



Real-time smooth trajectory generation for 3-axis blending tool-paths based on FIR filtering

Jun Fang^{1,2} · Bingran Li^{1,2} · Hui Zhang^{1,2} · Peiqing Ye^{1,3}

Received: 9 May 2022 / Accepted: 20 March 2023 / Published online: 1 April 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

Three-axis machining tool-paths are generated using computer-aided manufacturing (CAM) systems and programmed by a series of G01 codes mixed with long and short segments. The computer numerical control (CNC) system needs to analyze path information according to the G-codes through path smoothing, speed planning, and interpolation process; generate smooth trajectory; and ensure the high-order continuity of velocity. This study proposes a real-time smooth trajectory generation algorithm with a simple structure and low calculation for 3-axis blending machining tool-paths. Finite impulse response filters are used to generate a smooth trajectory with bounded acceleration in real-time. Due to the delayed response of the filter, it will inevitably bring contour error. The feed rate is adjusted by considering the influence of the adjacent segments on the contour error in the case of long segments. This idea is then extended to continuous micro-segments. Finally, a direct one-step smooth trajectory generation algorithm for the blending tool-paths is realized through pre-discretization with certain rules. The algorithm can effectively control the contour error but also is suitable for blending tool-paths. Consequently, the effectiveness of the proposed algorithms is validated in simulations and also experimented on a 5-axis machine tool.

Keywords CNC · Smooth trajectory · Velocity planning · FIR filter

1 Introduction

In typical 3-axis machining, to ensure the geometric dimension accuracy of the machined workpiece, computer-aided manufacturing (CAM) systems generally use linear lines instead of curves to discretize the tool-paths into a large number of G01 line segments [1]. To realize high-speed continuous motion, computer numerical control (CNC) systems usually adopt the corner smoothing algorithm based on pre-planning [2]. However, given the existence of multiple buffers, the real-time performance of the system is not strong, and spline interpolation has computational efficiency and accuracy issues. Thus, this study presents a

novel real-time smooth trajectory generation technology for accurate interpolation of 3-axis machining.

The G-code generated by CAM is a linear tool-path of point-to-point (P2P) and only has position continuity; thus, it must be completely stopped at the segment transfer, resulting in an extended processing time [3, 4]. To ensure the smoothness of the machining path and the high-order continuity of speed, CNC usually adopts “local blending/smoothing” or “global blending/smoothing” algorithms to improve the geometric continuity of the path and then carries out speed planning and interpolation. In related research, by inserting a line segment, arc, B-spline, and Bezier curve at the adjacent tool-path [5, 6], the smooth transition at the corner is realized and the speed continuity of the path corner is improved. The local smoothing has a good local property and can effectively control the contour error at the corner [6]. Compared with the local smoothing method, the global smoothing method adopts the fitting or interpolation to fit a large number of continuous short linear segments into one or more smooth spline curves under the contour error constraint; commonly used splines include Bezier, B-spline, and NURBS curves [7–9]. However, most algorithms adopt iterative calculation or even offline mode,

✉ Bingran Li
libingran1992@126.com

¹ Department of Mechanical Engineering, Lee Shau-Kei Science & Technology Building, Tsinghua University, Room A946-2, Beijing 100084, China

² Beijing Key Lab of Precision/Ultra-Precision Manufacturing Equipment and Control, Tsinghua University, Beijing, China

³ The State Key Laboratory of Tribology Tsinghua University, Beijing, China

which may affect the real-time performance of the system [2]. The tool-path is often complex and mixed with long and short segments in actual machining. Therefore, local smoothing and global smoothing algorithms must be used in segments according to the track type, and some tracks at the junction of the two fairing algorithms need to be repeatedly faired. The algorithm structure is complex, and the contour accuracy is not easy to control.

After the trajectory is smoothed, the CNC system needs to perform speed planning based on the kinematic constraints of the machine tool, the geometric constraints of the machining path, and the contour errors. The acceleration of S-shaped velocity is continuous, the vibration of the machine tool is reduced, and the surface quality of the workpiece is improved; thus, it is widely used in practical applications [10–12]. Furthermore, algorithms with continuous acceleration and even continuous jerk have also been proposed [13–15]. As the order of velocity continuity increases, the amount of calculation also increases geometrically.

Finite impulse response (FIR) technology has the characteristics of good smoothness, simple calculation, and no spline fitting. Song [16] proved that the velocity curve after FIR filtering is equivalent to the polynomial velocity curve. Thus, some scholars proposed to use the FIR filters to realize the trajectory control process of the CNC system in one step. The algorithm has a simple structure and is conducive to real-time calculation. For the long segment tool-paths, Sencer [17] and Liu [18] proposed a path-smoothing algorithm based on FIR filtering for adjacent corners. The dwell time is deduced according to the given contour error to adjust the overlapping speed area of the adjacent segments and realize the precise contouring control while meeting the requirements of vibration avoidance, high precision, and short machining time. Bearee [19] realized the maximum utilization of acceleration and shortened the processing time by changing the time constant of the filter. Furthermore, in order to realize jerk continuity, Li [20] further deduced the model of constant velocity sequence passing through the third filter and obtained the third-order continuous velocity curve, satisfying the contour error constraint. Another scheme of contour error control is adjusting the adjacent speed within the delay time, also called the velocity-controlled blending algorithm [21], which can further reduce velocity fluctuation at the corner.

The above studies only considered two adjacent trajectories at the corner but did not give continuous micro-segment processing methods. Hayasaka [22] proposed a block-splitting method to adjust the speed within the delay time and realized the machining of continuous micro-segments based on FIR filtering. Sencer [23, 24] proposed a

highly computationally efficient global smoothing interpolation method and determined the final velocity proportion coefficient by solving the influence of each corner on multi-stage speed. In the algorithm, trapezoidal shaped velocity (acceleration limited) is used to solve the contour error, but S-shaped velocity (jerk limited) is commonly used in processing.

This study proposes a universal algorithm of acceleration continuity based on FIR filtering for complex path machining of CNC systems. Using second-order FIR filters, the interpolation smoothing trajectory with continuous acceleration can be generated by adjusting the velocity proportion coefficient. Firstly, a non-stop velocity proportion compression algorithm for continuous long linear segments is proposed, and the adjustment coefficients of the speed of the front and rear segments are calculated according to the contour errors at the corners. On this basis, a blending velocity proportion compression algorithm for continuous micro-segments is proposed. Considering the influence of multiple trajectories before and after a single corner, the final velocity proportion compression coefficient is calculated, so that the generated trajectory satisfies the contour of each corner. Finally, a pre-discrete velocity proportion compression algorithm for blending tool-paths is proposed, which is suitable for complex trajectories in practical machining. Thus, the paper is organized as follows. Section 2 introduces the generation of acceleration continuous jerk-bounded tool-paths based on FIR filters. Section 3 presents the continuous trajectory generation that satisfies the contour error constraint. Section 4 verifies the effectiveness of the algorithm through simulation and actual machining experiments.

