



# A double-NURBS approach to the generation of trochoidal tool path

Chia-Hao Chang<sup>1</sup> · Ming Huang<sup>2</sup> · Hong-Tzong Yau<sup>1</sup>

Received: 25 May 2022 / Accepted: 23 November 2022 / Published online: 14 January 2023  
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

## Abstract

Trochoidal milling becomes a popular strategy in high-speed rough milling, thanks to its advantages in lowering cutting force (and tool wear) and increasing tool life. However, it also suffers from the disadvantages of low cutting efficiency and prolonged machining time due to its reduced cutter-workpiece engagement in the cutting area and the idle tool path in the non-cutting area. In this paper, a new trochoidal tool path generation method is proposed based on a double-NURBS curve. Unlike traditional circular trochoidal tool path that combines circular and linear paths to approximate the trochoidal path, the double-NURBS tool path can truly represent a trochoidal path with  $C^2$  continuity throughout the entire tool path. In the front cutting area, the control points of the NURBS are modified to increase the cutter-workpiece engagement angles; in the rear non-active area, the NURBS curve is modified to shorten the non-cutting distance and the machining time. Degree elevation of the NURBS circle from 2 to 3 maintains  $C^2$  continuity to ensure the machining is smooth throughout all transitions. Cutting force simulations and pocket machining experiments are carried out to validate that the proposed double-NURBS approach deliver the expected result and performance.

**Keywords** Trochoidal milling · NURBS · Cutting force · High-speed machining · Machining efficiency

## 1 Introduction

Rough machining process is a key process in high-speed CNC machining to remove large portion of materials. Its purpose is to quickly cut the workpiece to a specified shape. In choosing rough machining strategy, traditional method of the zig-zag or the contour parallel processing method by full immersion tend to cause large cutting forces and tool wear in marching. Trochoidal milling [1] becomes the popular strategy in the rough milling process, owing to its advantages in low cutting force, low tool wear, and high cutting efficiency. In trochoidal milling processing, as shown in Fig. 1, the tool path is composed of a circular and a linear motion which is called the circle based pattern. Each trochoidal cycle has two sections: active section and non-active section. In the

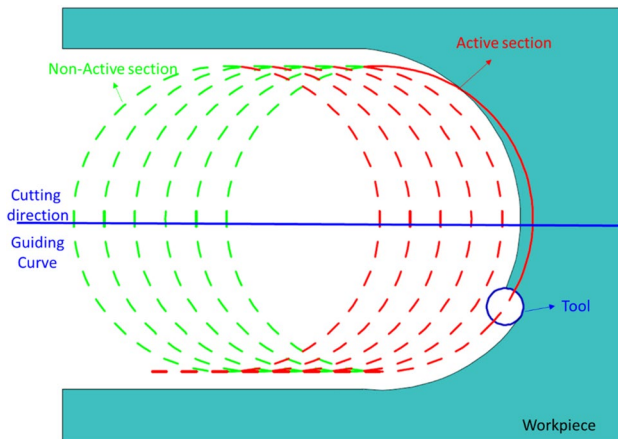
active section, the cutter contacts with the workpiece which avoids full immersion to reduce cutting force and increase heat dissipation. The non-active section connects each active section with no cutting force loading in this area. When the non-active section path is too long, it will reduce the cutting efficiency. The important issue in trochoidal milling is to improve the cutting effectiveness while reducing the path in non-active cutting section.

In the past, a trochoidal milling path is generated by the circular mathematical model, and each circular path is connected by a straight line to reduce the computation loading. As a result, the path has only the  $C^0$  continuity in the circular based trochoidal milling. Ibaraki et al. [2] proposed the use of straight lines for connection in the non-active area to improve cutting efficiency, but the path remains with  $C^0$  continuity. Otkur and Lazoglu [1] analyzed the mathematical model of the cutter workpiece contact area, estimated the engagement angle, and proposed a dual trochoidal tool path to improve the cutting efficiency. Rauch et al. [3] proposed the comparison of two kinds of trochoidal path generation methods and analyzed the force variations and tool parameters during machining. Their research also came up with a solution to deal with the corner and entry point issues of trochoidal machining.

✉ Hong-Tzong Yau  
imehty@gmail.com

<sup>1</sup> Department of Mechanical Engineering and Advanced Institute of Manufacturing with High Innovation, National Chung Cheng University, 168, University Rd., Ming-Hsiung, Chia-Yi, Taiwan, ROC

<sup>2</sup> Department of Mechanical Engineering, University of San Diego, San Diego, CA, USA



**Fig. 1** The area of trochoidal tool path

Because trochoidal machining path contains cutting and non-cutting portions, Ferreira and Ochoa [4] proposed an optimization algorithm to shorten the non-cutting path and processing time. Li et al. [5] proposed a spline-based new pattern trochoidal tool path in slotting process. They divided trochoidal path into active and non-active area, and using the Hermite curve to connect each active area to achieve  $C^1$  continuity. Huang et al. [6] proposed the new pattern trochoidal tool path based on an ellipse model and medial axis transformation for pocketing processing. They improved cutting efficiency by keeping the engagement angle in the active area. The above-mentioned two methods tried to increase the engagement angles in the cutting area by using parametric curves to replace traditional circular pattern. However, the tradeoff is the cost to increase computation load for the machine controller to process the many tiny segments interpolated by the spline curves.

For the generation of the guiding curve of the milling process, the medial axis can be used as a topological skeleton which allows the maximum allowable internal sphere rolling along the interior of the object. Some literatures have integrated the trochoidal path and medial axis theory for machining application. Yamaji et al. [7] found that the overloading path and medial axis coincide during traditional machining and they proposed a mixed machining strategy by combining trochoidal and conventional machining methods. Since the trochoidal path is the trajectory of a fixed point on a circle which rolls along its guiding curve, the medial axis can serve conveniently as the guiding curve. Traditional medial axis calculation is very time consuming and inefficient due to its nonlinear nature of solutions. Instead we applied a direct method, which did not need to solve nonlinear equations, to obtain an approximate medial axis based on an offset method in a computational geometry theorem [8] and the polar ball theory based on the characteristics of the graphics [9].

In practical applications, traditional CNC only offers straight line and circle interpolation operations. Hence, the circular type trochoidal path can utilize G1 and G2/G3 codes quite efficiently. However, if we want to increase the engagement angles by modifying the front-end cutting shape, such as the elliptical shape using a Hermite curve [6], the parametric curve needs to be broken down into a large number of short line segments. This could cause several drawbacks: (1) Velocity and acceleration discontinuities will result in variation of the feed rate in the two-segment junction points. (2) Acceleration discontinuity will result in machine vibration and reduce machine and tool life. (3) Massive NC data transfer may cause CNC system overloading. All of the above stifle the performance of high speed machining. To overcome these problems, many literatures in the past solved these issues by using a parametric curve interpolation algorithm known as Non-Uniform Rational B-Splines (NURBS) interpolation. Commercial CNC companies [10], [11] also integrated this technology into their controller products. It is interesting to know that with the availability of NURBS interpolation already adopted by modern controllers, there seems to find no research work in the past to use NURBS to represent a trochoidal tool path. Unlike cubic splines or Hermite curves used by [5] and [6], a NURBS curve can represent an exact circle and its shape can be modified by moving the control points. Traditional NURBS curve typically uses 9 control points with degree 2 to represent a full circle. Degree elevation can be used to increase its degree to 3 to maintain the  $C^2$  continuity. By combining two degree-3 NURBS curves (one to represent the medial axis and the other one the circular path), we can generate a doubleNURBS trochoidal path to maintain  $C^2$  continuity throughout the tool path, still with the full flexibility to modify the shape of the tool path. With modern controllers capable of processing NURBS, many research works have presented feed rate scheduling and interpolation schemes for NURBS-based tool paths. Shpitalni et al. [12] proposed their new CNC interpolator, which can handle general curves by reducing the order of the path interpolation algorithm. However, in NURBS interpolation, a fixed feed rate may lead to a great chord error in the sharp corners. Yeh et al. [13] proposed an interpolation algorithm that confines the chord error within a specific tolerance range during the process. The change in feed rate is based on the chord error and radius of the curvature. Farouki and Tsai [14] modified the Taylor second-order approximation interpolation and compared the curve accuracy at different orders of approximation. Xu et al. [15] proposed an adaptive feed rate algorithm by referencing the curvature of a path. Tsai and Cheng [16] presented a predictor–corrector interpolator algorithm to control the variation in feed rate and verify the convergences.

This paper proposes a new trochoidal tool path generation method based on double-NURBS curve. Unlike traditional circular trochoidal tool path that combines circular and linear paths to approximate the trochoidal path, the double-NURBS

tool path can truly represent a trochoidal path with  $C^2$  continuity throughout the entire tool path. In the front-end cutting area, the control points of the NURBS can be modified to increase the cutter-workpiece engagement angles; in the back-end non-active area, the NURBS curve can be modified to shorten the non-cutting distance and the machining time. With the existing feed rate scheduling and interpolation methods that have been developed and integrated with controllers, trochoidal tool path process parameters such as maximum cutting force (related to maximum engagement angle) and feed rate can be fully implemented using our proposed double NURBS approach.

Compared with traditional trochoidal tool paths, the proposed approach has three distinct features:

Use the Voronoi diagram and curve fitting method to generate the medial axis of the pocketing contours and the proposed trochoidal tool path by a degree-3 double-NURBS curve to achieve  $C^2$  continuity.

Enlarge and extend the engagement angles in the cutting area by modifying the front-end NURBS curve to improve the cutting productivity.

Modify the control points of the NURBS curve in the non-active area to shorten the non-cutting distance and improve the machining efficiency.

The remainder of this paper is organized as follows. The double-NURBS trochoidal tool path generation is developed in Section 2, including the guiding curve and NURBS curve. In Section 3, we analyze the engagement angle and optimize the NURBS curve in the active and non-active areas to improve cutting efficiency. Section 4 demonstrates pocketing machining with the double-NURBS algorithm, including tool path generation and cutting force analysis. Finally, in Section 5, contributions of this work are summarized and future research directions are also given.

## 2 Double NURBS-based trochoidal toolpath generation

In mathematics, the trochoidal path can be described by a circle rolling from point A to B along a guiding curve, typically a straight line. A point on the circumference will form a continuous path in the space in the rolling process and is called a trochoidal path as shown in Fig. 2.

The mathematical equation of a trochoidal path can be defined as [3]:

$$\begin{cases} X = X_i + v_{xi}S_{tr}\frac{\theta}{2\pi} + R_{trocho}\cos\theta \\ Y = Y_i + v_{yi}S_{tr}\frac{\theta}{2\pi} + R_{trocho}\sin\theta \end{cases} \quad (1)$$

As shown in Fig. 3, the corresponding parameters are: the center of the trochoidal path  $O_i(X_i, Y_i)$ , direction unit vector of center on the guide curve  $v_i(v_{xi}, v_{yi})$ , the radius of the

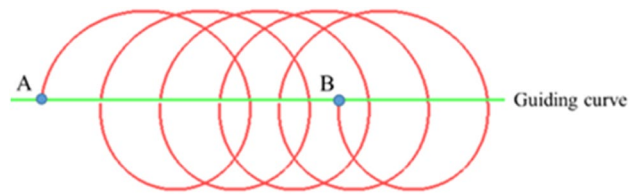


Fig. 2 Trochoidal curve

trochoidal path ( $R_{trocho}$ ), step length ( $S_{tr}$ ) and rotating angle ( $\theta$ ). Also in the figure,  $R_{tool}$  is the radius of the cutting tool and  $R_{dc}$  is the radial depth of cut.

The processing path calculated by a trochoidal algorithm has  $C^2$  continuity; this will be helpful for the milling machine due to a smoother path. Note that the radial depth of cut  $R_{dc}$  varies as angle changes and needs to be constrained to avoid the occurrence of full immersion ( $R_{dc} = R_{tool}$ ) during the roughing process. Using analytic geometry,  $R_{dc}$  can be calculated once the tool center and the two end engagement points defining the engagement angle are found (see [6, 17]) and thus its maximum value  $R_{dcmax}$  can be obtained. This, however, requires a numerical solution process which could be tedious to implement [17].

Alternatively, a simpler approximation can be used for  $R_{dcmax}$  based on a circular trochoidal-like model, which essentially connects two circles with a short linear segment. Rauch et al. [3] developed a closed-form expression of  $R_{dcmax}$  and established its validity as a conservative estimate for the true trochoidal  $R_{dcmax}$ . Such a circular trochoidal-like path is very easy to implement and hence, it is widely applied to existing CAM software. However, since this circle-based path is not really a trochoidal path, it still suffers a discontinuous acceleration problem in the junction point of the circle and the linear segment. Note that, as suggested in [3], we used this circular model based results (Eqs. (2) and (3) below) only for the purpose of effective parameter selection to aid in the trochoidal path planning.

