https://doi.org/10.1007/s00170-023-11991-0

ORIGINAL ARTICLE

A novel method to minimize the five-axis CNC machining error around singular points based on closed-loop inverse kinematics

Zehong Lu[1,2] · Guanying Huo[2,3,4] · Xin Jiang[3,4]

Received: 2 January 2023 / Accepted: 17 July 2023 / Published online: 4 August 2023

Abstract
Singularities in five-axis machining are a series of positions where the exact motion of the rotary axis becomes unpredictable or incalculable. In the neighborhood of a singular point, machining may cause unstable movements of drive axes and deteriorative dynamic performance of machine tools. In this paper, we proposed a closed-loop inverse kinematics method to solve accurately the five-axis joints' parameters around singular points. To achieve this purpose, the damped Jacobian pseudoinverse algorithm is introduced and the machine joints' velocities are calculated directly by the velocities of the five-axis cut points. The joints are then obtained by integrating the corresponding velocities. Furthermore, a feedback control method is applied to reduce the integrating error. In the end, simulations and experiments' results demonstrated the effectiveness of the closed-loop inverse kinematics method.

Keywords Five-axis CNC machining · Singularity points · Inverse kinematics

1 Introduction

Five-axis machining tools have the ability to combine position and orientation and complete machining without the repositioning process and have been widely used in machining dies, molds, and impellers. Some workpieces with a complex surface, such as a turbine blade, can only be manufactured by five-axis machining tools. The primary function of any five-axis CNC program is to transform cutter location data into machine commands (G-code), and the essential function is the inverse kinematics function (Fig. 1). Compared with three-axis machining tools, five-axis machining tools have two rotation axis, and the inverse kinematics of five-axis machining tools is a nonlinear function, which transforms the CL data $(x, y, z, o_i, o_j, o_k)^T$ into the machine commands of five-axis as $(P_x, P_y, P_z, \theta_A, \theta_C)$ or $(P_x, P_y, P_z, \theta_A, \theta_B)$ or $(P_x, P_y, P_z, \theta_B, \theta_C)$. The notation of the rotation axes is either (θ_A, θ_C) or (θ_A, θ_B) or (θ_B, θ_C) that depends on the structure of the five-axis machine.

While the orientation of the rotation axis is parallel to the tool direction, the exact solution of the inverse kinematics equation does not exist. Then the rotary axis will be unpredictable and may rotate 180° abruptly. These deadly positions are called singularities. Machining near such singularities may cause unstable movements of the drive axis and deteriorate the dynamic performance of machine tools. Furthermore, an abrupt change of the rotary axis will bring shock to the servo motor, lower the machining precision and even damage the machined quality. Therefore, the detection and avoidance of singularity should be conducted in five-axis machining.

To avoid singularities, there are currently four categories of methods, the locally deforming method [1–6] the forbidden region and acceptable region-based method [7–9], the workpiece position reorientation method [10–13] and the kinematics-based method [14–16].

The locally deforming method is based on the deforming of the tool path to devoid of the singularity. Affouard et al. [1] presented the concept of "singular cone" and use B-spline

✉ Guanying Huo
gyhuo@buaa.edu.cn

✉ Xin Jiang
jiangxin@buaa.edu.cn

Zehong Lu
luzehong@pku.edu.cn

[1] School of Mathematics and Physics, Anqing Normal University, Anqing, Anhui, China

[2] Zhengzhou Aerotropolis Institute of Artificial Intelligence, Zhengzhou, Henan, China

[3] LMIB & NLSDE & Institute of Artificial Intelligence, Beihang University, Beijing, China

[4] Zhongguancun Laboratory, Beijing, China

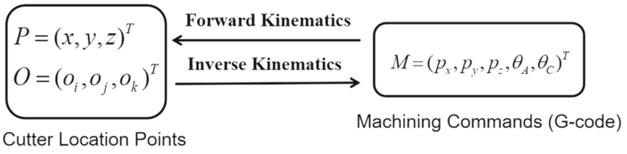


Fig. 1 Forward and inverse kinematics of five-axis machines

curves to describe tool trajectory. Singularity was detected and avoided in the studies of Yang and Altintas [2] based on the general kinematic model with screw theory and singularity was eliminated by modifying the B-spline control points. Wang et al. [3] projected the three dimension "singular cone" into two-dimension space, and the singularity phenomenon is avoided by changing the B-spline control points with an optimal method. The same method was also used for singularity avoidance of a 5-axis hybrid robot [4]. In this study, the sudden change of C-type rotation was eliminated and the tracking error could also be limited. Sun et al. [5] proposed a local gouge-free tool path modification method for singularity avoidance with consideration of compliant axes kinematics, and they also considered the singular phenomena in robotic machining [6].

The forbidden region method is mainly based on translating orientation polyline to avoid the singularity. Lin et al. [7] projected the orientation vector into the C-space, and the singularities are detected by contact checking between the orientation polyline and the tapper circle. Lin et al. [8, 9] also considered the surface texture in avoiding singularity, and the acceptable-texture orientation region concept (ATOR) was proposed.

The workpiece position reorientation method is based on resetting the workpiece orientation to make no orientation vertical and thus avoids the singularity. Cripps et al. [10] firstly analyzed the cause of singularity, and they proposed a position reorientation method to avoid the singularity. By analyzing the forces transmitted to the rotary drives as torque disturbances, Yang et al. [11] identified the workpiece location on the rotary to minimize tracking errors in five-axis machining. With the analysis of the machine's kinematic behavior, Pessoles et al. [12] designed the workpiece reorientation method to minimize the overall distance traveled by the rotary axes and avoided singularity. Recently, Gao et al. [13] developed a workpiece setup optimization method, which not only considered the singularity avoidance requirement but also considered the five-axis kinematic stability.

Kinematic based method is different from the previous method and avoids singularity by the optimization of inverse kinematics. Munlin et al. [14] considered the kinematic error near singular points, by the optimization of the required rotations on the inverse kinematics equation, the singularity was avoided. Shen et al. [15] proposed a double solutions-based optimisation method. The singularity was determined during inverse kinematics, which chooses the appropriate solution set within rotary ranges and avoided the disturbance of the rotary axis near singular points. My and Bohez [16] developed the five-axis inverse kinematic method near singular points, with the feedback control method, the rotary axis was directly evaluated.

There are also some other methods to avoid singularity. Sørby [17] proposed the kinematics of a five-axis machine with non-orthogonal rotary axes. Gray [18] proposed a $3\frac{1}{2}\frac{1}{2}$ axis machining method. In $3\frac{1}{2}\frac{1}{2}$ axis machining, only its three linear axes were moved and the two rotary axes were locked, and had the ability to pass the singular region. Grandguillaume et al. [19] proposed a tool path patching strategy method, which avoids the singular by modifying the tool axes orientation while respecting maximum velocity, acceleration, and jerk of the machine tools.

However, most of the mentioned solutions are based on the geometric approximation and regeneration of the tool path near the singular points. In this paper, a robust algorithm is designated for computing the tool path across singular points, and the singularity problems are changed to a matrix ill-conditioned problem. While the Jacobian matrix of five-axis kinematics is ill-conditioned, a small disturbance of the orientation vector may cause a huge change in the rotary axes in a five-axis machine, which means a small cutter point's velocity may cause a huge rotary velocity. To overcome the ill-conditioned problem of the Jacobian matrix, the damping least-square Jacobian method is used in this paper. Then the five-axis joints' positions (three linear joints and two rotary joints) can be calculated directly by integrating the joints' velocities numerically. When integrating the joints' velocities, a feedback control system is applied to limit the integration error.

The remainder of this paper is organized as follows. In Sect. 2, kinematic singularity and inverse kinematics of the five-axis CNC machine are presented. In Sect. 3, damping least-square Jacobian and feedback control methods are described detailed. Simulation and experimental results are given in Sect. 4. Section 5 provides some concluding remarks and future works.

2 Kinematic singularity of 5-axis CNC machine

The tool paths generated by CAM software are calculated basing the input on the part surface models, the surface quality required, the cutter definition, the tool path pattern, etc. [16] The output tool paths are defined in a workpiece coordinate system. However, the machine tool is controlled by the motor in each axes, which is based on the machine tool coordinate. The two coordinates are connected via a kinematic chain of transformation. In this paper, an A-C rotary type five-axis machine tool is used to illustrate the five-axis kinematics. The A-C type five-axis has a rotary table that rotates about the vertical Z-axes (C-rotation) and a tilting table that turns about the X-axis (A-rotation). Usually, the workpiece is mounted on the rotary table.