2 Jerk-bounded trajectory generation based on FIR filtering

2.1 FIR filter principle

The commonly used velocity acceleration curve of the numerical control system is S-shaped velocity; that is, the acceleration is continuous, and the jerk is bounded, which can be easily obtained by passing through two cascaded FIR filters through the equivalent speed sequence. The transfer function (TF) of the first-order FIR filter under Laplace(s) domain is as follows:

$$H_i(s) = \frac{1}{T_i} \frac{1 - e^{-sT_i}}{s}, i = 1 \dots N \quad (1)$$

where T_i is the time duration for the i th FIR filter. The transfer function consists of an integrator $\frac{1}{s}$ and a pure delay

element e^{-sT_i} . The impulse response is obtained by inverse Laplace transform:

$$h_i(t) = L^{-1}(H_i(s)) = \frac{u(t) - u(t - T_i)}{T_i} \text{ where } u(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (2)$$

$h_i(t)$ is a rectangular pulse of unit area, as shown in Fig. 1, revealing that the final position of the trajectory will not change after passing through the FIR filter, which is the “position invariance” of the FIR filter [18].

2.2 Generation of S-shaped velocity curve

In 3-axis machining, for each line of G-code, the start position coordinate $P_s(X_s, Y_s, Z_s)$, end position coordinate $P_E(X_E, Y_E, Z_E)$ and feed rate F can be obtained by compiling and analyzing. The speed planning of the rectangular-frame is obtained from the segment length L and the feed rate F :

$$L = \| P_E - P_s \| \text{ and } T_v = \frac{L}{F} \quad (3)$$

where T_v represents the duration of the rectangular frame speed pulse. When the velocity curve of the rectangular-frame is convolved with an FIR filter with a time constant of T_1 and then convoluted with an FIR filter with a time constant of T_2 , the S-shaped velocity curve commonly used in CNC system can be obtained [16]. To get a seven-segmented S-shaped velocity curve, that is, the acceleration curve of the system is a trapezoid and its jerk is bounded; it must also be ensured that $T_v > T_1 + T_2$ and $T_2 < T_1$. At this time, the maximum acceleration and jerk of the system are calculated as:

$$a_{\max} = \frac{F}{T_1} \text{ and } j_{\max} = \frac{F}{T_1 T_2} \quad (4)$$

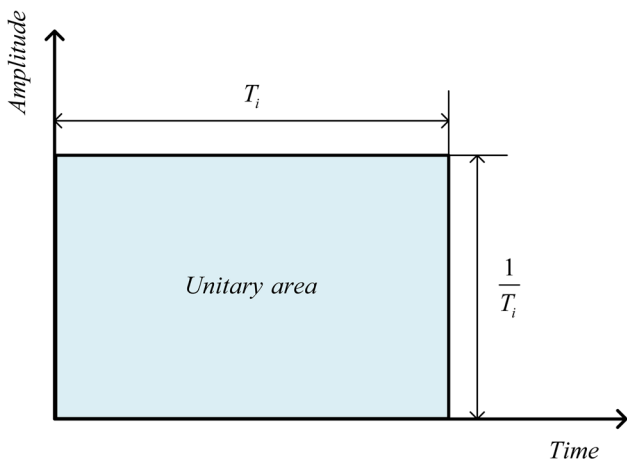


Fig. 1 Impulse response of FIR filter

After the filtering effect of these two cascaded FIR filters, the obtained speed is as follows:

$$v'(t) = v(t) * h_1(t) * h_2(t) = \begin{cases} \frac{1}{2} \frac{F}{T_1 T_2} t^2 & 0 \leq t < T_2 \\ \frac{1}{2} \frac{F T_2}{T_1} + \frac{F}{T_1} (t - T_2) & T_2 \leq t < T_1 \\ F - \frac{1}{2} \frac{F}{T_1 T_2} ((T_1 + T_2) - t)^2 & T_1 \leq t < T_1 + T_2 \\ F & T_1 + T_2 \leq t < T_v \\ F - \frac{1}{2} \frac{F}{T_1 T_2} (t - T_v)^2 & T_v \leq t < T_v + T_2 \\ F - \frac{1}{2} \frac{F T_2}{T_1} - \frac{F}{T_1} (t - (T_v + T_2)) & T_v + T_2 \leq t < T_v + T_1 \\ \frac{1}{2} \frac{F}{T_1 T_2} ((T_v + T_1 + T_2) - t)^2 & T_v + T_1 \leq t < T_v + T_1 + T_2 \\ 0 & t \geq T_v + T_1 + T_2 \end{cases} \quad (5)$$

The acceleration and position can be obtained by differentiating and integrating the velocity. The whole process is shown in Fig. 2.

The total processing time of the system is calculated as:

$$T = T_v + T_d = T_v + T_1 + T_2, \text{ where } T_d = T_1 + T_2 \quad (6)$$

where T_d is the delay time brought by the introduction of the filter. If the system introduces an n th order FIR filter, theoretically, the speed curve can achieve n th order continuity [17, 25], and the delay time T_d is calculated as:

$$T_d = T_1 + \dots + T_n \quad (7)$$

3 Continuous trajectory generation under contour error constraints

It has been introduced that a single-segment S-shaped velocity curve with continuous acceleration can be generated based on FIR filtering. Thus, this chapter will further introduce the trajectory generation method for continuous tool-paths following contour control. Section 3.1 first solves the velocity proportion coefficients of the two front and rear trajectories under the constraint of contour error at a single corner for long segments. The velocity proportion coefficient is used to adjust the speed of each track segment on the premise that the track segment length remains unchanged, so as to meet the contour accuracy at the track corner. Section 3.2 extends this idea to the case of continuous micro-segments, solves the influence of multiple trajectories at a single corner under the constraint of contour error one by one, and obtains the velocity proportion coefficient of each micro-segment by synthesizing the results at multiple corners. Section 3.3 adopts pre-discretization to scatter the complex trajectory into continuous micro-segments to satisfy the application of the algorithm in Sect. 3.2 and realize a universal continuous trajectory generation method based on FIR filtering.

