


An improved positioning method for flank milling of S-shaped test piece

Liwen Guan¹  · Jiao Mo¹ · Meng Fu¹ · Liping Wang¹

Received: 21 July 2016 / Accepted: 20 February 2017 / Published online: 11 March 2017
© Springer-Verlag London 2017

Abstract Accuracy detection for five-axis numerical control (NC) machine tools that can truly reflect their practical machining precision is of crucial importance. Standard test pieces are commonly employed for this purpose. However, poor accuracy detection performance is obtained when the test pieces used here are applied to five-axis NC machine tools. This paper introduces a new S-shaped test piece that is exclusively designed for the precision detection of five-axis NC machine tools. The S-shaped test piece integrates numerous characteristics associated with aviation parts and has been widely adopted by machine tool makers. This article presents a numerical model of the latest S-shaped test piece and shows that its side surfaces represent typical undevelopable ruled surfaces. The curvature changes along the ruled lines inevitably produce a theoretical error. Thus, machining methods seek to reduce the theoretical error as much as possible. Based on basic summaries of five existing positioning algorithms, a novel algorithm is proposed to position the tool head using three points, wherein two are tangential to the top and bottom boundary curves, respectively, and the third is tangential to the midpoint of the ruled line. The proposed positioning algorithm together with five existing positioning algorithms is applied to the S-shaped test piece, and a numerical error performance analysis is conducted. The results indicate that the machined surface reduces the theoretical error by at least 96% compared to all the existing numerical positioning algorithms except for Redonnet's algorithm. Compared with Redonnet's algorithm, the accuracy of the proposed algorithm is equivalent, although

the proposed algorithm reduces the calculation time by 62.7%, and is not sensitive to the initial values. Hence, the computational process demonstrates that the proposed method is efficient, robust, and universal. Finally, simulation results were confirmed through an actual machining experiment.

Keywords Five-axis numerical control machining · Flank milling · S-shaped test piece · Undevelopable ruled surface · Tool path generation

1 Introduction

Accuracy detection for NC machine tools is important for improving their performance. A number of significant developments have been achieved during the past decades worldwide. Direct methods of accuracy detection usually apply particular apparatus [1–6] for measurements. However, these methods are generally aimed at static accuracy testing or 3-axis linkage machining accuracy testing. In contrast, indirect methods utilize test pieces [7, 8], which reflect dynamic errors, and are convenient to implement. However, conventional test samples usually present several shortcomings, and are not appropriate for use in the accuracy detection of five-axis NC machine tools [9, 10]. Out-of-tolerance of surface quality are often obtained when processing parts with complex curved surfaces, though corresponding machine tools have already been certified using other test pieces [8].

Considering the difficulty of adequately evaluating the performance of five-axis NC machine tools with existing test samples, Chengdu Aircraft Industry Group developed an S-shaped test piece for accuracy detection [11]. Because the side surfaces of the S-shaped test piece are typical undevelopable ruled surfaces that integrate numerous characteristics associated with aviation parts, the test piece adequately evaluates the

✉ Liwen Guan
guanlw@mail.tsinghua.edu.cn

¹ Institute of Manufacturing Engineering, Department of Mechanical Engineering, Tsinghua University, Beijing 100084, China

accuracy and performance of five-axis NC machining. In 2012, application was made for including this S-shaped test piece as an international standard at the 74th meeting of the International Organization for Standardization (ISO) [12], and it was agreed to be added to the draft international standard (DIS) at the 79th meeting of the ISO/TC39SC2 in May 2016. Therefore, it is of great theoretical and practical significance to study this test piece and its processing. However, unlike other test pieces, theoretical error is generated due to the twist angle during the side milling process.

The side milling of ruled surfaces has been frequently studied by researchers [13–17]. The conventional single point offset (SPO) algorithm [18, 19] was proposed first. It became the standard method for the flank milling of the developable ruled surfaces in practice and is also presently the standard method given in CAD/CAM software systems. Subsequently, Liu [20] presented the double point offset (DPO) method that can reduce machining errors compared to the SPO algorithm. However, the error remains quite large. For this reason, Redonnet et al. [21, 22] developed a method that positions the cylindrical cutter with three-point tangential (TPT). Because seven transcendental equations must be solved simultaneously to obtain each tool position, the required system of equations limits the robustness of the TPT method, and a relatively long computation time is required. Then Bedi et al. [23] developed a strategy for rolling a cylindrical cutting tool along two boundary curves with double-point tangential (DPT). This method requires the solution of only two transcendental equations, and is therefore simpler than the TPT method. On the basis of the TPT and DPT algorithms, Liang et al. [24] proposed the new double points tangential (NDPT) method. The NDPT algorithm selects two points on the boundary curves at equal parametric values, and then offsets the two points at a distance d along the point's main normal. Therefore, only a single transcendental equation requires solution with the unknown value d . However, the transcendental equation in this case is complex, and it involves no obvious reduction in computation time.

The methods discussed above involve mathematical analysis, such that each tool position can be obtained by solving some number of equations. However, some methods have been developed that cannot be expressed by mathematical formulas, such as the statistical method [25–27], subsection optimization method [28], least square method [29, 30], and others [31], but these methods usually require greater computation time than the analytical methods or involve some limitations that have not been properly applied in the CAD/CAM systems.

The present study proposes new strategy, where the cylindrical cutting tool is rolled along the two boundary curves with three tangential points, which is similar to the TPT

method, and the method of placing the tool tangential to the two boundary curves is based on the DPT method. Hence, this algorithm is denoted as the new three-point tangential (NTPT) algorithm. With the proposed method, the tool position can thus be completely defined by solving only four simple transcendental equations, which makes the algorithm much more efficient and robust.

The remainder of this paper is organized as follows. In Section 2, the numerical model of the S-shaped test piece is presented in detail. The characteristics of the S-shaped test piece are given in Section 3, and in Section 4, the NTPT algorithm is proposed based on five existing side milling methods. The distribution of the theoretical error of the S-shaped test piece is analyzed, and the value compared with the errors associated with the other five algorithms in Section 5. The machining experiment verifying the results of the simulation is presented in Section 6. Finally, the conclusions of this paper are given in Section 7.

2 Description of the S-shaped test piece

A simplified theoretical modeling of the S-shaped test piece is established as follows. First, a Cartesian coordinate system is established at an arbitrary origin. Then, the ruled surface A is defined using two quasi-uniform cubic rational B-splines [32–34] in an “S” shape, which are further defined by two sets of control points, P_i and Q_i . Similarly, the ruled surface B is also defined by two sets of control points, M_i and N_i . Moreover, the ruled surfaces in the S-shaped test piece are obtained by linearly blending at equal parameter values between the two corresponding curves. Finally, the S-shape fillet defined by the two ruled surfaces is given with a rectangular base, as shown in Fig. 1. Next, the quasi-uniform cubic rational B-splines of surface A and B are shown in Fig. 2 and Fig. 3. The control points are also included in the figures. Then, the coordinates of arbitrary points with different values of Z can be deduced from the two quasi-uniform cubic rational B-splines at $Z = 0$ and $Z = 30$.

From the figures and the description presented above, it is clearly observed that the configuration of the S-shaped test

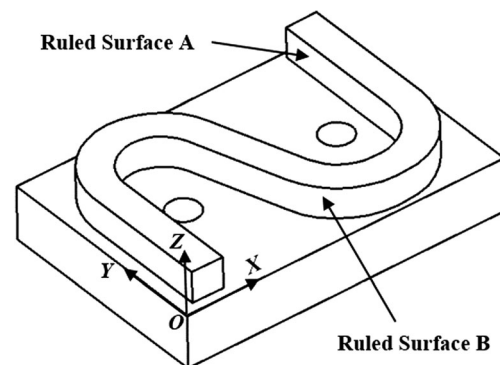


Fig. 1 The S-shaped test piece

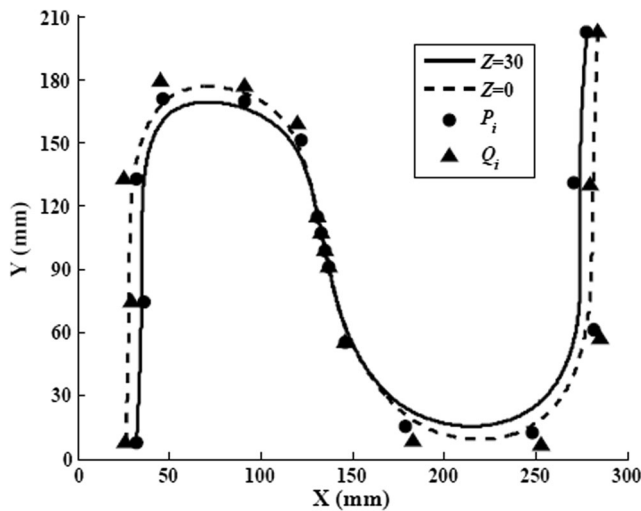


Fig. 2 Curves of surface A

piece offers significant advantages for accuracy detection, which include the following.

Both ruled surfaces comprise 13 B-spline curve sections, enabling the S-shaped test piece to integrate substantial characteristics.

Viewed from the positive direction of the Z-axis, the two boundary curves of each ruled surface are not coincident. Therefore, the orientation of the ruled line changes continuously.

When X values are small, the Y values of the bottom boundary curves are greater than those of the top boundary curves. Then, this relationship reverses as the X values increases. At the point of reversal from open-angle to closed-angle, the curves are straight lines, which avoids serious overcut.

