ORIGINAL ARTICLE

An improved Z-MAP method based on the SQP algorithm for fast surface topography simulation of ball-end milling

Yukun Xiao[1] · Guangyan Ge[1] · Zhilin Zeng[1] · Xiaobing Feng[1] · Zhengchun Du[1,2]

Abstract

Products with complex free-form surfaces, such as aerospace impellers and automotive engines, are typically manufactured with ball-end milling. The machined surface topography is critical to these key components' performance in harsh environments. Accurate prediction of the machined surface topography is essential to obtaining the optimized process parameters to achieve the best possible surface quality. An improved Z-MAP method based on the sequential quadratic programming (SQP) algorithm is proposed for efficient ball-end milling surface topography simulation in this paper. Compared to the traditional Z-MAP method, this improved method does not discretize the cutter edge and does not require a strict time-step limit. At each simulation time interval, the workpiece surface grid points falling into the cutting-edge instantaneous sweep polygon are searched to reduce the computation scale. An optimization problem, which combines these grid points' coordinates with the cutting-edge trajectory model, is established for the accurate cutting time and cutting-edge element. The SQP algorithm is used to solve this optimization problem due to its global search capability. The surface topography is then updated by comparing the height of the cutting-edge element with the corresponding grid point. Experimental results indicate that the simulated surface topography is consistent with the measured surface topography. The proposed method achieves the same level of accuracy as the traditional method while requiring less computing time. The proposed method for fast surface topography in ball-end milling shows potential in optimizing process parameters to guarantee the surface quality of critical components.

Keywords Ball-end milling · Surface topography · Improved Z-MAP method · SQP algorithm

1 Introduction

Ball-end milling is widely used in five-axis milling for the manufacture of complex free-form surface workpieces, such as blades in aerospace, and engines in automobiles [1]. The surface topography, directly decided by the ball-end milling, is crucial to the surface wear, fatigue resistance, reflection performance, and precision of assembly [2, 3]. To maximize workpiece performance, the investigation of surface topography generation has attracted researchers' attention and manufacturers' alike. Besides, the surface topography simulation

before processing not only reduces trial and error costs but also improves the machining efficiency and accuracy [4].

Generally, there are two mainstream methods for surface topography simulation. The Boolean operations method is the primary one to acquire the surface topography using the solid models of the tool and workpiece. Imani et al. [5, 6] developed B-rep solid modeling approaches to solve the geometric difficulties associated with ball-end milling simulation. The surface topography is simulated by executing successive Boolean operations between the updated part and the swept volume after the cutter swept volume has been scanned as an accurate B-rep model. To improve the calculation efficiency of the resection volume, Chung et al. [7] projected the workpiece surface on the X–Y plane, and this surface can be equivalently represented by a set of parallel lines. These parallel lines and tool envelopes are used to compute the scattered point location information on the machine surface. Liu et al. [8] gave an insight into the surface topography formed by the rotating cutter flutes, thus

✉ Zhengchun Du
zcdu@sjtu.edu.cn

1 State Key Laboratory of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

2 Ningbo SKYNC Five Axis Numerical Control Technology Co., Ltd., Ningbo 315000, China

extending the CAM system simulation with more residual height information. Artetxe et al. [9] achieved the surface prediction of integral blade rotors using the CAM solid modelers and the cutting force model considering tool runout and workpiece flexibility. Most CAM software, such as UG [10] and Pro/E [11], support customers to realize topography simulation under different workpieces, cutting tools, and tool paths. Vericut can achieve the advanced requirements of impeller machining, which considers the variable real machining conditions [12]. However, the Boolean operations method has the problem of how the software deal with the solid model storage. Besides, the Boolean operations method assumes a spherical shape for the ball-end cutter teeth, making it unable to generate the real surface topography. The impact of this simulation error is not negligible in high-speed and micro-milling conditions [13, 14], which restricts the development of the Boolean operations method.

The Z-MAP method is another method for machined surface topography prediction. Typically, the Z-MAP method meshes the workpiece in the X–Y plane and uses a Z-buffer to store the Z-height of these grid points. The tool's cutting edge is discretized. The tool's rotational motion is represented by its parametric equation. By comparing the height of the cutting-edge element with the corresponding workpiece grid point at each time step, the Z-buffer is updated, thus obtaining the final surface topography.

The Z-MAP method is first applied in ball-end milling of sculpture surfaces [15]. Both the workpiece and the cutter mesh into square elements so that the cutter/workpiece engagement can be obtained automatically with a given cutter path. After that, more comprehensive simulation approaches for surface topography are developed, which incorporate the effects of cutting parameters [16], tool runout [17], tool deflection [18], etc. The Z-MAP method is introduced into 5-axis milling due to its flexibility. The surface topography in 5-axis milling is found to be strongly affected by the tool inclination, the tool lead angle, and the tool tilt angle [19, 20]. To better understand the surface topography formation process in high-speed milling, the surface topography simulation based on the Z-MAP method, which considers the flank wear and the varying feed rates' effect, is developed [21], making the machining parameters can be effectively controlled to ensure the workpiece surface quality.

The Z-MAP method provides good insight into the surface topography simulation of ball-end milling. However, the simulation accuracy relies on the time-step precision, and the mesh discretization precision of both the workpiece and the cutting edge. The simulation efficiency decreases as the precision requirements increase. In other words, high-precision simulation in the Z-MAP method is much more time-consuming. Therefore, researchers are devoted to improving the traditional Z-MAP method. The accurate cutting time and cutting-edge element at each workpiece grid point are critical to surface topography simulation. In the following improved Z-MAP method, the primary nonlinear equations, which are determined by the cutting-edge trajectory and the coordinates of the workpiece grid points, are solved using the numerical iterative methods.

Gao et al. [22] proposed a numerical simulation method based on the Newton-Raphson algorithm in ball-end milling surface topography. This numerical simulation approach is mesh-independent and efficient, but the initial values must be selected properly. Based on Gao's work, Zhang et al. [23] applied the improved Z-MAP method to the multi-axis ball-end milling. The effects of tool tilt direction and tilt angle on the surface topography are investigated. Arizmendi et al. [24] used Chebyshev polynomial expansion to solve the equivalent polynomial equations, which combined the cutting-edge trajectory and the tool-swept envelope. The tool parallel axis offset is also considered. To reduce the computing scale, the surface grid points being milled at each time step are extracted by tool-workpiece contact area identification, such as the sweep surface method [25]. Shujuan [26] and Dong [14] combined Newton's iterative algorithm and the Z-MAP method to quickly calculate the height of the workpiece grid points falling into the tool/workpiece engagement. The kinematics model of the ball-end milling cutter is established based on the homogeneous coordinate transformation.

However, these improved Z-MAP methods using the classical numerical solutions (e.g., Newton's iterative algorithm) are prone to approximation errors. If the initial values and convergence accuracy of the classical numerical solutions are not reasonably set, the nonlinear equation may fall into a local optimal solution [27], resulting in the discontinuous height of the adjacent grid points on the workpiece. But the initial values are difficult to determine in the actual milling process. Besides, the solved parameters of the nonlinear equation, such as cutting time and the cutter's position angle, may lose meaning without physical constraints.

Aiming at the problem that the convergence of the classical numerical method depends on the initial values, the nonlinear equations for the cutting-edge trajectory coupled with the workpiece grid points' coordinates are transformed into an optimization problem. Compared with the classical numerical method, the sequential quadratic programming (SQP) algorithm is more advanced in global optimization searching [28], and the physical constraints during the cutting process can be easily introduced into the analytical model.

Based on the SQP algorithm, an improved Z-MAP method is proposed for the free-form machined surface topography simulation in multi-axis milling using a ball-end cutter. Compared with the traditional Z-MAP method, this method does not require discretizing the cutter edge, but only discretizing

the time. And the time interval does not need to be strictly controlled, thus significantly improving the computational efficiency. During the computation process, the nonlinear equation problem of the tool-workpiece contact point is transformed into an optimization problem by introducing a time step constraint and a tool angle constraint. This optimization issue is resolved using the SQP algorithm. In this way, the approximation errors caused by the initial value of the nonlinear equation can be avoided, thus ensuring the calculation accuracy.

The remaining parts of this paper are structured as follows. In Section 2, the cutting-edge trajectory model in multi-axis ball-end milling is established using homogeneous coordinate transformation. In Section 3, the tool-workpiece contact area identification is introduced, followed by the improved Z-MAP method using the SQP algorithm for surface topography calculation. The experimental verification is conducted in Section 4, and the major contributions are summarized in Section 5.

2 Cutting-edge trajectory model

The cutting-edge trajectory model is required for the surface topography in multi-axis ball-end milling. In this section, four rectangular reference coordinate systems are defined to describe the parametric equations of the ball-end milling cutter. The cutting-edge trajectory model in the workpiece coordinate system is then obtained through homogeneous coordinate transformation.

2.1 Reference coordinate systems

Four reference coordinate systems are defined in a ball-end milling operation (see Fig. 1).

(1) The workpiece coordinate system $O_W - X_W Y_W Z_W$, denoted $\{W\}$, is the reference of the NC program during the machining process. In this paper, the dynamic
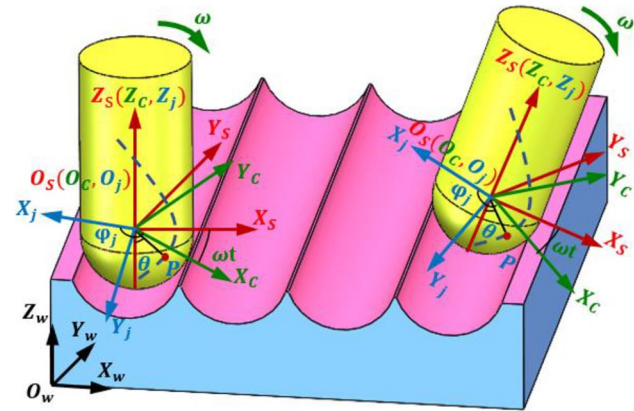

Fig. 1 Reference coordinate systems

displacement of the workpiece is neglected, so the workpiece coordinate system is also considered to be the global coordinate system.