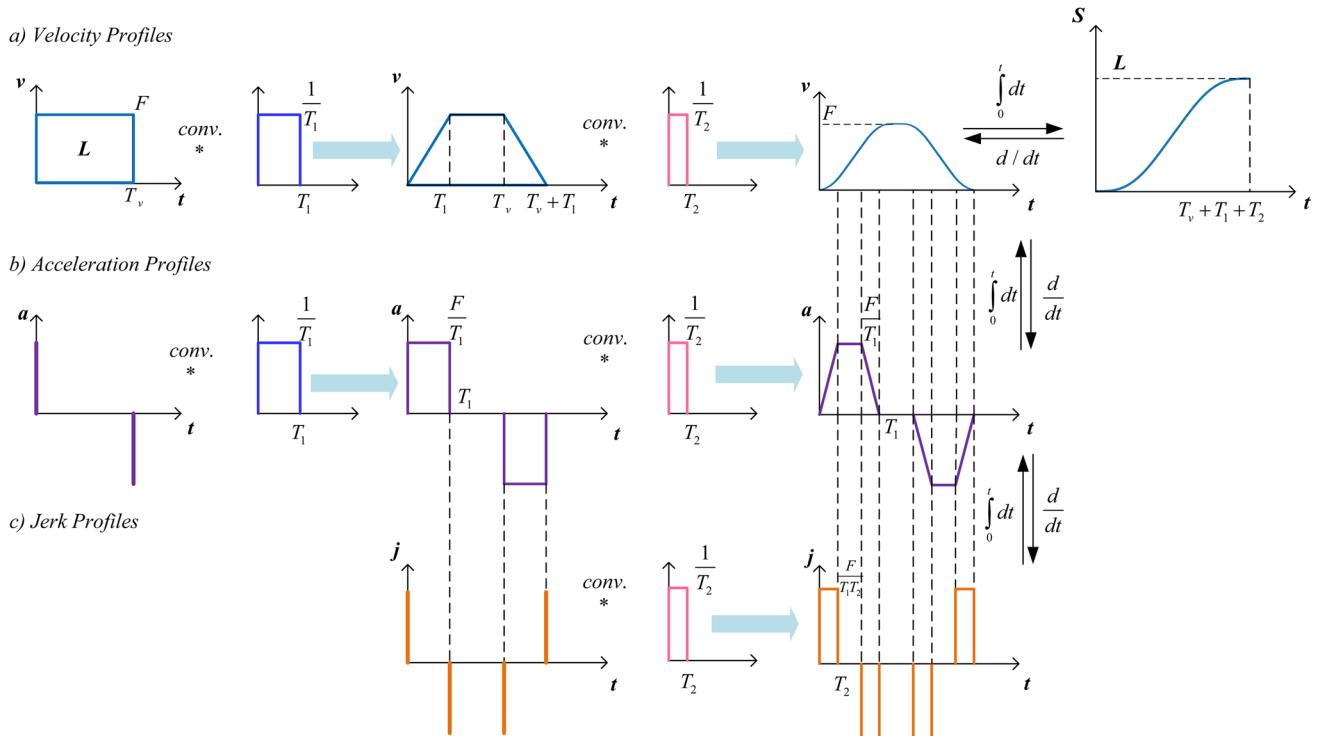


Fig. 2 S-shaped speed curve generation by FIR filtering

3.1 Non-stop velocity proportion compression for long segments

As described in Section 2, the premise of obtaining the S-shaped curve of the maximum feed speed F is to ensure $T_v > T_1 > T_2$ and $T_v \geq T_d = T_1 + T_2$. Then, for continuous trajectory, by carrying out rectangular-frame speed planning (RSP), a series of rectangular-frame speeds can be obtained. If the dwell time between adjacent rectangular-frame is greater than or equal to the delay time, a full-stop and an exact P2P motion can be obtained; that is, the system will continue to run the next track after each track has completely stopped, as shown in Fig. 3. At this time, the system shows no contour error, but this movement will seriously affect the machining efficiency of the system and greatly increase the machining time. Therefore, Ref. [21] proposed two methods: non-stop dwell-time control path blending (NS-DCB) and non-stop velocity-controlled blending (NS-VCB). By adjusting the dwell time between adjacent rectangular-frame (corresponding to NS-DCB) or the mixing velocity between adjacent rectangular-frame (corresponding to NS-VCB), the system can shorten the processing time to satisfy the contour error constraint. However, as pointed out in the literature, both methods are only suitable for long segments, which require $T_v \geq T_d$.

To suggest a general trajectory control algorithm based on FIR filtering for long segments, this paper proposes a

non-stop velocity proportion compression algorithm (NS-VPC), which makes the system dwell time $T_{dwell} = 0$ and the mixing time $T_{mix} = T_d - T_{dwell} = T_d$ at the corner. At this time, the system starts to perform the trajectory interpolation of the second segment when the first segment of the trajectory has not reached the end position, resulting in mixed corner and contour errors, as shown in Fig. 4.

Line N_1 corresponds to the starting point $P_1(X_1, Y_1, Z_1)$, line N_2 corresponds to the intermediate point $P_2(X_2, Y_2, Z_2)$, and line N_3 corresponds to the ending point $P_3(X_3, Y_3, Z_3)$. The unit vector of the N_1N_2 track segment is \vec{O}_1 , the unit vector of the N_2N_3 track segment is \vec{O}_2 , and it is assumed that the feed rate of the two tracks is the same, that is, $F_1 = F_2 = F$. The maximum contour error of the system occurs at the angle bisector of $\angle P_1P_2P_3$ [21], the corresponding time is set to T_b , and the maximum contour error is set to ϵ .

Where

$$\begin{cases} T_b = T_{v1} + \frac{1}{2}T_d \\ \epsilon = \|\vec{P}_2\vec{P}_b\| = \|\vec{l}_2 - \vec{l}_1\| \end{cases} \quad (8)$$

\vec{l}_1 corresponds to the vector of the post-influence area s_1 at the maximum contour error of the first track, and \vec{l}_2 corresponds to the vector of the pre-influence area s_2 at the maximum contour error of the second track.

