid Dynamics for Engineers*. Engineering Education System, USA, 1989.

[15] Vladimir D L: *Grid Generation Methods*. Springer, New York, 2010.

[16] Sarkar S, Dey P P: *A new Iso-parametric machining algorithm for free-form surface*. Proc. of the IMechE, Part E: J. Process Mechanical Engineering, DOI: 10.1177/0954408913495191, 2013.

[17] Sarkar S, Dey P P: *Construction of 2-D parametric surfaces bounded with four irregular curves*. Int. J. Computer Aided Engineering and Technology, 4(5): 474–487, 2012.

[18] Yamaguchi F: *Curves and Surfaces in Computer Aided Geometric Design*. Springer-Verlag, London, 1988.

[19] Faux I D, Pratt M J: *Computational Geometry for Design and Manufacture*. Ellis Horwood, Chichester, 1979.

[20] Stoker J J: *Differential Geometry*. Wiley-Interscience, New York London Sydney Toronto, 1989.

[21] Lin R S, Koren Y: *Efficient tool-path planning for machining free-from surfaces*. Transactions of the ASME, 118: 20–28, 1996.

Submitted: 13.8.2013

Accepted: 04.3.2015

Subhajit Sarkar
Partha Pratim Dey
Department of Mechanical Engineering
Indian Institute of Engineering Science and Technology (*Formerly Bengal Engineering and Science University*), Shibpur, Howrah-711103
West Bengal, India