Equation (2) shows the maximum radial depth of cut, based on the circular model [3], in relation to the tool radius, the trochoidal path radius, and the step length:

$$R_{dcmax} = \left( 1 + \frac{R_{tool}}{R_{trocho}} \right) S_{tr} - \frac{1}{2R_{trocho}} S_{tr}^2 \quad (2)$$

If the maximum depth of cut is specified, then the corresponding maximum step length  $S_{tr}$  can be obtained:

$$S_{tr} = R_{trocho} + R_{tool} - \sqrt{(R_{trocho} + R_{tool})^2 - 2R_{trocho}R_{dcmax}} \quad (3)$$

For completeness, the detail derivation of Eqs. (2)–(3) is given in Appendix A.

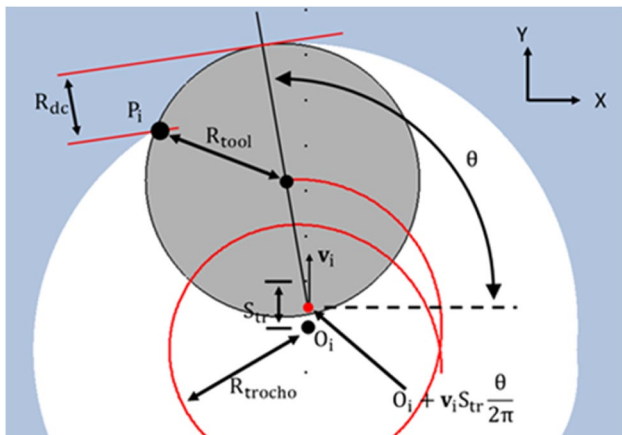


Fig. 3 Geometry of the trochoidal machining path [3]

Since the trochoidal tool path moves forward along its guiding curve, Eq. (3) can be readily used to set the maximum step length to generate a processing path, as shown in Fig. 4. Moreover, trochoidal machining paths can be integrated with traditional methods since they can avoid the full immersion condition of a cutter and ensure a secure entry point.

In practical CNC implementation, a trochoidal tool path must be split into a large number of short line segments and increase the loading of the controller during data importing. NURBS has the advantages of great flexibility and precision for modeling and has been integrated with many commercial CNC controllers. Since a degree-3 NURBS curve provides exact  $C^2$  continuity, in the following, we develop and redefine the trochoidal tool path along with its medial axis guiding curve (see Fig. 5) using a double-NURBS curve. In addition to its advantage of continuity, the NURBS-based trochoidal computation architecture can be integrated directly into a PC-based controller and enhance the value of the controller.

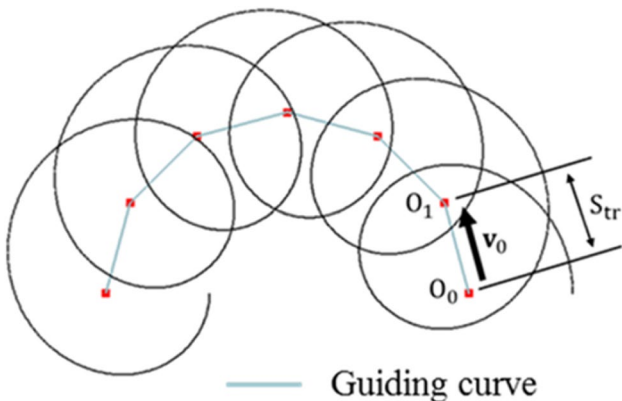


Fig. 4 Trochoidal tool path with arbitrary guiding curve

## 2.1 NURBS-based trochoidal path generation

In this sub-section, the equation of a trochoidal path using a double-NURBS curve will be presented. Since a trochoidal path is the trajectory of a fixed point on a circle which rolls along its guiding curve, we begin with using the NURBS curve [18] to define a circle which is formulated in Eq. (4):

$$C(t) = \frac{\sum_{j=0}^n N_{j,p}(t)w_j \vec{P}_j}{\sum_{j=0}^n N_{j,p}(t)w_j} \quad (4)$$

where  $N_{j,p}(t)$  is the basis function of degree- $p$  shown in Eqs. (5) and (6)  $P_j$  is the control points,  $w_j$  represents the weighting value and  $t$  is the parameter.  $U\{t_0, \dots, t_m\}$  is the knot sequence with the relation  $m = n + p + 1$ .

$$N_{j,0}(t) = \begin{cases} 1, & t_j \leq t \leq t_{j+1} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

$$N_{j,p}(t) = \frac{t - t_j}{t_{j+p} - t_j} N_{j,p-1}(t) + \frac{t_{j+p+1} - t}{t_{j+p+1} - t_{j+1}} N_{j+1,p-1}(t) \quad (6)$$

From [18], we see that a circle is a special case for a NURBS curve and can be established by 9 control points and specific weights and a knot vector. The detailed settings of a NURBS-based circle are listed as follows: degree-2, knot sequence  $U = \{0, 0, 0, 1/4, 1/4, 1/2, 1/2, 3/4, 3/4, 1, 1, 1\}$ , weights  $w_j = \{1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1\}$ , and radius  $R$ . Figure 6 shows the distribution of the 9 control points.

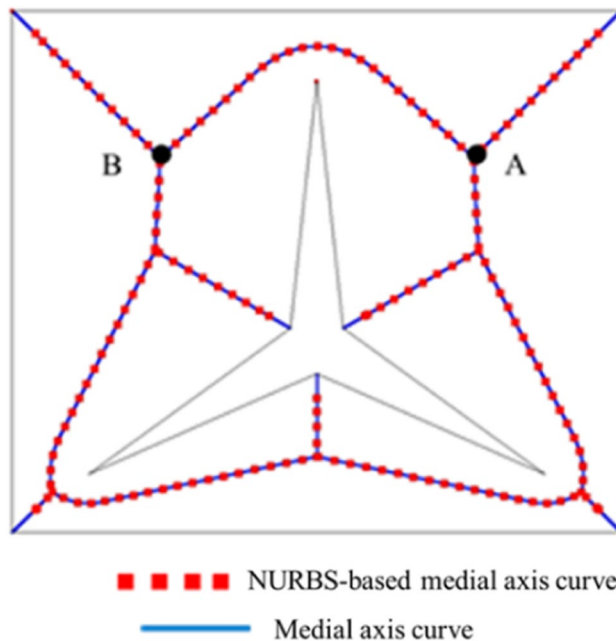


Fig. 5 The demo of approximating NURBS-based medial axis path

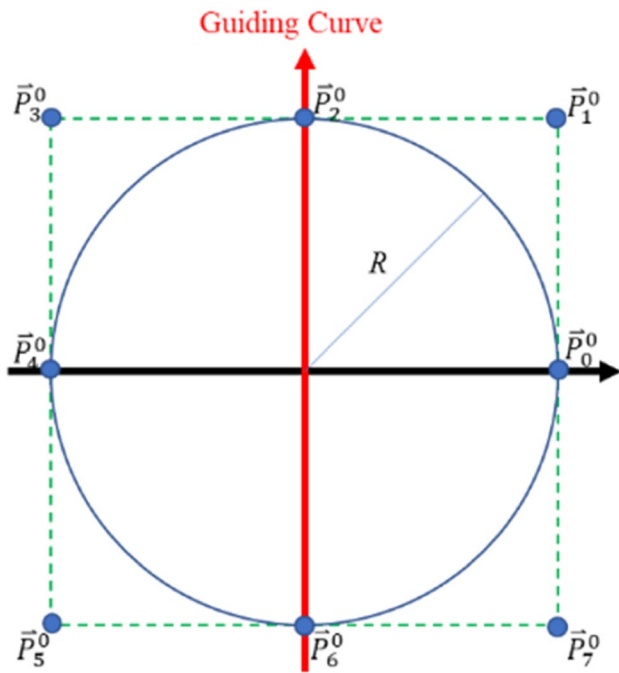


Fig. 6 Control point distribution of NURBS-based circle

Theoretically, the modeling of a circle using NURBS formulation can use either 9 control points or 6 control points (a triangle based control hull). Since our purpose is to create a NURBS parametric curve that represents a trochoidal circular path but also can be modified to extend the engagement angle (see Section 3), more control points will have more flexibility to alter or modify the shape. Therefore, in this work, we chose to use the NURBS circle with 9 control points rather than 6. We use de Boor algorithm to evaluate the NURBS curve which can be processed in parallel using FPGA or GPU (see [19]); hence, the computation load is almost equivalent even with more number of control points.

Figure 7 demonstrates the size variation of a NURBS-based circle when the radius changes along the guiding curve. By combining Eqs. (1) and (4), the NURBS-based trochoidal path can be written as:

$$t = \frac{\theta}{2\pi} - \text{int}\left(\frac{\theta}{2\pi}\right), t \in [0, 1] \tag{7}$$

$$T(u, t) = O_i + v_i S_{tr} u(\theta) + \frac{\sum_{j=0}^8 N_{j,2}(t) w_j \vec{P}_j}{\sum_{j=0}^8 N_{j,2}(t) w_j}$$

Equation (7) represents the trochoidal path of each step distance  $S_{tr}$  for any given center position. In other words, using this parametric NURBS equation, the complete set of trochoidal paths can be generated with a change of the circle center, which moves along the guiding curve. However, the connection between the trochoidal NURBS curve and the guiding NURBS curve has yet to be established. In

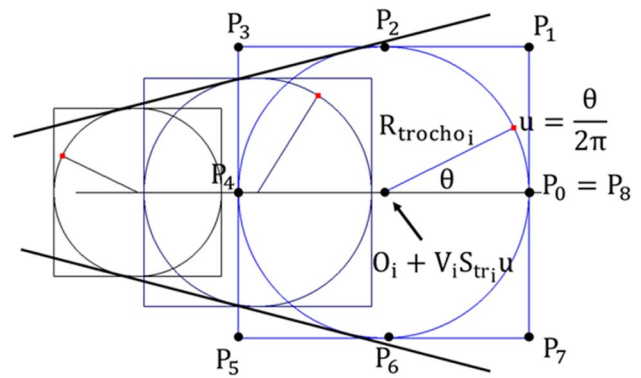


Fig. 7 NURBS-based circle changes depend on radius

Section 2.2, the equation of a double NURBS-based trochoidal machining path will be developed.

### 2.2 Double NURBS-based trochoidal path

The proposed double NURBS-based trochoidal path  $T(u)$  is a combination of two NURBS curves and is shown in Eq. (8):

$$T(u) = T(u, t(u)) = G(u) + C(t) \tag{8}$$

The two NURBS curves are guiding curve  $G(u)$  and rolling path  $C(t)$ . As mentioned in the above section,  $C(t)$  is a circle modeled by the NURBS curve and  $G(u)$  is an approximate medial axis path by the curve fitting method [22]:

$$G(u) = \frac{\sum_{i=0}^n N_{i,3}(u) w_i^G \vec{Q}_i}{\sum_{i=0}^n N_{i,3}(u) w_i^G} \tag{9}$$

$$C(t) = \frac{\sum_{j=0}^n N_{j,p}(t) w_j \vec{P}_j}{\sum_{j=0}^n N_{j,p}(t) w_j} \tag{10}$$

In Eq. (9),  $N_{i,3}(u)$  is the basis function of the B-spline and  $Q_i$  represents the control points of the NURBS-based medial axis path.  $w_i^G$  is the weighting of NURBS  $G(u)$  and  $u$  is the parameter. An approximate NURBS-based medial axis path is shown in Fig. 5.

In constructing the guiding curve from the discrete points of the medial axis, in addition to a NURBS curve, there could be other considerations of using linear G01 line segments, B-Splines, or Pythagorean Hodograph (PH) curve [20, 21] that has the advantage of providing precise interpolation positions based on variant feed rates. Since our goal is to convert the trochoidal circular path into parametric curves for machining, only NURBS can represent an exact circular path efficiently and directly without complex transformation. It would be natural and reasonable to convert the discrete points

of the medial axis to a NURBS based guiding curve so that the complete tool path representation is unified (and simplified).