From Eq. (5),

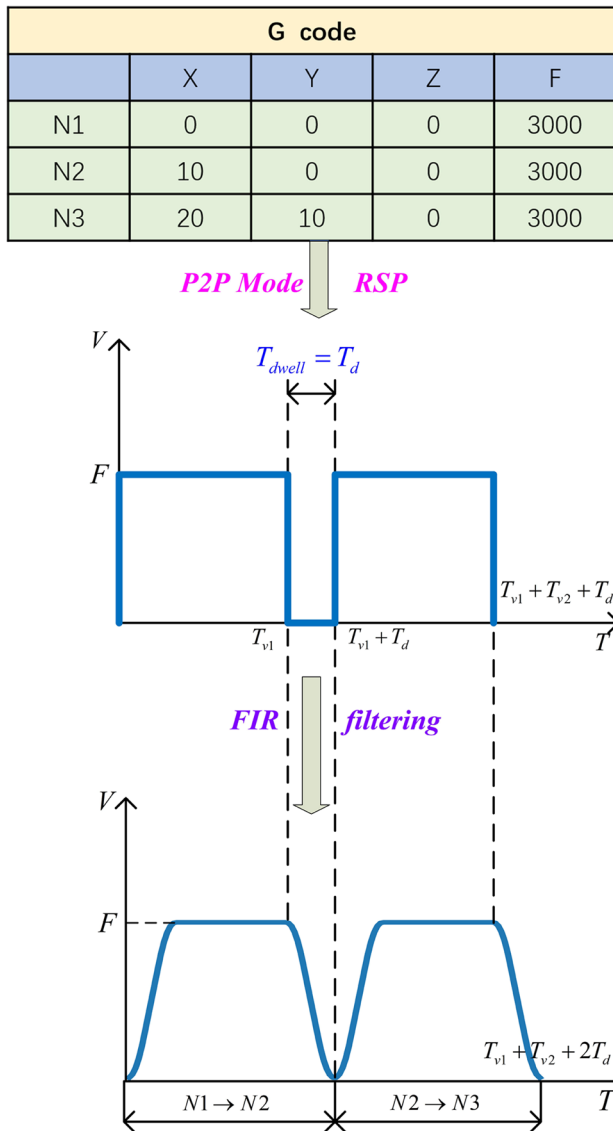
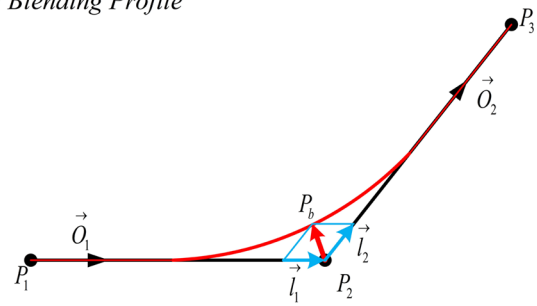


Fig. 3 P2P mode by FIR filtering

a) Blending Profile



b) Velocity Profile

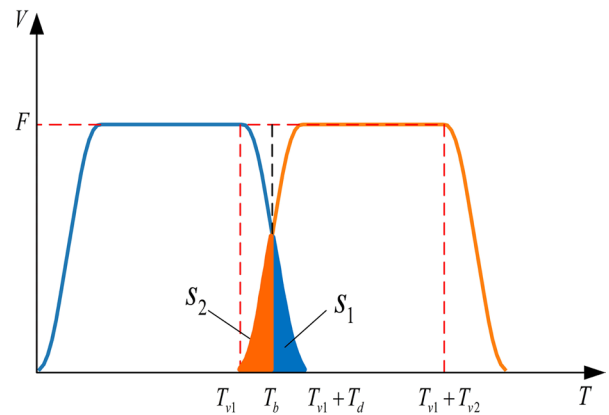


Fig. 4 Blending error under long segments

$$\epsilon = \| (s_2x_2 - s_1x_1, s_2y_2 - s_1y_1, s_2z_2 - s_1z_1) \| \tag{12}$$

Assuming that the allowable error given by the numerical control system is ϵ_{allow} , the velocity proportion coefficient k_v should be calculated as:

$$k_v = \frac{\epsilon_{allow}}{\epsilon} \tag{13}$$

To satisfy the contour error of the system, the velocity of the N_1N_2 trajectory segment and the N_2N_3 trajectory segment should be adjusted is calculated as:

$$s_1 = s_2 = \int_0^{t=\frac{T_d}{2}} v'(t)dt = \begin{cases} \frac{1}{6} \frac{FT_1^2}{T_1} + \frac{1}{2} \frac{FT_2}{T_1} (T_1 - T_2) + \frac{1}{2} \frac{F}{T_1} (t - T_2)^2 & T_d < 2T_1 \\ \frac{1}{6} \frac{FT_1^2}{T_1} + \frac{1}{2} \frac{FT_2}{T_1} (T_1 - T_2) + \frac{1}{2} \frac{F}{T_1} (T_1 - T_2)^2 - \frac{1}{6} \frac{F}{T_1 T_2} (t - T_1)^3 + \frac{1}{2} \frac{F}{T_1} (t - T_1)^2 + \frac{1}{2} \frac{F(T_1 - T_2)}{T_1} (t - T_1) & T_d \geq 2T_1 \end{cases} \tag{9}$$

Therefore,

$$\begin{cases} \vec{l}_1 = s_1 \cdot \vec{O}_1 \\ \vec{l}_2 = s_2 \cdot \vec{O}_2 \end{cases} \tag{10}$$

where

$$\begin{cases} \vec{O}_1 = \frac{(X_2 - X_1, Y_2 - Y_1, Z_2 - Z_1)}{L_1} = (x_1, y_1, z_1) \\ \vec{O}_2 = \frac{(X_3 - X_2, Y_3 - Y_2, Z_3 - Z_2)}{L_2} = (x_2, y_2, z_2) \end{cases} \tag{11}$$

Thus, by plugging Eqs. (9)–(12) into Eq. (8), ϵ is calculated as:

$$\begin{cases} v'_1 = k_v \cdot F_1 \\ v'_2 = k_v \cdot F_2 \end{cases} \quad (14)$$

To ensure that the length of the track segment remains unchanged, the time of these two segments should also be adjusted accordingly.

$$\begin{cases} T'_{v,1} = \frac{L_1}{v'_1} \\ T'_{v,2} = \frac{L_2}{v'_2} \end{cases} \quad (15)$$

It should be noted that the above derivation is based on $F_1 = F_2$. If $F_1 \neq F_2$, the maximum contour error at the corner is not at the angle bisector of $\angle P_1P_2P_3$, but the actual contour error is greater than the calculated value of Eq. (12), and the above algorithm can still meet the contour error constraint of the system [21].

3.2 Multi-stage mixing velocity proportion compression for micro-segments

In CNC machining, CAM usually breaks up the spline into a large number of continuous tiny linear segments. Then, because of $T_v < T_2 < T_1$, the NS-VPC algorithm is no longer applicable. Therefore, based on the NS-VPC algorithm in Section 3.1, this section proposes a multi-stage mixing velocity proportion compression (MM-VPC) algorithm for continuous micro-segments, which can effectively ensure that the system meets the contours. Under the premise of error constraints, a smooth transition of the angular velocity is achieved.

Given $T_v < T_2 < T_1$, the maximum speed and acceleration that the system can reach at this time is calculated as:

$$\begin{cases} v_{\max} = \frac{FT_v}{T_1} \\ a_{\max} = \frac{FT_v}{T_1T_2} \end{cases} \quad (16)$$

The jerk is the same as that in the case of the long segment, which is still $J_{\max} = \frac{F}{T_1T_2}$. Then, after the rectangular-frame speed passes through the second-order FIR filter, the S-shaped speed curve is shown in Fig. 5.