The tool path is defined as a piecewise curve passing CL points (cutter location points), which are generated by CAM system, and defined in standardised format (x, y, z, i, j, k), where (x, y, z) is tool tip position and (i, j, k) is the orientation vector. The transform function between axes parameters and CL points coordinates can be found

$$\begin{bmatrix} x \\ y \\ z \\ i \\ j \\ k \end{bmatrix} = \begin{bmatrix} C_C P_x + S_C C_A P_y + S_C S_A P_z + L_x \\ -S_C P_x + C_C C_A P_y + C_C S_A P_z + L_y \\ -S_A P_y + C_A P_z + L_z \\ S_C S_A \\ C_C S_A \\ C_A \end{bmatrix} \tag{1}$$

In the above equation, (L_x, L_y, L_z) defines the offsets of the pivot point of the two rotary axes A and C relative to the workpiece coordinate system origin and it is a constant vector. (P_x, P_y, P_z) represents the five-axis joint coordinates. A, C are the two rotary coordinates and S_A, S_C, C_C, C_A represent $sin(A), sin(C), cos(C), cos(A)$ respectively.

According to equation Eq. 1, the rotary axis parameters A and C can be calculated directly:

$$\begin{aligned} &A = cos^{-1}(k) \quad (0 \leq A \leq \pi) \\ &C = atan2(i, j) \quad (-\pi < C \leq \pi) \end{aligned} \tag{2}$$

Combine equation Eq. 2 and Eq. 1, the prismatic axis parameters (P_x, P_y, P_z) can be calculated as follows:

$$\begin{aligned} P_x &= C_C(x - L_x) - S_C(y - L_y) \\ P_y &= S_C C_A(x - L_x) + C_C C_A(y - L_y) - S_A(z - L_z) \\ P_z &= S_C S_A(x - L_x) + C_C S_A(y - L_y) + C_A(z - L_z) \end{aligned} \tag{3}$$

As $k = \pm\sqrt{1 - i^2 - j^2}$, the tool path can be represented by $\boldsymbol{r} = (x, y, z, i, j) \in W^5 \subset \mathbb{R}^5$ in work space. We denote $\boldsymbol{q} = [q_1, q_2, q_3, q_4, q_5]^T \in Q^5 \subset \mathbb{R}^5$ as an axis variable in joint (axes) space, where q_1, q_2, q_3, q_4, q_5 represent P_x, P_y, P_z, A, C respectively. The forward kinematic function of five-axis tool machine can be written as follows:

$$\boldsymbol{r} = f(\boldsymbol{q}) = \begin{bmatrix} C_{q5}q_1 + S_{q5}C_{q4}q_2 + S_{q5}S_{q4}q_3 + L_x \\ -S_{q5}q_1 + C_{q5}C_{q4}q_2 + C_{q5}S_{q4}q_3 + L_y \\ -S_{q4}q_2 + C_{q4}q_3 + L_z \\ S_{q5}S_{q4} \\ C_{q5}S_{q4} \end{bmatrix}, \tag{4}$$

where $f : Q^5 \rightarrow W^5$ is a nonlinear map. The kinematic differential equation can be obtained as follows:

$$\dot{\boldsymbol{r}} = J(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{5}$$

where $J(\boldsymbol{q}) = \frac{\partial f(\boldsymbol{q})}{\partial \boldsymbol{q}} \in \mathbb{R}^{5\times5}$ is the Jacobian matrix. Equation 5 represents the relationship between the joint velocities and the cutter point's velocities.

According to the equation Eq. 4 and Eq. 5, we can also calculate the joint variable $\boldsymbol{q}$ from $\boldsymbol{r}$ and joint velocity $\dot{\boldsymbol{q}}$ from cutter point velocity $\dot{\boldsymbol{r}}$.

$$\begin{aligned} &\boldsymbol{q} = f^{-1}(\boldsymbol{r}) \\ &\dot{\boldsymbol{q}} = J^{-1}(q)\dot{\boldsymbol{r}} \end{aligned} \tag{6}$$

For both equation Eq. 4 and equation Eq. 5, the condition of inverse function consistence is as follows:

$$Det(J(\boldsymbol{q})) \neq 0,$$

therefore, the singularity position can be determined by solving the equation $Det(J(\boldsymbol{q})) = 0$. The Jacobian matrix in equation Eq. 5 can be calculated directly from equation Eq. 4:

$$J(\boldsymbol{q}) = \begin{bmatrix} C_{q5} & S_{q5}C_{q4} & S_{q5}S_{q4} & q_3C_{q4}S_{q5} - q_2S_{q4}S_{q5} & q_2C_{q4}C_{q5} - q_1S_{q5} + q_3C_{q5}S_{q4} \\ -S_{q5} & C_{q4}C_{q5} & C_{q5}S_{q4} & q_3C_{q4}C_{q5} - q_2C_{q5}S_{q4} & -q_1C_{q5} - q_2C_{q4}S_{q5} - q_3S_{q4}S_{q5} \\ 0 & -S_{q4} & C_{q4} & -q_2C_{q4} - q_3S_{q4} & 0 \\ 0 & 0 & 0 & C_{q4}S_{q5} & C_{q5}S_{q4} \\ 0 & 0 & 0 & C_{q4}C_{q5} & -S_{q4}S_{q5} \end{bmatrix} \tag{7}$$

According to the above equation Eq. 7, we get the determinate of $J(\boldsymbol{q})$:

$$Det(J(\boldsymbol{q})) = -C_{q4}S_{q4}. \tag{8}$$

Usually the angle range of q_4, q_5 is: $q_4 \in [0, \pi]$, $q_5 \in (-\pi, \pi)$, and the singular points are the points which corresponding to $q_4 = 0$. In those points $\boldsymbol{r} = (r_1, r_2, r_3, 0, 0)$, which corresponds to the CL points $(r_1, r_2, r_3, 0, 0, 1)$ in CAM system.

3 Singularity avoidance method

3.1 Inverse kinematics at singularity

Consider curve $\boldsymbol{r}(t)$ that passes through all of the CL points, this curve is called the desired trajectory when cutting across the singular points. While the $Det(J(\boldsymbol{q})) = 0$, the analytic solution for $\boldsymbol{q} = f^{-1}(\boldsymbol{r})$ is not defined, and joint variable $\boldsymbol{q}$ cannot be computed according to the given CL points.

It should be noticed that if the joint velocity $\dot{\boldsymbol{q}}$ can be calculated, the joint variable $\boldsymbol{q}$ can be calculated by integrating numerically. If the $\dot{\boldsymbol{q}}$ can be determined more conveniently than $\boldsymbol{q}(t)$, then the inverse kinematics could consider $\dot{\boldsymbol{q}}(t)$ as known, instead of $\boldsymbol{q}(t)$.

To solve the equation Eq. 6, an appropriate inverse of the Jacobian matrix must be applied. The Jacobian Pseudoinverse (JP) algorithm [20] is widely used and a generalized inverse for the Jacobian matrix can be defined as follows:

$$J^+ = J^T(JJ^T)^{-1}.$$

Then the joint velocity vector can be calculated by

$$\dot{\boldsymbol{q}} = J^+(\boldsymbol{q})\dot{\boldsymbol{r}} \tag{9}$$

However, when a five-axis machine tool reaches a singularity, the pseudoinverse matrix may have too large condition values. By computing the Singular Value Composition (SVD) of Jacobian matrix J:

$$J = U\Sigma V^T = \sum_{i=1}^{n} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T, \tag{10}$$

where σ_i is the singular value of J^+, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$, $\boldsymbol{u}_i, \boldsymbol{v}_i$ is the ith column of the matrix U and V. By the equation Eq. 10, the condition number of J is $\kappa(J) = \frac{\sigma_1}{\sigma_n}$. As U and V are orthogonal matrices, then the pseudoinverse of J is

$$J^+ = V\Sigma^+U^T = \sum_{i=1}^{n} \frac{1}{\sigma_i} \boldsymbol{v}_i \boldsymbol{u}_i^T, \tag{11}$$

And then we can find out that the condition number of the pseudoinverse matrix J^+ is $\kappa(J^+) = \frac{\sigma_1}{\sigma_n}$, which tends to be infinity at singularity points.