3 Characteristics of the S-shaped test piece

We must first establish related definitions before we focus on the characteristics of the S-shaped test piece. It is clear from

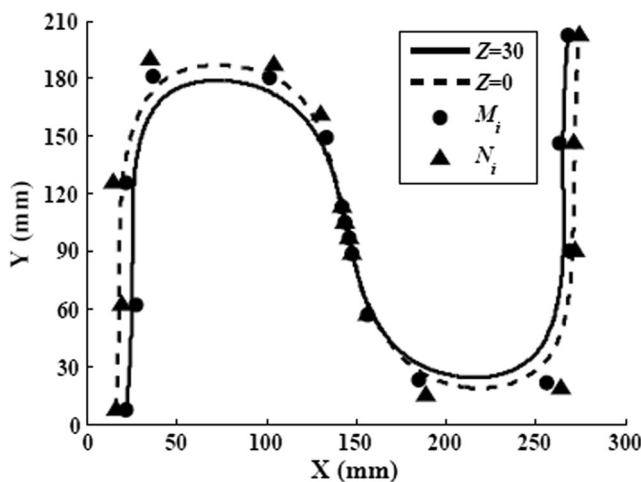


Fig. 3 Curves of surface B

the figures and previous description that each ruled surface is generated by a set of straight lines based on two boundary curves $C_1(u)$ and $C_2(u)$. At equal parametric values u_0 , vectors $N_1(u_0)$ and $N_2(u_0)$ are the main normal vectors of the top and bottom boundary curves, respectively.

Figure 4 presents the distributions of the curvature of the ruled surfaces. The inconsistent size and direction of curvature are the primary characteristics of the S-shaped test piece. On the one hand, these inconsistencies increase the difficulty of error estimation by mathematical analysis; whereas, on the other hand, non-uniform changes in the curvature will directly lead to changes in the milling force during processing, resulting in the vibration of tools and parts, which destroys the stability of the milling machine [10]. Hence, compared to other test pieces, the S-shaped test piece incurs higher requirements for an NC machine tool.

However, we expect that the measurement of processing error can effectively reflect the performance of the machine tool. Therefore, it is required to decrease the theoretical error as much as possible. The theoretical error is mainly produced by the twist angle γ , which is a typical feature of undevelopable ruled surface, as shown in Fig. 5, where the projection of the upper and lower S-shaped curves in the view of the ruled line crossed each other rather than coinciding, resulting in a finite value of γ .

As shown in Fig. 6, the change in γ is irregular along the boundary curves. Therefore, sharp changes in the tool axis vector are required when flank milling with cylindrical tools. This further explains the detection ability of multi-axis linkage due to the high demand of processing capacity.

Considering the cause of theoretical error, the algorithm should fit the ruled surface to the greatest extent possible. In addition, the formulation should be expressed regularly to reduce the vibration of tools and parts. These considerations will be further investigated in the following section.

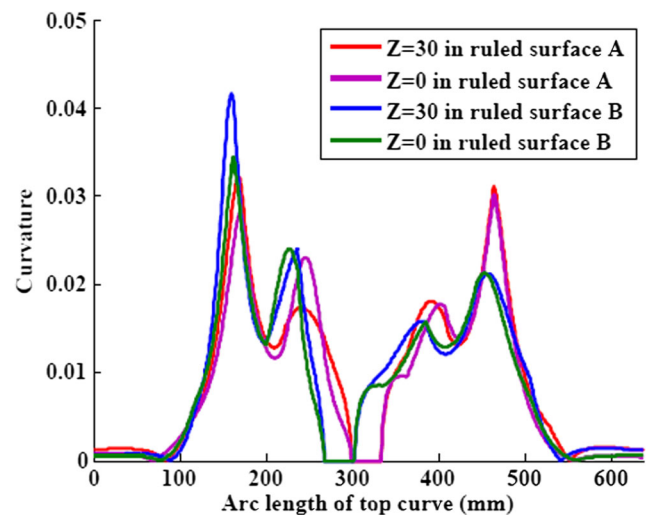


Fig. 4 The distribution of curvature

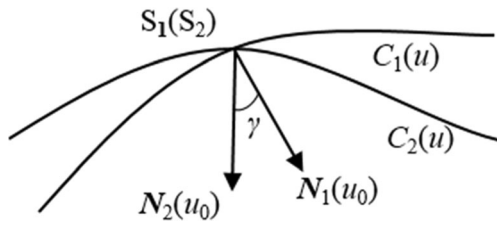


Fig. 5 The projection of curves

4 The NTPT Algorithm

In this section, we first provide basic summaries of five existing positioning algorithms and then present the NTPT algorithm in detail.

4.1 Existing positioning algorithms

With the SPO algorithm, the tool axis is positioned collinear to the ruled line with an offset distance R , which is the radius of the cylindrical tool, and the perpendicular vector from S_1 to the tool axis is $N_1(u_0)$. (note that the main normal of another curve is sometimes also applied.) Then, as shown in Fig. 7, where we take the top curves as an example, the tool axis vector and the tool tip are expressed as $S_1 - S_2$ and $S_2 + RN_1(u_0)$, respectively.

The DPO algorithm is illustrated in Fig. 8. Here, the tool axis vector $M_1 - M_2$ is obtained, and the tool tip is calculated by making a vertical line from S_2 to the tool axis and finding the crossing point.

- S_0 — The midpoint of S_1S_2 .
- P_1 — The midpoint of S_0S_1
- P_2 — The midpoint of S_0S_2
- $N_{P1}(u_0)$ —The normal vector at point P_1
- $N_{P2}(u_0)$ —The normal vector at point P_2
- M_1 —Calculated by $M_1 - P_1 = RN_{P1}(u_0)$
- M_2 —Calculated by $M_2 - P_2 = RN_{P2}(u_0)$

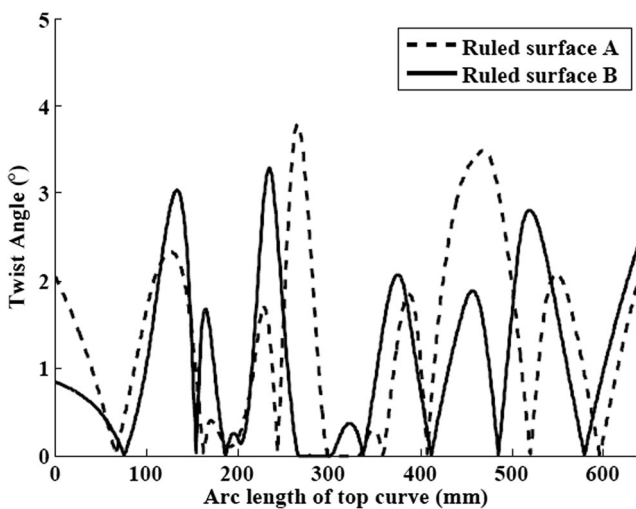


Fig. 6 The distribution of twist angle

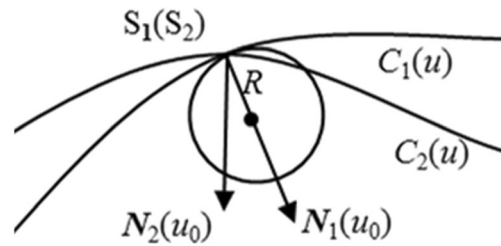


Fig. 7 Schematic of the SPO algorithm

Next, according to the DPT method (illustrated in Fig. 9), the two equations below can be obtained to maintain the tool tangential to the designed surface at two points (S_1 and S_2) with the parameter $u = u_0$:

$$a_1 \cos(\theta) + b_1 \sin(\theta) + R = 0$$

$$a_2 \cos(\varphi) + b_2 \sin(\varphi) - R = 0$$

where

$$a_1 = (M_1 - M_2) \cdot N_1(u_0) - RB_2(u_0) \cdot N_1(u_0) \sin \varphi - RN_2(u_0) \cdot N_1(u_0) \cos \varphi$$

$$b_1 = (M_1 - M_2) \cdot B_1(u_0) - RB_2(u_0) \cdot B_1(u_0) \sin \varphi - RN_2(u_0) \cdot B_1(u_0) \cos \varphi$$

$$a_2 = (M_1 - M_2) \cdot N_2(u_0) - RB_1(u_0) \cdot N_2(u_0) \sin \theta + RN_1(u_0) \cdot N_2(u_0) \cos \theta$$

$$b_2 = (M_1 - M_2) \cdot B_2(u_0) - RB_1(u_0) \cdot B_2(u_0) \sin \theta + RN_1(u_0) \cdot B_2(u_0) \cos \theta$$

To obtain the tool position, only two transcendental equations are needed to numerically solve for the two unknowns θ and φ . The tool position is easily determined by the tool axis vector ($M_1 - M_2$), and tool tip was obtained by the same means as employed by the DPO algorithm.

The NDPT algorithm is illustrated in Fig. 10. The differences between the NDPT algorithm and the DPO algorithm include that the NDPT algorithm substitutes P_1 , P_2 , $N_1(u_0)$, and $N_2(u_0)$ with S_1 and S_2 , and their normal vectors $N_1(u_0)$ and $N_2(u_0)$, respectively. In addition, M_1 and M_2 reside on $N_1(u_0)$ and $N_2(u_0)$, respectively, with an unknown distance parameter d , rather than with the known value of R . Therefore, $\|S_1 - M_1\|_2 = \|S_2 - M_2\|_2 = d$, and M_1 and M_2 are joined as the tool axis. The only equation of constraint is that the distances from S_1 and S_2 to the tool

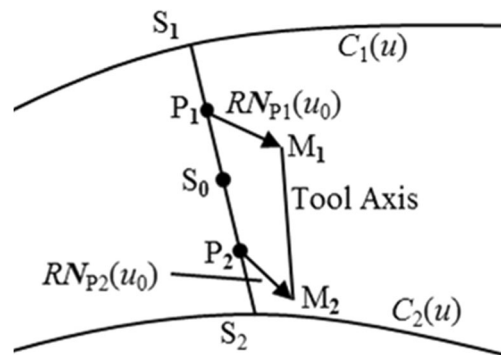


Fig. 8 Schematic of the DPO algorithm

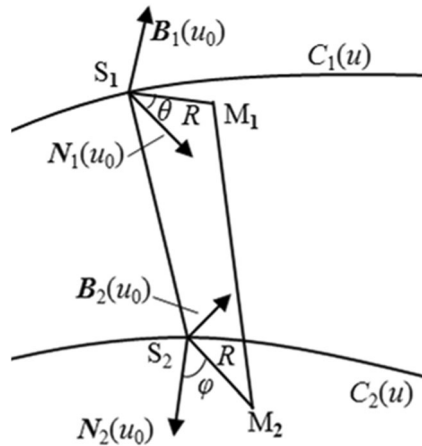


Fig. 9 Schematic of the DPT algorithm

axis are equal to R (See Fig. 10). Thus, according to the established geometric relationships, we obtain

$$\frac{\sqrt{d^2 - R^2}}{d} = \frac{2d \sin^2(\gamma/2)}{(l^2 + 4d^2 \sin^2(\gamma/2))^{1/2}}$$

where l is the length of the rule and γ is established in accordance with the parametric relationship $u = u_0$. Finally, the tool position is determined by the tool tip O_2 and the tool axis vector $M_1 - M_2$ when the value of d is calculated.