(2) The spindle coordinate system $O_S - X_S Y_S Z_S$, denoted $\{S\}$, is fixed on the machine tool spindle. This spindle coordinate translates and rotates with the spindle's position movement and posture adjustment, respectively. The origin O_S is located at the center of the ball-end milling cutter's spherical part.

(3) The tool coordinate system $O_C - X_C Y_C Z_C$, denoted $\{C\}$, is fixed at the cutter to define the tool rotation. The coordinate axis $\overrightarrow{O_C Z_C}$ is aligned with $\overrightarrow{O_S Z_S}$. The directions of $\overrightarrow{O_C X_C}$ and $\overrightarrow{O_C Y_C}$ change as the machine tool spindle rotates.

(4) The tool tooth coordinate system $O_j - X_j Y_j Z_j$, denoted $\{j\}$, is mainly used to describe the three-dimensional curve of the cutting edge. The origin O_j is located at the center of the ball-end milling cutter's spherical part, and the plane $X_j O_j Y_j$ is the spherical equatorial plane. The coordinate axis $\overrightarrow{O_j Z_j}$ is consistent with the coordinate axis $\overrightarrow{O_C Z_C}$.

2.2 Cutting-edge definition

Cutting-edge definition aims to describe the cutter's geometry using parametric equations. The cutting edge of a ball-end milling cutter is complex in shape and is made up of two parts: the side edge and the spherical edge. In general, the spherical edge is utilized for actual machining. Depending on the geometry of the tool, the spherical edge is divided into a helical edge and a flat edge (see Fig. 2). For the ball-end cutter with a helical edge, the tool sphere vertex P_0 is at the bottom. The arbitrary point P, which is located at the $j -$ th cutter tooth, is described as follows.

$$\begin{cases} x_p^j = R \sin \theta \cos \psi \\ y_p^j = R \sin \theta \sin \psi \\ z_p^j = -R \cos \theta \end{cases} \tag{1}$$

where R is the cutter's radius. θ is the axial immersion angle (the angle between $\overrightarrow{O_c P_0}$ and $\overrightarrow{O_c P}$). ψ is the lag angle, which is defined as:

$$\psi = \tan \gamma_0 (1 - \cos \theta) \tag{2}$$

where γ_0 is the helix angle. When γ_0 is equal to zero, Eq. 1 can be used to describe the arbitrary point P on the flat-edged ball-end cutter.

In the cutting process, the coordinate system $\{C\}$ rotates at all times, changing with the machine tool spindle rotation. The homogeneous coordinate transformation matrix $\mathbf{M_{Cj}}$ from $\{j\}$ to $\{C\}$ is defined as:

Fig. 2 Ball-end milling cutters with **a** helical edge and **b** flat edge

(a) (b)
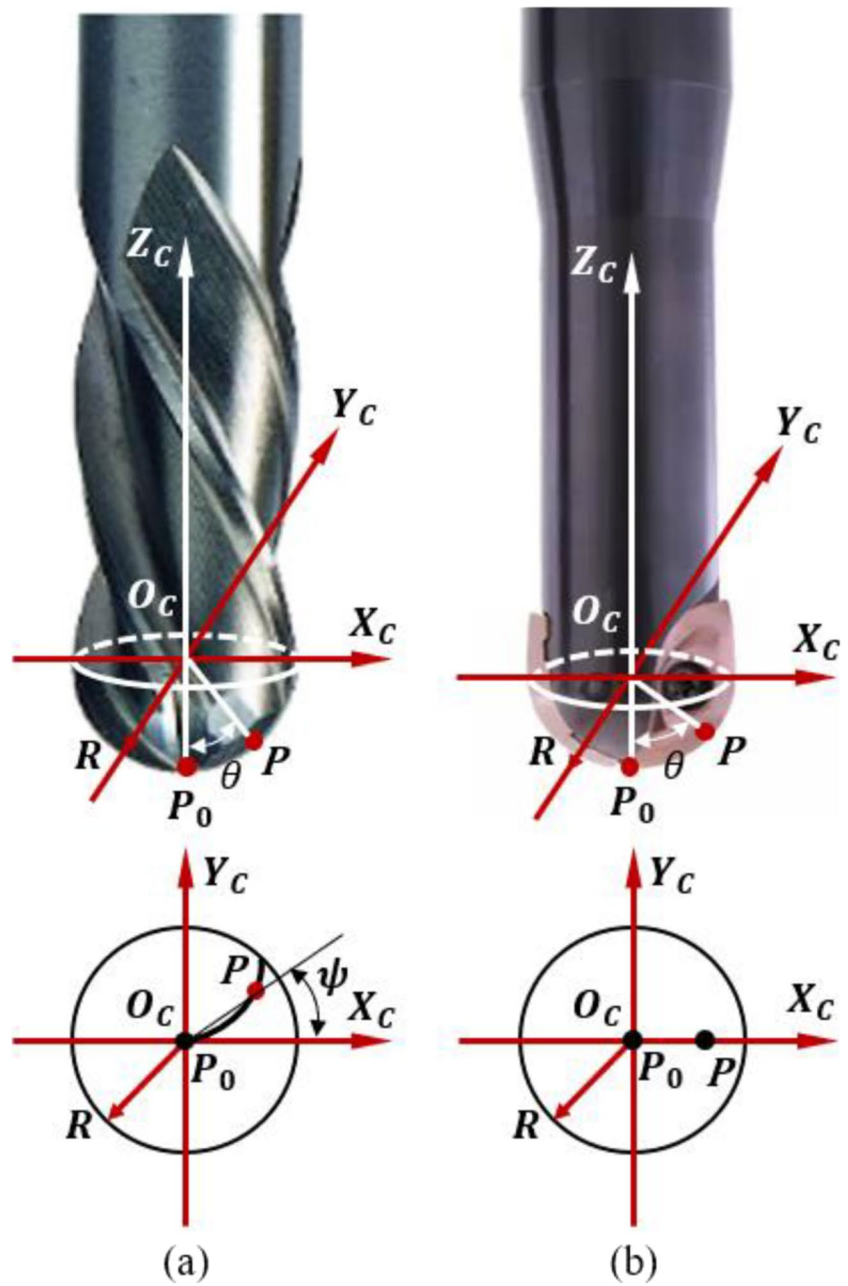
$$\mathbf{M_{Cj}} = \begin{bmatrix} \cos \varphi_j^C & -\sin \varphi_j^C & 0 & 0 \\ \sin \varphi_j^C & \cos \varphi_j^C & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (3)$$

$$\varphi_j^C = \varphi_j + \omega t = (j-1)2\pi/N + \omega t_1 \qquad (4)$$

where φ_j^C is the angle at the present time t_1 in the coordinate system $\{C\}$, which includes the intersection angle φ_j and the rotation angle ωt_1. φ_j is the angle between the first and the

$j-$th cutter tooth. N is the number of cutter teeth. ω is the spindle speed (r/min). t_1 is the rotation time of the spindle (s).

2.3 Tool path definition

The tool path definition includes two aspects, the cutter's posture adjustment, as well as position movement (see Fig. 3) Both aspects need to be properly considered to prevent position interference between the cutter and the workpiece in the actual machining process.

Fig. 3 Cutter's **a** posture adjustment and **b** position movement

(a)   (b)
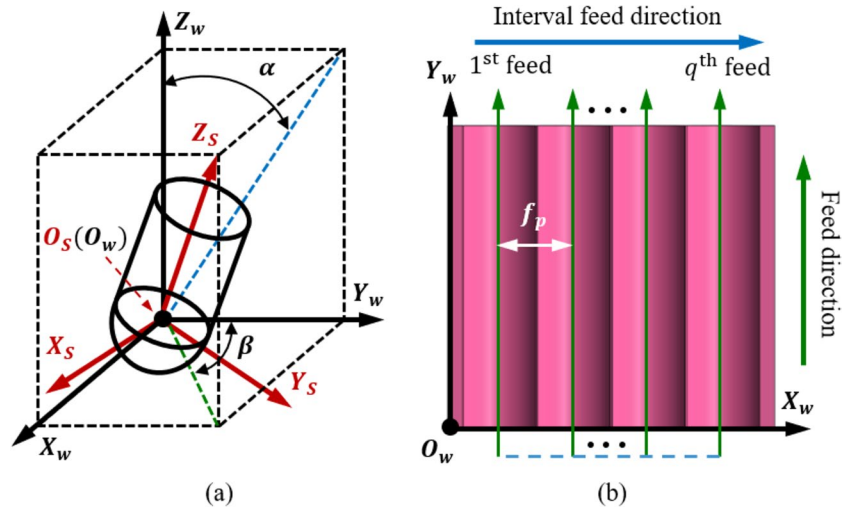
Posture adjustment refers to adjusting the cutter so that its axis forms a specific angle with the normal line of the machined surface (see Fig. 3(a)). The side tilt angle, denoted α, is the angle between the projection line of the vector $\overrightarrow{O_S Z_S}$ on the plane $Y_w O_w Z_w$ and the vector $\overrightarrow{O_W Z_W}$. The forward tilt angle, denoted β, is the angle between the projection line of the vector $\overrightarrow{O_S Z_S}$ on the plane $X_w O_w Y_w$ and the vector $\overrightarrow{O_W Y_W}$ of the coordinate system $\{W\}$. To rotate the cutter to the target position depicted in Fig. 3(a), the steps are detailed as follows.