To avoid the big condition number near singular points, the Jacobian Damping (JD) pseudoinverse was proposed in [21–23]:

$$J_D^+ = (J^TJ + \lambda^2 I)^{-1} J^T = \sum_{i=1}^{n} \frac{\sigma_i}{{\sigma_i}^2 + \lambda^2} \boldsymbol{v}_i \boldsymbol{u}_i^T, \tag{12}$$

where $\lambda \geq 0$ is the damping factor, I is the identity matrix. Then the solution corresponds to the JD is

$$\dot{\boldsymbol{q}} = J_D^+ \dot{\boldsymbol{r}} = (J^TJ + \lambda^2 I)^{-1} J^T \dot{\boldsymbol{r}}. \tag{13}$$

Notice that, if $\lambda = 0$, then $J_D^+ = J^+$ and it is ill-conditioned near the singularity. It is important to point out that Eq. equationJD satisfies the following equation:

$$\min_{\dot{\boldsymbol{q}}} \left\{ \|\dot{\boldsymbol{r}} - J\dot{\boldsymbol{q}}\|^2 + \lambda^2 \|\dot{\boldsymbol{q}}\|^2 \right\} \tag{14}$$

Since the sum in equation Eq. optimal can be written as follows:

$$\left\| \begin{pmatrix} J \\ \lambda I \end{pmatrix} \dot{\boldsymbol{q}} - \begin{pmatrix} \dot{\boldsymbol{r}} \\ 0 \end{pmatrix} \right\|^2 \tag{15}$$

The unique minimizer $\dot{\boldsymbol{q}}$ is given by the solution of normal equation:

$$(J^TJ + \lambda^2 I)\dot{\boldsymbol{q}} = J^T \dot{\boldsymbol{r}} \tag{16}$$

It can be easily shown that the solution to equation Eq. 16 can be formally written as follows:

$$\dot{\boldsymbol{q}} = (J^TJ + \lambda^2 I)^{-1} J^T \dot{\boldsymbol{r}}. \tag{17}$$

For a five-axis machine tool, the velocity of the C-axes may become very large. In Eq. 14, λ is a tradeoff between least-square condition and the least norm condition. In Eq. 14, small values of λ will cause a high accuracy solution but low robustness in the neighbor of singularity, while high values of λ will result in low accuracy. It is essential to choose a suitable value for λ.

It is noticed that while the CL point is not in the singular region, the normal inverse can be used directly. A singular region can be defined and the DJ is applied only to the CL point entering the region. To this purpose, an adapted factor damped Jacobian method is proposed in [23]:

$$\lambda^2 = \begin{cases} 0 & \sigma_n \geq \epsilon \\ \left(1 - \left(\frac{\sigma_n}{\epsilon}\right)^2\right) \lambda_{max}^2 & \sigma_n < \epsilon \end{cases} \tag{18}$$

where σ_n is the minimum singular value of the Jacobian matrix, ϵ is the width of the singular region, and λ_{max} is the maximum allowed damping factor.

3.2 Integrator method

If the initial joint value $\boldsymbol{q}(0)$ is known, the joint position could be computed by integrating velocity over time:

$$\boldsymbol{q}(t) = \boldsymbol{q}(0) + \int_0^t \dot{\boldsymbol{q}}(t)dt \tag{19}$$

The integration can be calculated in discrete time by resorting to numerical techniques. The simplest method is based on the Euler integrator method. If the joint positions and velocities at time t_k are known, the joint position at time $t_{k+1} = t_k + \Delta t$ can be computed as follows:

$$\boldsymbol{q}_{k+1} = \boldsymbol{q}_k + \dot{\boldsymbol{q}}(t_k)\Delta t. \tag{20}$$

However, the $\boldsymbol{q}(t)$ obtained by equation Eq. 20 could not satisfy equation Eq. 4 due to the accumulated error and the integrator error may exceed the preset tolerance. To overcome the drawback, a feedback correction term [24] can be introduced by replacing the end-effector velocity by

$$\dot{\boldsymbol{r}}_d + \boldsymbol{K}\boldsymbol{e}, \tag{21}$$

where $\dot{\boldsymbol{r}}_d$ denotes the desired velocity, $\boldsymbol{K}$ is a positive definite matrix, usually a diagonal 5 × 5 matrix, and $\boldsymbol{e}$ represents the error between the desired and the evaluated CL point: $\boldsymbol{e}(t) = \boldsymbol{r}_d(t) - f(\boldsymbol{q}(t))$. According to the above analysis, the inverse kinematics of the Five-axis machine tools can be written by the equation:

$$\begin{cases} \boldsymbol{q}(t) = \boldsymbol{q}(0) + \int_0^t \dot{\boldsymbol{q}}(t)\,dt \\ \dot{\boldsymbol{q}}(t) = \left(J^T J + \lambda^2 I\right)^{-1} J^T \left(\dot{\boldsymbol{r}}_d + \boldsymbol{K}\boldsymbol{e}\right) \end{cases} \tag{22}$$

The equation Eq. 22 can be converted to discrete form:

$$\begin{cases} \boldsymbol{e}(t_i) = \boldsymbol{r}_d(t_i) - f(\boldsymbol{q}(t_i)) \\ \dot{\boldsymbol{q}}(t_i) = [J^T(\boldsymbol{q}(t_i))J(\boldsymbol{q}(t_i)) + \lambda^2(\boldsymbol{q}(t_i))I]^{-1}J^T(\boldsymbol{q}(t_i))[\dot{\boldsymbol{r}}_d(t_i) + \boldsymbol{K}\boldsymbol{e}(t_i)] \\ \boldsymbol{q}(t_{i+1}) = \boldsymbol{q}(t_i) + \Delta t \cdot \dot{\boldsymbol{q}}(t_i) \\ \left(t_i = \frac{i}{N}, \Delta t = \frac{1}{N}, t_0 = 0, i = 0, 1, 2, \cdots, N\right) \end{cases} \tag{23}$$

The flowchart of the proposed method is shown in Fig. 2

In the equation Eq. 23, $\boldsymbol{e}(t) = \boldsymbol{r}_d(t) - f(\boldsymbol{q}(t))$, which includes both tool tip position error and tool orientation error. $\boldsymbol{r}_d(t)$ is the desired tool position. Slerp is shorthand for spherical linear interpolation. Let p_0 and p_1 be the first and last points of the arc, and let t be the parameter, $0 \leq t \leq 1$. Let Ω is the angle of the arc, and $cos(\Omega) = p_0 \cdot p_1$, then the geometric Slerp of the two vectors is as follows:

$$Slerp(p_0, p_1; t) = \frac{sin[(1-t)\Omega]}{sin(\Omega)}p_0 + \frac{sin[t\Omega]}{sin(\Omega)}p_1 \tag{24}$$

In fact, a Slerp path is a spherical geodesic. A Quaternion can be used to represent a vector. If we transform normal vectors p_0, p_1 to quaternion q_0, q_1, then the quaternion Slerp of the two quaternions is as follows:

$$Slerp(q_0, q_1; t) = q_0(q_0^{-1}q_1)^t = (q_1^{-1}q_0)^t q_0 \tag{25}$$

In Fig. 2, the quaternion is used to represent tool orientation. After quaternion Slerp, we use b-spline to interpolate those discrete quaternions, and evaluates the desired velocities $\dot{\boldsymbol{r}}_d(t_i)$.

On the assumption that $J_D^+ = (J^T J + \lambda^2 I)^{-1} J^T$, so J_D^+ is square and nonsingular, while equation

$$\dot{\boldsymbol{q}}(t) = J_D^+(\dot{\boldsymbol{r}}_d + \boldsymbol{K}\boldsymbol{e}) \tag{26}$$

leads to the equivalent linear system

$$\dot{\boldsymbol{e}} + \boldsymbol{K}\boldsymbol{e} = 0 \tag{27}$$

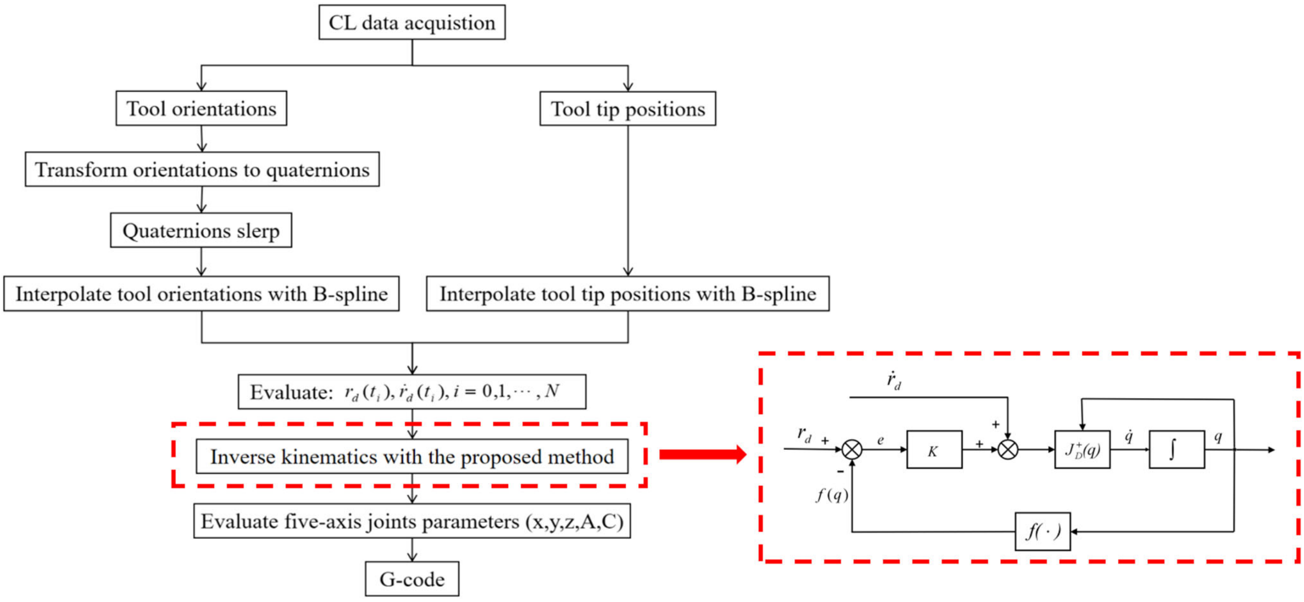
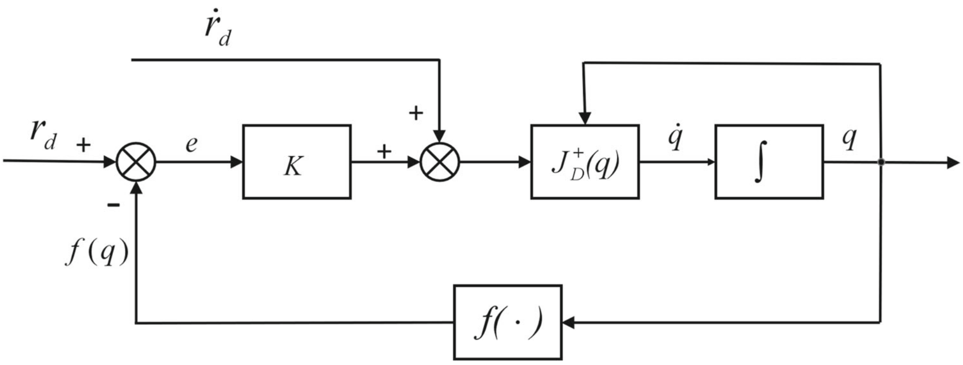