The trochoidal radius  $R_{tro}$  is a function of  $u$  and varies depending on its center location. The step distance, given by Eq. (3), is also a function of  $u$  and is modified as follows:

$$S_{tr}(u) = R_{tro}(u) + R_{tool} - \sqrt{(R_{tro}(u) + R_{tool})^2 - 2R_{tro}(u)R_{dc_{max}}} \quad (11)$$

Next the synchronization of  $G(u)$  and  $C(t)$  is addressed; that is, it is necessary to relate  $u$  and  $t$  so that the full trochoidal path can be generated with a single parameter. The double NURBS-based trochoidal tool path is the combination of a NURBS-based medial axis path  $G(u)$  and a rolling path  $C(t)$ . For a trochoidal-based path, a complete trochoidal path cycle could be achieved for each step distance moving forward. The exact step distance value on a guiding curve can be evaluated by Eq. (11). Here we assume that there is a normalized trochoidal parameter  $\hat{u}_k \in [0, 1]$  on  $G(u)$ . The next parameter  $\hat{u}_{k+1}$  can be calculated by second order Taylor approximation interpolation algorithm [14] and shown in Eq. (12):

$$\hat{u}_{k+1} = \hat{u}_k + \frac{S_{tr}(\hat{u}_k)}{\|G'(\hat{u}_k)\|} + \frac{A(\hat{u}_k)}{\|G'(\hat{u}_k)\|} \cdot \frac{T_s^2}{2} - \frac{G'(\hat{u}_k) \cdot G''(\hat{u}_k)}{\|G'(\hat{u}_k)\|^4} \cdot \frac{S_{tr}(\hat{u}_k)^2}{2} \quad (12)$$

$A(\hat{u}_k)$  is the acceleration and  $G'(\hat{u}_k)$  and  $G''(\hat{u}_k)$  are the first and second derivatives of  $G(u)$ , respectively. Here,  $A(\hat{u}_k)$  can be taken to be zero assuming uniform feed rate along the curve and  $S_{tr}(\hat{u}_k)$  is the step distance, and Eq. (13) then simplifies to:

$$\hat{u}_{k+1} = \hat{u}_k + \frac{S_{tr}(\hat{u}_k)}{\|G'(\hat{u}_k)\|} - \frac{G'(\hat{u}_k) \cdot G''(\hat{u}_k)}{\|G'(\hat{u}_k)\|^4} \cdot \frac{S_{tr}(\hat{u}_k)^2}{2} \quad (13)$$

We normalize  $u$  as  $\hat{u}$  to be between  $[0, 1]$  and the parameter  $\hat{u}_{k+1}$  at the start of the next trochoidal cycle of the NURBS-based guiding curve can be obtained by second order Taylor approximation interpolation. This means the guiding curve step distance will also depend on the varying of the trochoidal radius. Figure 8 shows the relationship between the trochoidal radius  $R_{tro}(\hat{u}_k)$  change and the step distance  $S_{tr}(\hat{u}_k)$ .

After the trochoidal parameter is defined, the connection between  $u$  and  $t$  can also be established readily as follows:

$$t(u) = \frac{u - \hat{u}_k}{\hat{u}_{k+1} - \hat{u}_k} \quad (14)$$

Combining Eq. (8), (11), (13), (14), the two NURBS-based curves can then be combined and rewritten as:

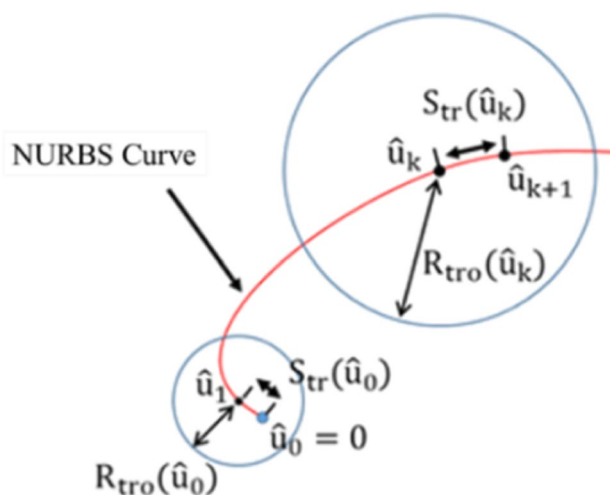


Fig. 8 The relation between  $R_{tro}(\hat{u}_k)$  and  $S_{tr}(\hat{u}_k)$

$$T(u) = \frac{\sum_{i=0}^n N_{i,p}(u)w_i^G \vec{Q}_i}{\sum_{i=0}^n N_{i,p}(u)w_i^G} + \frac{\sum_{j=0}^8 N_{j,2}(t(u))w_j \vec{P}_j}{\sum_{j=0}^8 N_{j,2}(t(u))w_j} \quad (15)$$

Figure 8 shows the machining tool path generated by the double NURBS-based trochoidal generation algorithm using Eq. (15).

In constructing the guiding curve using the medial axis, it is worth mentioning the method we used to construct the discrete medial axis. Early works of Held [23] and Lee [24] show that given sufficient sample points from the boundaries of an enclosed shape, the Voronoi vertices approximate the medial axis. Therefore, the method we used to obtain the medial axis is by using the discrete sample points of the boundary curves to obtain the dual Voronoi vertices after we applied the Delaunay triangulation. These Voronoi vertices serve as the approximation of the discrete medial axis. However, not all Voronoi vertices lie on the interior medial axis which is what we need. Amenta, et al. [25] proposed the medial axis transform (MAT) by using the Delaunay triangulation to calculate not only the Voronoi vertices but also separate them into inner and outer poles, both for 3D and 2D. Following Amenta’s work, we showed that by using only the outer poles, we can sculpt out the concave regions of the triangulated surface from the sample points [26]. With further refinement, the method can be extended to reconstruct surface with sharp edges [27]. But in this work, by using the same method to obtain the inner poles (the Voronoi vertices inside the machined region), not only we can use them to approximate the medial axis for the guiding curve, we can also use the polar radius in 2D to obtain the radius of the NURBS circle for the trochoidal circular path.

### 3 Improve the trochoidal tool path

In Section 2, we develop a new trochoidal tool path generation algorithm, based on a double-NURBS curve, for the  $C^1$  continuity. The processing efficiency is the same with the circle-based trochoidal tool path. To improve the efficiency, we will deform the original circular NURBS curve to have more engagement angles in the cutting area, and shorten the travel distance in the non-active area. To achieve this goal while still maintaining the  $C^2$  continuity, we will elevate the degree of the original NURBS circle from 2 to 3, so that not only more control points can be used to deform the shape but also help reducing unnecessary machine vibration by keeping the  $C^2$  continuity.

#### 3.1 Degree elevation of the circle

In the last section, we use the degree-2 NURBS to represent the trochoidal circle. The curve has  $C^1$  continuity. In the path planning, it leads to the jerk discontinuity in the jerk and increases the cutting force in the milling process. To improve this issue, we use degree elevation to raise degree to 3 without changing the shape of the circle, as is shown in Fig. 9.

We show the quadrant of the circle in Fig. 10. The circle arc is a NURBS, whose degree  $p$  is 2. Based on Eq. (16), the control points  $\{\vec{p}_i^0\} = \{\vec{p}_0^0, \vec{p}_1^0, \vec{p}_2^0\}$ ,  $\{w_i^0\} = \{1, 1/\sqrt{2}, 1\}$ , and the knots  $U = \{0, 0, 0, 1, 1, 1\}$ .

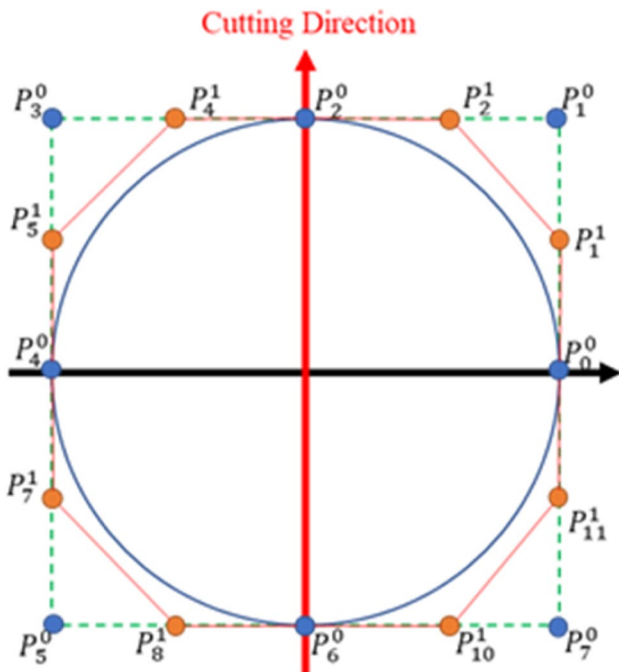


Fig. 9 Degree elevation of the circle

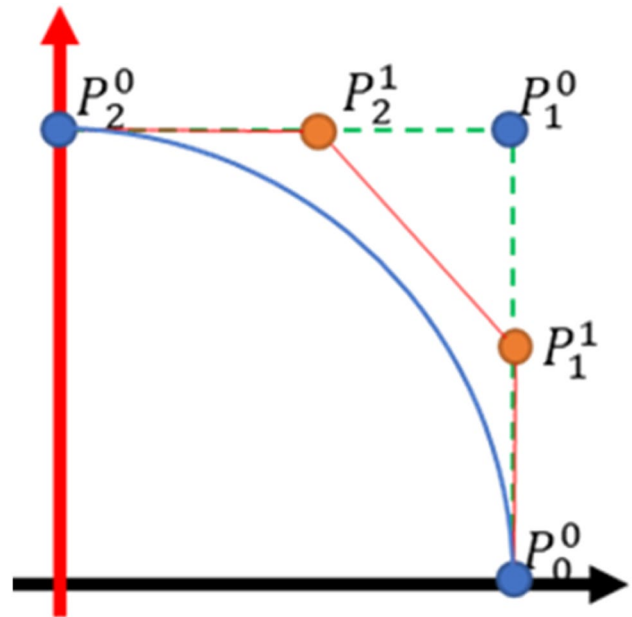


Fig. 10 The quadrant of the circle

$$C(u) = \frac{\sum_{i=0}^n N_{i,2}(u)w_i^0 \vec{P}_i^0}{\sum_{i=0}^n N_{i,2}(u)w_i^0}, 0 \leq u \leq 1 \tag{16}$$

Using the degree elevation algorithm [18], we reformulate the control points, weights and curve (Eq. (17)-(20)).

$$C(u) = \frac{\sum_{i=0}^n N_{i,2}(u)w_i^0 \vec{P}_i^0}{\sum_{i=0}^n N_{i,2}(u)w_i^0} = \frac{\sum_{i=0}^n N_{i,3}(u)w_i^1 \vec{P}_i^1}{\sum_{i=0}^n N_{i,3}(u)w_i^1}, 0 \leq u \leq 1 \tag{17}$$

$$w_i^1 \vec{P}_i^1 = (1 - \alpha_i)w_i^0 \vec{P}_i^0 + \alpha_i w_{i-1}^0 \vec{P}_{i-1}^0 \tag{18}$$

$$w_i^1 = (1 - \alpha_i)w_i^0 + \alpha_i w_{i-1}^0 \tag{19}$$

$$\alpha_i = \frac{i}{p + 1} \tag{20}$$

First, we insert the knot of the curve:

$$U' = \{0, 0, 0, 0, 1, 1, 1, 1\} \tag{21}$$

The degree of the circle is raised to 3, and the weights of the new curve are:

$$w_0^1 = w_0^0 \tag{22}$$

$$w_1^1 = \frac{2}{3}w_1^0 + \frac{1}{3}w_0^0 \tag{23}$$

$$w_2^1 = \frac{1}{3}w_2^0 + \frac{2}{3}w_1^0 \tag{24}$$

$$w_3^1 = w_2^0 \tag{25}$$

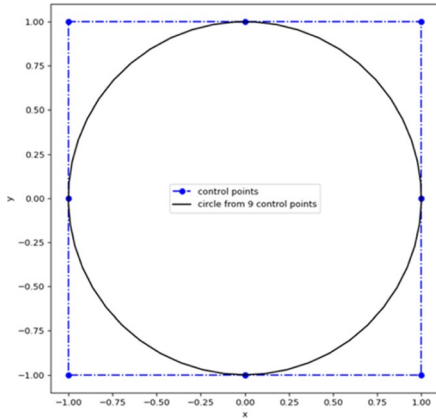
After getting the new weights of the curve, we recalculate the control points of the curves  $\{\vec{P}_i^1\} = \{\vec{P}_0^1, \vec{P}_1^1, \vec{P}_2^1, \vec{P}_3^1\}$ :

$$P_0^{-1} = \frac{w_0^0 P_0^{-0}}{w_0^1} \tag{26}$$

$$P_1^{-1} = \frac{\frac{2}{3}w_1^0 P_1^{-0} + \frac{1}{3}w_0^0 P_0^{-0}}{w_1^1} \tag{27}$$

$$P_2^{-1} = \frac{\frac{1}{3}w_2^0 P_2^{-0} + \frac{2}{3}w_1^0 P_1^{-0}}{w_2^1} \tag{28}$$

$$P_3^{-1} = \frac{w_2^0 P_2^{-0}}{w_3^1} \tag{29}$$



$$\begin{cases} U = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\} \\ w = \left\{1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1, \frac{\sqrt{2}}{2}, 1\right\} \\ \vec{P}^0 = \{(1,0), (1,1), (0,1), (-1,1), (-1,0), (-1,-1), (0,-1), (1,-1), (1,0)\} \end{cases}$$