(1) Rotate $\{S\}$ around the vector $\overrightarrow{O_W Y_W}$ by a tilt angle $\beta\prime$, where $\beta\prime = \arctan(\tan\beta\cos\alpha)$.
(2) Rotate $\{S\}$ around the vector $\overrightarrow{O_W X_W}$ by a side tilt angle α.
(3) Define the counterclockwise rotation around the positive direction of the respective reference coordinate system as positive, otherwise as negative.

The homogeneous coordinate transformation matrixes of the cutter's rotation can be expressed as:

$$\mathbf{R_\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{R_{\beta\prime}} = \begin{bmatrix} \cos\beta\prime & 0 & \sin\beta\prime & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta\prime & 0 & \cos\beta\prime & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Therefore, the posture adjustment transformation matrix $\mathbf{M_{SC}}$ of $\{C\}$ with respect to $\{S\}$ is

$$\mathbf{M_{SC}} = \mathbf{R_\alpha R_{\beta\prime}} = \begin{bmatrix} \cos\beta\prime & 0 & \sin\beta\prime & 0 \\ \sin\alpha\sin\beta\prime & \cos\alpha & -\sin\alpha\cos\beta\prime & 0 \\ -\cos\alpha\sin\beta\prime & \sin\alpha & \cos\alpha\cos\beta\prime & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Position movement reflects the translation of the cutter during milling (see Fig. 3(b)), The direction of the vector

$\overrightarrow{O_W X_W}$ and the vector $\overrightarrow{O_W Y_W}$ are defined as the interval feed direction and the feed direction, respectively. The unidirectional linear feed cutting is applied in the simulation. f_p is the feed spacing (mm), and q represents the feed order. The position movement transformation matrix $\mathbf{M_{WS}}$ of $\{S\}$ with respect to $\{W\}$ is expressed as:

$$\mathbf{M_{WS}} = \begin{bmatrix} 1 & 0 & 0 & x_0 + (q-1)f_p \\ 0 & 1 & 0 & y_0 + v_f t_2 \\ 0 & 0 & 1 & z_0 + w_h + R - a_p \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

where $[x_0 y_0 z_0]$ is the initial cutter location of O_S in the coordinate system $\{W\}$. v_f is the feed rate (mm/s), which equals to $\omega f_z N$. f_z is the feed per tooth (mm/(r·z)). t_2 is the feeding duration in the current feed order (s). The workpiece is initialed as a flat surface with a height of w_h (mm). a_p is the cutting depth (mm).

The cutting-edge trajectory model is then obtained based on the above homogeneous coordinate transformation matrices. The trajectory model is expressed as

$$[\, x_P^W(t,\theta) \;\; y_P^W(t,\theta) \;\; z_P^W(t,\theta) \;\; 1 \,]^T = \mathbf{M_{WS}M_{SC}M_{Cj}} \left[x_P^j \; y_P^j \; z_P^j \; 1 \right]^T \quad (8)$$

3 Surface topography simulation

The improved Z-MAP method based on the SQP algorithm is described in detail in this section. In the traditional Z-MAP method, both the cutting edge and the workpiece need to be discretized simultaneously for simulation precision. At each time step, each cutting-edge element's positions are calculated based on the trajectory model, making it time-consuming. The improved Z-MAP method avoids cutting-edge discretization and does not require strict time

step limits. The X–Y coordinates of the workpiece grid points are used to establish nonlinear equations with the cutting-edge trajectory model. These nonlinear equations are then transformed into optimization problems by introducing constraints on the time step and axial immersion angle. These optimization issues are resolved using the SQP algorithm. The Z coordinates of the grid point and the cutting edge are then compared to update the workpiece surface. The integrated surface topography simulation algorithm is summarized at last.

3.1 Workpiece surface discretization

The principle of the Z-MAP method is to discretize the workpiece surface into X–Y grid points with height information. As shown in Fig. 4, the workpiece's dimension is $L_x \times L_y \times w_h$, and the flat surface is divided into $m \times n$ grid points. The matrix $\mathbf{Z}[\mathbf{ii}, \mathbf{jj}]$, $(ii = 1, 2, \ldots, m+1, jj = 1, 2, \ldots, n+1)$ stores the Z coordinates of these grid points and is initialized with the workpiece's height w_h. The surface topography simulation continuously updates the matrix $\mathbf{Z}[\mathbf{ii}, \mathbf{jj}]$ during the milling process based on the cutting-edge trajectory model (Eq. 8).

The workpiece discretization directly affects the simulation accuracy and computational efficiency. If m and n are set too small, the micro surface topography is difficult to be reflected in simulation. On the other hand, when m and n are set too large, the calculation time cost increases significantly with less improvement of simulation accuracy. To balance both accuracy and efficiency, the workpiece discretization satisfies the following condition.

$$\max\left(L_x/m, L_y/n\right) \leq \frac{1}{10}\min\left(f_z, f_p\right) \tag{9}$$

The time step is another important simulation parameter to be determined. The traditional Z-MAP method requires a small time step to ensure that each workpiece grid point is milled. With the improved Z-MAP method, restrictions on the time step are loosened. A time step Δt in the range
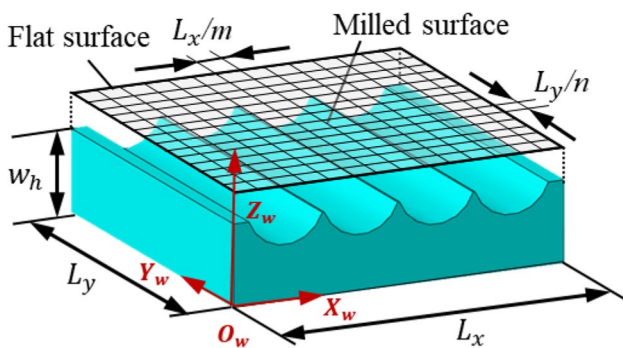
of $(5° \sim 10°)/\omega$ can generally produce the desired result since the accurate cutting-edge element and cutting time are inversed with the grid point coordinates and the cutting-edge trajectory model.

3.2 Tool-workpiece contact area identification

Identification of the tool-workpiece contact area is the most important part of topography simulation. In this section, the instantaneous sweep polygon of the ball-end cutter in the milling process is established. The overlapping area between swept polygon and the workpiece surface is exactly the contact area at that time interval. So, the identification of the contact area becomes the problem of determining whether the workpiece grid point is inside the sweep polygon. A simple solution to this geometric problem is represented at last.

The cutting depth a_p is generally smaller than the cutter's radius R during milling (see Fig. 5). The axial immersion angle θ_{im} is calculated as

$$\theta_{\text{im}} = 2\arccos\left(\left(R - a_p\right)/R\right) \tag{10}$$

The cutter inclination angle A_t is determined by both the side tilt angle α and the forward tilt angle β. A_t is denoted as

$$A_{\text{t}} = \arccos\left(\cos\alpha\cos\beta\right) \tag{11}$$

Not all cutting edges are involved in milling, and the minimum and maximum axial immersion angles $\theta_{\min}, \theta_{\max}$ are solved by

$$\begin{cases} \theta_{\min} = A_{\text{t}} - 0.5\theta_{\text{im}}, \theta_{\min} \in \left[0, \pi/2\right] \\ \theta_{\max} = A_{\text{t}} + 0.5\theta_{\text{im}}, \theta_{\max} \in \left[0, \pi/2\right] \end{cases} \tag{12}$$

Suppose that the current cutting time is t_c and the next cutting time is $t_c + \Delta t$, the four endpoints $P_k(k = 1, 2, 3, 4)$
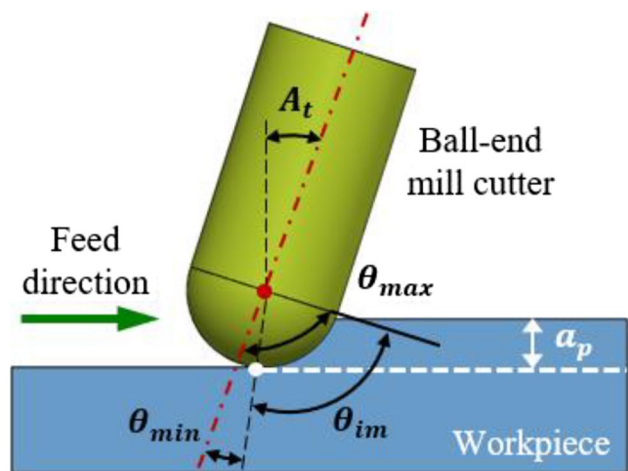

Fig. 4 Discretization of the workpiece

Fig. 5 Ball-end cutter milling workpiece

forming the cutting-edge sweep polygon can be calculated using Eq. 13 with θ_{min} and θ_{max}.

$$P_k = \left[x_p^W(t_k, \theta) y_p^W(t_k, \theta_k) z_p^W(t_k, \theta_k) \right]^T$$
$$\begin{cases} k = 1, t_k = t_c, \theta_k = \theta_{min} \\ k = 2, t_k = t_c, \theta_k = \theta_{max} \\ k = 3, t_k = t_c + \Delta t, \theta_k = \theta_{min} \\ k = 4, t_k = t_c + \Delta t, \theta_k = \theta_{max} \end{cases} \qquad (13)$$

The next step is to create the cutting-edge instantaneous sweep polygon and search for the workpiece grid points falling into it. In this way, the number of grid points, whose Z coordinates need to be updated, is significantly reduced. The four endpoints $P_k (k = 1, 2, 3, 4)$ need to be arranged in the order in which they make up a closed polygon ABCD. Otherwise, the grid point originally in the sweep polygon may be incorrectly identified as outside the sweep polygon, as shown in Fig. 6(a)–(b). When the cutter cuts in or out of the workpiece, the sweep polygon is not completely inside the workpiece surface, as shown in Fig. 6(c). In this case, the tool-workpiece contact area becomes the overlap area between the sweep polygon and the workpiece surface.