CL data acquistion
Tool orientations
Tool tip positions
Transform orientations to quaternions
Quaternions slerp
Interpolate tool orientations with B-spline
Interpolate tool tip positions with B-spline
Evaluate: $r_d(t_i), \dot{r}_d(t_i), i = 0,1,\cdots,N$
Inverse kinematics with the proposed method
Evaluate five-axis joints parameters (x,y,z,A,C)
G-code
$\dot{r}_d$
r_d
+
-
e
K
+
+
$J_D^+(q)$
$\dot{q}$
$\int$
q
$f(q)$
$f(\cdot)$

Fig. 2 Procedure of the proposed singularity avoidance method and the block scheme corresponding to the feedback control system

Fig. 3 Inverse kinematic algorithm with Jacobian Damping (JD) pseudoinverse



If $\boldsymbol{K}$ is a positive definite (usually diagonal) matrix, the system Eq. 27 is asymptotically stable. The error trends to zero with a convergence rate that depends on the eigenvalues of $\boldsymbol{K}$. The larger eigenvalues, the faster convergence. However, since the system is discrete-time system, there will be a limit for the maximum eigenvalue of $\boldsymbol{K}$ under which asymptotic stability of the error system is guaranteed. The block scheme corresponding to the inverse kinematics algorithm in Eq. 27 is illustrated in Fig. 3, where block $\boldsymbol{K}$ represents the diagonal matrix. The block J_D^+ represents the JD matrix. The block scheme shows the presence of a string of integrators on the forward loop, which depends on the discrete Euler integration method. Further, the feed forward action provided by $\dot{\boldsymbol{r}}_d$ for a time-varying reference ensures that the error is kept to zero along the whole trajectory. The discrete form of the feed back control system is the equation Eq. 23. For more details, please refer to [25].

4 Simulation and experimental results

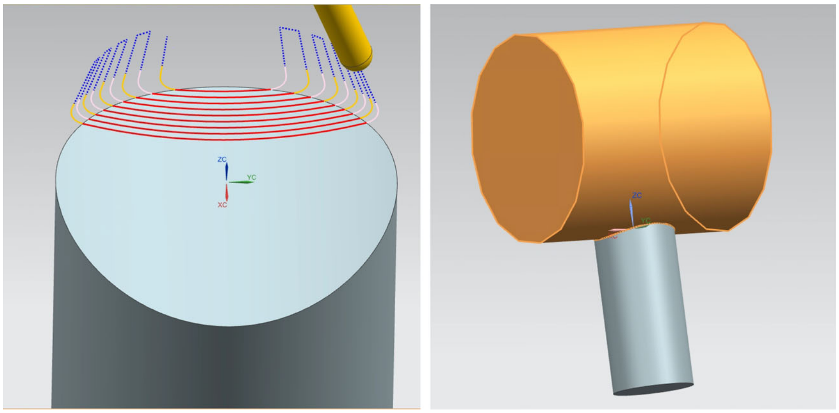
In the following examples, the workpiece models are constructed by Siemens NX 12.0 software, and the CL points and the G-codes are generated by the CAM module in the Siemens NX system. Before real cutting, VERICUT software is used to simulate CNC machining. Finally, the whole machining process is verified by an A-C type five-axis machine tool.

The proposed method in this paper is a kinematics-based method. We compared our method with My and Bohez [16] theoretically. And the inverse kinematics system in [16] is as follows:

$$\begin{cases} \boldsymbol{q}(t) = \boldsymbol{q}(0) + \int_0^t \dot{\boldsymbol{q}}(t)dt \\ \dot{\boldsymbol{q}}(t) = J^+(\dot{\boldsymbol{r}}_d + \boldsymbol{K}\boldsymbol{e}), \end{cases} \tag{28}$$

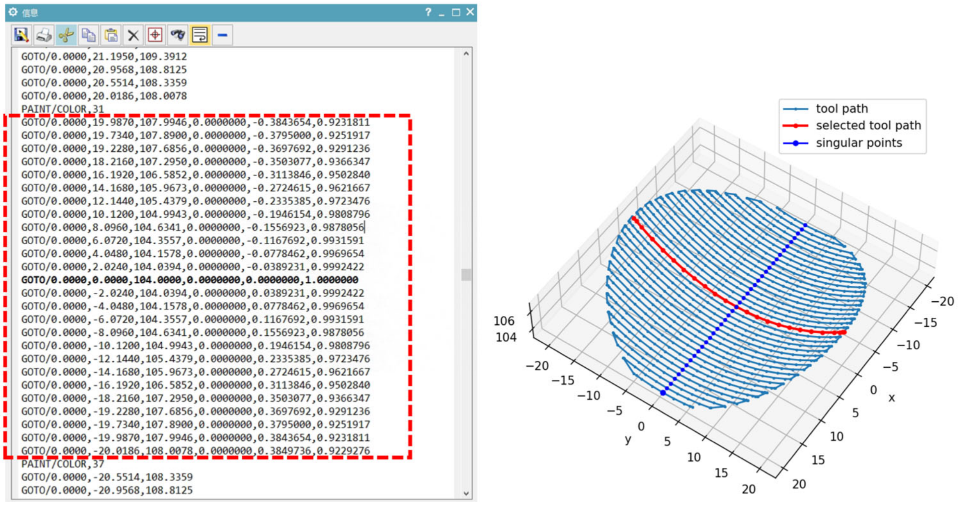
where $J^+ = (J^TJ)^{-1}J$. As it is discussed in Sect. 3.1, the condition number of the pseudoinverse matrix J^+ is $\kappa(J) = \frac{\sigma_1}{\sigma_n}$, which tends to be infinity at singular points, and it not suit for our experiments. According to the conclusion in [26], λ should have a high value to avoid the maximum $\frac{1}{2\sigma}$ of the condition number at $\sigma_n = \lambda$, but on the other hand, λ must also have a very small value to avoid conditioning tends to infinity as σ_n decreases. By those reasons, we choose the adapted factor damped Jacobian method. The maximum damped factor $\lambda_max = 0.001$, and it's an empirical numbers.