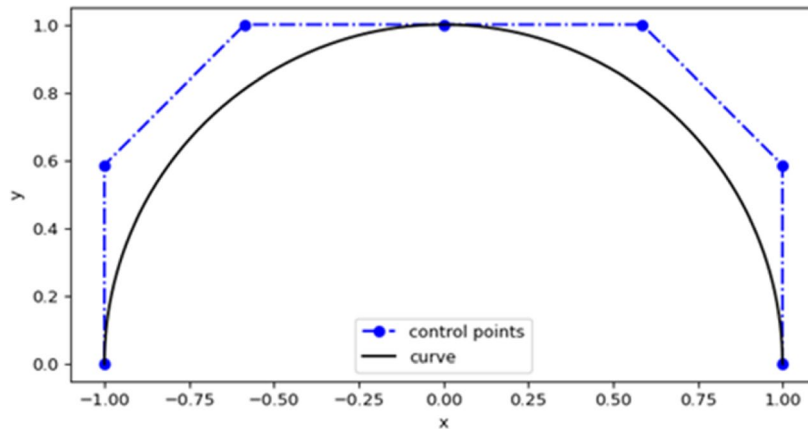
Fig. 11 The full circle of NURBS in  $p=2$

Figure 11 shows the NURBS circle based on  $p=2$  and Fig. 12 shows the semicircle NURBS circle based on  $p=3$ . After getting the control points of the degree-3 NURBS, we recalculate the weights of the curve  $w'$ . Figure 13 compares the NURBS circle with  $p=2$  and  $p=3$

### 3.2 Calculate the engagement angle in the active area

The engagement angle in the cutting process affects the cutting force and tool life. With the same depth of cut, the cutting force increases and tool life reduces when the engagement angle increases. The engagement angles  $\alpha$ ,

Fig. 12 The semicircle of NURBS in  $p=3$



$$\begin{cases} U' = \{0, 0, 0, 0, 1, 1, 1, 2, 2, 2, 2\} \\ w' = \left\{1, \frac{1+\sqrt{2}}{3}, \frac{1+\sqrt{2}}{3}, 1, \frac{1+\sqrt{2}}{3}, \frac{1+\sqrt{2}}{3}, 1\right\} \\ \vec{P}^1 = \left\{(1,0), \left(1, \frac{\sqrt{2}}{1+\sqrt{2}}\right), \left(\frac{\sqrt{2}}{1+\sqrt{2}}, 1\right), (0,1), \left(-\frac{\sqrt{2}}{1+\sqrt{2}}, 1\right), \left(-1, \frac{\sqrt{2}}{1+\sqrt{2}}\right), (-1,0)\right\} \end{cases}$$



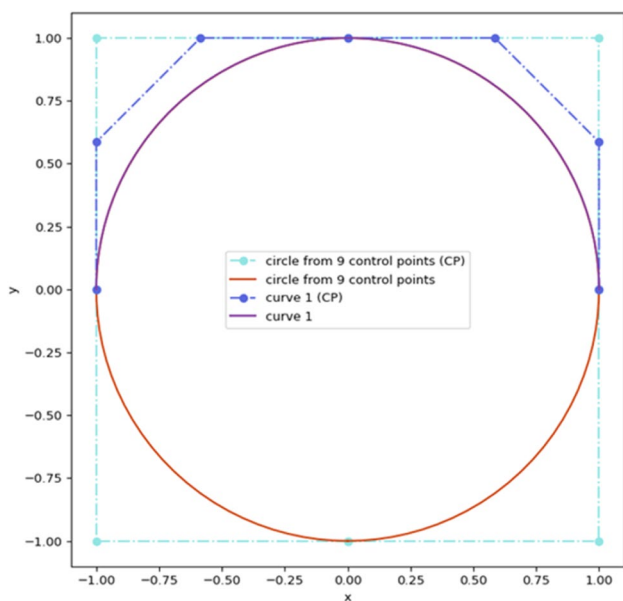


Fig. 13 Compare the NURBS circle in  $p=2$  and  $p=3$

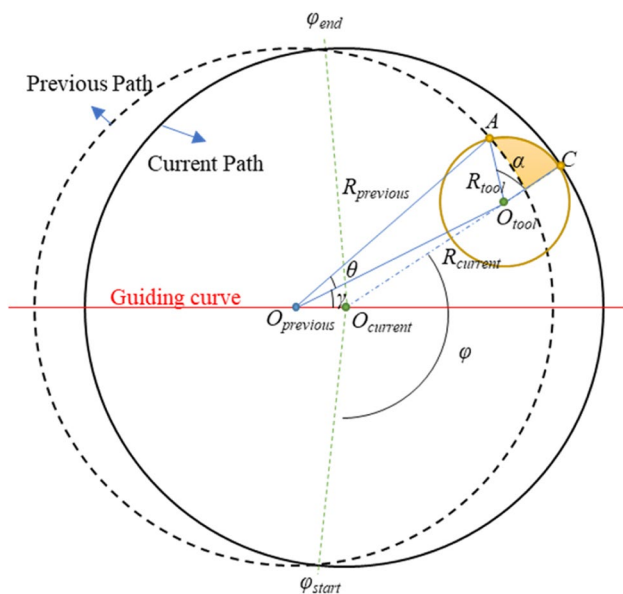


Fig. 14 Engagement angle in trochoidal tool path

as shown in Fig. 14, is calculated between the current and previous path of the trochoidal tool path.

In Fig. 14,  $O_{tool}$  is the current tool center. Point A is the contact point with tool and previous path. Point C is the contact point with tool and current path.  $O_{current}$  and  $O_{previous}$  are the points on the guiding curve. Based on the Eqs.(30)-(34), we can obtain positions of the point C and A.[17]:

$$O_{tool} = O_{current} + R_{current}e^{i(\varphi-\varphi_{start})} \quad \varphi_{start} \leq \varphi \leq \varphi_{end} \quad (30)$$

$$C = O_{current} + (R_{current} + R_{tool})e^{i(\varphi-\varphi_{start})} \quad (31)$$

$$\theta = \cos^{-1}\left(\frac{R_{previous}^2 + \overline{O_{tool}O_{previous}}^2 - R_{tool}^2}{2R_{current} \| \overline{O_{tool}O_{previous}} \|}\right) \quad (32)$$

$$\gamma = \cos^{-1}\left(\frac{\overline{O_{previous}O_{current}}^2 + \overline{O_{tool}O_{previous}}^2 - R_{current}^2}{2 \| \overline{O_{previous}O_{current}} \| \| \overline{O_{tool}O_{previous}} \|}\right) \quad (33)$$

$$A = O_{previous} + R_{previous}e^{i(\gamma+\theta)} \quad (34)$$

We obtain the engagement angles  $\alpha$  by Eq. (35).

$$\alpha = \cos^{-1} \frac{\overline{O_{tool}C} \cdot \overline{O_{tool}A}}{\| \overline{O_{tool}C} \| \| \overline{O_{tool}A} \|} = \cos^{-1} \left( \frac{\overline{O_{tool}C} \cdot \overline{O_{tool}A}}{R_{tool}^2} \right) \quad (35)$$

The engagement angles  $\alpha$  profile in the traditional trochoidal tool path is shown in Fig. 15. The engagement angle starts from zero and increases after the cutter-and-workpiece contact. When the maximum engagement angle has arrived, the profile drops immediately. The cutting efficiency is not very satisfactory.

### 3.3 Improve the cutting active section

In the previous section, we calculate the maximum engagement angles  $\alpha$  in the circular based trochoidal path. To improve the cutting efficiency, we propose a new method based on the bell shape planning to adjust the engagement angles  $\alpha$ . Figure 16 shows the relationship between the engagement angle and rotation angle of the trochoidal path. We divide the engagement angle curve into acceleration, extended maximum angle, and deceleration sections.

The planning in the acceleration section is by Eqs. (36), (37):

$$\alpha(s) = \alpha_{start} + C_1\varphi^2, \quad \varphi < \frac{\varphi_1 - \varphi_{start}}{2} \quad (36)$$

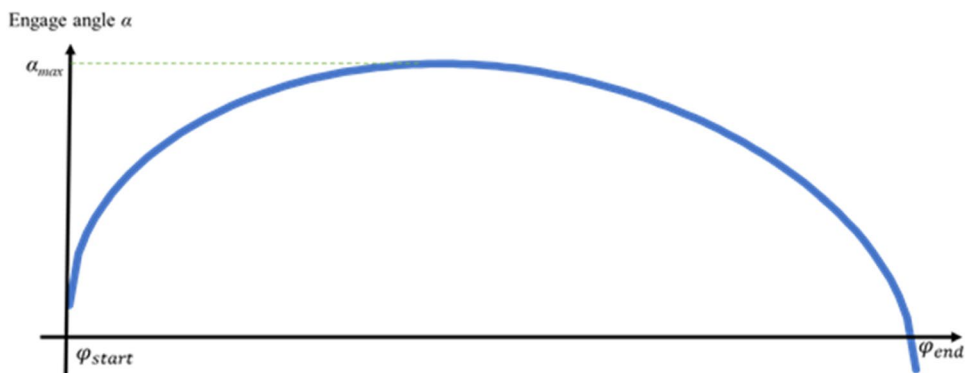
$$\alpha(s) = \alpha_{max} - C_1(\varphi_1 - \varphi)^2, \quad \frac{\varphi_1 - \varphi_{start}}{2} < \varphi < \varphi_1 \quad (37)$$

The boundary condition in the acceleration section:

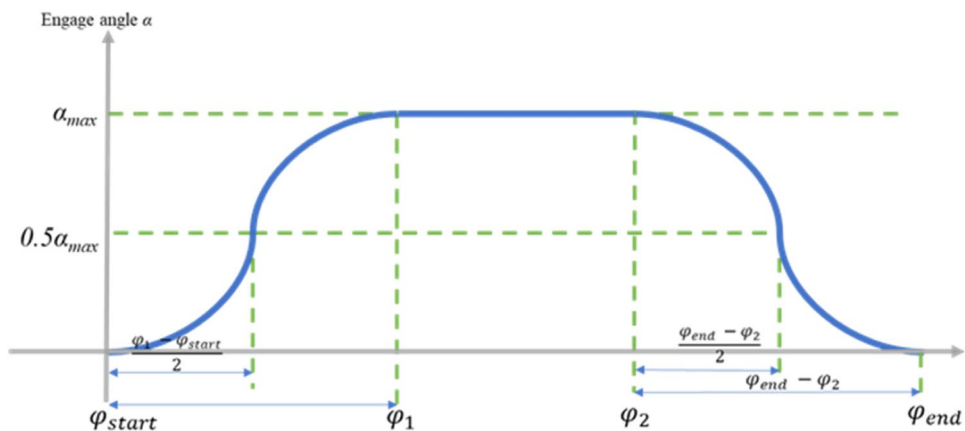
$$C_1 = \frac{2(\alpha_{max} - \alpha_{start})}{(\varphi_1 - \varphi_{start})^2} \quad (38)$$

The planning in the deceleration section is by Eqs. (39), (40):

**Fig. 15** Engage angle in circle pattern



**Fig. 16** Bell-shaped engage angle planning



$$\alpha(s) = \alpha_{max} - C_2(\varphi_{end} - \varphi)^2, \varphi_2 < \varphi < \varphi_2 + \frac{(\varphi_{end} - \varphi_2)}{2} \quad (39)$$

$$\alpha(s) = \alpha_{end} - C_2(\varphi_{end} - \varphi)^2, \varphi_2 + \frac{(\varphi_{end} - \varphi_2)}{2} < \varphi < \varphi_{end} \quad (40)$$

The boundary condition in the deceleration section:

$$C_2 = \frac{2(\alpha_{max} - \alpha_{end})}{(\varphi_{end} - \varphi_2)^2} \quad (41)$$

Finally, based on Eqs. (36)–(41), we organize the engage angle planning model as follows:

$$\alpha(s) = \begin{cases} \alpha_{start} + C_1\varphi^2, \varphi < \frac{\varphi_1 - \varphi_{start}}{2} \\ \alpha_{max} - C_1(\varphi_1 - \varphi)^2, \frac{\varphi_1 - \varphi_{start}}{2} < \varphi < \varphi_1 \\ \alpha_{max}, \varphi_1 < \varphi < \varphi_2 \\ \alpha_{max} - C_2(\varphi_{end} - \varphi)^2, \varphi_2 < \varphi < \varphi_2 + \frac{(\varphi_{end} - \varphi_2)}{2} \\ \alpha_{end} - C_2(\varphi_{end} - \varphi)^2, \varphi_2 + \frac{(\varphi_{end} - \varphi_2)}{2} < \varphi < \varphi_{end} \end{cases} \quad (42)$$