Tool-workpiece contact area identification is now a point containment problem for polygons, which can be described as, given a polygon *ABCD* and an arbitrary point Q, determining whether Q is inside the polygon [29]. The vector cross-multiplication method is computationally efficient in solving the point containment problem and robust in dealing with complicated constraints during the actual milling process [30]. The vector cross-multiplication values of the point Q to two sequential adjacent points $\left\{ \overrightarrow{QA} \times \overrightarrow{QB}, \overrightarrow{QB} \times \overrightarrow{QC}, \overrightarrow{QC} \times \overrightarrow{QD}, \overrightarrow{QD} \times \overrightarrow{QA} \right\}$ are calculated. If these cross-multiplication values are all positive or all negative and not zero, the point is in the polygon. If these cross-multiplication values are all positive or all negative and have zero, the point is on the polygon. Otherwise, the point is outside the polygon. Therefore, the vector cross-multiplication method is applied in this paper to extract the grid points inside the sweep polygon, thus realizing the tool-workpiece contact area identification.

3.3 The SQP algorithm

After extracting the grid points $\left(x_{ii}, y_{jj} \right)$ inside the sweep polygon, the first and second rows of Eq. 8 are used to solve the cutting time t^* and the axial immersion angle θ^*. The nonlinear constraint equation system is defined as follows.

$$\begin{cases} f_x(t, \theta) = x_P^W(t, \theta) - x_{ii} = 0 \\ f_y(t, \theta) = y_P^W(t, \theta) - y_{jj} = 0 \end{cases} \qquad (14)$$

The results of the above nonlinear equations (t^*, θ^*) are substituted into Eq. 8, and the height of the cutter edge element at the current cutting time is obtained.

$$z_{P^*}^W = z_P^W(t^*, \theta^*) \qquad (15)$$

$Z[ii, jj]$ is then replaced with a smaller value by contrast with $z_{P^*}^W$ and $Z[ii, jj]$, thus updating the workpiece surface topography.

The nonlinear equation system in Eq. 14 is transformed into a functional optimization problem, which is rewritten as

Fig. 6 Identification of tool-workpiece contact area. **a** Incorrect and **b** correct identification of grid point within sweep polygon. **c** Identification of grid points in the contact area
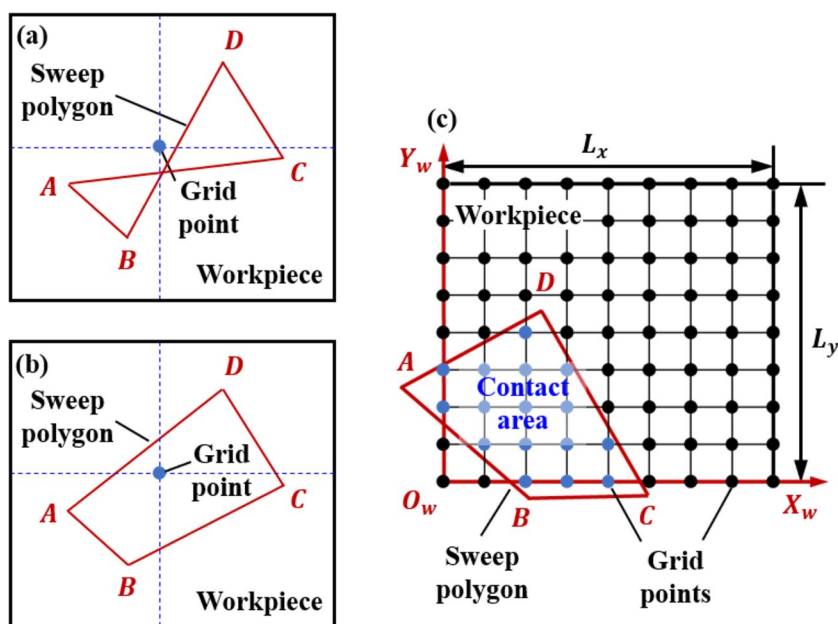
$$\min\ F(t,\theta) = f_x^2(t,\theta) + f_y^2(t,\theta)$$
$$\text{s.t.}\qquad t \in \left[t_c, t_c + \Delta t\right] \qquad (16)$$
$$\theta \in \left[\theta_{\min}, \theta_{\max}\right]$$

If solutions to the nonlinear equation system in Eq. 14 exist, the optimal value of the evaluation function $F(t,\theta)$ in Eq. 16 is expected to be zero. Unlike the nonlinear equation system, the functional optimization problem gives these parameters physical meaning by introducing cutting time and axial immersion angle constraints. In addition, this transformation avoids determining the initial values of t and θ, making it easier to obtain a global solution.

For constrained optimization issues like Eq. 16, the SQP algorithm is better suited for global searching. To accommodate the solution form of the SQP algorithm, the quadratic programming model is defined as:

$$\min\quad F(t,\theta) = f_x^2(t,\theta) + f_y^2(t,\theta)$$
$$\text{s.t.}\quad g_u(t,\theta) = 0\ \ u = 1, 2, \ldots, n_g \qquad (17)$$
$$h_v(t,\theta) \geq 0\ \ v = 1, 2, \ldots, n_h$$

where $g_u(t,\theta)$ are the equality constraints, and n_g is the number of $g_u(t,\theta)$. There are no equality constraints in the optimal model but $g_u(t,\theta)$ is still retained in the following description for generality. The inequality constraints $h_v(t,\theta)$ are applied to limit the range of parameters, where $t \in \left[t_c, t_c + \Delta t\right]$ and $\theta \in \left[\theta_{\min}, \theta_{\max}\right]$. n_h is the number of $h_v(t,\theta)$.

In general, the SQP algorithm possesses three main steps for solving constrained optimization problems.

(1) SQP subproblem construction

$$\min \nabla F(\mathbf{x})^T \boldsymbol{\delta} + \tfrac{1}{2}\boldsymbol{\delta}^T \mathbf{Q}\boldsymbol{\delta}$$
$$\text{s.t.}\ \ g_u(\mathbf{x}) + \nabla g_u(\mathbf{x})\boldsymbol{\delta} = 0 \qquad (18)$$
$$h_v(\mathbf{x}) + \nabla h_v(\mathbf{x})\boldsymbol{\delta} \geq 0$$

where $\mathbf{x} = [t, \theta]$. $\boldsymbol{\delta}$ is the search direction in the range of the vector $\mathbf{x}$. $\mathbf{Q}$ is the iterative Hessian matrix.

(B) Iterative Hessian matrix calculation

Using the Lagrange multiplier method, Eq. 18 is rewritten:

$$L(\mathbf{x}, \kappa, \hbar) = F(\mathbf{x}) - \sum_{u=1}^{p} \kappa^u g_u(\mathbf{x}) - \sum_{v=1}^{q} \hbar^v h_v(\mathbf{x}) \qquad (19)$$

where $L(\mathbf{x}, \kappa, \hbar)$ is the Lagrange function with equal variables. κ^u is the Lagrange factor of the equality constraints. Correspondingly, $\hbar^v$ is the Lagrange factor of the inequality constraints.

In the solution of Eq. 19, the search direction and corresponding point are updated:

$$\mathbf{x}_{\tau+1} = \mathbf{x}_\tau + \boldsymbol{\delta}_\tau \qquad (20)$$

where τ is the number of iterations.

Suppose that the search direction and the derivative of the Lagrange function are specified as:

$$\begin{cases} \mathbf{s}_\tau = \mathbf{x}_{\tau+1} - \mathbf{x}_\tau \\ \wp_\tau = \nabla_x L\left(\mathbf{x}_{\tau+1}, \kappa_{\tau+1}, \hbar_{\tau+1}\right) - \nabla_x L\left(\mathbf{x}_\tau, \kappa_\tau, \hbar_\tau\right) \end{cases} \qquad (21)$$

the iterative Hessian matrix $\mathbf{Q}$ can be approximately calculated as:

$$\mathbf{Q}_{\tau+1} = \mathbf{Q}_\tau + \frac{\overline{Q}_\tau \overline{Q}_\tau^T}{s_\tau^T \overline{\zeta}_\tau} - \frac{\mathbf{Q}_\tau s_\tau s_\tau^T \mathbf{Q}_\tau^T}{s_\tau^T \mathbf{Q}_\tau s_\tau} \qquad (22)$$

(C) Evaluation function calculation

$V_L(\mathbf{x}, \gamma, \psi)$ is formulated as follows to guarantee the descending searching direction, and is called the evaluation function.

$$V_L(\mathbf{x}, \Upsilon, \psi) = z(\mathbf{x}) + \sum_{u=1}^{p} \Upsilon^{(u)} \left| g^{(u)}(\mathbf{x}) \right| + \sum_{v=1}^{q} \psi^{(v)} \left| \min\left\{0, h^{(v)}(\mathbf{x})\right\} \right| \qquad (23)$$

where Υ represent the penalty factor. ψ is given as

$$\psi_\tau^{(v)} = \begin{cases} \hbar_0^v \tau = 0 \\ max\left\{ \left|\hbar_{\tau+1}^v\right|, \tfrac{1}{2}\left[\psi_{\tau-1}^{(v)} + \left|\hbar_{\tau+1}^v\right|\right] \right\} \tau \geq 1 \end{cases} \qquad (24)$$

where τ is the number of iterations, and $\hbar_0^v$ is the initial value of $\hbar^v$.

3.4 Surface topography simulation algorithm

Figure 7 gives the detailed surface topography simulation algorithm in the form of a flow chart.

(1) First of all, the machining parameters and cutter parameters are initialed, which include the cutter's radius R, the helix angle γ_0, the number of cutter teeth N, the side tilt angle α, the forward tilt angle β, the spindle speed ω, the feed per tooth f_z, the feed spacing f_p, the feed order q, and the cutting depth a_p. The simulation time interval Δt is set in $(5° \sim 10°)/\omega$.