In accordance with the TPT algorithm illustrated in Fig. 11 and Fig. 12, the tool position can be completely defined in the Plane $A_1A_2B_2B_1$ by the angle δ made between the tool axis and the ruled line and the position of the point S_0 . The TPT algorithm employs a system of seven transcendental equations that must be solved. The condition for tangency between the tool and the boundary curve $C_1(u)$ is translated to the condition of unicity of the intersection between $C_1(u)$ and the ellipse $\mathbf{EII}_1(w)$ of the tool. The vector $\mathbf{tanEII}_1(w)$ is tangent to $\mathbf{EII}_1(w)$, so that the unicity of the intersection can be expressed by the fact that $\mathbf{tanEII}_1(w)$ and $T_1(u_0)$ have the

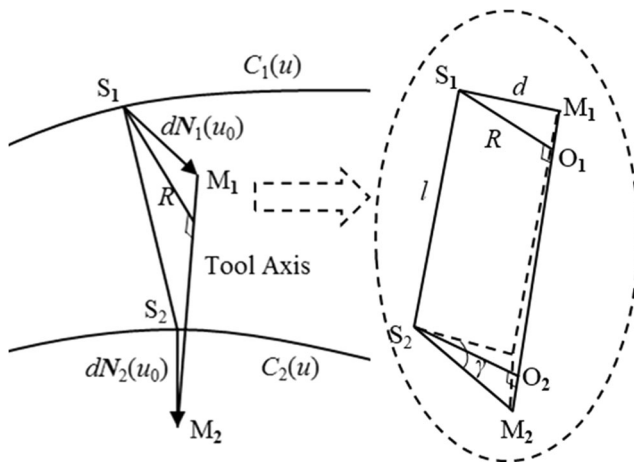


Fig. 10 Schematic of the NDPT algorithm

same slope at the point of intersection, as does $\mathbf{tanEII}_2(w)$ and $T_2(u_0)$. Thus, we obtain the following equations in the same reference:

$$\begin{aligned} C_1(u_1)_x &= \mathbf{EII}_1(w_1)_x & C_1(u_1)_y &= \mathbf{EII}_1(w_1)_y \\ C_2(u_2)_x &= \mathbf{EII}_2(w_2)_x & C_2(u_2)_y &= \mathbf{EII}_2(w_2)_y \\ \frac{T_{1y}}{T_{1x}} &= \frac{\mathbf{tanEII}_{1y}(w_1)}{\mathbf{tanEII}_{1x}(w_1)} & \frac{T_{2y}}{T_{2x}} &= \frac{\mathbf{tanEII}_{2y}(w_2)}{\mathbf{tanEII}_{2x}(w_2)} & l_1 + l_2 &= l \end{aligned}$$

where the subscripts express the component along the x , y , or z , axes, and l is the length of the rule divided into segments l_1 and l_2 by point S_0 . Then, the tool position is determined after solving the system of these seven equations.

4.2 Mathematic definition of the NTPT algorithm

The NTPT algorithm is proposed based on an investigation of the TPT algorithm. The TPT method establishes of four coordination references, and the equations should be finally transformed with respect to only one reference. It is readily visible that the more tangential points involved in a method, the less theoretical error the undeveloped surface will show, so the TPT algorithm can be expected to involve less theoretical error than the SPO, DPO, DPT and NDPT algorithms. However, the TPT method is difficult to understand and inconvenient to calculate. Due to the complexity of the expression, there is no solution to these equations when the given initial values are inappropriate, which is one reason why the

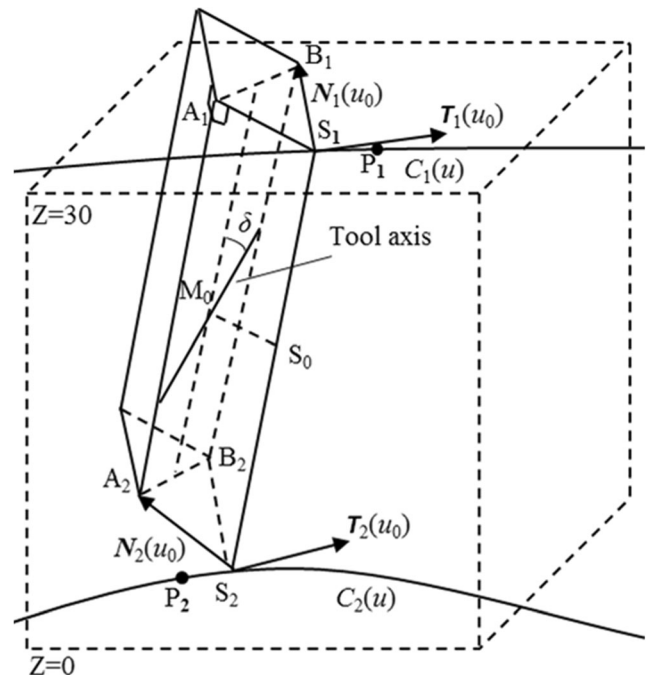


Fig. 11 Schematic of the TPT algorithm

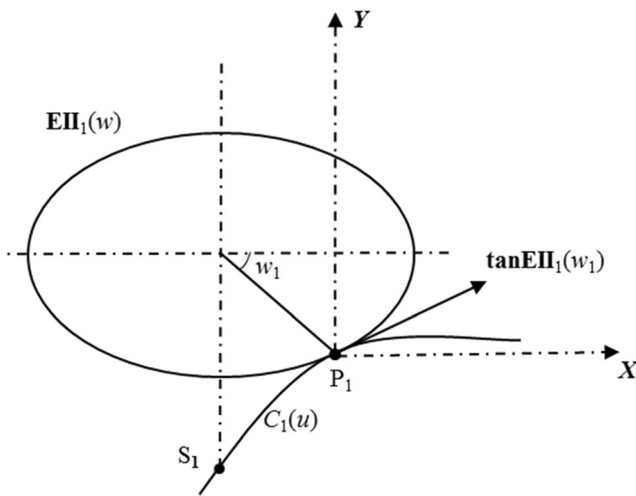


Fig. 12 The ellipse $EII_1(w)$ of the tool

initial values are so sensitive. In addition, the TPT method includes an absolute value function and division, produces a breakpoint $\gamma = 0$ within the function. However, numerical methods for solving nonlinear equations require that the function be continuous on the interval, so the solution must provide both positive and negative initial values for each position in order to suitable for different conditions, and this is the second reason why the initial values are sensitive. Another cause of this problem is the order of convergence of the nonlinear equations. Nevertheless, it is extremely difficult to obtain the order of convergence because of the intricate formulas. Thus, the approximate order of convergence is derived by calculating the actual iterative process and fitting the results, as shown in Fig. 13. Then, the order of convergence can be found to be about 1.043, near a linear iteration, which demonstrates the slow convergence speed and sensitivity of the initial values [35].

The TPT algorithm is constructed using two kinds of constraints. The first constraint is the position of S_0 , expressed by an unknown v ($0 \leq v \leq 1$, $v = 0$ is S_1 and $v = 1$ is S_2). The second constraint is the tangential relationship at P_1 and P_2 , which is equivalent to the relationship between θ and φ in the DPT algorithm. Therefore, we must obtain the unknowns v, θ, φ, u_1 (determine P_1) and u_2 (determine P_2). Next, we consider the equations: The first two equations are defined by the three collinear points M_0, M_1 , and M_2 (see Figs. 9 and 11). The vertical relation between the tool axis and $(M_i - S_i)$ ($0 \leq i \leq 2$) also provides three equations. Finally, we obtain five equations expressing equivalent constraints to those of the TPT method. However, the essential parameter γ still exists, resulting in a limit for solution. Therefore, we propose an improved NTPT algorithm.

The NTPT algorithm is illustrated in Fig. 14, where the position of the ruled line is defined by $u = u_0$. The first step is to establish the Frenet frames for both boundary curves at