By changing the engagement angle, we improve the cutting efficiency in the trochoidal tool path. The result of the engagement angle is shown in Fig. 17, where the blue curve is generated by the circular model and the orange curve is generated by the new pattern model. In the boundary condition of the engagement angle planning,

the tool path will be greatly modified when the change of  $\alpha$  is larger. It is easy to cause overcutting in the milling process. To overcome this problem, we set the boundary condition by Eq. (43).

$$\alpha'_{max} - \alpha < 0.2\alpha_{max} \quad (43)$$

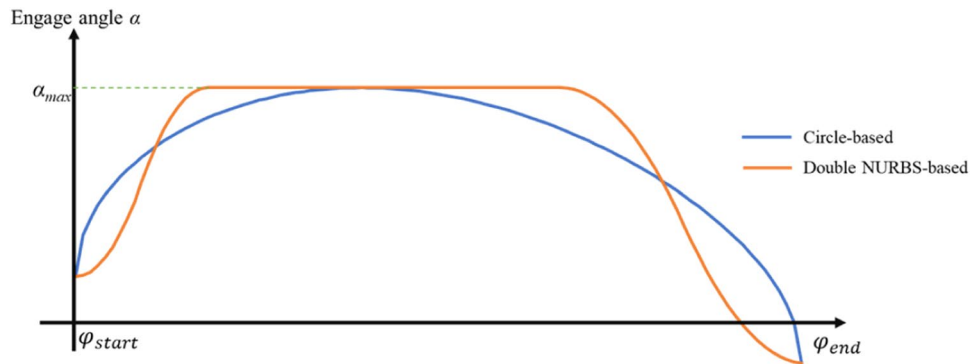
### 3.4 Tool path generation in the active section

In the last section, we update the engagement angle by our proposed method. Based on Eq. (44), we obtain the new contact points  $A'_i = \{A'_1, A'_2, \dots, A'_n\}$  in  $C'_A(u)$ , where  $R_{z, O_{tool,i}}(\alpha'_i)$  is the rotation matrix with  $O_{tool,i}$  as the center and  $\alpha'_i$  is the updated engagement angle. Point  $C_i$  is the contact point in the current path in Fig. 14.

$$A'_i = R_{z, O_{tool,i}}(\alpha'_i)C_i \quad (44)$$

After getting the new contact points, we use the pseudo-inverse solution for the formulation of least-squares problem, of which the steps are shown in Eqs. (45)–(48), to obtain the new control points  $P^1 = \{P^1_0, P^1_1, \dots, P^1_6\}$  of which the knots are  $U^1 = \{0, 0, 0, 0, 1, 1, 1, 2, 2, 2, 2\}$  and the weights  $\{1, \frac{1+\sqrt{2}}{3}, \frac{1+\sqrt{2}}{3}, 1, \frac{1+\sqrt{2}}{3}, \frac{1+\sqrt{2}}{3}, 1\}$  in degree-3 NURBS in Section 3.1 (Fig. 18).

**Fig. 17** Circular and new pattern engagement angle comparison



$$\vec{A}' = \frac{\sum_{i=0}^6 N_{i,3}(u)w'_i \vec{P}_i}{\sum_{i=0}^6 N_{i,3}(u)w'_i} = \sum_{i=0}^6 H_{i,3}(u) \vec{P}_i \tag{45}$$

$$H_{i,3}(u) = \frac{N_{i,3}(u)w'_i}{\sum_{i=0}^6 N_{i,3}(u)w'_i} \tag{46}$$

$$\vec{A}'_{n \times 1} = H_{n \times 7} \vec{P}_{7 \times 1} \tag{47}$$

$$\vec{P}_{7 \times 1} = (H_{7 \times n}^T H_{n \times 7})^{-1} H_{7 \times n}^T \vec{A}'_{n \times 1} \tag{48}$$

We generate the new tool path based on the new control points  $\vec{p}_1$  by Eq. (49).

$$C'_{A}(u) = \frac{\sum_{i=0}^6 N_{i,3}(u)w'_i \vec{P}_i}{\sum_{i=0}^6 N_{i,3}(u)w'_i}, 0 \leq u \leq 1 \tag{49}$$

In Fig. 19, we compare the curve-fitting result with the curve created from the engagement angle planning. The NURBS curve with  $p = 3$  from the curve fitting matches quite well with the green one which was built from the engagement angle planning.

Based on [19], we calculate the radius of curvature to find the trochoidal radius of the new trochoidal path:

$$R'_{previous} = \frac{\|\dot{C}'_A(u) \times \ddot{C}'_A(u)\|}{\|\dot{C}'_A(u)\|^3} = R'_{troch} + R_{tool} \tag{50}$$

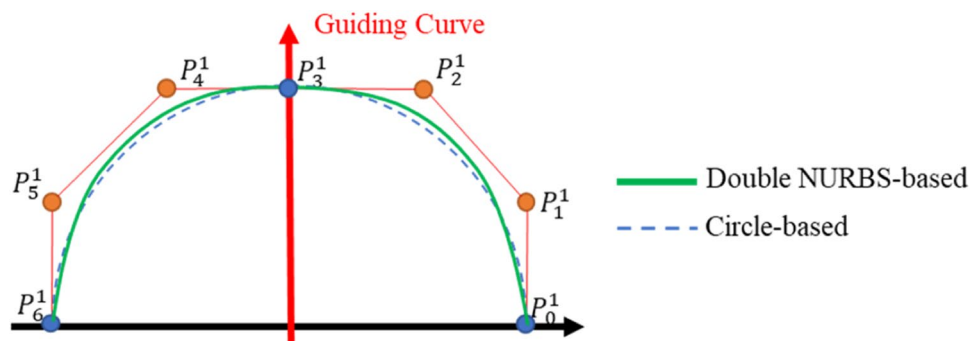
$$R'_{troch} = \frac{\|\dot{C}'_A(u) \times \ddot{C}'_A(u)\|}{\|\dot{C}'_A(u)\|^3} - R_{tool} \tag{51}$$

The result of the tool paths is shown in Fig. 20, where the blue curve is generated by circular model and the green one is generated by the proposed new pattern model. In the figure, the green curve has more cutting area compared with the blue one to improve the cutting efficiency.

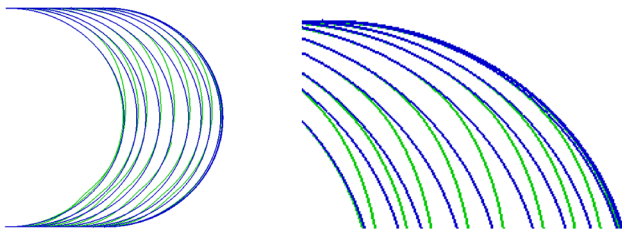
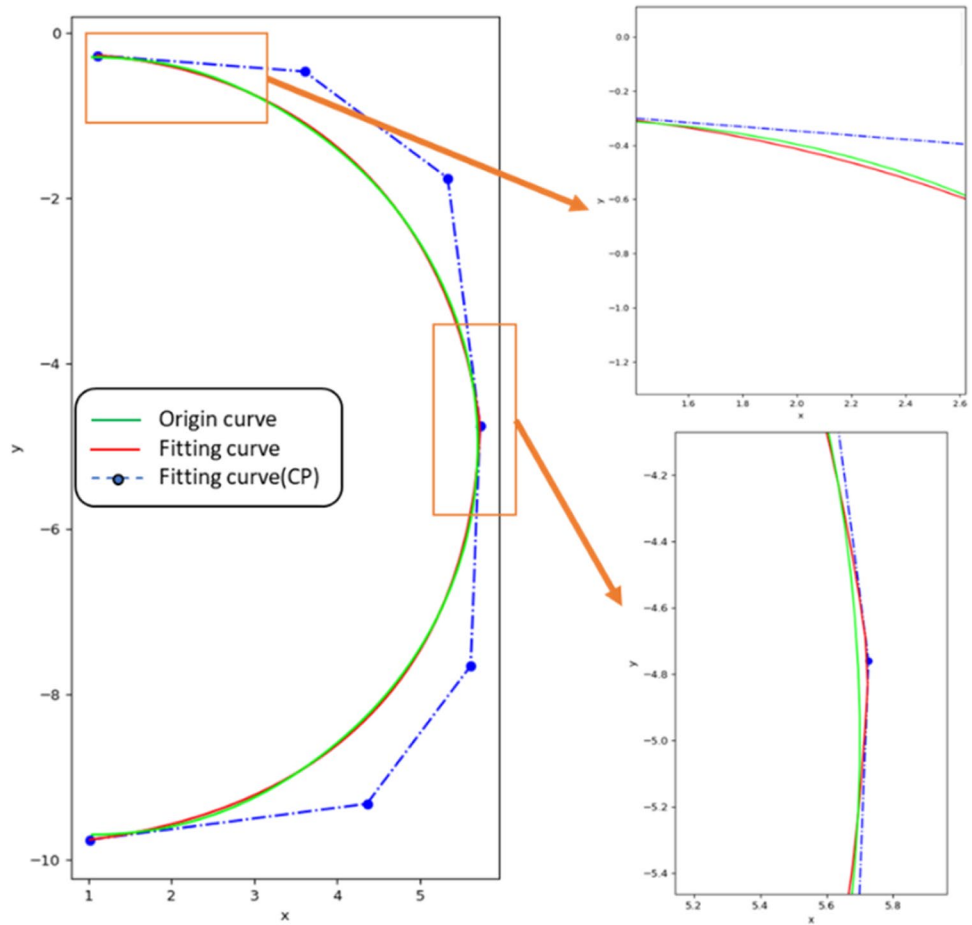
### 3.5 Tool path improvement in the non-active section

In the non-active section, we use the degree 3 NURBS curve to fit the section in the trochoidal path. As shown in Fig. 21,  $C_{A,i}(u)$  and  $C_{A,i+1}(u)$  are the tool paths of the active section.  $P_{C_{A,i}(2)}$  and  $P_{C_{A,i+1}(0)}$  are the end points of each curve.  $S_{C_{A,i}(2)}$  and  $S_{C_{A,i+1}(0)}$  are the end slopes of each

**Fig. 18** Circular and new pattern toolpath comparison



**Fig. 19** Curve fitting result: the green curve is the original curve from engagement angle planning; the red curve is the curve from curve fitting



**Fig. 20** Compare the tool paths between circular model and proposed new pattern model

curve. We recalculate the control points of the NURBS in the non-active section.