(2) To begin with, the workpiece surface, with a dimension of $L_x \times L_y$, is discretized into $m \times n$ grid points with height information. The height matrix $Z\left[ii,jj\right]$ is initialed as w_h.

(3) The simulation algorithm carries out the interval feed cycle, feed time cycle, and cutter teeth cycle. These three cycles are dependent on the feed order q, the sin-

Fig. 7 Flow chart of the surface topography simulation algorithm
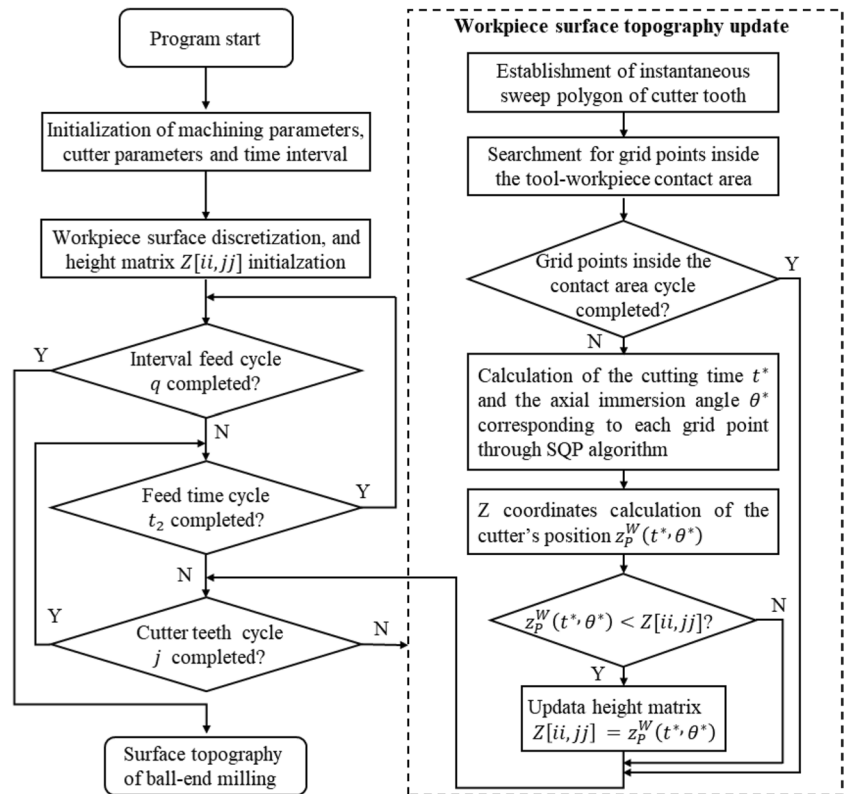
gle feed time $max(t_2)$, and the number of cutter teeth $j = 1, \ldots, N$, respectively.

(4) At each time step, the cutting-edge instantaneous sweep polygon on the $j-$ th tooth is created and then the grid points falling into the tool-workpiece contact area are searched.

(5) For each grid point inside the contact area, the cutting time t^* and the axial immersion angle θ^* are solved by the SQP algorithm. So that the cutting-edge's position $z_P^W(t^*, \theta^*)$ is obtained.

(6) By contrast $z_P^W(t^*, \theta^*)$ with the workpiece height `, the workpiece height matrix is updated with a smaller value.

(7) With all cycles completed, the simulated surface topography in ball-end milling is generated.

The most novel part of this improved Z-MAP simulation algorithm is the workpiece surface topography update. The traditional Z-MAP method calculates each cutting-edge element's position using the trajectory model and then updates its z-coordinate to the nearest workpiece grid point, which means all cutting-edge elements need to participate in the calculation at each time interval whether they are cutting the workpiece or not. In the proposed simulation algorithm, the instantaneous sweep polygon of the cutter tooth is established to search for grid points inside the tool-workpiece contact area, making that the number of grid points to be updated is significantly reduced. The cutting time and the axial immersion angle of each workpiece grid point are calculated by the SQP algorithm, which not only avoids cutting-edge discretization but also gives an accurate solution to the updated z coordinates.

4 Time complexity analysis and case study

According to the algorithm flow described in the previous section, this section provides a time complexity analysis to demonstrate the efficiency of the proposed algorithm. Additionally, a case study is included for comparing the accuracy and stability of the proposed method with other Z-MAP methods.

4.1 Time complexity analysis

The differences between the three surface topography algorithms are introduced firstly, including the proposed algorithm, the improved Z-MAP method based on the Newton iteration approach [26], and the traditional Z-MAP method. The proposed algorithm has been given in Fig. 7. The improved Z-MAP method using the Newton iteration approach only differs from the proposed algorithm in the calculation of the cutting time and the axial immersion angle. The traditional Z-MAP method has a cutting-edge element

cycle nested in the cutter teeth cycle, and the workpiece surface topography update process is more intuitive. At each time step, all the cutting-edge elements' positions are calculated through the cutting-edge trajectory model, and the workpiece surface is updated by comparing the height of the cutting-edge element and the height of the adjacent workpiece grid point.

Table 1 presents the time complexity of three topography simulation methods The time complexity of the proposed algorithm is denoted as $O(C_1 \cdot K_1 \cdot I_1)$, where C_1 represents the time complexity of the interval feed cycle, feed time cycle, and cutter teeth cycle. K_1 is the number of grid points to be updated at the time step. I_1 is the time complexity of the SQP algorithm. I_1 is considered to be $O(n_1^3)$, where $n_1 = 2$ is the number of variables [31]. Similarly, the time complexity of the Newton-based Z-MAP method is denoted as $O(C_1 \cdot K_1 \cdot I_2)$, where I_2 is the time complexity of the classical Newton iteration algorithm. I_2 is also considered to be $O(n_2^3)$, where $n_2 = 2$ is the dimension of equations [32]. As for the traditional Z-MAP method, the time complexity is denoted as $O(C_2 \cdot K_2 \cdot I_3)$, where C_2 has the same meaning as C_1, but $C_2 = 10^3 C_1$. Since that, the traditional Z-MAP method requires a strict time step, which is set as 10^{-6} s, while the improved Z-MAP method has the time step set as 10^{-3} s. K_2 is the time complexity of the cutting-edge element cycle. In the traditional Z-MAP method, the axial immersion angle $\theta \in [0, pi/2]$ is discretized into 10^{-5} rad elements, thus $K_2 \approx 10^5$. I_3 is the time complexity of the matrix operation in the cutting-edge trajectory model. I_3 is considered to be $O(n_3^3)$, where $n_3 = 4$ is the dimension of the homogeneous coordinate transformation matrix.

From the time complexity analysis, it can be seen that the improved Z-MAP algorithm is less than that of the traditional Z-MAP algorithm. The proposed method and the Newton-based Z-MAP method have similar time complexity, indicating that the proposed method is able to improve simulation efficiency.

4.2 Case study

To compare the efficiency and the accuracy, a simulation case is presented. The simulation parameters are almost the same as those described in Section 5.1. The forward tilt angle β, the feed per tooth f_z, and the feed spacing f_p are set as 0°, 0.8 mm/(r·z), and 0.4 mm, respectively. In the Newton-based method [26], the initial values are set as $[t_0, \theta_0] = [t_c - \Delta t, 0.5(\theta_{min} + \theta_{max})]$, where $\theta_{min} = 0$ rad, and $\theta_{max} = 0.451$ rad are calculated by Eq. 12. The termination condition of the iteration is $[t_k - t_{k-1}, \theta_k - \theta_{k-1}] = [10^{-5}, 10^{-5}]$.

Figure 8 gives the topography simulations of the three methods. The time cost of the proposed algorithm, the Newton-based algorithm, and the traditional Z-MAP method are 230 s, 200 s, and 500 s, respectively. Although the proposed method takes longer time than the Newton-based algorithm, it can achieve the same simulation accuracy as the traditional method without the unexpected unsmooth areas. These unsmooth areas show that the approximate error generated from the Newton iteration method damages the simulation surface's continuity. The line $y = 0.11$ passing through the unsmooth area is selected for comparison. In most cases, the prediction results of the three methods are consistent, but there are errors in the Newton method in special areas. At the workpiece grid point $(x, y) = (0.84, 0.11)$, the predicted height of the proposed algorithm, the Newton-based algorithm, and the traditional Z-MAP method are -7.4114 μm, -4.7962 μm, and -7.9643 μm, respectively. The traditional Z-MAP method is used as a benchmark for simulation accuracy due to its high accuracy of cutter teeth and time discretization. It can be seen that the Newton-based algorithm has

Table 1 Time complexity of three topography simulation methods

Method	Calculation procedure	Execution times	Time complexity
The proposed	Interval feed cycle Feed time cycle Cutter teeth cycle	C_1	$O(C_1 \cdot K_1 \cdot I_1)$
	Grid points to be updated	K_1	
	SQP algorithm	$I_1 = O(2^3)$	
Newton-based Z-MAP	Interval feed cycle Feed time cycle Cutter teeth cycle	C_1	$O(C_1 \cdot K_1 \cdot I_2)$
	Grid points to be updated	K_1	
	Newton iteration algorithm	$I_2 = O(2^3)$	
Traditional Z-MAP	Interval feed cycle Feed time cycle Cutter teeth cycle	$C_2 = 10^3 C_1$	$O(C_2 \cdot K_2 \cdot I_3)$
	Cutting-edge element cycle	$K_2 \approx 10^5$	
	Matrix operation	$I_3 = O(4^3)$	