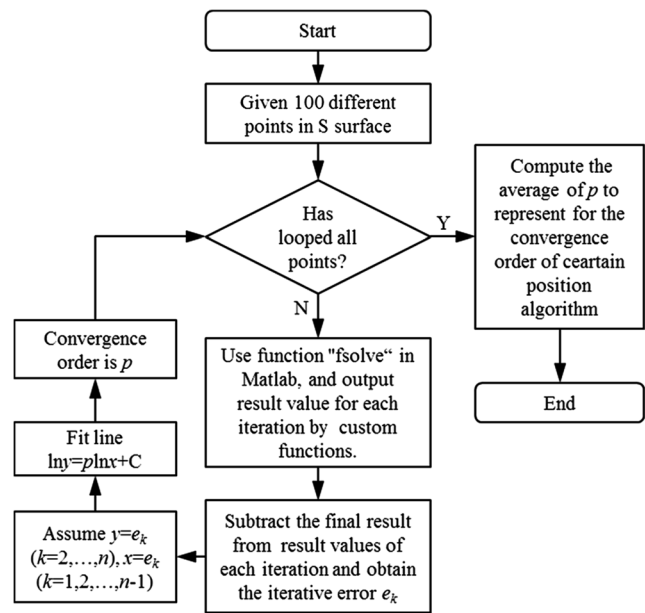


Fig. 13 Method to solve the order of convergence

tangential points P_1 and P_2 . The Frenet frame for P_1 is defined by the tangent $T_1(u_1)$, the main normal $N_1(u_1)$, and the binormal $B_1(u_1)$. Similarly, $T_2(u_2), N_2(u_2)$, and $B_2(u_2)$ establish the Frenet frame at P_2 . To position a cylindrical tool tangential to curve $C_1(u)$ at P_1 , the vector $(M_1 - P_1)$ must pass through a plane described by $N_1(u_1)$ and $B_1(u_1)$ at an angle θ towards $N_1(u_1)$. Here, for $T_1(u_1)$ perpendicular out of the page, a positive rotation is counter-clockwise. Analogously, the vector $(M_2 - P_2)$ must pass through a plane described by $N_2(u_2)$ and $B_2(u_2)$ at an angle φ towards $N_2(u_2)$. Therefore, we obtain

$$M_1 - P_1 = R \cos(\theta) N_1(u_1) + R \sin(\theta) B_1(u_1) \tag{1}$$

$$M_2 - P_2 = R \cos(\varphi) N_2(u_2) + R \sin(\varphi) B_2(u_2) \tag{2}$$

In addition, for cylindrical tools, the axis vector $(M_1 - M_2)$ must be perpendicular to the two vectors $(M_1 - P_1)$ and $(M_2 - P_2)$, such that $(M_1 - M_2) \cdot (M_1 - P_1) = (M_1 - M_2) \cdot (M_2 - P_2) = 0$. Thus, we eliminate M_1 and M_2 from Eqs. (1) and (2), which yields

$$a_1 \cos(\theta) + b_1 \sin(\theta) + R = 0 \tag{3}$$

$$a_2 \cos(\varphi) + b_2 \sin(\varphi) - R = 0 \tag{4}$$

where

$$\begin{aligned} a_1 &= (M_1 - M_2) \cdot N_1(u_1) - R B_2(u_2) \cdot N_1(u_1) \sin \varphi - R N_2(u_2) \cdot N_1(u_1) \cos \varphi \\ b_1 &= (M_1 - M_2) \cdot B_1(u_1) - R B_2(u_2) \cdot B_1(u_1) \sin \varphi - R N_2(u_2) \cdot B_1(u_1) \cos \varphi \\ a_2 &= (M_1 - M_2) \cdot N_2(u_2) - R B_1(u_1) \cdot N_2(u_2) \sin \theta + R N_1(u_1) \cdot N_2(u_2) \cos \theta \\ b_2 &= (M_1 - M_2) \cdot B_2(u_2) - R B_1(u_1) \cdot B_2(u_2) \sin \theta + R N_1(u_1) \cdot B_2(u_2) \cos \theta \end{aligned}$$

Equations (3) and (4) are two transcendental equations including four unknown variables. Further, we know that point M_0 is unused at the tool axis with a distance R from S_0 .

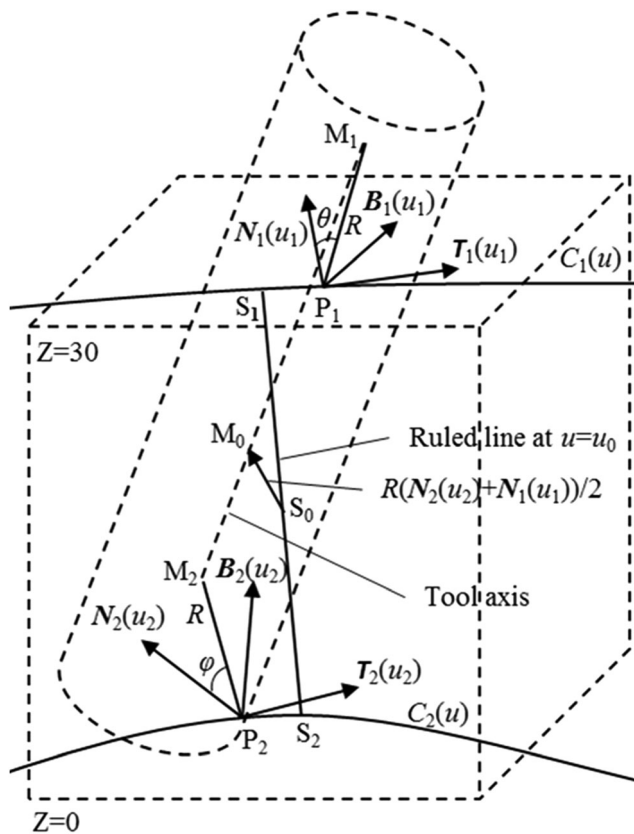


Fig. 14 Schematic of the NTPT algorithm

Therefore, we obtain another equation with the three collinear points M_0 , M_1 , and M_2 :

$$\frac{M_{0x}-M_{1x}}{M_{2x}-M_{1x}} = \frac{M_{0y}-M_{1y}}{M_{2y}-M_{1y}} = \frac{M_{0z}-M_{1z}}{M_{2z}-M_{1z}} \tag{5}$$

which include two equations,

$$\frac{M_{0x}-M_{1x}}{M_{2x}-M_{1x}} = \frac{M_{0y}-M_{1y}}{M_{2y}-M_{1y}} \tag{6}$$

$$\frac{M_{0y}-M_{1y}}{M_{2y}-M_{1y}} = \frac{M_{0z}-M_{1z}}{M_{2z}-M_{1z}} \tag{7}$$

The system of four equations (Eqs. (3), (4), (6), and (7)) with four unknown parameters u_1 , u_2 , θ , and φ can be quickly solved. Finally, we can establish the position of the cylindrical tool with the tool tip M_2 and the tool axis vector $(M_1 - M_2)$.

- S_0 —The midpoint of S_1S_2 .
- S_1 —At $C_1(u)$ with parameter $u = u_0$.
- S_2 —At $C_2(u)$ with parameter $u = u_0$.
- P_1 —At $C_1(u)$ with parameter $u = u_1$.
- P_2 —At $C_2(u)$ with parameter $u = u_2$.
- M_0 —At the tool axis with a distance R from S_0 .
- M_1 —At the tool axis with a distance R from S_1 .
- M_2 —At the tool axis with a distance R from S_2 .

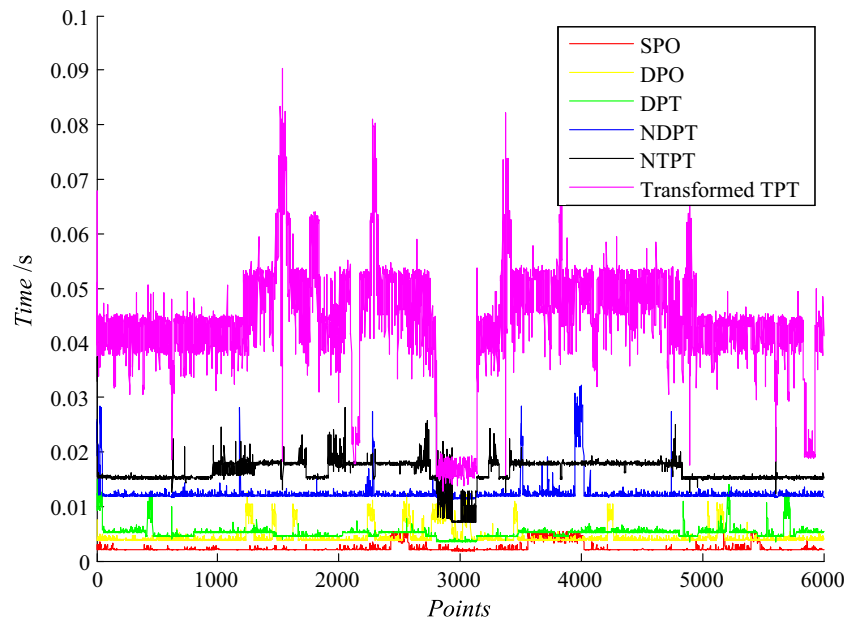
In a sense, the proposed NTPT algorithm is similar to DPT and TPT algorithms. However, The DPT tangential positioning strategy places the cutting tool tangential to the top and bottom boundary curves of the ruled surface at an equal parametric value u_0 , while the NTPT algorithm places the cutting tool tangential to the top and bottom boundary curves of the ruled surface at different parametric values u_1 and u_2 . In addition, the TPT tangential positioning strategy positions the tool axis in the Plane $A_1A_2B_2B_1$ (see Fig. 11), while the NTPT algorithm abolishes the vertical relation between the tool axis and $(M_0 - S_0)$ to make the tool axis free. Furthermore, for the purpose of reducing the number of equations, we define S_0 as the midpoint of S_1S_2 , so the NTPT algorithm requires solving a system of four transcendental equations rather than seven.

Precisely comparing the geometric differences between NTPT and TPT algorithms showed the only difference to be the definitions related to S_0 . Because the initial value of the v was equal to 0.5 (for fastest solution) in the TPT algorithm, it is also the solution for NTPT if the TPT gives solution at $v = 0.5$, which increases the solving speed to an extent. The NTPT method also fixes the variable v when the vertical restraint between tool axis and (M_0-S_0) is removed. This renders results of TPT and NTPT algorithms similar, as shown in next section. However, it increases the convergence order to 1.89, which speeds up convergence and improves the robustness of initial values.

For the purpose of efficient direct comparisons, MATLAB was employed to solve all six algorithms at the same 6000 points for an equivalent S-shaped test piece. Considering that the computation time is related to computer performance, MATLAB settings, and the environmental temperature were uniformly maintained, and equivalent steps were repeated 20 times for each algorithm. A computer with an Intel® Core™ i7 Q720 1.6 GHz CPU, 16 GB memory, and 256 GB SSD was employed in the numerical study. In addition, the initial values can seriously affect the results, so we set the zero (most algorithms) and the value of R (only the NDPT algorithm) as the generic initial value. However, due to the sensitivity of results with respect to initial values, the TPT method required different initial values. Nevertheless, even though we employed initial values near the final values, the TPT method also requires more time and iterations. For example, when initial values are (0.81, 0.81, -1.5, -1.5, 15.5, 15.5, 0.002) and the solution is (0.8062, 0.8052, -1.5764, -1.5652, 15.5453, 15.5024, 0.0018), the TPT method still required around 0.2401 s and 14 iterations, which is much greater than those required by other algorithms. For this reason, when comparing the time cost and iterations of NTPT to TPT method, only the transformed TPT is used. The computation time is shown in Fig. 15, and the average computation time and average number of iterations are listed in Table 1.