Fig. 4 Machining surface in Siemens NX



(a) Machining surface and tool path (b) The Boolean operation of two cylinders

Fig. 5 **a** CL points file generated by Siemens NX CAM systems; **b** tool paths created by the CL points file, the red tool path corresponds to the circled CL points in (a) and the blue points are the singular points, whose orientation is (0, 0, 1)



```
GOTO/0.0000,21.1950,109.3912
GOTO/0.0000,20.9568,108.8125
GOTO/0.0000,20.5514,108.3359
GOTO/0.0000,20.0186,108.0078
PAINT/COLOR,31
GOTO/0.0000,19.9870,107.9946,0.0000000,-0.3843654,0.9231811
GOTO/0.0000,19.7340,107.8900,0.0000000,-0.3795000,0.9251917
GOTO/0.0000,19.2280,107.6856,0.0000000,-0.3697692,0.9291236
GOTO/0.0000,18.2160,107.2950,0.0000000,-0.3503077,0.9366347
GOTO/0.0000,16.1920,106.5852,0.0000000,-0.3113846,0.9502840
GOTO/0.0000,14.1680,105.9673,0.0000000,-0.2724615,0.9621667
GOTO/0.0000,12.1440,105.4379,0.0000000,-0.2335385,0.9723476
GOTO/0.0000,10.1200,104.9943,0.0000000,-0.1946154,0.9808796
GOTO/0.0000,8.0960,104.6341,0.0000000,-0.1556923,0.9878056
GOTO/0.0000,6.0720,104.3557,0.0000000,-0.1167692,0.9931591
GOTO/0.0000,4.0480,104.1578,0.0000000,-0.0778462,0.9969654
GOTO/0.0000,2.0240,104.0394,0.0000000,-0.0389231,0.9992422
GOTO/0.0000,0.0000,104.0000,0.0000000,0.0000000,1.0000000
GOTO/0.0000,-2.0240,104.0394,0.0000000,0.0389231,0.9992422
GOTO/0.0000,-4.0480,104.1578,0.0000000,0.0778462,0.9969654
GOTO/0.0000,-6.0720,104.3557,0.0000000,0.1167692,0.9931591
GOTO/0.0000,-8.0960,104.6341,0.0000000,0.1556923,0.9878056
GOTO/0.0000,-10.1200,104.9943,0.0000000,0.1946154,0.9808796
GOTO/0.0000,-12.1440,105.4379,0.0000000,0.2335385,0.9723476
GOTO/0.0000,-14.1680,105.9673,0.0000000,0.2724615,0.9621667
GOTO/0.0000,-16.1920,106.5852,0.0000000,0.3113846,0.9502840
GOTO/0.0000,-18.2160,107.2950,0.0000000,0.3503077,0.9366347
GOTO/0.0000,-19.2280,107.6856,0.0000000,0.3697692,0.9291236
GOTO/0.0000,-19.7340,107.8900,0.0000000,0.3795000,0.9251917
GOTO/0.0000,-19.9870,107.9946,0.0000000,0.3843654,0.9231811
GOTO/0.0000,-20.0186,108.0078,0.0000000,0.3849736,0.9229276
PAINT/COLOR,37
GOTO/0.0000,-20.5514,108.3359
GOTO/0.0000,-20.9568,108.8125
```

(a) CL points coordinates

(b) Tool path of the CL points

4.1 Experiment 1

The workpiece is a cylinder with a height of 80 mm and a diameter of 40 mm and the origin of the workpiece coordinate system is the center of the upper circle of the cylinder, which is shown in Fig. 4(a). The model is generated by boolean operation of two cylinders and it is shown in Fig. 4(b).

In this example, the coordinate of the rotary center of the A-C axis in a 5-axis tool machine is $(0, 0, -108)$. A ball nose end cutter R2 (radius 2 mm) is selected. The feed rate is constant and it is $250mm/min$.

In the Siemens NX CAM systems, the zigzag tool path pattern is applied and the tool direction parallels the surface normal. The singular points can be found in the neighbor of the middle points in each tool path. The CL points are outputted by the Siemens NX CAM systems, and the CL points including singular points are detailed in Fig. 5.

Based on the CL points, the tool tip can be computed using the inverse kinematic method as usual.

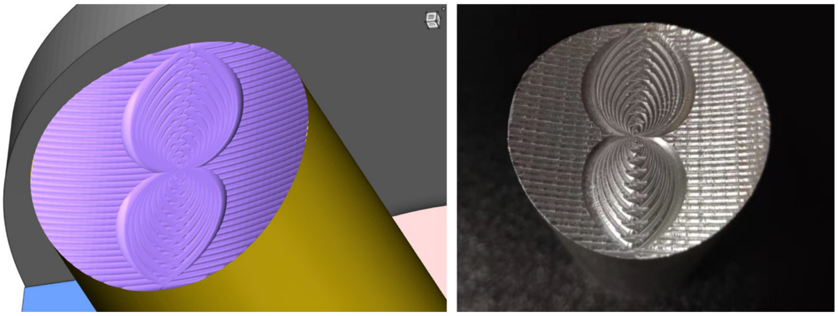
The CL points are used to evaluate the 5-axis joint parameters, and G-code is created by the result of the postprocessor. We find that the postprocessor in Siemens NX CAM systems is based on the inverse kinematic equation (2) and (3).

In the NX CAM generated G-codes, the C rotary axis changes from 180° to 0° and then changes to 180° from 0° and repeat this process. In the actual machine process, the sudden changes of C-axes cause the machine error, which is shown in Fig. 6.

The VERICUT software is used to simulate the G-code, and we find the cut path includes some circles in Fig. 6. The five-axis machine tool used in this example doesn't have the RTCP (Rotation Tool Center Point) function, those circles are the overcut part, which is caused by the sudden changes in the C-axis. In Fig. 5(b), there are 25 points on the selected tool path. B-spline is used to interpolate the positions coordinate of the tool path and quaternion is used to represent the orientations, then quaternion slerp (spherical linear interpolation) is used to interpolate two adjacent orientations.

In equation Eq. 23, the damping factor λ is evaluated directly by the equation Eq. 18. In this equation, ϵ and λ_{max} represent the width of the singular region and maximum

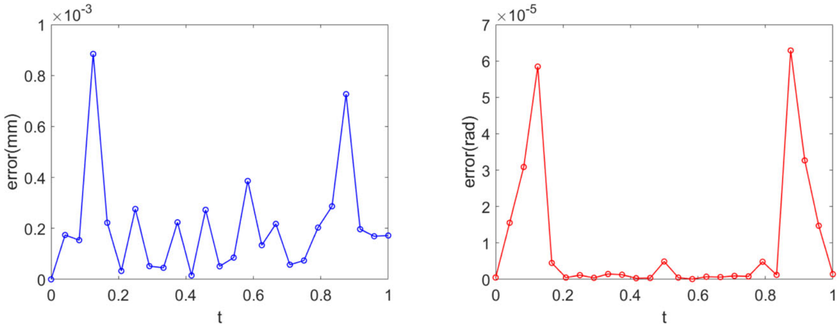
Fig. 6 The simulation and real cut result of the NX CAM generated G-codes



(a) G-code simulation in VERICUT

(b) Real machining result with the G-code.

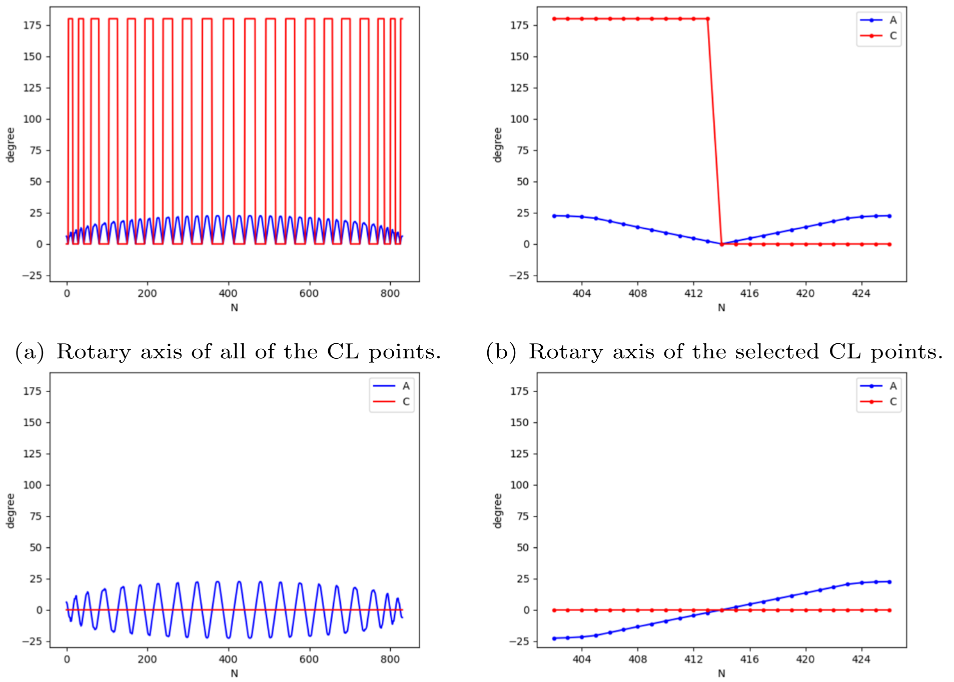
Fig. 7 The deviation between the proposed method and the original 25 CL points when each adjacent point is divided into 50 segments. **a** The maximum tool tip error is $8.8446 \times 10^{-4}mm$, the average tool tip error is $2.0449 \times 10^{-4}mm$; **b** the maximum tool orientation error is $6.2862 \times 10^{-5}rad$, the average tool orientation error is $9.6662 \times 10^{-6}rad$



(a) Tool tip error

(b) Tool orientation error

Fig. 8 The red line is the angle of the C rotary axis and the blue line is the angle of the A rotary axis. The horizontal coordinate represents the order of CL points and the vertical coordinate represents the degree of the rotary axis



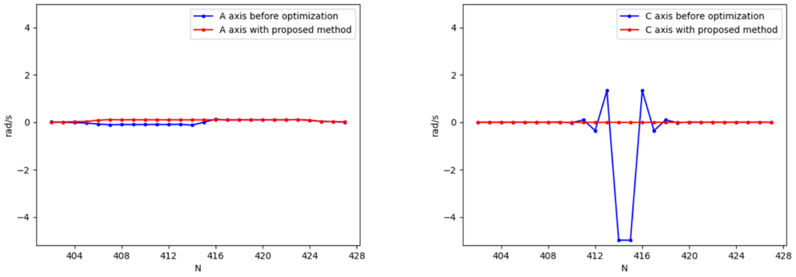
(a) Rotary axis of all of the CL points.