Its speed formula is calculated as:

$$v''(t) = v(t) * h'_1(t) * h'_2(t)$$

$$= \begin{cases} \frac{1}{2} \frac{F}{T_1T_2} t^2 & 0 \leq t < T_v \\ \frac{1}{2} \frac{FT_v^2}{T_1T_2} + \frac{FT_v}{T_1T_2} (t - T_v) & T_v \leq t < T_2 \\ \frac{FT_v}{T_1} - \frac{1}{2} \frac{F}{T_1T_2} ((T_v + T_2) - t)^2 & T_2 \leq t < T_v + T_2 \\ \frac{FT_v}{T_1} & T_v + T_2 \leq t < T_1 \\ \frac{FT_v}{T_1} - \frac{1}{2} \frac{F}{T_1T_2} (t - T_1)^2 & T_1 \leq t < T_v + T_1 \\ \frac{FT_v}{T_1} - \frac{1}{2} \frac{FT_v^2}{T_1T_2} - \frac{FT_v}{T_1T_2} (t - (T_v + T_1)) & T_v + T_1 \leq t < T_1 + T_2 \\ \frac{1}{2} \frac{F}{T_1T_2} ((T_v + T_1 + T_2) - t)^2 & T_1 + T_2 \leq t < T_v + T_1 + T_2 \\ 0 & t \geq T_v + T_1 + T_2 \end{cases} \quad (17)$$

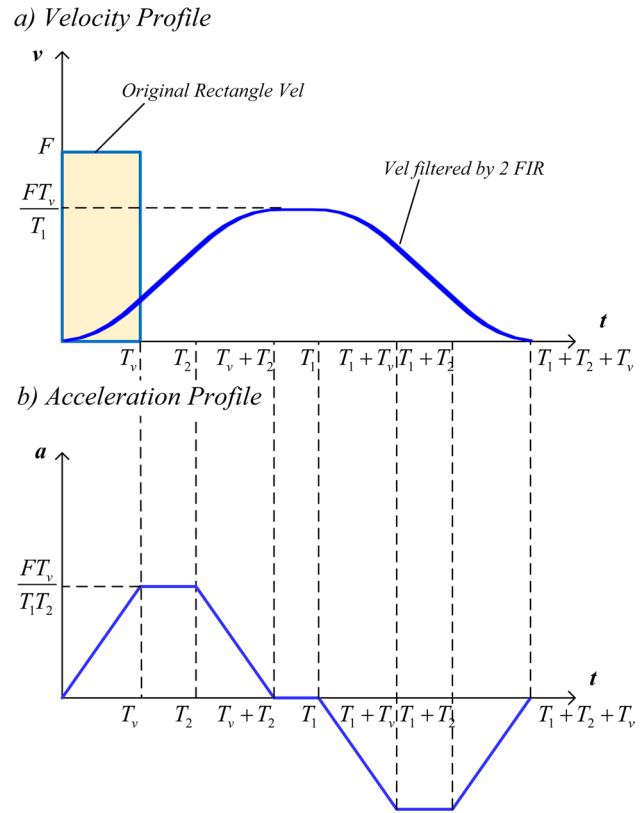


Fig. 5 S-shaped velocity curve

For a continuous micro-segment trajectory, the maximum value of the contour error at the k th contour point is generated in $T_{b,k}$, and the maximum contour error ϵ_k is affected by the displacement vector of the rear area of the front m rectangular-frames and the front area displacement of the rear n rectangular-frames, as shown in Fig. 6.

Where,

$$\begin{cases} T_{b,k} = T_k + \frac{T_d}{2} \\ \epsilon_k = \left\| \sum_{i=0}^{n-1} \overrightarrow{l_{k+i}} - \sum_{j=0}^{m-1} \overrightarrow{l_{k-j-1}} \right\| \end{cases} \quad (18)$$

where T_k represents the starting time of the k th rectangular-frame. Then, the key of the algorithm is to solve the displacement vectors $\overrightarrow{l_{k+i}}$ and $\overrightarrow{l_{k-j-1}}$.

$$\begin{cases} \overrightarrow{l_{k+i}} = \left(\int_0^{t_{k+i}} v''(t) dt \right) \cdot \overrightarrow{O_{k+i}}, i = 0, \dots, n - 1 \\ \overrightarrow{l_{k-j-1}} = \left(L_{k-j-1} - \int_0^{t_{k-j-1}} v''(t) dt \right) \cdot \overrightarrow{O_{k-j-1}}, j = 0, \dots, m - 1 \end{cases} \quad (19)$$

where L_{k-j-1} represents the length corresponding to the $(k - j - 1)$ th rectangular-frame, t_{k+i} represents the influence time of each rectangular-frame in the previous m rectangular-frames on the contour error point and $\overrightarrow{O_{k+i}}$ is its direction vector. t_{k-j-1} represents the influence time of each of the

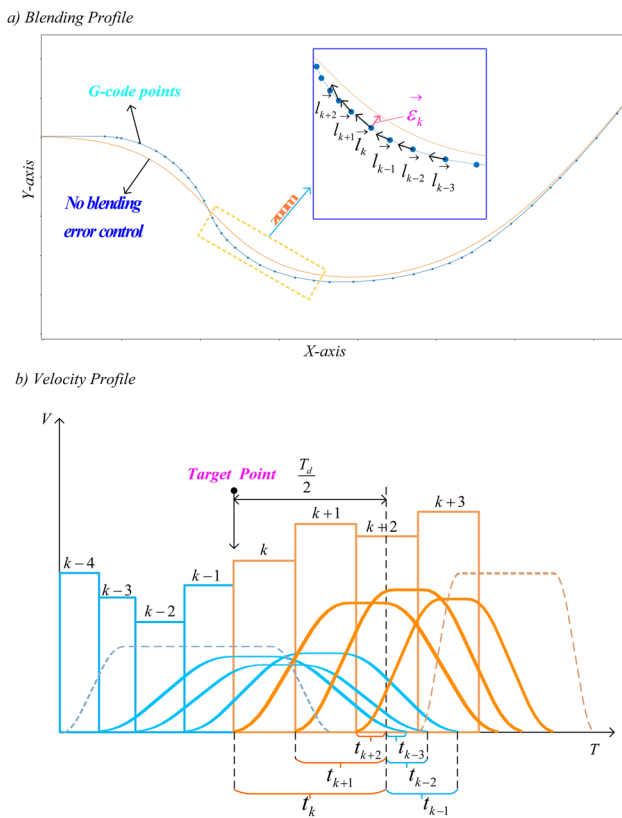


Fig. 6 Blending error under micro-segments

following n rectangular-frames on the contour error point, and O_{k-j-1} is its direction vector.

$$\begin{cases} t_{k+i} = T_{b,k} - T_{k+i}, i = 0, \dots, n - 1 \\ t_{k-j-1} = T_{k-j-1} + T_{v,k-j-1} + T_d - T_{b,k}, j = 0, \dots, m - 1 \end{cases} \quad (20)$$