Although the three tangential points employed by the TPT and NTPT algorithms are very similar, the NTPT reduced the computation time by 62.7%, in roughly equivalent to the time

Fig. 15 Computation time at 6000 points as indicated using different methods



required to calculate the tool position by the NDPT algorithms. In addition, the convergence rate of the NTPT algorithm is very fast, and position calculations using the NTPT algorithm were less sensitive to initial values than for the case of the TPT. In the following section, we show that the NTPT algorithm reduces theoretical errors by greater than 96 and 99%, compared with the theoretical errors of the NDPT (or DPO or DPT) and SPO algorithms, respectively, and provides a similar performance to that of the TPT algorithm.

5 Error analysis of S-shaped test piece

The theoretical errors using the NTPT algorithm on the S-shaped test piece cannot be calculated like that of the SPO or DPO algorithms, because the adjacent tool positions affect the error with each other. Two methods were employed to calculate the theoretical error for the sake of reliable error calculation. One method (Fig. 16) is based on the cutter's swept volume, where the error is calculated by comparing the envelope surface of the tool

Table 1 Average computation time and average number of iterations using different methods

Algorithm	Computation time (s)	Number of iterations
SPO	0.0023	Null
DPO	0.0044	Null
DPT	0.0051	1.5422
NDPT	0.0123	6
Transformed TPT	0.0437	3.5746
NTPT	0.0163	3.4081

with the designed surface. Another method (Fig. 17) computes the minimum distance between each tool position and the designed surface of the S-shaped test piece.

Prior to conducting error analysis, we must first address a related issue. For the DPO method, if we replace points P_1 and P_2 with points S_1 and S_2 [20], we can obtain the more basic DPO algorithm. However, this does not change the limit of error (LE), defined as the difference between the highest and the lowest error values, but only changes the overcut error values and undercut error values, as illustrated in Fig. 18. Similarly, the overcut and undercut error values can be adjusted for any algorithm by changing the points used along the ruled line. Therefore, the overcut and

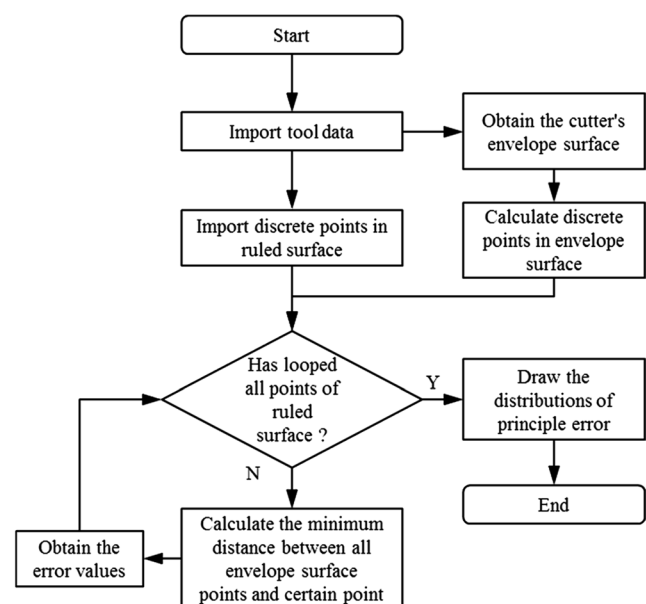


Fig. 16 Swept volume method used to assess theoretical errors

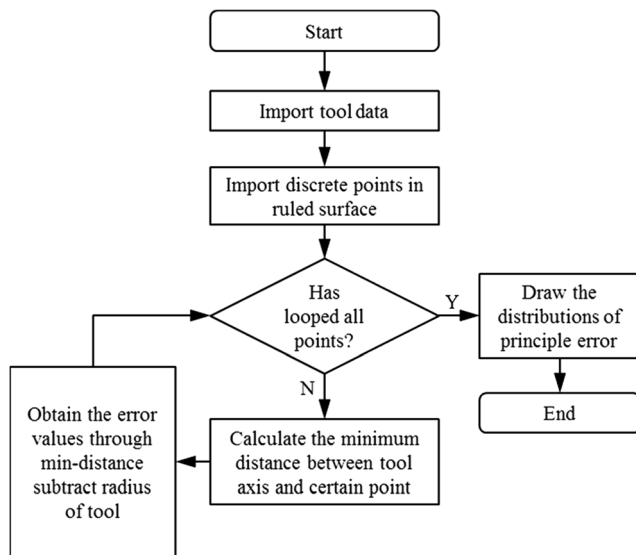


Fig. 17 Minimum distance method used to assess theoretical errors

undercut error values are not particularly useful for error analysis. In addition, given that γ is slightly larger for ruled surface A than for ruled surface B, only the error distribution of ruled surface A is discussed owing to space limitations. Our error analysis results indicate that the error distribution for ruled surface B is similar to that for ruled surface A, but is slightly smaller in the peak position. The global theoretical errors for the S-shaped test piece using the six different algorithms (SPO, DPO, DPT, NDPT, TPT, and NTPT) with $R = 10$ mm are shown in Fig. 19.

The general features and trends of the distributions are discussed as follows.

For the five existing methods, relatively larger error values appear in three areas where X is approximately equal to 40, 140, and 250 (the homologous arc lengths are about 130, 270, and 480, respectively). This indicates that the positioning strategies have high consistency. In addition, compared with Fig. 6, we find that these larger error values coincide with the maxima of γ , as would be expected.

The SPO algorithm yields the largest theoretical error, with an LE of roughly $25 \mu\text{m}$. Therefore, use of the SPO algorithm should be avoided due to its high theoretical error.

Although the working principles of the DPO, DPT, and NDPT methods differ, their theoretical error distributions

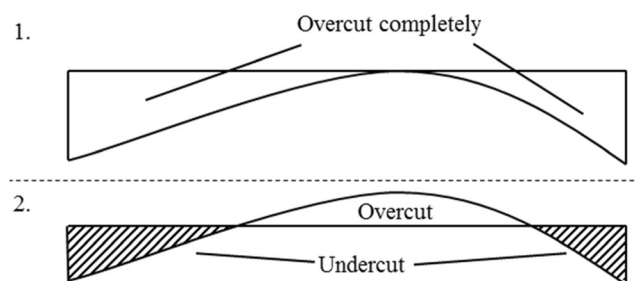


Fig. 18 Different error distributions along the ruled line

are similar, and provide LEs that are all about $5 \mu\text{m}$. Therefore, for machining the S-shaped test piece, the DPO algorithm is the best of the three due to its lower calculation time (see Subsection 3.1).

The TPT and NTPT algorithms provide similar results. Investigation of the G-code files for these two algorithms indicates that the values of u_1 and u_2 are not far from u , or more accurately, Δu_1 and Δu_2 are about 10^{-3} . In addition, the position of S_0 solved in the TPT algorithm is near the midpoint of S_1S_2 . For these reasons, the TPT and NTPT algorithms produce very similar results, which lead to the close tool-axis vector, as shown in Fig. 20. However, the NTPT algorithm provides for more rapid calculation and easier execution than the TPT algorithm.

For all the algorithms considered, the error value is about zero in most regions, while the different methods present slightly different trends. The SPO algorithm provides a peak error value near the boundary curve, while those of the DPT and NDPT algorithms are around the middle curve. In addition, the DPO algorithm presents a high undercut error in the boundary curves and a high overcut error in the middle curve. Error concentration is primarily avoided when X is about 40 and 250. The position $X \approx 270$ is where the open-angle to closed-angle transformation occurs [10]. Therefore, error concentration still exists.

In general, the TPT and NTPT algorithms provide the lowest theoretical errors, and the LE is decreased to approximately $0.2 \mu\text{m}$ by these two methods. The scope of the experimental test accuracy for the S-shaped test piece lies at around $60 \mu\text{m}$, as indicated from trial cut results obtained by the Chengdu Aircraft Industrial Group. Therefore, with the TPT and NTPT algorithms, the theoretical error can be ignored in accuracy detection.

To clarify the global theoretical errors shown in Fig. 19, we plot the error distribution curves obtained with the six algorithms at five typical heights in Fig. 21. Here, the results obtained from the DPO algorithm have been transformed from undercut to overcut to compare the LE directly.

The general features and trends of the distributions given in Fig. 21 are discussed as follows.

Compared with Fig. 6, the error distribution curves coincide well with the calculated distribution of γ . Generally, a larger theoretical error value coincides with a larger value of γ .

With the NTPT algorithm, the LE has been reduced by more than 99% compared to that of the SPO algorithm. In addition, the LE has been reduced in this case by up to 96% compared to the LE values of the DPO, DPT, and NDPT algorithms. Therefore, the NTPT algorithm is effective for the side milling of undevelopable ruled surfaces.