(b) Rotary axis of the selected CL points.

(c) Rotary axis of all of the points with the proposed method.

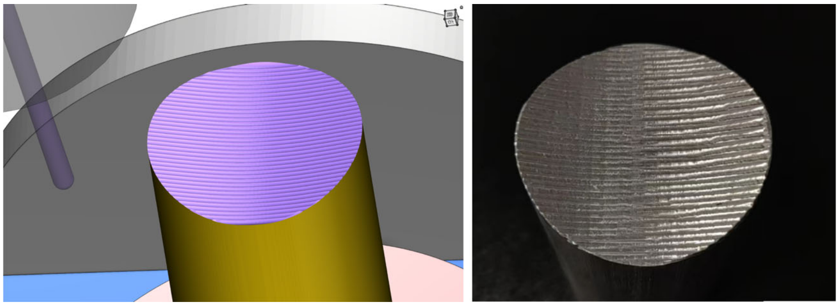
(d) Rotary axis of the selected line with the proposed method.

Fig. 9 The blue line is the initial velocity of the A-C axis, and the red line is the velocity of the A-C axis with the proposed method. The horizontal coordinate represents the order of CL points and the vertical coordinate represents the angular velocity



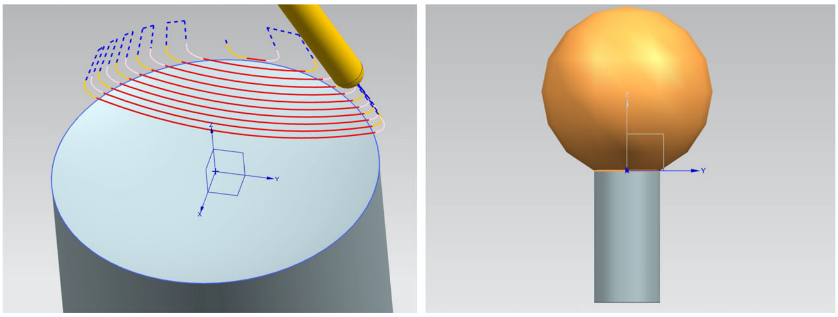
(a) Velocity of A axis in the selected line.

(b) Velocity of C axis in the selected line



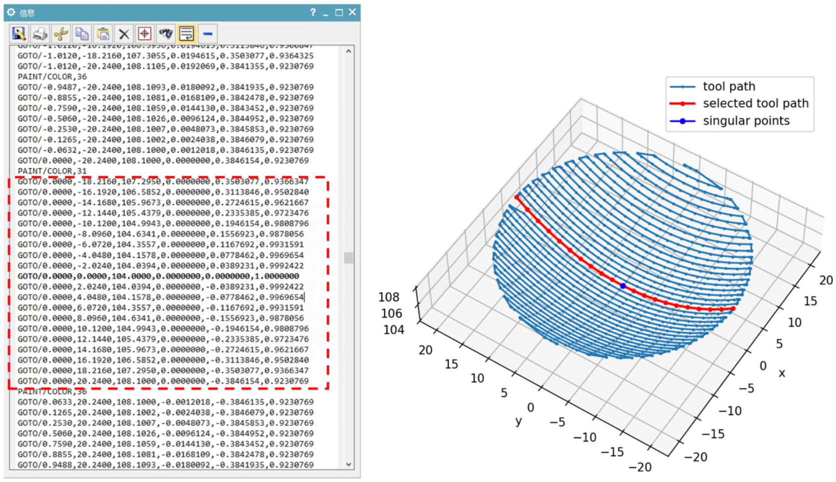
(a) Simulation result of the proposed method. (b) Real cut result of the proposed method.

Fig. 10 The simulation and real cut results of the proposed method



(a) Machining surface and tool path (b) The Boolean operation between ball and cylinder

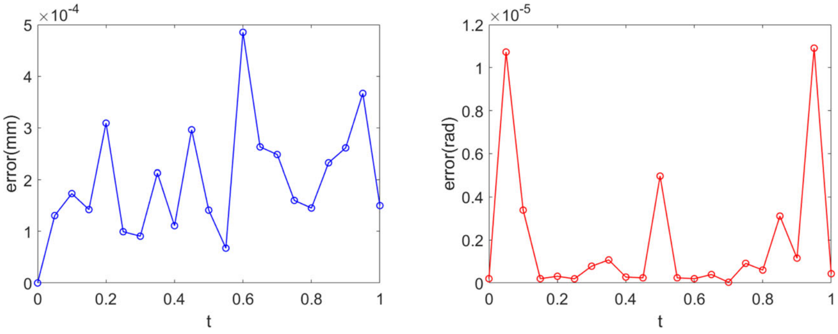
Fig. 11 Machining surface in Siemens NX



```
GOTO/-1.0120,-18.2160,107.3055,0.0194615,0.3503077,0.9364325
GOTO/-1.0120,-20.2400,108.1105,0.0192069,0.3841355,0.9230769
PAINT/COLOR,36
GOTO/-0.9487,-20.2400,108.1093,0.0180092,0.3841935,0.9230769
GOTO/-0.8855,-20.2400,108.1081,0.0168109,0.3842478,0.9230769
GOTO/-0.7590,-20.2400,108.1059,0.0144130,0.3843452,0.9230769
GOTO/-0.5060,-20.2400,108.1026,0.0096124,0.3844952,0.9230769
GOTO/-0.2530,-20.2400,108.1007,0.0048073,0.3845853,0.9230769
GOTO/-0.1265,-20.2400,108.1002,0.0024038,0.3846079,0.9230769
GOTO/-0.0632,-20.2400,108.1000,0.0012018,0.3846135,0.9230769
GOTO/0.0000,-20.2400,108.1000,0.0000000,0.3846154,0.9230769
PAINT/COLOR,31
GOTO/0.0000,-18.2160,107.2950,0.0000000,0.3503077,0.9366347
GOTO/0.0000,-16.1920,106.5852,0.0000000,0.3113846,0.9502840
GOTO/0.0000,-14.1680,105.9673,0.0000000,0.2724615,0.9621667
GOTO/0.0000,-12.1440,105.4379,0.0000000,0.2335385,0.9723476
GOTO/0.0000,-10.1200,104.9943,0.0000000,0.1946154,0.9808796
GOTO/0.0000,-8.0960,104.6341,0.0000000,0.1556923,0.9878056
GOTO/0.0000,-6.0720,104.3557,0.0000000,0.1167692,0.9931591
GOTO/0.0000,-4.0480,104.1578,0.0000000,0.0778462,0.9969654
GOTO/0.0000,-2.0240,104.0394,0.0000000,0.0389231,0.9992422
GOTO/0.0000,0.0000,104.0000,0.0000000,0.0000000,1.0000000
GOTO/0.0000,2.0240,104.0394,0.0000000,-0.0389231,0.9992422
GOTO/0.0000,4.0480,104.1578,0.0000000,-0.0778462,0.9969654
GOTO/0.0000,6.0720,104.3557,0.0000000,-0.1167692,0.9931591
GOTO/0.0000,8.0960,104.6341,0.0000000,-0.1556923,0.9878056
GOTO/0.0000,10.1200,104.9943,0.0000000,-0.1946154,0.9808796
GOTO/0.0000,12.1440,105.4379,0.0000000,-0.2335385,0.9723476
GOTO/0.0000,14.1680,105.9673,0.0000000,-0.2724615,0.9621667
GOTO/0.0000,16.1920,106.5852,0.0000000,-0.3113846,0.9502840
GOTO/0.0000,18.2160,107.2950,0.0000000,-0.3503077,0.9366347
GOTO/0.0000,20.2400,108.1000,0.0000000,-0.3846154,0.9230769
PAINT/COLOR,36
GOTO/0.0633,20.2400,108.1000,-0.0012018,-0.3846135,0.9230769
GOTO/0.1265,20.2400,108.1002,-0.0024038,-0.3846079,0.9230769
GOTO/0.2530,20.2400,108.1007,-0.0048073,-0.3845853,0.9230769
GOTO/0.5060,20.2400,108.1026,-0.0096124,-0.3844952,0.9230769
GOTO/0.7590,20.2400,108.1059,-0.0144130,-0.3843452,0.9230769
GOTO/0.8855,20.2400,108.1081,-0.0168109,-0.3842478,0.9230769
GOTO/0.9488,20.2400,108.1093,-0.0180092,-0.3841935,0.9230769
```