$$P_0 = P_{C_A(u)}(1) \tag{52}$$

$$P_1 = \frac{3P_0 + S_{C_A(u)}(1)}{3} \tag{53}$$

$$P_2 = \frac{3P_3 - S_{C_A(u+1)}(0)}{3} \tag{54}$$

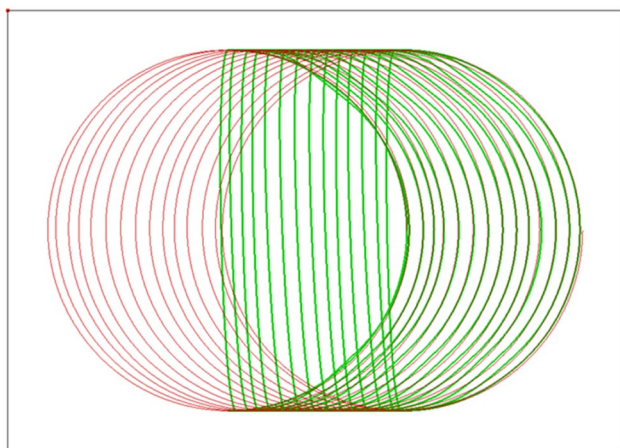
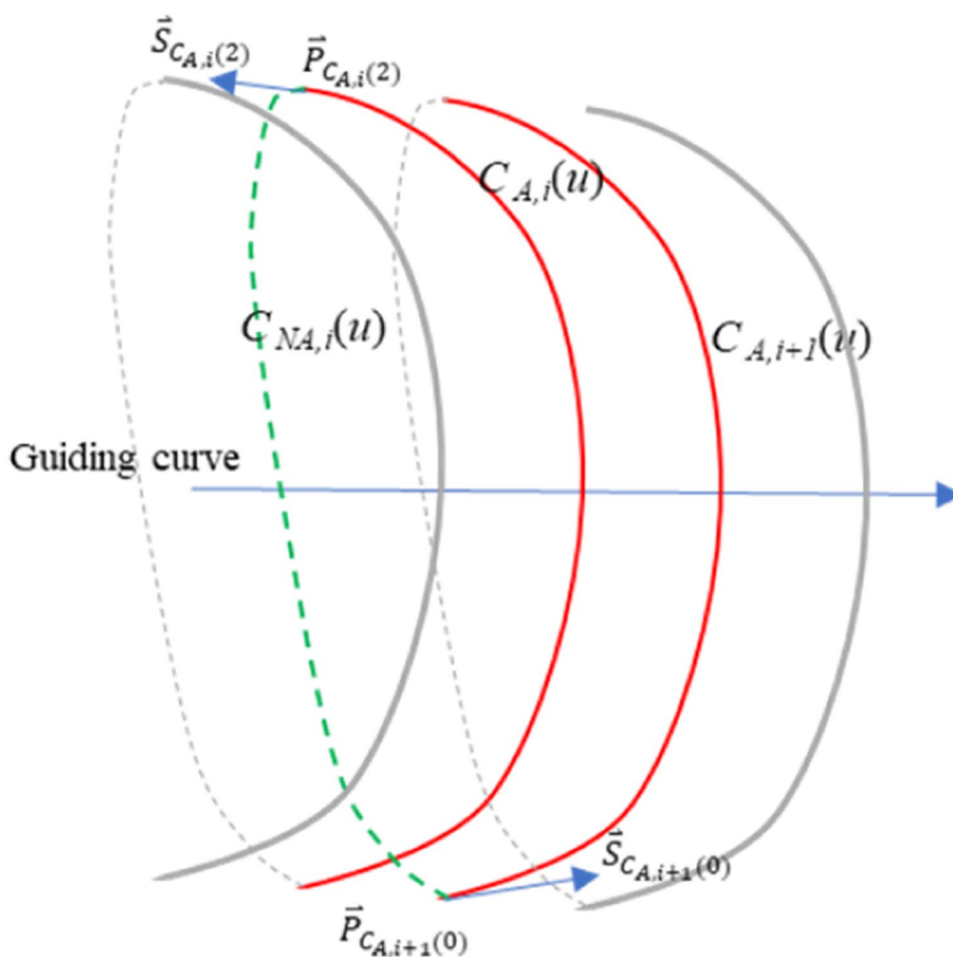
$$P_3 = P_{C_A(u+1)}(0) \tag{55}$$

The knots of curve  $U = \{0, 0, 0, 0, 1, 1, 1, 1\}$  and the weights  $w = \{1, 1, 1, 1\}$ . The NURBS of the non-active section is:

$$C_{NA,i}(u) = \frac{\sum_{i=0}^n N_{i,3}(u)w_i \vec{P}_i}{\sum_{i=0}^n N_{i,3}(u)w_i}, 0 \leq u \leq 1 \tag{56}$$

As shown in Fig. 22, we compare two tool paths by different methods. The green curve is our proposed method, and the red curve is the circle-based method. In terms of tool path length comparison, as shown in Table 1, the path length in the non-active section is 115.679 mm by our proposed method and 202.105 mm by traditional trochoidal path method. The path length in non-active section is reduced by 42.8%. The result shows our proposed method can significantly shorten the processing time compared with the circle-based method.

**Fig. 21** The toolpath by NURBS in non-active section



**Fig. 22** Compare the trochoidal paths by different methods. The green curve is our proposed method, and the red curve is the circle-based method

### 3.6 Cutting force simulation

In order to verify the proposed method can achieve the purpose of improving the cutting efficiency, we use the

**Table 1** Compare the length of tool path from Fig. 22

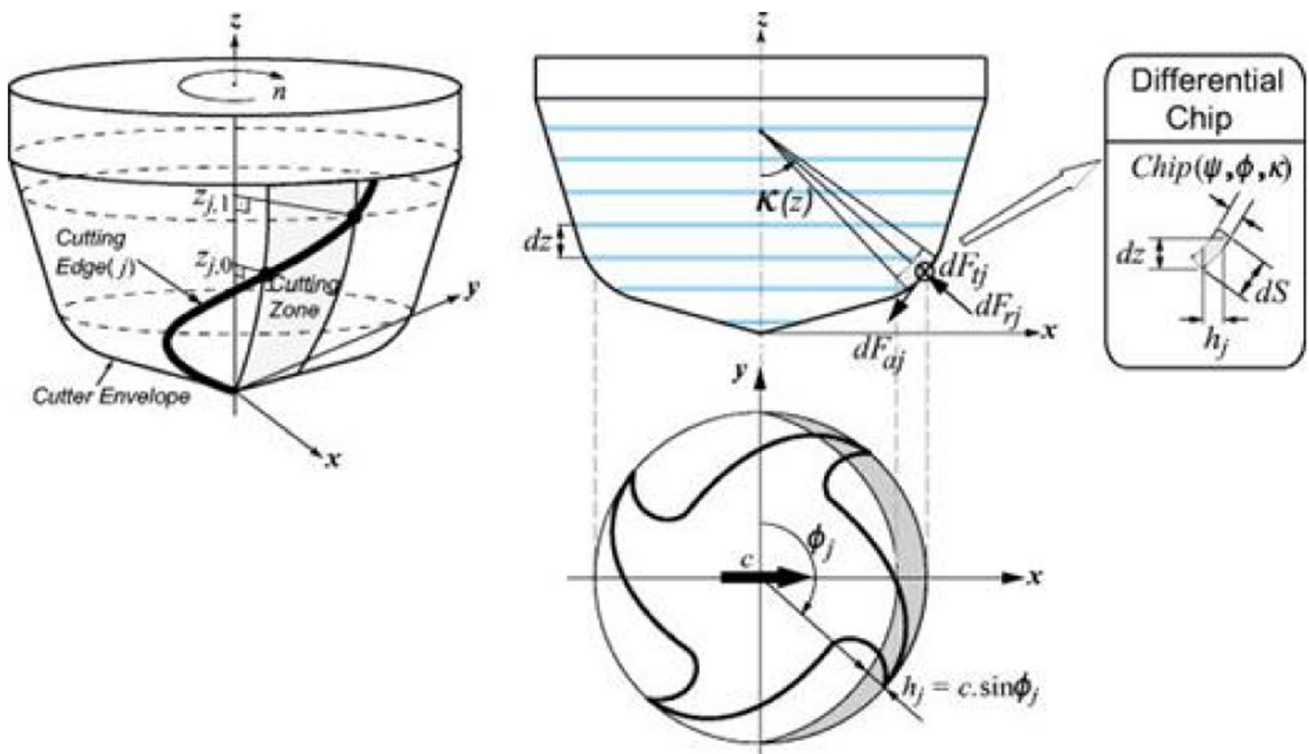
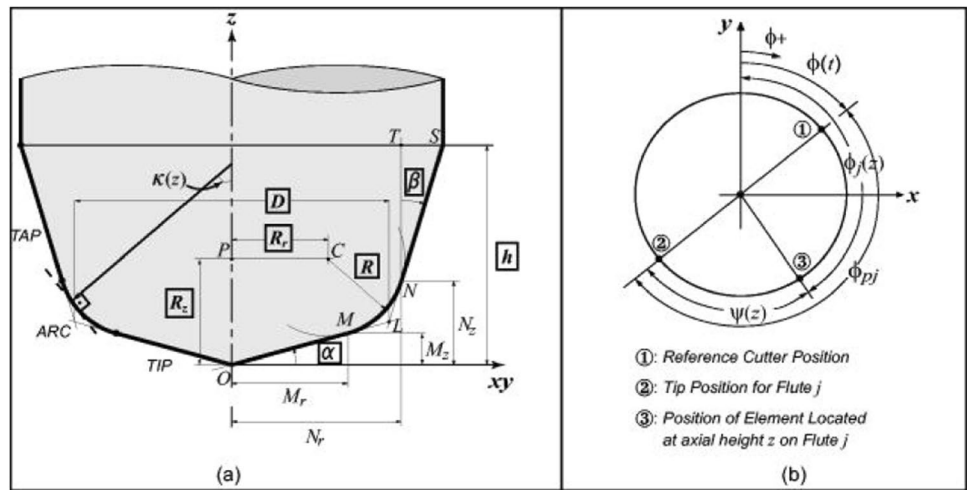
	Active section (mm)	Non-active section (mm)	Total (mm)
Proposed method	188.816	115.679	304.495
Circle-based method	184.262	202.105	386.367

mechanistic model [29] to analyze and simulate the cutting force to see whether the change of the cutting force conforms to the change of the engagement angle calculated by our algorithm. Before calculating the cutting force, we refer to [30] and use the APT model of a generalized cutter, with  $\alpha=0, \beta=0$ , and  $R=0$ , to define the geometric shape of a flat end mill in Fig. 23. When the cutter contacts with the workpiece, as shown in Eq. (57), the engagement angle  $\phi$  between the entry angle  $\theta_{st}$  and the departure angle  $\theta_{ex}$  is calculated.

$$\theta_{st} \leq \phi \leq \theta_{ex} \tag{57}$$

In cutting simulation, we adopt a direct computation of cutting force method [31] and analyze the cutting force of each point on the layer along the Z axis of the cutter.

**Fig. 23** Generalized cutting tool geometry (a) and the cutting angle convention (b)



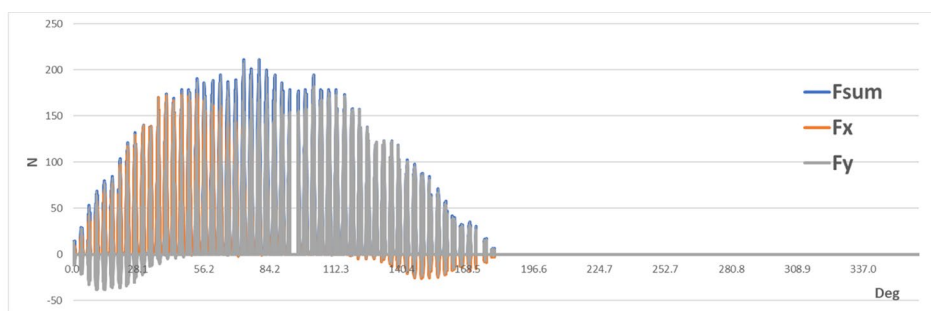
**Fig. 24** Mechanics and kinematics of milling process [29]

According to the trochoidal tool path trajectory, the differential rotating angle, and the instantaneous chip thickness, the instantaneous cutting force is evaluated using the following mechanistic model (Fig. 24) [29]:

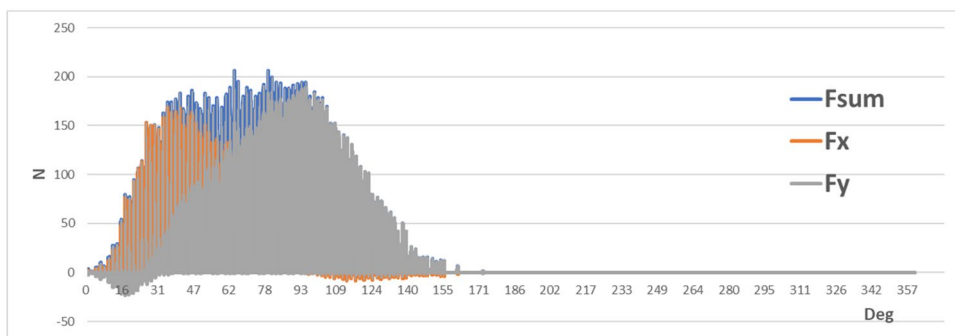
$$\begin{bmatrix} dF_{rj}(\phi, z) \\ dF_{tj}(\phi, z) \\ dF_{aj}(\phi, z) \end{bmatrix} = \begin{bmatrix} K_{re} \\ K_{te} \\ K_{ae} \end{bmatrix} dS(z) + \begin{bmatrix} K_{rc} \\ K_{tc} \\ K_{ac} \end{bmatrix} h_j(\phi, z) dz \quad (58)$$

where  $K_{rc}$ ,  $K_{tc}$ ,  $K_{ac}$  are cutting force coefficients due to the shear and  $K_{re}$ ,  $K_{te}$ ,  $K_{ae}$  are the edge friction coefficients between the cutter and the workpiece in radial, tangential, and axial directions. Based on Eq. (58), we compare the simulated cutting forces between the circle-based tool path and our proposed NURBS-based tool path.