The LE obtained with NTPT algorithm has been decreased to less than 0.8% in most areas, except for the three peak areas (X is about 40, 140, and 250), where the LE is approximately

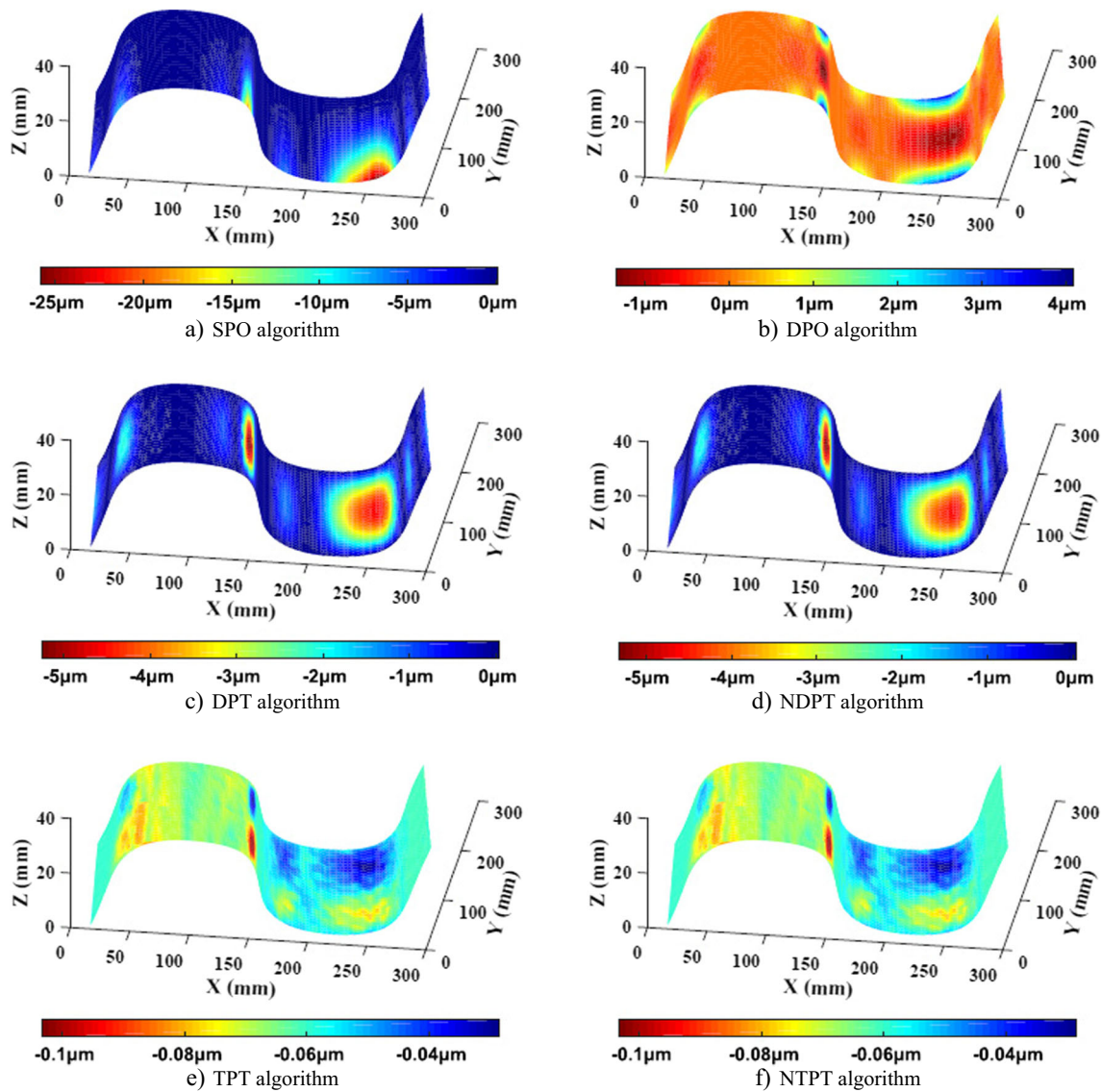


Fig. 19 The global theoretical errors obtained using the six different algorithms for the S-shaped test piece

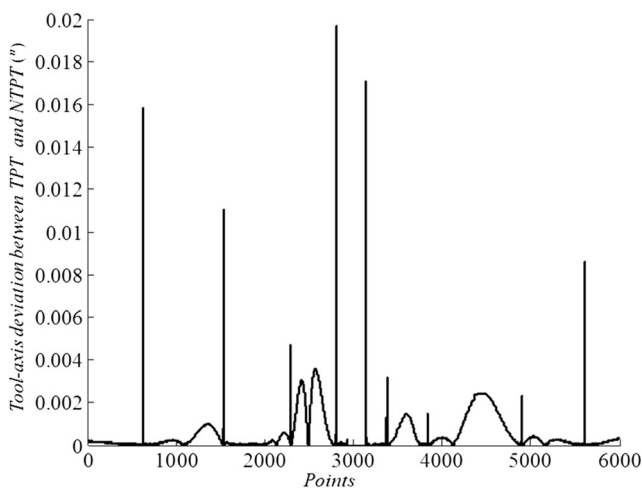


Fig. 20 The deviation of tool-axis vector between TPT and NTPT

4%. This result indicates that the NTPT algorithm performs better when γ is less than 2° .

Similar to the conclusion drawn earlier, theoretical error is reduced more at arc lengths of 130 and 480 than at 270. Therefore, the results of this study indicate that the NTPT algorithm performs much better under conditions where a transformation from open-angle to closed-angle does not occur.

The NTPT and TPT algorithms produced nearly equivalent results. However, the TPT algorithm must simultaneously solve a greater number of equations and is sensitive to initial values due to the breakpoint within the function. When solving nonlinear transcendental equations in MATLAB, the TPT algorithm requires an appropriate initial value while the NTPT algorithm does not, and the former requires several iterations to obtain a solution.

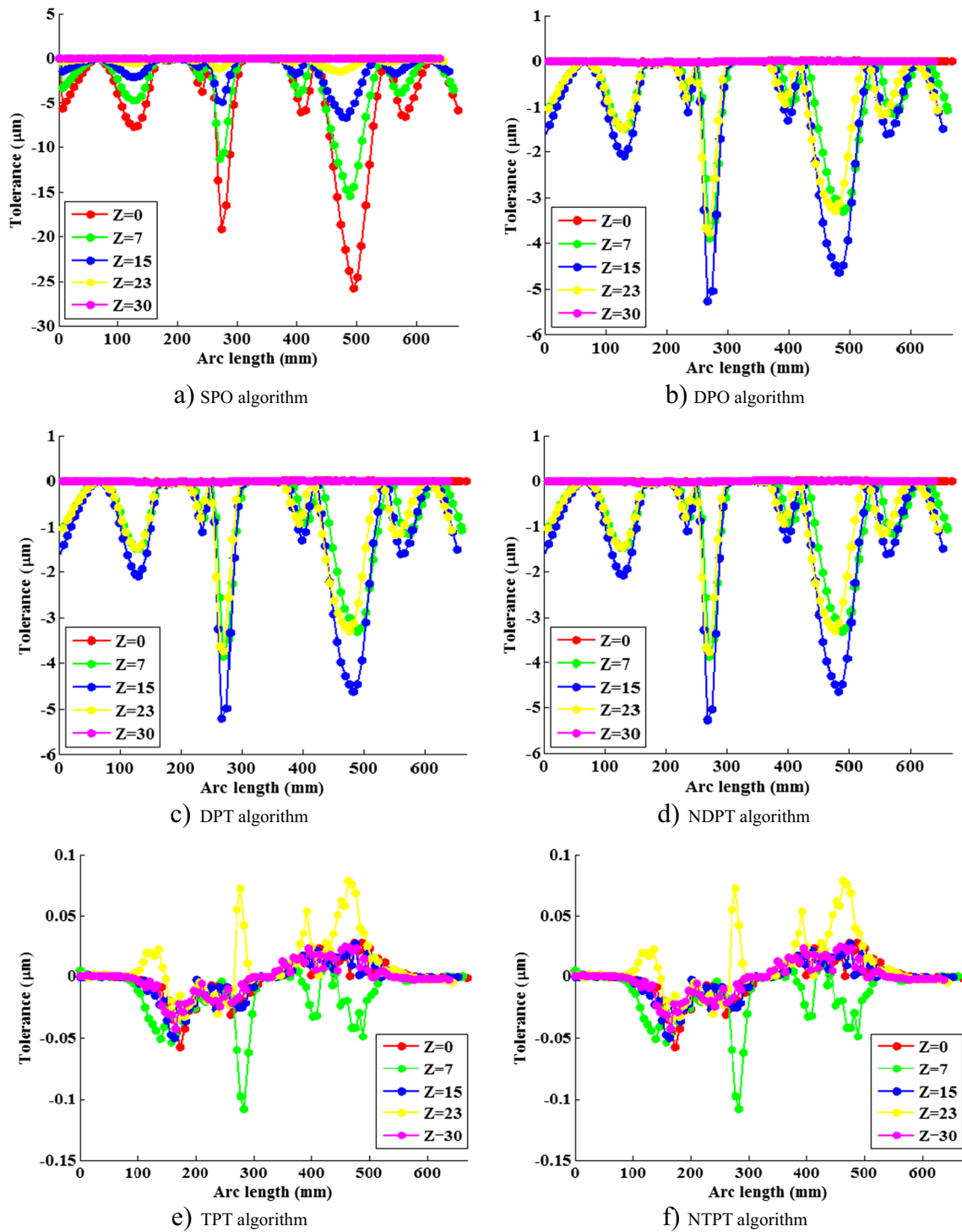
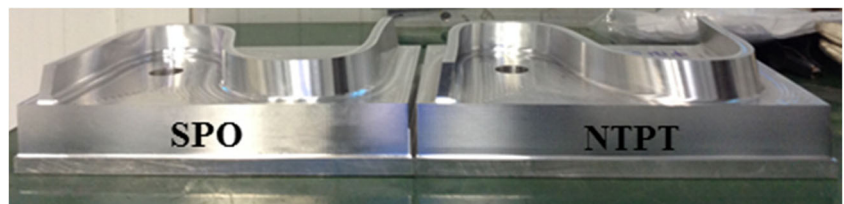


Fig. 21 The error distribution curves for the six different algorithms at different heights

Fig. 22 Two test pieces produced using the SPO and the NTPT methods



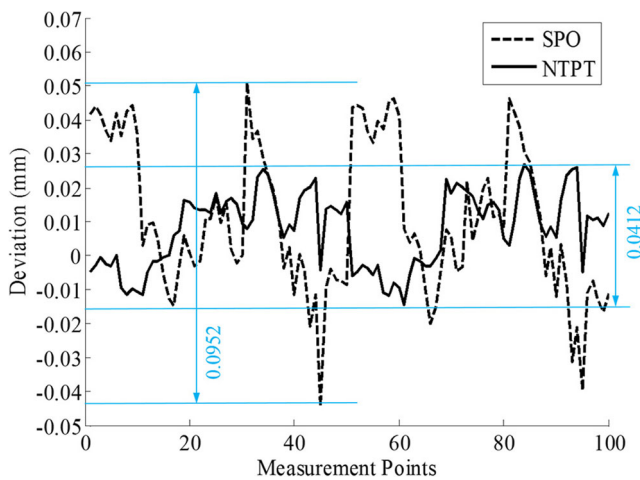


Fig. 23 The deviation of two test pieces

6 Machining experiment

A machining test was conducted to verify the results of the simulation. Two test pieces were flank milled on a ZOJE GMC2060u five-axis machine using the SPO and the NTPT methods, as shown in Fig. 22. The finish machining required about 30 min under both methods, which demonstrates that the NTPT method is effective. Then, measurements points identified using according to DIS (Appendix A) were obtained using a coordinate measuring machine, and the deviation of the machined surfaces from the designed ruled surfaces is graphed in Fig. 23. The picture shows that the NTPT algorithm reduced actual LE approximately by 50 μm , which indicates that the NTPT algorithm is better than ever expected.