(a) CL points coordinates (b) Tool path of the CL points

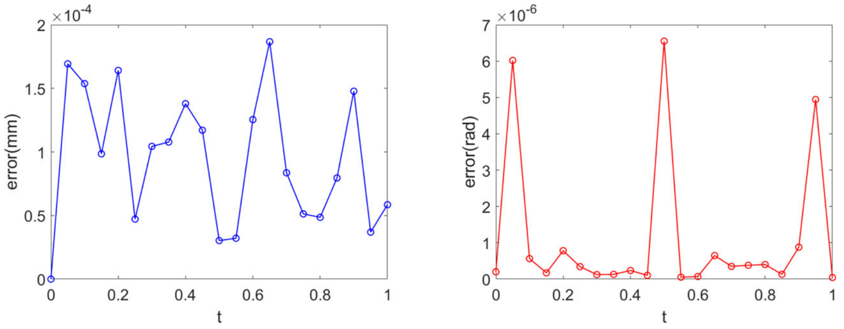
Fig. 12 **a** CL points file generated by Siemens NX CAM systems; **b** tool paths created by the CL points file, the red tool path corresponds to the circled CL points in (a) and the blue points are the singular points, whose orientation is (0, 0, 1)



(a) Tool tip error (b) Tool orientation error

Fig. 13 The deviation between the proposed method and the original 21 CL points when each adjacent point is divided into 50 segments. **a** The maximum tool tip error is $4.8516 \times 10^{-4} mm$, the average tool tip error is $1.9456 \times 10^{-4} mm$; **b** the maximum tool orientation error is $1.0908 \times 10^{-5} rad$, the average tool orientation error is $1.9214 \times 10^{-6} rad$

Fig. 14 The deviation between the proposed method and the original 21 CL points when each adjacent point is divided into 100 segments. **a** The maximum tool tip error is $1.8669 \times 10^{-4} mm$, the average tool tip error is $9.4322 \times 10^{-5} mm$; **b** the maximum tool orientation error is $6.5500 \times 10^{-5} rad$, the average tool orientation error is$1.1011 \times 10^{-6} rad$



(a) Position error

(b) Orientation error

allowed damping factor respectively. Based on the paper [26], we set $\epsilon = \lambda_{max} = 0.001$.

In this example, the diagonal matrix K is set to be $200 * diag[1, 1, 1, 4, 4]$. The $\boldsymbol{r}_d(t_i)$ and $\dot{\boldsymbol{r}}_d(t_i)$ are evaluated by B-spline interpolation and quaternion Slerp.

The tool tip error and tool orientation error compared with the original 25 CL points is shown in Fig. 7. In Fig. 7, horizontal ordinate t is the parameter of B-spline. In this experiment, each adjacent point is divided into 50 segments. In Eq. 23, $N = 24 \times 50 = 1200$, $\Delta t = \frac{1}{N} = \frac{1}{1200}$ and $t \in [0, 1]$. The maximum tool tip error is $8.8446 \times 10^{-4} mm$ and the maximum tool orientation error is $6.2862 \times 10^{-5} rad$, which is small enough for the real five-axis machine tool.

In this example, each tool path has one singular point. The proposed method can also be used for all tool paths, and the A-C values can be optimized globally.

In Fig. 8(a), the C axis changes suddenly in each singular point, while in Fig. 8(c), the C axis almost keeps constant. By comparing Fig. 8(b) and Fig. 8(d), it is easy to find that function A-C axis changes smoothly when crossing the singular point.

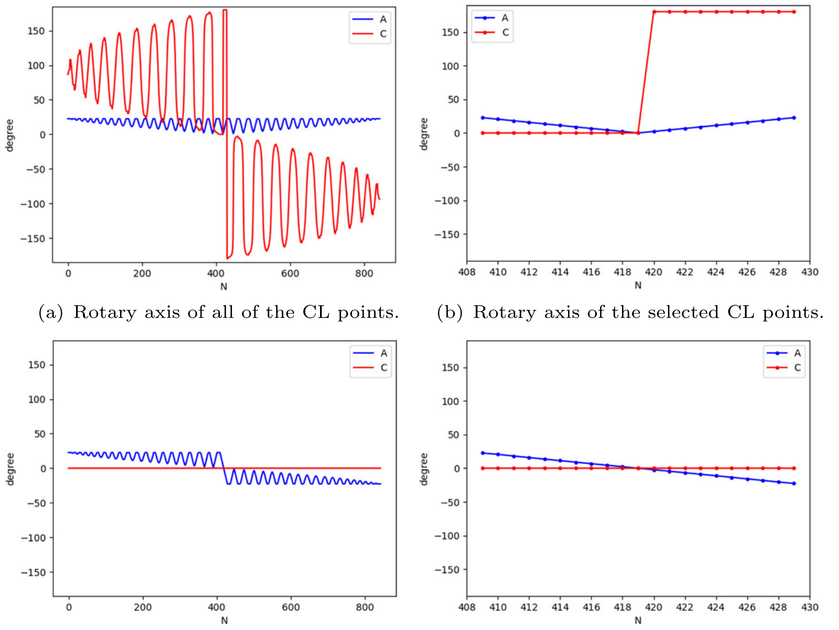
The velocity of the A-C axis after optimization is shown in Fig. 9. In original G-codes, the C axis changes abruptly in singular points. While with the proposed method, the angular velocity of the C axis has decreased significantly.

In Fig. 10(b), there is the real cut result of the proposed method. Compared with Fig. 6, the real cut result with the proposed method is better.

4.2 Experiment 2

The machining surface is designed in Siemens NX software and it is presented in Fig. 11. The model is generated by Boolean operation between a ball and a cylinder, which is shown in Fig. 11(b).

Fig. 15 The red line is the C axis and the blue line is the A axis. The horizontal coordinate represents the order of CL points and the vertical coordinate represents the degree of the rotary axis. The number of the CL points is 841, and the number of the CL points is 21



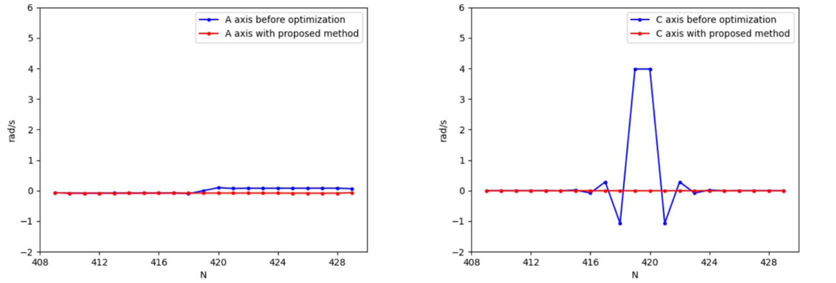
(a) Rotary axis of all of the CL points.

(b) Rotary axis of the selected CL points.

(c) Rotary axis of all of the CL points with the proposed method.

(d) Rotary axis of the selected line with the proposed method.

Fig. 16 The blue line is the initial velocity of the A-C axis, and the red line is the velocity of the A-C axis with the proposed method. The horizontal coordinate represents the order of CL points and the vertical coordinate represents the angular velocity



(a) Velocity of A axis in the selected line. (b) Velocity of C axis in the selected line

The CL points including singular points are shown in Fig. 12.

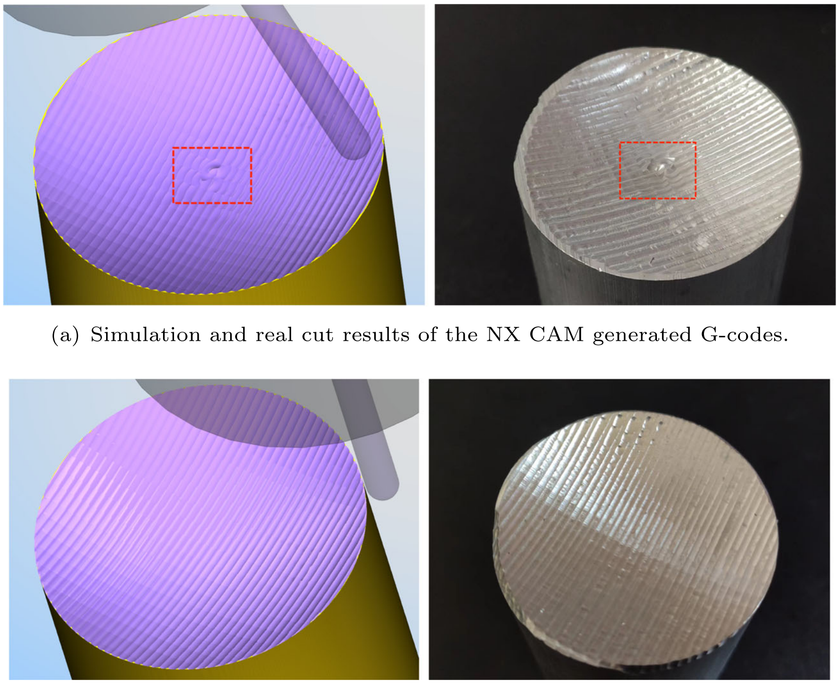
In this example, there is only one singular point, however, the C rotary axis still changes abruptly in each tool path, which is shown in Fig. 15(a). In this example, we also set $\epsilon = \lambda_{max} = 0.001$, $\boldsymbol{K}$=100*diag[1,1,1,4,4]. There are 21 points on the selected tool path and each adjacent point is divided into 50 segments. The tool tip error and tool orientation error compared with the original 21 CL points is shown in Fig. 13. In Fig. 13, the horizontal ordinate is the parameter of B-spline, and $t \in [0, 1]$. In this experimental, $N = 20 \times 50 = 1000$, $\Delta t = \frac{1}{N} = \frac{1}{1000}$. The maximum tool tip error is $4.8516 \times 10^{-4} mm$ and the maximum tool orientation error is $1.0908 \times 10^{-5} rad$, which is small enough for the real five-axis machine tool.