Fig. 8 Topography simulation of the three methods. **a** Newton-based Z-MAP method. **b** Traditional Z-MAP method. **c** The proposed method. **d** Selected line comparison
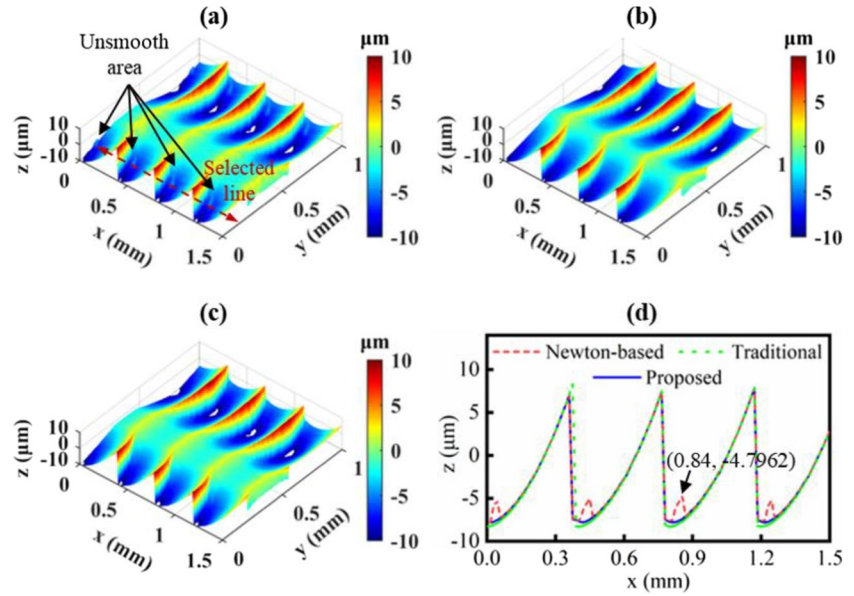

Table 2 Results of SQP and Newton numerical solver under different initial values

Method	Initial Value	(t^*, θ^*)	$Z_P^W(t^*, \theta^*)$
SQP	$t_c - \Delta t$	(0.0070, 0.0128)	–7.4114
	t_c	(0.0070, 0.0128)	–7.4114
	$t_c + \Delta t$	(0.0070, 0.0128)	–7.4114
Newton numerical solver	$t_c - \Delta t$	(0.0043, -0.0348)	–4.7962
	t_c	(0.0070, 0.0134)	–7.3724
	$t_c + \Delta t$	(0.0070, 0.0134)	–7.3724

almost 3 μm approximation error, which is not allowed in high-precision simulation. In contrast, the proposed method is more stable in simulation results than the Newton-based algorithm.

The false selection of the initial value may be the reason why the classical Newton numerical solver generates unexpected approximation errors. The workpiece grid point $(x, y) = (0.84, 0.11)$ is taken as an example. This point is updated when the feed order $q = 4$, the single feed time $t_c = 0.006$s, $\Delta t = 10^{-3}$s, and the cutter tooth $j = 2$. Table 2 gives the results of SQP and Newton numerical solver under different initial values. The initial value $\theta_0 = 0.5(\theta_{min} + \theta_{max}) = 0.226$rad. The SQP algorithm has constraints $t^* \in [t_c, t_c + \Delta t] = [0.006, 0.007], \theta^* \in [\theta_{min}, \theta_{max}] = [0, 0.451]$ and can converge to the same solution stably at different initial values. While the Newton numerical solver converges to (0.0043, -0.0348) when the initial value is $t_0 = t_c - \Delta t$, which is beyond the range of the cutting time and axial immersion angle. Therefore, the SQP method with physical constraints introduced in the simulation can reduce the

influence of the initial value on the convergence results, making the simulation results more stable.

5 Ball-end milling experiments

Several ball-end milling experiments with various cutting parameters are carried out to validate both the accuracy and the efficiency of the proposed method in this section.

5.1 Experiment setup

The ball-end milling experiment setup is illustrated in Fig. 9. The experiments are performed utilizing a DMU50 five-axis machining center (Fig. 8(a)–(b)). The ball-end milling cutter, whose diameter is 10 mm, is composed of tungsten steel. Besides, the cutter has two teeth with a helix angle of 35°. The workpiece material is aluminum alloy 7050 with high strength and small plastic deformation. Four milling experiments have been conducted, and the workpiece size of each experiment is 3 mm × 3 mm (Fig. 8(c)). After milling, the machined surfaces are measured by the VK-X3000 surface profiler from KEYENCE (Fig. 8(d)), with a height resolution of 1 nm.

The cutting parameters are listed in Table 3. Different surface topography factors are considered, which include the forward tilt angle β, the feed per tooth f_z, and the feed spacing f_p. Among the common parameters of these experiments, the spindle speed ω is 3000 r/min. The cutting depth a_p is set as 0.5 mm.

The simulation parameters are presented as follows. The simulated surface dimension is set as 1.5 mm × 1 mm, which matches the visual field of the surface profiler system. The

Fig. 9 Experimental setup

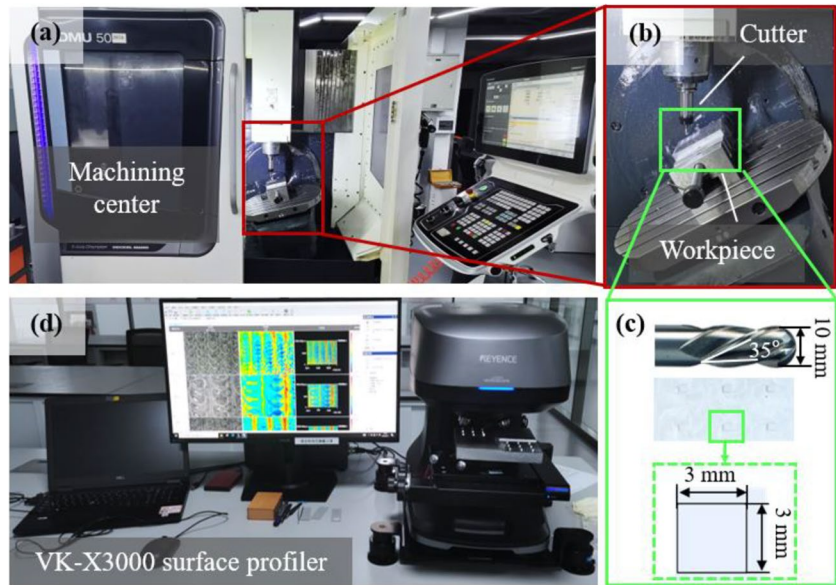**Table 3** Experiment cutting parameters

No	β°	$f_z mm/(r \cdot z)$	$f_p mm$	Common parameters
1	0	0.2	0.4	spindle speed
2	30	0.2	0.4	$\omega = 3000 r/min$
3	0	0.5	0.4	cutting depth
4	0	0.2	0.8	$a_p = 0.5mm$

surface is divided into 100×100 grid points with dimensions 0.03×0.03^2 mm. In the traditional Z-MAP method, the time interval is 1×10^{-5} s, and the cutting-edge position angle unit is 1×10^{-4} rad to ensure that all grid points can be milled. In the proposed method, the time interval is 1×10^{-3} s, and the cutting edge doesn't need to be discretized. In addition, the computer information used for surface topography simulation is as follows. The computer processor is 12th Gen Intel(R) Core(TM) i7-12700 CPU @ 2.10 GHz, memory capacity 64 GB, Windows 10 system, software environment Matlab 2021a.

5.2 Results and discussion

The surface topography simulation accuracy and efficiency of the proposed method are identified by comparing with both the experimental measurements and the traditional Z-MAP method simulations. The profiles in the interval feed direction under different cutting parameters are selected to further illustrate the differences between measurement and simulation.

The experimental measurements and the surface topography simulations are presented in Figs. 10, 11, 12, and 13. In addition to 3D height maps, 2D profiles of 0.7 mm length, extracted at the center of the area, along the interval feed direction are presented. Although numerous burrs appear on the measured surface (especially on the workpiece edges), the effect of the cutting parameters on the machined surfaces can be observed. Multiple periodic peaks are shown in the feed direction, and the 2D profile fluctuates with a range of [-5, 5] μm (see Fig. 10). Under the influence of the tilt angle, the workpiece is unevenly milled by the ball-end cutter, resulting in periodic spherical depressions on the measured surface topography (see Fig. 11). When the feed per tooth increases from 0.2 mm/(rz) to 0.5 mm/(rz), the number of peaks in the feed direction decrease, and the 2D profile fluctuates greatly with a range of [-10, 10] μm (see Fig. 12). In experiment No. 5, fewer cycles are shown in the interval feed direction than that of experiment No. 1. Only one peak appears on the surface topography at the 0.8 mm feed spacing, as the visual field in the interval feed direction is 1.5 mm (see Fig. 13).

Even though the measured profiles fluctuate, the simulated profiles are in good agreement with the measurement. In Fig. 10(d) about 0.55 mm, the measured profile includes much vibration. The possible reason may be the tool vibration and the cutting force. In Fig. 11(d) about 0.45 mm, the measured profile is much smaller than that of the simulation, which may be caused by the workpiece plastic deformation and the slenderness of the cutting tool.