7 Conclusions

This paper introduced a new standard S-shaped test piece for testing the performance of five-axis NC machining tools, and summarized the main existing machining methods (i.e., the SPO, DPO, DPT, NDPT, and TPT algorithms) using mathematical expressions for flank milling the non-developable surface. The main goals of this study were to introduce the new S-shaped test piece and implement an optimized numerical method for flank milling a ruled surface. The proposed NTPT algorithm is based on the DPT and TPT tool-positioning strategies. In contrast to the TPT algorithm, this algorithm requires the simultaneous solution of only four simple transcendental equations, which greatly reduces the calculation time required for tool positioning, and makes the algorithm more robust. In addition, the working principle of this method is simpler and the physical meaning of the parameters is more intuitive. The NTPT algorithm reduced the theoretical error by more than 99, 96, 96, and 96% compared with the SPO, DPO, DPT, and NDPT algorithms, respectively. Compared with the TPT algorithm, the LE obtained by the

NTPT algorithm was similar, while the calculation time was reduced by 62.7%, and the setting of initial values could also be ignored. The S-shaped test piece integrates a variety of practical features of interest, such that the low theoretical error of this method implies that the NTPT algorithm has extensive adaptability for different surfaces. In this way, these results provide sufficient reason to use the NTPT algorithm in CAD/CAM software systems.

The NTPT algorithm, in conjunction with the S-shaped test piece, can provide better accuracy detection measurements for five-axis NC machine tools. In addition, the ideas employed to establish the constraint equation for unknown arguments and to maintain a number of equations equal to the number of unknown arguments can be studied in depth. This may facilitate the development of similar algorithms (e.g., four points tangential), which may provide optimal tool positioning for the flank milling of undevelopable ruled surfaces. Finally, whether this method is effective for other surfaces with more important twists provides a good foundation for further investigation.

Acknowledgements This research was financially supported by the Major National S&T Program (Grant No. 2014ZX04014-031).

Appendix

Definition parameters of the S-shaped test piece used in this paper are listed in Table 2 and Table 3, where the ruled surface A is defined by two sets of control points, P_i and Q_i , presented in Table 2, and the ruled surface B is defined by two sets of control points, M_i and N_i . Measurement points according to DIS are presented in Table 4.

Table 2 Control points of ruled surface A

P_i	X	Y	Z	Q_i	X	Y	Z
P_0	22	7.5	30	Q_0	16	7.5	0
P_1	27	62	30	Q_1	19	62	0
P_2	22	126	30	Q_2	15	126	0
P_3	37	181	30	Q_3	35	190	0
P_4	102	180	30	Q_4	104	187	0
P_5	133	149	30	Q_5	130	161	0
P_6	142	113	30	Q_6	142	113	0
P_7	144	105	30	Q_7	144	105	0
P_8	146	97	30	Q_8	146	97	0
P_9	148	89	30	Q_9	148	89	0
P_{10}	156	57	30	Q_{10}	156	57	0
P_{11}	185	23	30	Q_{11}	189	15	0
P_{12}	256	22	30	Q_{12}	264	19	0
P_{13}	269	90	30	Q_{13}	272	90	0
P_{14}	263	146	30	Q_{14}	271	146	0
P_{15}	268	202.5	30	Q_{15}	274	202.5	0

Table 3 Control points of ruled surface B

M_i	X	Y	Z	N_i	X	Y	Z
M_0	32	7.5	30	N_0	26	7.5	0
M_1	36	74	30	N_1	29	74	0
M_2	32	133	30	N_2	25	133	0
M_3	46	171	30	N_3	45	179	0
M_4	91	170	30	N_4	91	177	0
M_5	122	151	30	N_5	120	159	0
M_6	131	115	30	N_6	131	115	0
M_7	133	107	30	N_7	133	107	0
M_8	135	99	30	N_8	135	99	0
M_9	137	91	30	N_9	137	91	0
M_{10}	146	55	30	N_{10}	146	55	0
M_{11}	179	15	30	N_{11}	183	8	0
M_{12}	248	12	30	N_{12}	253	6	0
M_{13}	282	61	30	N_{13}	285	57	0
M_{14}	271	131	30	N_{14}	280	130	0
M_{15}	278	202.5	30	N_{15}	284	202.5	0

Table 4 Measurements points in machining experiment

X	Y	Z	I	J	K
19.6103	32.7247	11	-0.974431	0.042103	0.220708
20.3919	57.9766	11	-0.971666	0.018916	0.235601
20.6858	83.2398	11	-0.971398	0.007038	0.237353
21.0788	108.5007	11	-0.974209	0.033341	0.223172
23.1626	133.6656	11	-0.969376	0.144681	0.198436
29.6884	157.9968	11	-0.898594	0.396065	0.18884
45.492	177.1908	11	-0.504397	0.830749	0.235458
69.6297	183.7191	11	-0.042217	0.965967	0.255196
94.6185	180.7799	11	0.263289	0.930146	0.255945
116.7231	169.0187	11	0.654739	0.725257	0.212882
131.0207	148.4131	11	0.909993	0.409811	0.062988
139.0988	124.5053	11	0.967295	0.25365	0.001463
145.2498	100.0009	11	0.970143	0.242536	0
151.5789	75.5446	11	0.955442	0.295179	0.000481
161.8422	52.5303	11	0.853035	0.521824	-0.00548
178.3731	33.6086	11	0.611259	0.788443	-0.068699
201.0421	22.9118	11	0.2263	0.961465	-0.156118
226.139	21.5356	11	-0.123295	0.973735	-0.191412
249.4168	30.5318	11	-0.60768	0.763343	-0.219164
262.8441	51.5056	11	-0.931217	0.305856	-0.198208
267.5502	76.2764	11	-0.981229	0.09377	-0.168512
268.7953	101.5043	11	-0.984529	0.017921	-0.174303
269.0294	126.7679	11	-0.979339	0.008105	-0.202063
269.4382	152.0294	11	-0.975474	0.025272	-0.218658
270.364	177.2764	11	-0.975304	0.046078	-0.216006
280.2761	177.3317	11	0.973585	-0.04886	0.223037
279.2772	152.1371	11	0.971303	-0.028402	0.236144
278.7725	126.9276	11	0.9725	-0.011808	0.232603
278.5188	101.714	11	0.977374	-0.013113	0.211114

Table 4 (continued)

X	Y	Z	I	J	K
277.559	76.5222	11	0.98042	-0.079948	0.17996
273.0981	51.7576	11	0.940102	-0.2919	0.176077
261.4583	29.5795	11	0.746942	-0.629449	0.214178
241.0226	15.3022	11	0.338712	-0.911887	0.231813
216.2639	11.3567	11	-0.017199	-0.978929	0.203477
191.5629	15.8281	11	-0.337359	-0.930755	0.141012
170.042	28.6794	11	-0.658414	-0.750977	0.050238
153.8217	47.8753	11	-0.843229	-0.537545	0.003176
142.8072	70.4985	11	-0.941581	-0.336784	-0.001311
136.055	94.7799	11	-0.970143	-0.242536	0
129.9395	119.2419	11	-0.970143	-0.242536	0
122.3741	143.2603	11	-0.914104	-0.40406	-0.033893
107.1894	163.0031	11	-0.582558	-0.794132	-0.17315
84.0993	172.6409	11	-0.194253	-0.953513	-0.230389
59.1012	172.5839	11	0.250724	-0.939677	-0.23269
39.8735	157.4177	11	0.857717	-0.483992	-0.173417
32.4815	133.4534	11	0.971071	-0.1452	-0.189574
30.6826	108.3198	11	0.976722	-0.022232	-0.213353
30.391	83.1069	11	0.975561	-0.00884	-0.21955
30.0275	57.8948	11	0.976241	-0.020668	-0.215701
29.301	32.6908	11	0.977681	-0.03566	-0.20704
22.7492	32.1473	25	-0.973948	0.054528	0.220116
23.7577	56.8357	25	-0.971622	0.025338	0.235179
24.0994	81.5432	25	-0.971337	0.004159	0.23767
24.238	106.2527	25	-0.974358	0.015992	0.224434
25.6587	130.9113	25	-0.973199	0.115068	0.199107
31.2847	154.8975	25	-0.909867	0.370309	0.187121
46.4787	173.8216	25	-0.500151	0.833219	0.235787
70.1303	180.0408	25	-0.035653	0.966218	0.255248
94.5398	176.9478	25	0.275178	0.926854	0.255379
116.1693	165.3857	25	0.634973	0.74099	0.218504
131.1137	145.9091	25	0.888678	0.451592	0.079478
139.5433	122.7313	25	0.966899	0.255155	0.001599
145.5589	98.7646	25	0.970143	0.242536	0
151.7859	74.8553	25	0.9541	0.299487	0.000568
162.0048	52.4357	25	0.843133	0.537657	-0.007141
178.8296	34.5619	25	0.569148	0.818388	-0.079449
201.4746	25.1126	25	0.198344	0.966996	-0.159931
226.0493	24.2708	25	-0.142637	0.971313	-0.190278
248.4607	33.8329	25	-0.6479	0.729665	-0.218666
260.9623	54.7987	25	-0.935488	0.295277	-0.194096
265.4126	79.0565	25	-0.981989	0.087607	-0.167399
266.3236	103.7416	25	-0.984347	0.000073	-0.176242
266.1314	128.4509	25	-0.979033	-0.005289	-0.203633
266.3254	153.1599	25	-0.975471	0.022567	-0.218966
267.2947	177.8495	25	-0.974999	0.053575	-0.215655
277.1006	177.8744	25	0.972942	-0.061409	0.222738
275.9072	153.2047	25	0.971191	-0.032497	0.236074
275.437	128.5105	25	0.972525	-0.005444	0.232733
275.5035	103.8112	25	0.977457	0.005581	0.211059