The accuracy can also be improved by dividing the adjacent points into more segments. In Fig. 14, the adjacent points are divided into 100 segments, the maximum tool tip error decreases to $1.8669 \times 10^{-4} mm$ and the maximum tool orientation error decreases to $6.5500 \times 10^{-5} rad$.

The proposed method can also be used for all of the 841 CL points, and the A-C axis values are optimized. In Fig. 15(a), the C axis changes abruptly in each tool path, while in Fig. 15(c), the C axis almost keep constant. By comparing Fig. 15(b) and Fig. 15(d), it is easy to find that C axis changes smoothly when crossing the singular point. The feed rate is set to be $250 mm/min$, and the velocity of A-C axis with the proposed method is shown in Fig. 16.

In Fig. 17(a), the part circled by the red rectangular is the region with singular points. In Fig. 17(b), there are the simulation and real cut results of the proposed method. The result with the proposed method is smoother than the original result, which verifies the effect of the proposed method.

Fig. 17 The simulation and real cut results



(a) Simulation and real cut results of the NX CAM generated G-codes.

(b) Simulation and real cut results of the proposed method.

5 Conclusion

The whole machining process is verified by an A-C type five-axis machine tool. While the orientation of the rotation axis is parallel to the tool direction, the exact solution of the inverse kinematics equation does not exist. Then B-C and A-B type machines also have the singular phenomenon. In the singular points, the Jacobi matrix will be singular or ill-conditioned. This paper proposed a kinematic-based method to avoid singularity for five-axis machine tools by using Jacobian Damping (JD) pseudoinverse, which is suit for all kinds of ill-conditioned problems. And it is natural that the proposed method can be extended to A-B and B-C type machines. The main contributions are as follows:

(1) Jacobian Damping pseudoinverse technology is first used to overcome the ill-condition problem in the five-axis machine tool.
(2) The method optimizes the joint velocities, which means it not only avoids singularity but also makes the rotary axis smoother, and it is shown in the previous two experiments.
(3) Experimental results confirm that the tool tip error and tool orientation error are small enough and the real cut results prove the effectiveness of the proposed method.

This paper provides a general inverse kinematics method for CL points near singular region and the rotary axis can be optimized by a discrete integrator. However, the kinematics method in this paper considers only pose errors. Further improvement can be achieved by taking the acceleration limit into consideration, and the feedback integrator method will be extended to a second-order integrator. In the future, we will extend our kinematics method by considering above aspects to develop better machining strategy.

Author Contributions All authors contributed to the study's conception and design. Material preparation and data collection were performed by Zehong Lu. Analysis was performed by Zehong Lu, Guanying Huo, and Xin Jiang. The first draft of the manuscript was written by Zehong Lu and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding This project is supported by Scientific Research Projects of Colleges and Universities in Anhui Province (No.2022AH051037) and the National Natural Science Foundation of China (Grants No. 12001028, No. 62102013, and No. 62141605).

Declarations

Competing interests The authors declare no competing interests.

References

1. Affouard A, Duc E, Lartigue C, Langeron JM, Bourdet P (2004) Avoiding 5-axis singularities using tool path deformation. Int J Mach Tools Manuf 44(4):415–425
2. Yang J, Altintas Y (2013) Generalized kinematics of five-axis serial machines with non-singular tool path generation. Int J Mach Tools Manuf 75:119–132
3. Wan M, Liu Y, Xing WJ, Zhang WH (2018) Singularity avoidance for five-axis machine tools through introducing geometrical constraints. Int J Mach Tools Manuf 127:1–13
4. Liu Q, Huang T (2019) Inverse kinematics of a 5-axis hybrid robot with non-singular tool path generation. Robotics and Computer-Integrated Manufacturing 56:140–148
5. Sun S, Sun Y, Lee YS (2019) A gouge-free tool axis reorientation method for kinematics compliant avoidance of singularity in 5-axis machining. J Manuf Sci Eng 141(5)
6. Yuwen S, Jinjie J, Jinting X, Mansen C, Jinbo N (2022) Path, feedrate and trajectory planning for free-form surface machining: a state-of-the-art review. Chinese J Aeronaut 35(8):12–29
7. Lin Z, Fu J, Shen H, Gan W (2014) Non-singular tool path planning by translating tool orientations in c-space. Int J Adv Manuf Technol 71(9):1835–1848
8. Lin Z, Fu J, Yao X, Sun Y (2015) Improving machined surface textures in avoiding five-axis singularities considering tool orientation angle changes. Int J Mach Tools Manuf 98:41–49
9. Lin Z, Fu J, Shen H, Xu G, Sun Y (2016) Improving machined surface texture in avoiding five-axis singularity with the acceptable-texture orientation region concept. Int J Mach Tools Manuf 108:1–12
10. Cripps R, Cross B, Hunt M, Mullineux G (2017) Singularities in five-axis machining: cause, effect and avoidance. Int J Mach Tools Manuf 116:40–51
11. Yang J, Aslan D, Altintas Y (2018) Identification of workpiece location on rotary tables to minimize tracking errors in five-axis machining. Int J Mach Tools Manuf 125:89–98
12. Pessoles X, Landon Y, Segonds S, Rubio W (2013) Optimisation of workpiece setup for continuous five-axis milling: application to a fiveaxis bc type machining centre. Int J Adv Manuf Technol 65(1):67–79
13. Gao S, Zhou H, Hu P, Chen J, Yang J, Li N (2020) A general framework of workpiece setup optimization for the five-axis machining. Int J Mach Tools Manuf 149:103508
14. Munlin M, Makhanov SS, Bohez EL (2004) Optimization of rotations of a five-axis milling machine near stationary points. Computer-Aided Design 36(12):1117–1128
15. Shen H, Fu J, Lin Z (2015) Five-axis trajectory generation based on kinematic constraints and optimisation. Int J Comput Integr Manuf 28(3):266–277
16. My CA, Bohez EL (2016) New algorithm to minimise kinematic tool path errors around 5-axis machining singular points. Int J Prod Res 54(20):5965–5975
17. Sørby K (2007) Inverse kinematics of five-axis machines near singular configurations. Int J Mach Tools Manuf 47(2):299–306
18. Gray PJ, Ismail F, Bedi S (2007) Arc-intersect method for 31212-axis tool paths on a 5-axis machine. Int J Mach Tools Manuf 47(1):182–190
19. Grandguillaume L, Lavernhe S, Tournier C (2016) A tool path patching strategy around singular point in 5-axis ball-end milling. Int J Prod Res 54(24):7480–7490
20. Ben-Israel A, Greville TN (2003) Generalized inverses: theory and applications. Springer Science & Business Media 15
21. Nakamura Y, Hanafusa H (1986) Inverse kinematic solutions with singularity robustness for robot manipulator control. J Dyn Syst Meas Control 108(3):163–171

22. Wampler CW (1986) Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. IEEE Trans Syst Man, and Cybern 16(1):93–101
23. Chiaverini S, Egeland O, Kanestrom R (1991) Achieving user-defined accuracy with damped least-squares inverse kinematics. In: Fifth International Conference on Advanced Robotics' Robots in Unstructured Environments, pp. 672–677. IEEE
24. Wampler C, Leifer L (1988) Applications of damped least-squares methods to resolved-rate and resolved-acceleration control of manipulators. Journal of Dynamic Systems, Measurement, and Control 110(1):31–38
25. Siciliano B, Sciavicco L, Villani L, Oriolo G (2009) Differential kinematics and statics. Robotics: Modelling, Planning and Control, 105–160
26. Colomé A, Torras C (2014) Closed-loop inverse kinematics for redundant robots: comparative assessment and two enhancements. IEEE/ASME Transactions on Mechatronics 20(2):944–955

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