By substituting Eqs. (19) and (20) into Eq. (18), the contour error ϵ_k of the system at the k th contour point can be obtained, and the velocity proportion coefficient $k_{v,k}$ is calculated as:

$$k_{v,k} = \frac{\epsilon_{allow,k}}{\epsilon_k} \quad (21)$$

Therefore, the velocity scale coefficient of the $m + n$ rectangular-frames at the k th contour point is $k_{v,k}$. Given that each rectangular-frame may affect more than one corner, it may have multiple speed scale factors. The minimum value is taken as the final velocity proportion coefficient among the multiple velocity proportion coefficients of each rectangular-frame to ensure that the contour error of each contour point meets the constraint.

$$k'_v = \min\{k_{v,0}, \dots, k_{v,k}, \dots, k_{v,n}\} \quad (22)$$

Its speed and duration are also adjusted accordingly as:

$$\begin{cases} F' = k'_v \cdot F \\ T'_v = \frac{L}{F'} \end{cases} \quad (23)$$

3.3 Pre-discrete velocity proportion compression algorithm for blending tool-paths

In actual processing, most G-codes are mixed paths with long segments and micro-segments. In this case, neither the NS-VPC nor MM-VPC algorithms are applicable anymore. Thus, this section proposes a pre-discrete velocity proportion compression algorithm (PD-VPC) for mixed paths, which can meet the needs of various machining paths and improve machining efficiency by effectively ensuring contour error constraints.

The key to the PD-VPC algorithm is to pre-discretize in advance, break the long G-codes into micro-segments based on certain rules, and then put the broken G-code into the MM-VPC algorithm to realize speed adjustment and interpolation. The algorithm flow is shown in Fig. 7.

The premise of the MM-VPC algorithm is to satisfy $T_v < T_2 < T_1$, the critical condition $T_v = T_2$, and the system acceleration reaching the maximum value at this time. Thus, the minimum time for segmentation is $T_{div,min} = T_2$. Firstly, the entire G-code is read in advance, the speed planning of the rectangular-frame is carried out, the line segment $T_v \geq T_{div,min}$ is marked as a long segment, and linear interpolation is performed on the long segment and discretize into micro-segments. To break up the long segment into a certain number of micro-segments, a discrete coefficient k_s is introduced, and the duration of each micro-segment is denoted as $T_{v,s}$,

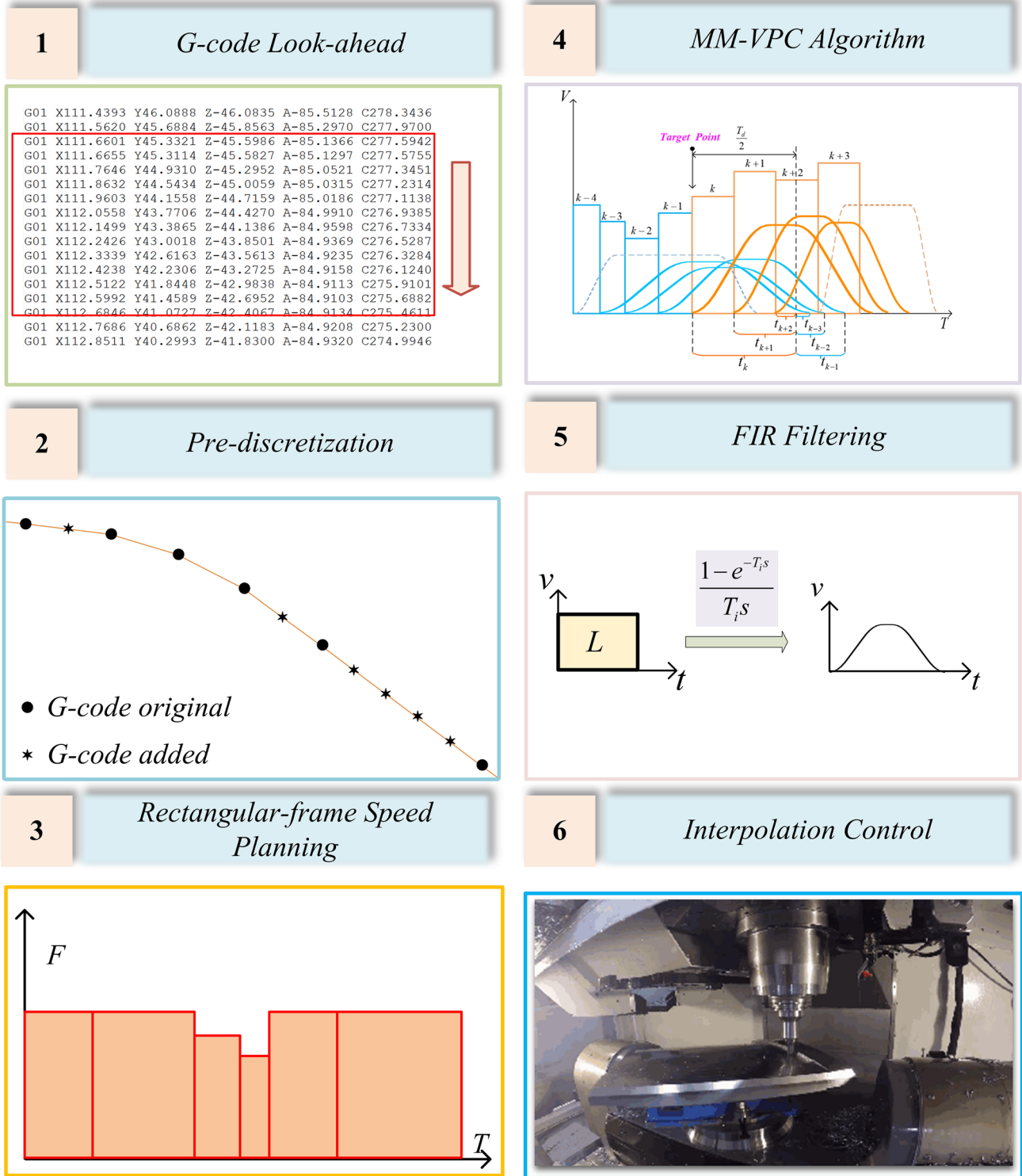
$$T_{v,s} = k_s \cdot T_2, k_s \in (0, 1) \quad (24)$$

It should be noted that the value of k_s will not affect the actual running time and velocity acceleration trend of G-code trajectory but only break the long segment into micro-segments for application to the MM-VPC algorithm. However, the value of k_s will directly affect the number of broken segments, thus affecting the operational efficiency of the algorithm.