**Fig. 25** Comparing cutting forces under different toolpaths (a) Circular method (b) Our proposed method



(a)



(b)

**Table 2** The composition ratio of Al6061

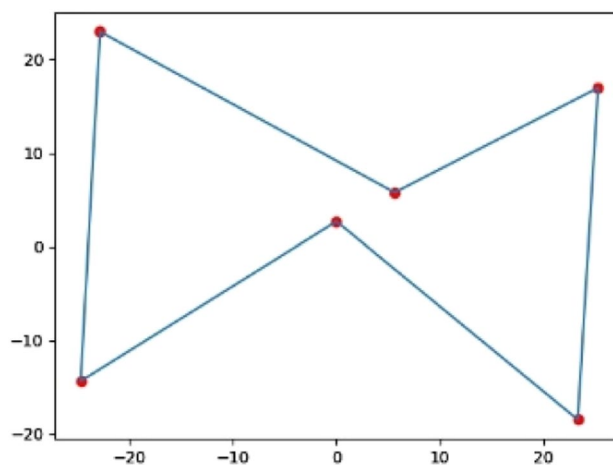
Si	Fe	Cu	Mn	Mg	Cr	Zn	Other
0.4	0.4	0.1	0.4~1.0	4.0~4.9	0.05~0.25	0.25	0.05~0.15

As shown in Fig. 25, under the same cutting shape by Fig. 22, our proposed path generation algorithm can maintain the maximum cutting efficiency for a longer period of time and improve the cutting efficiency compared with the traditional trochoidal machining path.

## 4 Experimental validation and discussion

### 4.1 Experiment setting

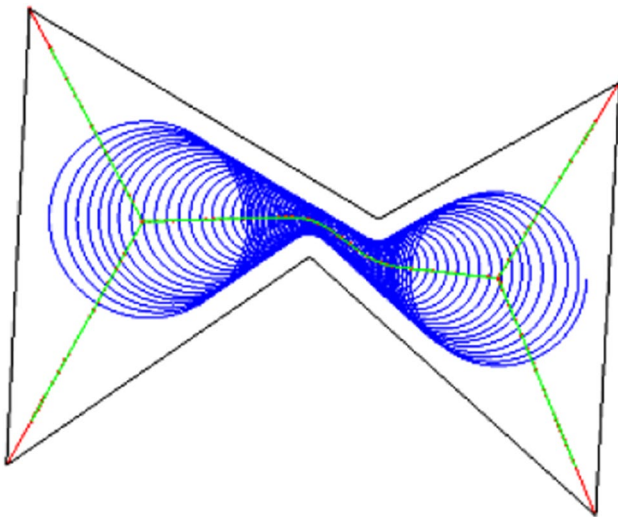
To verify the proposed method in this paper, we use Tongtai VC-608 3-axis machine tool for testing. The controller of the machine is the LNC M6800D. The tool we used is NACHI 2SE  $\phi$ 3mm end-mill tool. The dynamometer is Kistler type 9257B. The sampling rate is 200 Hz. The material we use is aluminum alloy as shown in Table 2. We use the cutting parameters in the milling process suggested by the tool supplier[32]. The spindle speed is 5333 rpm, the feedrate is 350 mm/min, and the axial depth of cut is 3 mm. As shown in Fig. 26, the size of the test workpiece is 50.06 mm  $\times$  41.4 mm.



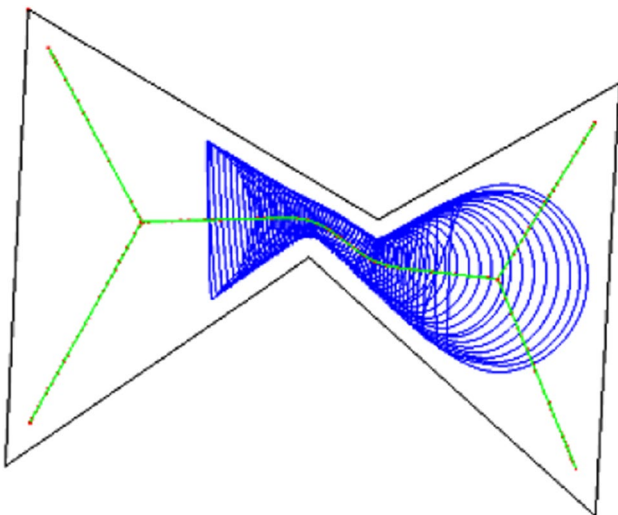
**Fig. 26** Pocketing case

## 5 Result and discussion

We compare the proposed new trochoidal tool path with the circular trochoidal tool path, as shown in Figs. 27 and 28.



**Fig. 27** Circle-based trochoidal toolpath



**Fig. 28** NURBS-based trochoidal tool path

The total length of the tool path by our proposed method achieves a reduction of 22.5%, with an increase of 1.2% in the active area, and a reduction of 47.2% in the non-active area. The total machining time of the tool path achieves a reduction of 17.4%. The detail results are listed in Table 3 and show that the cutting efficiency is increased by increasing the total engagement area in the cutting region and reducing the tool path in the non-active area.

**Table 3** Comparison of two different toolpaths

Toolpath type	Cutting path length(mm)	Non-Cutting path length(mm)	Total length(mm)	Machining Time(s)
Circle-Based	690.697	663.258	1353.955	386
NURBS-based	699.087	350.414	1049.501	319

The cutting force of the total path in the demonstration case is shown in Fig. 29. For the analysis of a single cycle, as shown in Fig. 30, we compare the cutting time to the non-cutting time in the NURBS trochoidal tool path and the circular trochoidal tool path. Table 4 shows the ratio of active time to non-active time; the circular model is 0.52 and the NURBS model is 1.34. The result shows that the NURBS trochoidal tool path improves the cutting efficiency by saving the non-cutting time significantly.

Figure 31 shows two more pocketing cases with one island and two islands. As shown in Table 5, the result shows that the NURBS trochoidal tool path improves the cutting efficiency by saving the non-cutting time significantly, indicated by the ratio of the cutting path to the non-cutting path.

## 6 Conclusions

In this paper, we propose a novel double-NURBS trochoidal path generation method. The proposed method has three characteristics:

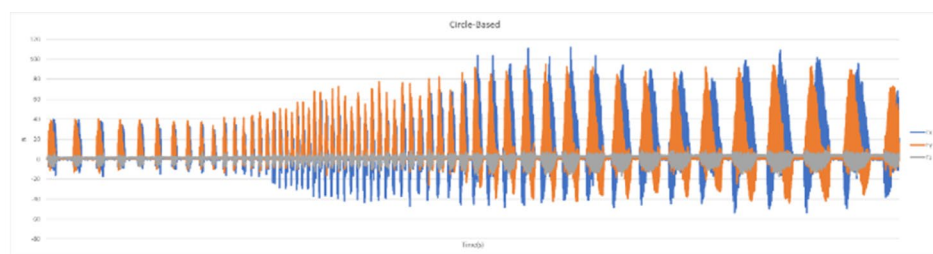
- 1 Using the double-NURBS model to represent the trochoidal tool path that combines the (medial-axis) guiding-curve tool path and the circular tool path with a degree-3 NURBS curve to maintain  $C^2$  continuity throughout the entire trochoidal tool path.
- 2 By degree elevation of the degree-2 NURBS circle to a degree-3 NURBS curve, we increase the cutter-workpiece engagement bandwidth by modifying the front-end cutting curve and its control points. This increases the cutting efficiency while maintaining the  $C^2$  continuity to improve smooth tool path transition and reduce undesired vibration.
- 3 For the back-end retraction tool path in the non-active region, we modify the degree-3 NURBS curve to shorten the travel distance to save non-active time. The reduction of time saving from a ratio of 0.52 to 1.34 is quite significant.

Compared with the traditional circular trochoidal tool path, the double-NURBS method we present has a reduction of 22.5% of the tool path length which has an increase of 1.2% in the active area and a reduction of 47.2% in the non-active area. The total cutting time is reduced by 17.4%.

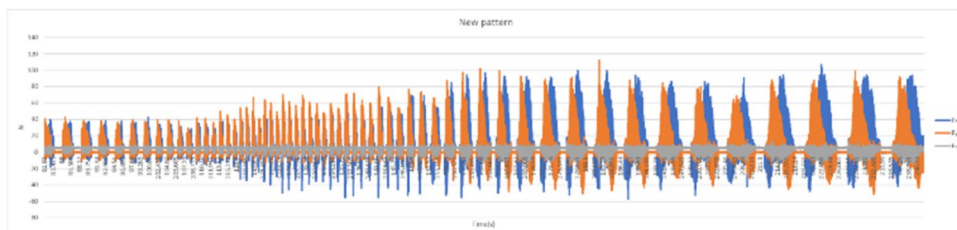
Since the trochoidal machining path has non-cutting path for every cycle, it is known that trochoidal machining will not be suitable for the entire machining region. Hence good



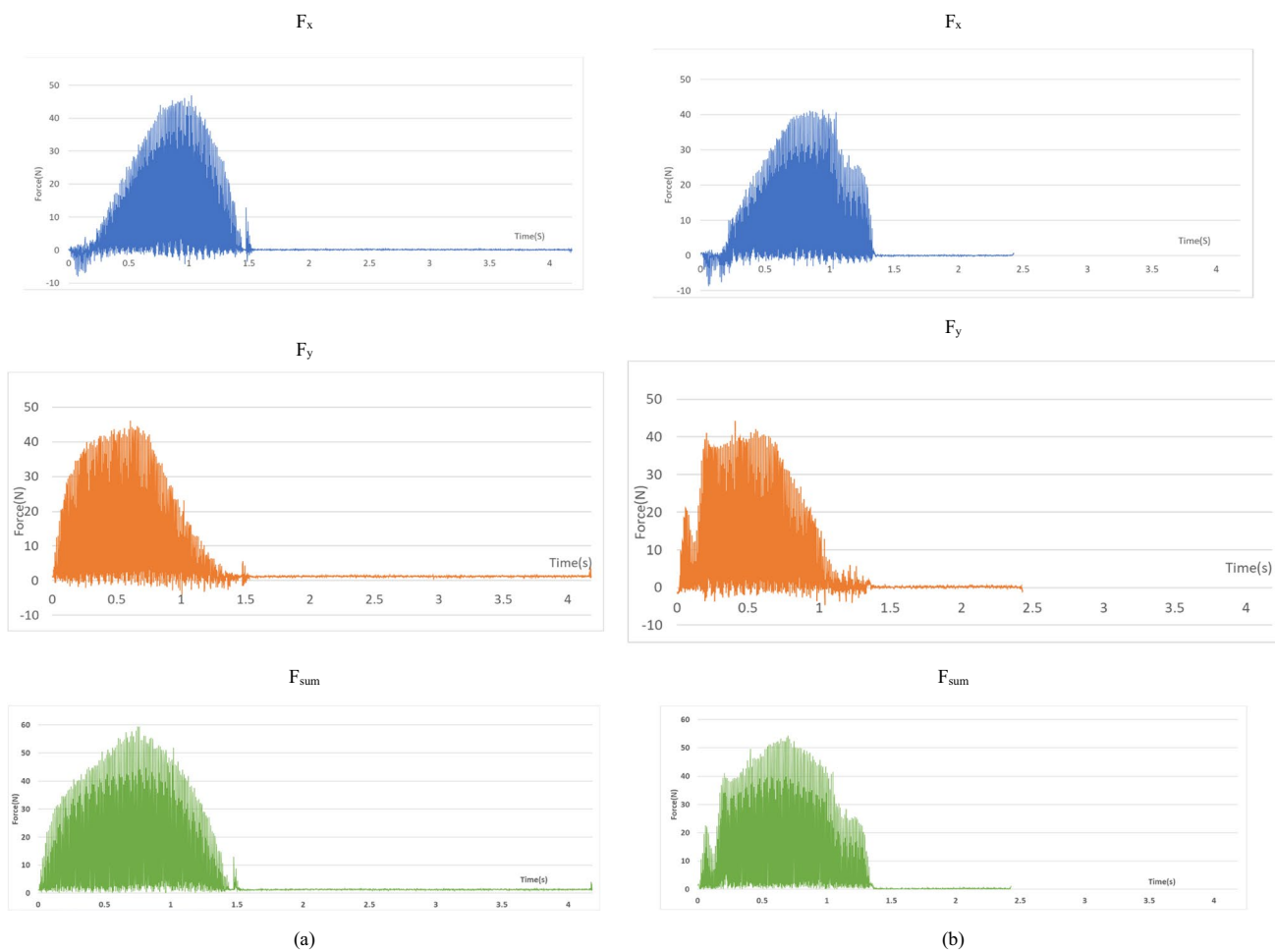
**Fig. 29** Cutting force during machining (a) Circle-based path (b) NURBS-based path



(a)



(b)



**Fig. 30** Cutting force in a single cycle (a) Circle-based path (b) NURBS-based path

**Table 4** The cutting time and ratio of active and non-active area in a single trochoidal cycle

Toolpath	Time of active area (s)	Time of non-active area(s)	Ratio
Circular model	1.59	3.02	0.52
New pattern model	1.65	1.23	1.34