There are many factors that affect the topography of the machined surface in the actual cutting process, such as material elasticity, tool wear, tool runout, the machine tool volumetric error, and measuring instrument noise. Moreover, the initial cutting angles of the adjacent tool paths are different in real machining, which also contributes to the errors between the simulation and the measurement. These

Fig. 10 Surface topography measurement and simulation of No. 1. **a** Measurement. **b** Traditional Z-MAP simulation. **c** Improved Z-MAP simulation. **d** Profiles in the interval feed direction
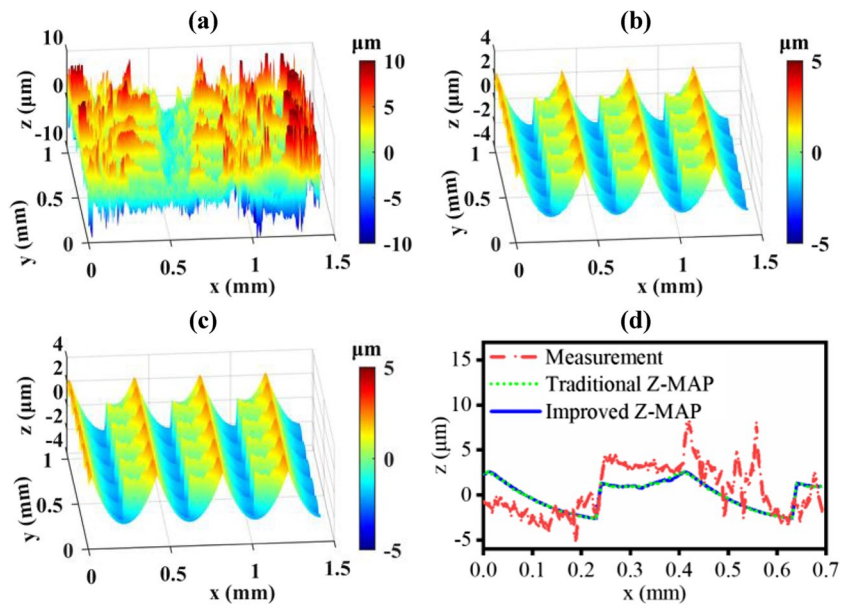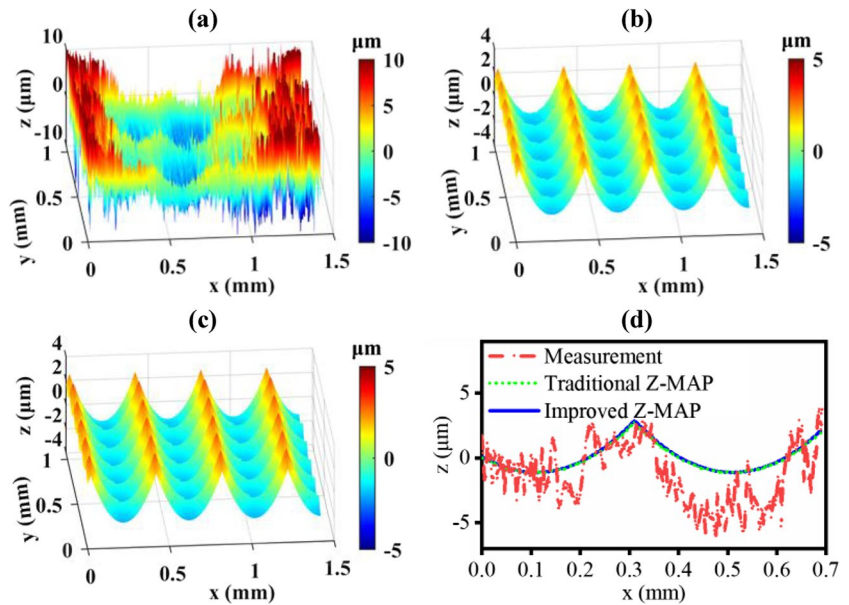

Fig. 11 Surface topography measurement and simulation of No. 2. **a** Measurement. **b** Traditional Z-MAP simulation. **c** Improved Z-MAP simulation. **d** Profiles in the interval feed direction

error sources are complex and strongly random, which are difficult to simulate, and the deviations between simulated and measured topography are acceptable.

The simulated topographies by the proposed method and the traditional Z-MAP method are almost identical in most cases. These two simulation methods are based on the same kinematics model. In the traditional Z-MAP method, the theoretical topography can be obtained as long as the units of time-step and cutting edge are small enough (without considering the tool runout, tool defection, etc.). While the proposed method obtains the topography by solving the system of trajectory equations, which avoids cutting-edge discretization. From the simulation results, there are minor

differences between these two calculation methods, showing the accuracy of the proposed method.

The Root-Mean-Square Error (RMSE) of the proposed method and the traditional Z-MAP method on the selected profile is listed in Table 4. The RMSE of the four experiments is within 0.15 µm, illustrating that the proposed method has almost the same accuracy level as the traditional Z-MAP method.

The time costs of three topography simulation methods in the four experiments are given in Fig. 14. The simulation parameters and environment are already illustrated in the Experiment Setup section. Although the proposed method takes a little longer time than the Newton-based

Fig. 12 Surface topography measurement and simulation of No. 3. **a** Measurement. **b** Traditional Z-MAP simulation. **c** Improved Z-MAP simulation. **d** Profiles in the interval feed direction
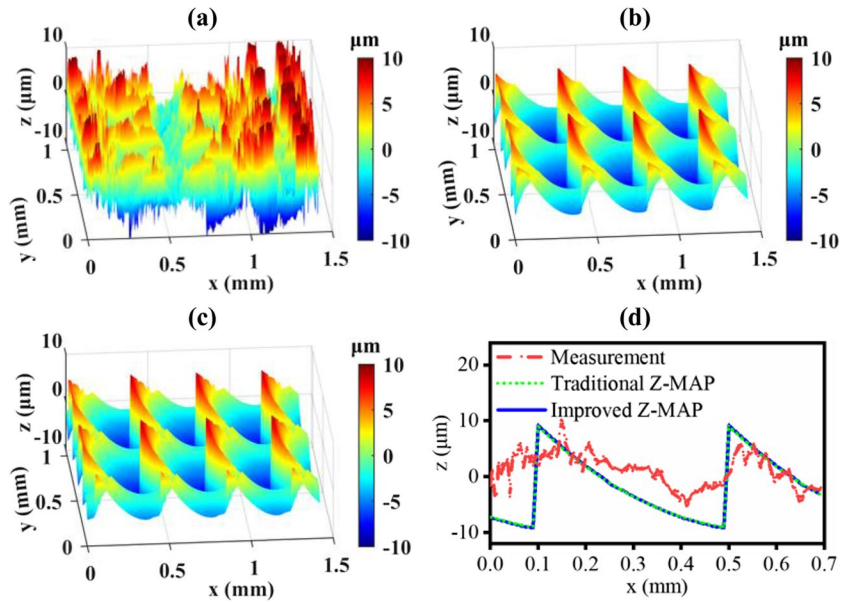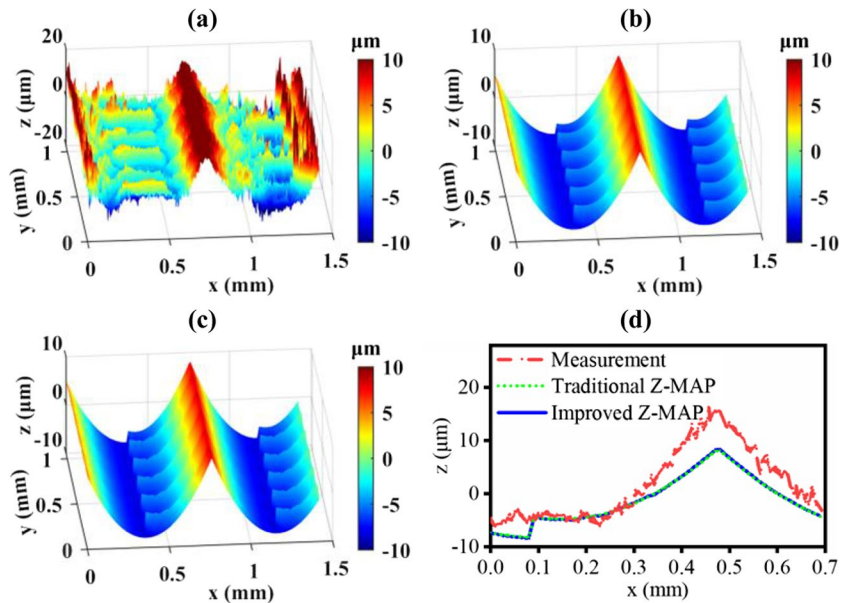

Fig. 13 Surface topography measurement and simulation of No. 4. **a** Measurement. **b** Traditional Z-MAP simulation. **c** Improved Z-MAP simulation. **d** Profiles in the interval feed direction

Table 4 RMSE of the proposed method and the traditional Z-MAP method in the selected profile

No	RMSE (μm)
1	0.101
2	0.097
3	0.150
4	0.139

Z-MAP method, considering the stability of the solution, this time loss is acceptable. In the traditional Z-MAP method, the computing time of experiments No. 1–2 and No. 4 are 58 min, 57 min, and 53 min, respectively.

The feed per tooth of the No. 3 is bigger than the other experiments, resulting in faster feed rates and less simulation time, which is 27 min. In the proposed method, the computing time of the four experiments is within 30 min, which are 15 min, 21 min, 11 min, and 20 min, respectively. The proposed method reduces the time cost by 74.1%, 63.2%, 59.3%, and 62.3% compared to the traditional Z-MAP method. The proposed method costs only half as much time as the traditional Z-MAP method, showing higher efficiency for surface topography simulation.
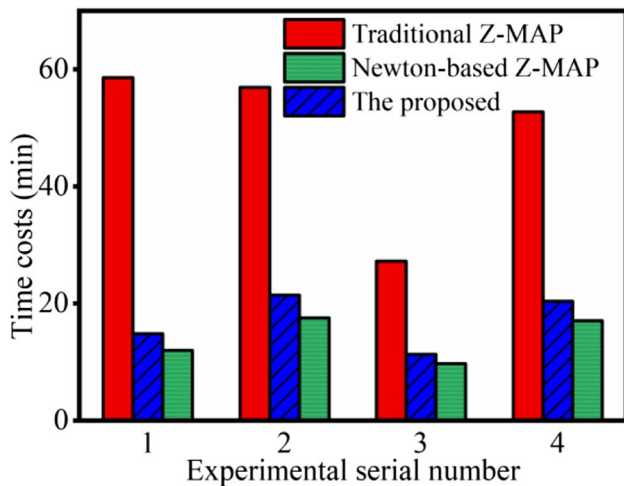
Fig. 14 Time costs of three topography simulation methods

6 Conclusions

In this paper, an improved Z-MAP method based on the SQP algorithm for fast surface topography simulation of ball-end milling is proposed. Experiments and numerical simulations are conducted to validate this proposed method. The following are the primary conclusions:

(1) The proposed method establishes a cutting-edge trajectory model based on the homogeneous coordinate transformation, which includes both the cutter's posture adjustment and position movement. The cutting-edge instantaneous sweep polygon is presented, followed by the identification of the workpiece grid points falling into it. In this way, the number of grid points, whose Z coordinates need to be updated, is significantly reduced.