Assuming that the running time of a long segment of G-code is denoted as T_i , the coordinates of the starting point are $P_{i-1}(X_{i-1}, Y_{i-1}, Z_{i-1})$, and the coordinates of the end point are $P_i(X_i, Y_i, Z_i)$. Then, the number of short segments that will be broken up into the following:

$$N = \text{ceil}\left(\frac{T_i}{T_{v,s}}\right), N \in N^* \text{ and } N \geq 2 \quad (25)$$

The ceil function means rounding up. Through linear interpolation, the newly added intermediate position G-code point $P_n = (X_n, Y_n, Z_n)$ can be obtained. The coordinates of each axis are solved as follows where $n = 1, 2, \dots, N - 1$.



$$\begin{aligned}
 X_n &= X_{i-1} + (X_i - X_{i-1}) * \frac{n}{N} \\
 Y_n &= Y_{i-1} + (Y_i - Y_{i-1}) * \frac{n}{N} \\
 Z_n &= Z_{i-1} + (Z_i - Z_{i-1}) * \frac{n}{N}
 \end{aligned}
 \tag{26}$$

4 Experimental results

This study uses an actual 5-axis machine tool for practical verification in combination with the interpolation point data generated by the simulation experiment. The experimental machine tool is shown in Fig. 8. The machine tool adopts five-axis full closed-loop control and supports locking the rotating axes. The maximum feed speed of the

linear axis is 48,000 [mm/min], and the control resolution is 0.001 [mm]. In this section, the rotating axes are locked in the whole experiment, and the closed-loop sampling period of the servo control system is set to 1 [kHz].

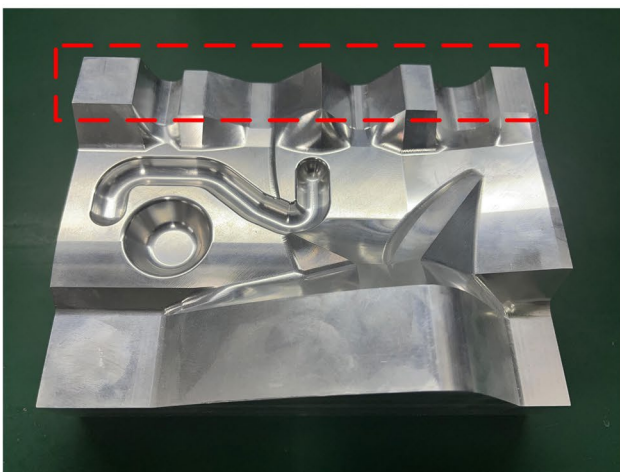
The Mercedes-Benz mold testing sample is the standard test sample of 3-axis milling in the industry [2]. It is recognized by many countries in Europe and the USA and is widely used to evaluate NC machine tools' machining efficiency and accuracy. The specimen's surface contains various complex tool-path features such as lines, curves and surfaces, as shown in Fig. 9a. By importing the G-code of the Mercedes Benz mold testing sample into the PD-VPC algorithm proposed in this paper, given the FIR filter parameters and contour error (the specific parameters are the same as below), the interpolation points that meet the actual processing requirements can be obtained. The speed expressed by the coloring method is shown in Fig. 9b. The dark red represents the maximum speed of 3000 [mm/min], and the dark blue represents the speed approaching 0. It can be seen from the figure that the speed trend conforms to the track characteristics of the Mercedes Benz sample. There are different degrees of deceleration at each corner, and the maximum speed is basically obtained at long segments and small curvature.

As the Mercedes Benz sample is composed of several regions, and the number of G-code points for finish machining exceeds 620 thousand, this study needs to select a representative path for specific comparison. The G-code tool-paths in the red dotted box in Fig. 9 has a large curvature change, and the complexity of the track segment length is the largest. Thus, this study uses a section of the line-cutting path as the processing path, as shown in Fig. 10, to verify the feasibility and effectiveness of the algorithm. The red point represents the command position of G-code, and the



Fig. 8 Experimental machine tool

a) Benz Pattern



b) The distribution of the path speed on the Benz Pattern

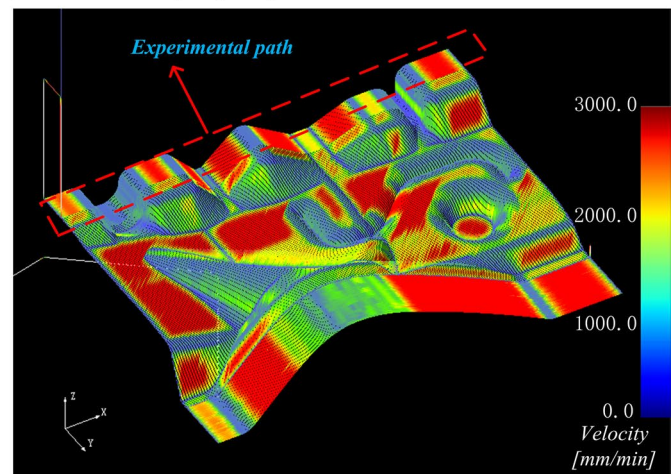


Fig. 9 The Mercedes Benz pattern

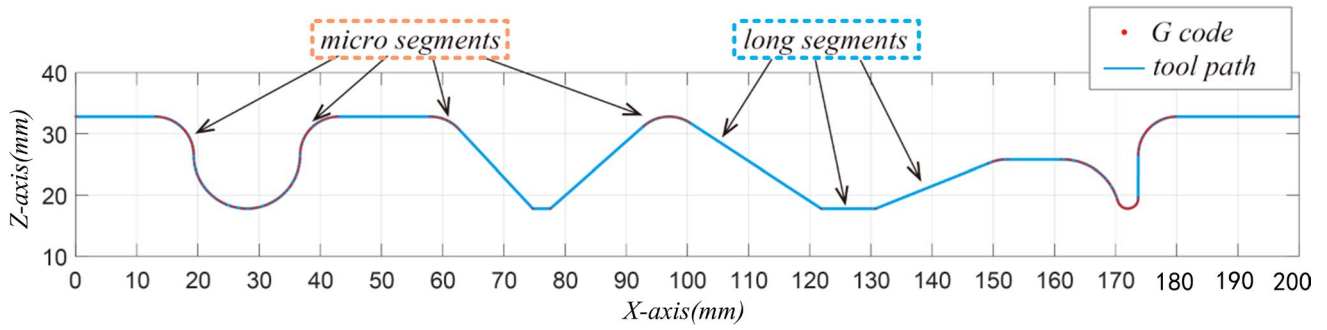


Fig. 10 The line-cutting of Benz pattern

blue line is the tool-path formed by the command position. The tool-path is composed of long segments and continuous micro-segments, and the trajectory contour includes lines, curves, sharp corners, and rounded corners. This multi-feature complex tool-path is representative of practical NC machining.