Table 4 (continued)

X	Y	Z	I	J	K
275.1729	79.1171	25	0.98247	-0.051858	0.179064
271.3982	54.7653	25	0.9452	-0.275735	0.174834
260.2569	32.9131	25	0.753196	-0.622473	0.21266
240.4266	18.6451	25	0.354908	-0.905731	0.231713
216.2523	14.2904	25	0.006602	-0.978398	0.206626
191.9149	17.8853	25	-0.296849	-0.94303	0.15025
170.2732	29.4785	25	-0.627246	-0.776628	0.058412
153.9045	47.8422	25	-0.836208	-0.548397	0.00424
142.989	69.936	25	-0.940586	-0.339553	-0.001428
136.3236	93.7057	25	-0.970143	-0.242536	0
130.333	117.668	25	-0.970143	-0.242536	0
122.7841	141.1402	25	-0.904168	-0.425437	-0.038516
107.2132	159.9179	25	-0.560067	-0.809057	-0.178189
84.4947	169.1778	25	-0.194589	-0.953455	-0.230343
59.9875	169.347	25	0.230407	-0.944675	-0.233456
41.1632	154.6556	25	0.873107	-0.45631	-0.171657
34.8848	130.9078	25	0.975477	-0.109454	-0.190959
33.7117	106.2467	25	0.976722	-0.011286	-0.214212
33.527	81.5477	25	0.975558	-0.009609	-0.219531
33.0943	56.8525	25	0.976197	-0.02531	-0.215404
32.2432	32.168	25	0.977492	-0.041817	-0.206787

References

- Linares JM, Chaves-Jacob J, Schwenke H, Longstaff A, Fletcher S, Flore J et al (2014) Impact of measurement procedure when error mapping and compensating a small CNC machine using a multilateration laser interferometer. *Precis Eng* 38(3):578–588. doi:10.1016/j.precisioneng.2014.02.008
- Pan FY, Li M, Yin J (2012) Error identification for 3-axis machine tool based on laser interferometer. *Adv Mater Res* 490-495:309–314
- Zhang Z, Hu H (2013) A general strategy for geometric error identification of multi-axis machine tools based on point measurement. *Int J Adv Manuf Technol* 69(5):1483–1497. doi:10.1007/s00170-013-5094-7
- Xiang S, Yang J (2014) Using a double ball bar to measure 10 position-dependent geometric errors for rotary axes on five-axis machine tools. *Int J Adv Manuf Technol* 75(1):559–572. doi:10.1007/s00170-014-6155-2
- Li J, Xie F, Liu XJ, Dong Z, Song Z, Li W (2016) A geometric error identification method for the swiveling axes of five-axis machine tools by static R-test. *International Journal of Advanced Manufacturing Technology*. doi:10.1007/s00170-016-9199-7
- Fan KC, Wang H, Shiou FJ, Ke CW (2004) Design analysis and applications of a 3D laser ball bar for accuracy calibration of multiaxis machines. *J Manuf Syst* 23(3):194–203. doi:10.1016/S0278-6125(05)00009-9
- Jiang Z, Tang X, Zhou X, Zheng S (2015) Machining tests for identification of location errors on five-axis machine tools with a tilting head. *Int J Adv Manuf Technol* 79(1):245–254. doi:10.1007/s00170-015-6838-3
- NAS 979: Uniform cutting test, NAS series, metal cutting equipment specifications. 1969:34–7.
- Su Z, Wang L (2015) Latest development of a new standard for the testing of five-axis machine tools using an s-shaped test piece. *Proceedings of the Institution of Mechanical Engineers Part B Journal of Engineering Manufacture* 229(7). doi:10.1177/0954405414560780
- Li DU, Zheng CZ, Bian ZY, Zhao XD, Wang W (2015) Research on reconstruction and optimization of the “s” shaped test piece. *Modular Machine Tool & Automatic Manufacturing Technique*
- Mou WP, Song ZY, Guo ZP, Tang LM (2012) A machining test to reflect dynamic machining accuracy of five-axis machine tools. *Adv Mater Res* 622-623:414–419
- Mou WP 2012. Test conditions for machining centers, part 7: accuracy of a finished test piece-M5. In: the 74th meeting of ISO/TC39/SC2, Hangzhou, China, 24–28
- Sprott K, Ravani B (2008) Cylindrical milling of ruled surfaces. *Int J Adv Manuf Technol* 38(7):649–656. doi:10.1007/s00170-007-1133-6
- Roth D, Bedi S, Ismail F, Mann S (2001) Surface swept by a toroidal cutter during 5-axis machining. *Comput Aided Des* 33(1):57–63. doi:10.1016/S0010-4485(00)00063-4
- Lartigue C, Duc E, Affouard A (2003) Tool path deformation in 5-axis flank milling using envelope surface. *Comput Aided Des* 35(4):375–382. doi:10.1016/S0010-4485(02)00058-1
- Li C, Bedi S, Mann S (2006) Flank milling of a ruled surface with conical tools—an optimization approach. *Int J Adv Manuf Technol* 29(11):1115–1124. doi:10.1007/s00170-005-0002-4
- Chu CH, Hsieh HT, Lee CH, Yan C (2015) Spline-constrained tool-path planning in five-axis flank machining of ruled surfaces. *Int J Adv Manuf Technol* 80(9):2097–2104. doi:10.1007/s00170-015-7201-4
- Marciniak K (1991) *Geometric modelling for numerically controlled machining*. Oxford University Press
- Rubio DW, Lagarrigue P, Dessein G, Pastor F (1998) Calculation of tool paths for a torus mill on free-form surfaces on five-axis

- machines with detection and elimination of interference. *Int J Adv Manuf Technol* 14(1):13–20. doi:10.1007/BF01179412
20. Liu XW (1995) Five-axis NC cylindrical milling of sculptured surfaces. *Comput Aided Des* 27(12):887–894. doi:10.1016/0010-4485(95)00005-4
 21. Redonnet JM, Rubio W, Dessein G (1998) Side milling of ruled surfaces: optimum positioning of the milling cutter and calculation of interference. *Int J Adv Manuf Technol* 14(7):459–465. doi:10.1007/BF01351391
 22. Monies F, Felices JN, Rubio W, Redonnet JM, Lagarrigue P (2002) Five-axis NC milling of ruled surfaces: optimal geometry of a conical tool. *Int J Prod Res* 40(12):2901–2922. doi:10.1080/00207540210133516
 23. Bedi S, Mann S, Menzel C (2003) Flank milling with flat end milling cutters. *Comput Aided Des* 35(3):293–300. doi:10.1016/S0010-4485(01)00213-5
 24. Quan L, Wang Y, Fu H, Han Z (2008) Cutting path planning for ruled surface impellers. *Chin J Aeronaut* 21(5):462–471
 25. Menzel C, Bedi S, Mann S (2004) Triple tangent flank milling of ruled surfaces. *Comput Aided Des* 36(3):289–296. doi:10.1016/S0010-4485(03)00118-0
 26. Tsay DM, Ming JH (2001) Accurate 5axis machining of twisted ruled surfaces. *J Manuf Sci Eng* 123(4):731–738. doi:10.1115/1.1402628
 27. Tsay DM, Yan WF, Ho HC (2001) Generation of five-axis cutter paths for turbomachinery components. *J Eng Gas Turbines Power* 123(1):50–56. doi:10.1115/1.1340639
 28. Chiou JCJ (2004) Accurate tool position for five-axis ruled surface machining by swept envelope approach. *Comput Aided Des* 36(10):967–974. doi:10.1016/j.cad.2003.10.001
 29. Gong H, Cao LX, Liu J (2005) Improved positioning of cylindrical cutter for flank milling ruled surfaces. *Comput Aided Des* 37(12):1205–1213. doi:10.1016/j.cad.2004.11.006
 30. Gong H, Fang FZ, Hu XT, Cao LX, Liu J (2010) Optimization of tool positions locally based on the BCELTP for 5-axis machining of free-form surfaces. *Comput Aided Des* 42(6):558–570. doi:10.1016/j.cad.2010.02.006
 31. Senatore J, Monies F, Redonnet JM, Rubio W (2007) Improved positioning for side milling of ruled surfaces: analysis of the rotation axis's influence on machining error. *Int J Mach Tools Manuf* 47(6):934–945. doi:10.1016/j.ijmachtools.2006.07.008
 32. Charles Anderson (1990) *Curves and Surfaces in Computer Aided Geometric Design*. Springer-Verlag. doi:10.1007/978-3-642-48952-5
 33. Li C, Bedi S, Mann S (2012) Nurbs approximation to the flank-milled surface swept by a cylindrical NC tool. *Int J Adv Manuf Technol* 61(1):35–51. doi:10.1007/s00170-011-3712-9
 34. Lu Y, Ding Y, Peng Z, Chen ZC, Zhu L (2016) A spline-based method for stability analysis of milling processes. *The International Journal of Advanced Manufacturing Technology*. doi:10.1007/s00170-016-9757-z
 35. Esch L, Kieffer R, Lopez T, Berbé C, Damel P, Debay M., Hannosset JF (2013) *Numerical Methods for Solving Nonlinear Equations*. doi:10.1002/9781118673515.app8