**Table 5** Comparison of two different toolpaths in pocketing case with two islands

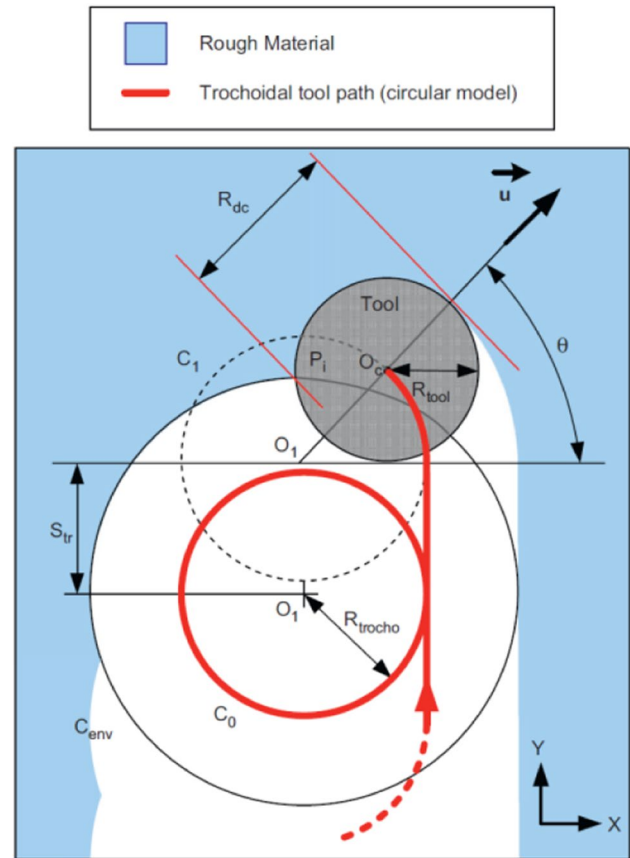
Toolpath type	Cutting path length(mm)	Non-Cutting path length(mm)	Total length(mm)	Ratio
One island	2136.214	1379.196	3711.214	1.548
Two islands	5638.133	3559.813	9197.946	1.584

strategies need to be designed to integrate trochoidal tool path with other types of tool path such as pocket contouring path. This will be one of our future work directions. Other future work includes the feed rate optimization of the trochoidal tool path based on the cutting force constraints. This is because as the trochoidal radius varies along the guiding curve to approximate pocketing contours, cutting force becomes smaller as the trochoidal radius increases. If we are to maintain a steady cutting force (so a steady tool wear progress), we can actually speed up the feed rate and increase productivity. In our future work, we plan to integrate real-time cutting force prediction with optimized feed rate scheduling to maximize trochoidal milling productivity. This might be further integrated with the look-ahead function of a machine tool controller that is equipped with the NURBS interpolation capability.

### Appendix A

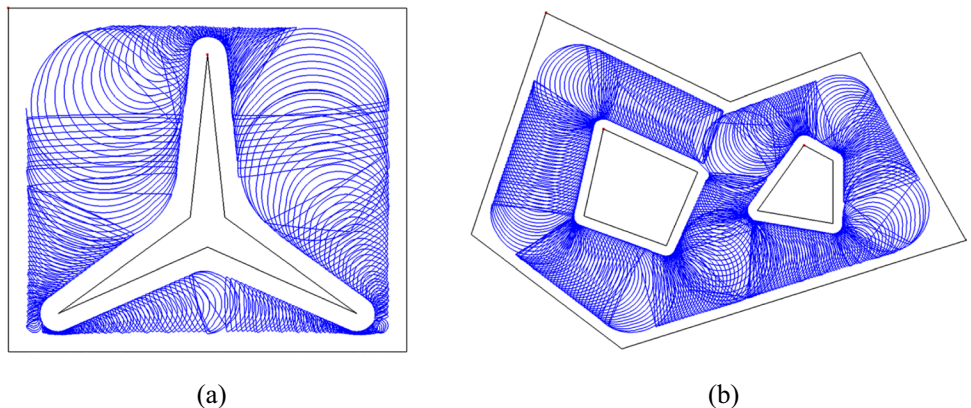
The derivation for the mathematical model of the maximum radial depth of cut ( $R_{dc_{max}}$ ) and step length ( $S_{tr}$ )

For completeness, we provide the derivation of the maximum radial depth of cut ( $R_{dc_{max}}$ ) and step length ( $S_{tr}$ ) in [3]. As shown in Fig. 32, the trochoidal circular path is modeled by circle  $C_1$ , whose center is  $O_1(0, S_{tr})$  and the trochoidal radius  $R_{trocho}$ . The center of tool  $O_c(X_c, Y_c)$  is parameterized by angle  $\theta$ :



**Fig. 32** Radial depth of cut calculation model and parameterization [3]

**Fig. 31** Pocketing cases (a) one island (b) two islands



$$\begin{cases} X_c = R_{trocho} \cos \theta \\ Y_c = S_{tr} + R_{trocho} \sin \theta \end{cases} \quad (59)$$

Point  $P_i(X_i, Y_i)$ , the intersection between  $C_{env}$  and tool, needs to be determined before calculating the radial depth of cut:

$$\begin{cases} X_i^2 + Y_i^2 = (R_{trocho} + R_{tool})^2 \\ (X_i - R_{trocho} \cos \theta)^2 + (Y_i - S_{tr} - R_{trocho} \sin \theta)^2 = R_{tool}^2 \end{cases} \quad (60)$$

As soon as  $P_i$  position is obtained, radial depth of cut  $R_{dc}$  can be calculated using:

$$R_{dc} = \vec{P}_p \vec{O}_c \vec{u} + R_{tool} = (X_i - X_c) \cos \theta + (Y_i - Y_c) \sin \theta + R_{tool} \quad (61)$$

In trochoidal milling process, the most common configuration is Eq. (62). Radial depth of cut increases continuously on the linear segment and in the beginning of the circle until its maximal value.

$$R_{tool} + S_{tr} < R_{trocho} \quad (62)$$

Then its value decreases. For this configuration, maximal depth of cut is reached when  $O_0, O_1$  and  $P_i$  are aligned:

$$R_{dc_{max}} = \left(1 + \frac{R_{tool}}{R_{trocho}}\right) S_{tr} - \frac{1}{2R_{trocho}} S_{tr}^2 \quad (63)$$

$$\frac{1}{2} S_{tr}^2 - (R_{trocho} + R_{tool}) S_{tr} + R_{trocho} R_{dc_{max}} = 0 \quad (64)$$

$$S_{tr} = (R_{trocho} + R_{tool}) \pm \sqrt{(R_{trocho} + R_{tool})^2 - 2R_{trocho} R_{dc_{max}}} \quad (65)$$

To comply with Eq. (62), the step length of trochoidal milling is obtained:

$$S_{tr} = (R_{trocho} + R_{tool}) - \sqrt{(R_{trocho} + R_{tool})^2 - 2R_{trocho} R_{dc_{max}}} \quad (66)$$

**Author contribution** All authors contribute equally to the theoretical development and experimental design and implementation of this work.

**Funding** The authors express their thanks to the Ministry of Science and Technology of Taiwan for the funding support (Grant number [MOST 110–2221-E-194–038]).

**Data availability** Data and material can be available upon request.

**Code availability** Code can be available upon request.

### Declarations

**Ethics approval** The authors declare that the content of this work is original and there is no conduct of violating the principle of ethics in this work.

**Consent to participate** The authors were granted and approved to participate in this work.

**Consent for publication** The authors give full consent to the publisher for the publication of this work.

**Conflict of interest** The authors declare no competing interests.

### References

- Otkur M, Lazoglu I (2007) Trochoidal milling. *Int J Mach Tools Manuf* 47(9):1324–1332
- Ibaraki S, Yamaji I, Kakino Y, Inoue D, Nishida S (2003) Tool path planning using trochoid cycles for hardened steel in die and mold manufacturing (1st report)—tool path generation for trochoid cycles based on Voronoi diagram. In *Proceedings of the 2003 international conference on agile manufacturing, ICAM 2003*, Beijing, pp 435–442
- Rauch M, Duc E, Hascoet JY (2009) Improving trochoidal tool paths generation and implementation using process constraints modelling. *Int J Mach Tools Manuf* 49(5):375–383
- Ferreira JC, Ochoa DM (2013) A method for generating trochoidal tool paths for 2½D pocket milling process planning with multiple tools. *227(9):1287–1298*
- Li Z, Xu K, Tang K (2019) A new trochoidal pattern for slotting operation. *Int J Adv Manuf Technol* 102(5):1153–1163
- Huang X, Wu S, Liang L, Li X, Huang N (2020) Efficient trochoidal milling based on medial axis transformation and inscribed ellipse. *Int J Adv Manuf Technol* 111(3):1069–1076
- Yamaji I, Ibaraki S, Kakino Y, Nishida S (2003) Tool path planning using trochoid cycles for hardened steel in die and mold manufacturing (2nd report). In *Proc 2003 Int Conf Agile Manuf*, Beijing, pp 443–450
- Mortara M, Spagnuolo M (2001) Similarity measures for blending polygonal shapes. *Comput Graph* 25(1):13–27
- Amenta N, Bern M, Kamvysselis M (1998) A new Voronoi-based surface reconstruction algorithm. *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pp 415–421
- Shima A, Sasaki T, Ohtsuki T, Wakimoto Y (1996) 64-bit RISC-based Series 15 NURBS interpolation. *FANUC technical review* 9(1):23–28
- Herweck P (1996) Siemens SINUMERIK 840-D, vol 10. *Siemens Autom Technol Mag*, 4 p
- Shpitalni M, Koren Y, Lo CC (1994) Realtime curve interpolators. *Comput Aided Des* 26(11):832–838
- Yeh SS, Hsu PL (2002) Adaptive-feedrate interpolation for parametric curves with a confined chord error. *Comput Aided Des* 34(3):229–237
- Farouki RT, Tsai YF (2001) Exact Taylor series coefficients for variable-feedrate CNC curve interpolators. *Comput Aided Des* 33(2):155–165
- Xu ZM, Chen JC, Feng ZJ (2002) Performance evaluation of a real-time interpolation algorithm for NURBS curves. *The International Journal of Advanced Manufacturing Technology* 20(4):270–276
- Tsai MC, Cheng CW (2003) A real-time predictor-corrector interpolator for CNC machining. *J Manuf Sci Eng* 125(3):449–460
- Akhavan Niaki F, Pleta A, Mears L (2018) Trochoidal milling: investigation of a new approach on uncut chip thickness modeling and cutting force simulation in an alternative path planning strategy. *Int J Adv Manuf Technol* 97(1):641–656
- Piegle L, Tiller W (1997) *The NURBS book*. Springer, New York

19. Yau HT, Lin MT, Tsai MS (2006) Real-time NURBS interpolation using FPGA for high speed motion control. *Computer Aided Design* 38(10):1123–1133
20. Farouki RT, Tsai YF, Yuan GF (1998) Contour machining of free-form surfaces with real-time PH curve CNC interpolators. *Computer Aided Geometric Design* 16(1):61–76
21. Tsai YF, Farouki RT, Feldman B (2001) Performance analysis of CNC interpolators for time-dependent feedrates along PH curves. *Computer Aided Geometric Design* 18(3):245–265
22. Yau HT, Chen JS (1997) Reverse Engineering of Complex Geometry Using Rational B-Splines. *The International Journal of Advanced Manufacturing Technology* 13(10):548–555
23. Held M (1998) Voronoi diagrams and offset curves of curvilinear polygons. *Computer Aided Design* 30(4):281–300
24. Lee DT (1982) Medial axis transformation of a planar shape. *IEEE Pattern Anal Mach Intell PAMI-4(4)*:363–369
25. Amenta N, Choi S, Kolluri RK (2001) 2001, “‘The Power Crust, Unions of Balls, and the Medial Axis Transform’”, *Computational Geometry: Theory and Applications* 19(2–3):127–153
26. Kuo CC, Yau HY (2003) Reconstruction of Virtual Parts from Unorganized Scanned Data for Automated Dimensional Inspection. *J Comput Inf Sci Eng (JCISE) Trans ASME* 3(1):76–86
27. Kuo CC, Yau HY (2006) A New Combinatorial Approach to Surface Reconstruction with Sharp Features. *IEEE Transactions on Visualizations and Computer Graphics* 12(1):73–83
28. Farin G, Sapidis N (1989) Curvature and the fairness of curves and surfaces. *IEEE Comput Graphics Appl* 9(2):52–57
29. Merdol SD, Altintas Y (2008) Virtual simulation and optimization of milling operations—Part I: process simulation. *J Manuf Sci Eng* 130(5):510041–5100412
30. Altintas Y, Engin S (2001) Generalized Modeling of Mechanics and Dynamics of Milling Cutters. *CIRP Ann* 50(1):25–30
31. Yau HT, Wang SY, Chang HC, Chang CH (2022) Direct computation of instantaneous cutting force in real-time multi-axis NC simulation. *Int J Adv Manuf Technol* 119(11):6967–6978
32. (2018) *Cutting Tools 2019–2020 NACHI*, ed, pp 132–137

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.