(2) At each simulation time interval, the workpiece grid points' coordinates are used to establish nonlinear equations with the cutting-edge trajectory model, so that the accurate cutting time and the axial immersion angle can be obtained. The proposed method transforms these nonlinear equations into an optimization problem and the SQP algorithm is applied for the solution. Therefore, the updated Z coordinates of the grid points are no longer affected by the initial values and exhibit global optimality.

(3) The ball-end milling experiments and numerical simulations are conducted for validation. Experimental results indicate that the simulated surface topography is consistent with the measured surface topography. Compared to the traditional Z-MAP method, the proposed method shows almost the same level of accuracy and needs less computing time.

The improved Z-MAP method has a broad application prospect in the ball-end milling surface quality estimation and cutting parameters optimization. The improved Z-MAP method is validated with a rectangular plate in this paper but can be easily applied in projectable free-form surface milling, which only changes the workpiece's initial height and the axial immersion angle during milling. The shortcoming of the proposed method is that it cannot support non-projectable surfaces, such as the surface of the impeller. This method can be also applied to more types of tool shapes such as torus, flat-end, etc. by changing the cutting-edge definition and the axial immersion angle. Furthermore, this method can be strengthened by considering the spindle eccentricity, the cutter runout, and the workpiece deformation, which will be our future work.

Declarations

Ethics approval Not applicable.

Consent to participate The authors declare that all authors have read and approved submitting this manuscript to IJAMT.

Consent for publication The authors declare that all authors agree to sign the transfer of copyright for the publisher to publish this article upon acceptance.

Conflict of interest The authors declare no competing interests.

References

1. Lu Y-A, Chen S-F, Wang C-Y (2022) Collision-free and smooth path generation method of five-axis ball-end milling considering effective tool diameter change. J Manuf Process 84:435–447. https://doi.org/10.1016/j.jmapro.2022.10.011

2. Zhuo Y, Han Z, An D, Jin H (2021) Surface topography prediction in peripheral milling of thin-walled parts considering cutting vibration and material removal effect. Int J Mech Sci 211:106797. https://doi.org/10.1016/j.ijmecsci.2021.106797

3. Tong J, Zhang Z, Chen P, Zai P, Yang S, Bian P (2022) Study on surface morphology of titanium alloy curved thin-walled parts by longitudinal-torsional composite ultrasonic assisted milling. J Manuf Process 84:316–326. https://doi.org/10.1016/j.jmapro.2022.09.049

4. Wang L, Ge S, Si H, Guan L, Duan F, Liu Y (2020) Elliptical model for surface topography prediction in five-axis flank milling. Chin J Aeronaut 33:1361–1374. https://doi.org/10.1016/j.cja.2019.06.007

5. Imani BM, Sadeghi MH, Elbestawi MA (1998) An improved process simulation system for ball-end milling of sculptured surfaces. Int J Mach Tools Manuf 38:1089–1107. https://doi.org/10.1016/S0890-6955(97)00074-6

6. Imani BM, Elbestawi MA (2000) Geometric simulation of ball-end milling operations. J Manuf Sci Eng 123:177–184. https://doi.org/10.1115/1.1347034

7. Chung YC, Park JW, Shin H, Choi BK (1998) Modeling the surface swept by a generalized cutter for NC verification. Comput-Aided Des 30:587–594. https://doi.org/10.1016/S0010-4485(97)00033-X

8. Liu N, Loftus M, Whitten A (2005) Surface finish visualisation in high speed, ball nose milling applications. Int J Mach Tools Manuf 45:1152–1161. https://doi.org/10.1016/j.ijmachtools.2004.12.007

9. Artetxe E, Dv O, de Lacalle LNL, Campa FJ, Dn O, Lamikiz A (2017) Solid subtraction model for the surface topography prediction in flank milling of thin-walled integral blade rotors (IBRs). Int J Adv Manuf Technol 90:741–752. https://doi.org/10.1007/s00170-016-9435-1

10. Guo Q, Sun Y, Jiang Y, Yan Y, Zhao B, Ming P (2016) Tool path optimization for five-axis flank milling with cutter runout effect using the theory of envelope surface based on CL data for general tools. J Manuf Syst 38:87–97. https://doi.org/10.1016/j.jmsy.2015.11.003

11. Ding S, Yang DCH, Han Z (2005) Boundary-conformed machining of turbine blades. Proc Inst Mech Eng B J Eng Manuf 219:255–263. https://doi.org/10.1243/095440505X28981

12. Yan B, Hao Y, Zhu L, Liu C (2022) Towards high milling accuracy of turbine blades: A review. Mech Syst Signal Process 170:108727. https://doi.org/10.1016/j.ymssp.2021.108727

13. Klauer K, Altherr N, Eifler M, Kirsch B, Böß V, Seewig J, Aurich JC (2021) Kinematic simulation to investigate the influence of the cutting edge topography when ball end micro milling. Proc CIRP 102:109–114. https://doi.org/10.1016/j.procir.2021.09.019

14. Dong Y, Li S, Zhang Q, Li P, Jia Z, Li Y (2021) Modeling and Analysis of micro surface topography from ball-end milling in a trochoidal milling mode. Micromachines 12:1203. https://doi.org/10.3390/mi12101203

15. Lazoglu I (2003) Sculpture surface machining: a generalized model of ball-end milling force system. Int J Mach Tools Manuf 43:453–462. https://doi.org/10.1016/S0890-6955(02)00302-4

16. Buj-Corral I, Vivancos-Calvet J, Domínguez-Fernández A (2012) Surface topography in ball-end milling processes as a function of feed per tooth and radial depth of cut. Int J Mach Tools Manuf 53:151–159. https://doi.org/10.1016/j.ijmachtools.2011.10.006

17. Liu X, Soshi M, Sahasrabudhe A, Yamazaki K, Mori M (2005) A Geometrical simulation system of ball end finish milling process and its application for the prediction of surface micro features. J Manuf Sci Eng 128:74–85. https://doi.org/10.1115/1.2039098

18. Han SG, Zhao J, Zhang XF (2009) Surface topography and roughness simulations for 5-axis ball-end milling. Adv Mat Res 69–70:471–475. https://doi.org/10.4028/www.scientific.net/AMR.69-70.471

19. Quinsat Y, Lavernhe S, Lartigue C (2011) Characterization of 3D surface topography in 5-axis milling. Wear 271:590–595. https://doi.org/10.1016/j.wear.2010.05.014

20. Layegh KSE, Lazoglu I (2017) 3D surface topography analysis in 5-axis ball-end milling. CIRP Ann 66:133–136. https://doi.org/10.1016/j.cirp.2017.04.021

21. Gao H, Yue C, Liu X, Nan Y (2019) Simulation of surface topography considering cut-in impact and tool flank wear. Appl Sci 9:732. https://doi.org/10.3390/app9040732

22. Gao T, Zhang W, Qiu K, Wan M (2005) Numerical simulation of machined surface topography and roughness in milling process. J Manuf Sci Eng 128:96–103. https://doi.org/10.1115/1.2123047

23. Zhang W-H, Tan G, Wan M, Gao T, Bassir DH (2008) A new algorithm for the numerical simulation of machined surface topography in multiaxis ball-end milling. J Manuf Sci Eng 130:011003. https://doi.org/10.1115/1.2815337

24. Arizmendi M, Fernández J, de Lacalle LNL, Lamikiz A, Gil A, Sánchez JA, Campa FJ, Veiga F (2008) Model development for the prediction of surface topography generated by ball-end mills taking into account the tool parallel axis offset. Exp Valid CIRP Ann 57:101–104. https://doi.org/10.1016/j.cirp.2008.03.045

25. Wang R, Zhang S, Ge R, Luan X, Zhang Q, Wang J, Lu S (2021) Modified iterative approach for predicting machined surface topography in ball-end milling operation. Int J Adv Manuf Technol 115:1783–1794. https://doi.org/10.1007/s00170-021-07245-6

26. Shujuan L, Dong Y, Li Y, Li P, Yang Z, Landers RG (2019) Geometrical simulation and analysis of ball-end milling surface topography. Int J Adv Manuf Technol 102:1885–1900. https://doi.org/10.1007/s00170-018-03217-5

27 de Jesús Rubio J, Islas MA, Ochoa G, Cruz DR, Garcia E, Pacheco J (2022) Convergent newton method and neural network for the electric energy usage prediction. Inf Sci 585(89):112. https://doi.org/10.1016/j.ins.2021.11.038

28. Zhang Z, Cheng Q, Qi B, Tao Z (2021) A general approach for the machining quality evaluation of S-shaped specimen based on POS-SQP algorithm and Monte Carlo method. J Manuf Syst 60:553–568. https://doi.org/10.1016/j.jmsy.2021.07.020

29. Atallah MJ (1986) Review of computational geometry–an introduction. Math Comput 47:763–764. https://doi.org/10.2307/2008200

30. Hormann K, Agathos A (2001) The point in polygon problem for arbitrary polygons. Comput Geom-Theory Appl 20:131–144. https://doi.org/10.1016/S0925-7721(01)00012-8

31. Nocedal J, Wright SJ (2006) Numerical optimization, 2nd edn. Springer, New York

32. Burden RL, Faires JD, Burden AM (2016) Numerical analysis, 10th edn. Cengage Learning, Boston