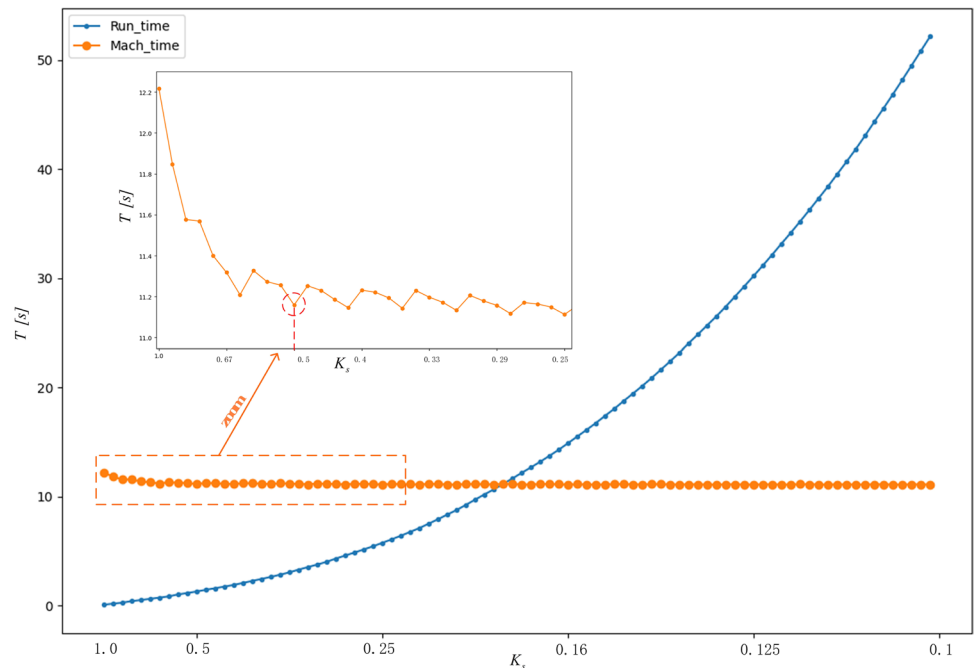
The line-cutting trajectory includes 233 linear segments, of which the shortest path length is 0.021 [mm], the longest path length is 29.078 [mm], the total path length is 221.849 [mm], and the average length is 0.952 [mm]. The given feed rate during machining is 3000 [mm/min], and the set contour error is 10 [μm]. To make a fair and comprehensive comparison, this study uses the following FIR filter interpolation algorithms for the line-cutting trajectory: (1) the P2P mode, where the dwell time is equal to the delay time; (2) the NS-DCB algorithm proposed in Ref. [21], which controls the dwell time according to the contour error; (3) the proposed

PD-VPC algorithm; (4) velocity directly blending mode, whose dwell time is zero and hereinafter referred to as VDB mode. It is worth mentioning that the NS-VCB algorithm proposed in Ref. [21] is only applicable to long segments; thus, it does not participate in the comparative experiment.

For the above three algorithms, the same filter constant must be used. In this study, $T_1 = 20$ [ms] and $T_2 = 10$ [ms] are set. The PD-VPC algorithm needs to discretize the G-code points in advance. Through the simulation of the line-cutting trajectory, the running time T_{run} of the algorithm and the total interpolation time T_{interp} of G-code under different k_s are obtained, as shown in Fig. 11.

It can be seen from the figure that when k_s is close to 1, the running time of the algorithm T_{run} is short. Due to the number of discrete segments of the system is small, the speed of some regions is depressed and the duration time is long, so the total interpolation time of G-code T_{interp} is

Fig. 11 The influence law of discrete coefficient



long. When k_s is close to 0.67, the total interpolation time T_{interp} is significantly shorter, but the running time T_{run} is slightly increased because of too many discrete segments of the system. Later, as k_s exceeds 0.5 and approaches 0.1, the total interpolation time T_{interp} basically remains unchanged, while the running time T_{run} increases exponentially. Thus, the running time of the algorithm and the total interpolation time of G-code are usually appropriate when $k_s \in [0.5, 0.75]$ is usually taken. In this study, the local optimal solution $k_s = 0.5$ is chosen.

The G-code track before and after discretization is shown in the Fig. 12. The blue point is the original G-code point, and the orange point is the discrete G-code point. It is obvious that long segments are regularly broken into short segments, while the micro-segments remain unchanged. After discretization, the number of track segments is 799, and the length of the shortest segment remains unchanged because this segment is not discretized. The longest segment length is 0.503 [mm], and the duration of its rectangular-frame speed can be calculated to be 10.12 [ms], which is approximately equal to the delay time of the second-order filter. As shown in the red dotted line box in the figure, on the one hand, in the area of rapid curvature change, such as corner [1], the track segments are micro-segments and will not be discrete. On the other hand, the micro-segment on a path close to a straight line, such as corner [2], also remains unchanged. Discretization realizes consistent processing of trajectory characteristics, making the processed trajectory meet the input requirements of the MM-VPC algorithm.

Figure 13a shows the distribution of trajectory velocity on the interpolation contour obtained by the PD-VPC algorithm. RGB, three primary colors, are selected to represent the speed. Red color represents high speed, and light colors

represent low speed. There are 13 obviously light colored areas, corresponding to the deceleration part in Fig. 13b. For large corners, that is, close to a straight line, such as corners [2, 4, 12], the speed will drop slightly, and then quickly return to the maximum speed. However, for areas with large curvature changes, such as corners [3, 7, 11], the corresponding deceleration time is longer, and there are multiple speed extreme points. Consequently, the result of velocity planning conforms to the geometric characteristics of the trajectory.

Figure 14 shows the machining contour calculated by interpolation with four algorithms, in which the blue point represents the G-code point of the Benz sample. The 3-axis contouring performance for the interpolation schemes are measured in Fig. 15. As can be seen from the figures, the P2P mode performs point-to-point precise motion. It completely coincides with the G-code point trajectory. Thus, no contour errors are observed. The VDB mode directly mixes the speed without considering the contour error constraint. So out-of-tolerance is evident at the corner. The proposed PD-VPC algorithm complies with the set contour error requirement of 10 [μm] in all processing areas, whereas the NS-DCB algorithm has an out-of-tolerance phenomenon in local areas. The corners [1–6] in the Fig. 14 show that the contour error of the PD-VPC algorithm is smaller than that of the NS-DCB algorithm at various corners. Figure 15 also shows that the contour error of the PD-VPC algorithm is smaller than NS-DCB algorithm in most areas of the whole trajectory. Summarily, the PD-VPC algorithm can effectively and accurately control contour error.

The composite velocity, acceleration, and jerk of the four algorithms are shown in Fig. 16. No dwell time is observed in VDB mode. Thus, the processing time is

Fig. 12 G-code pre-discretization

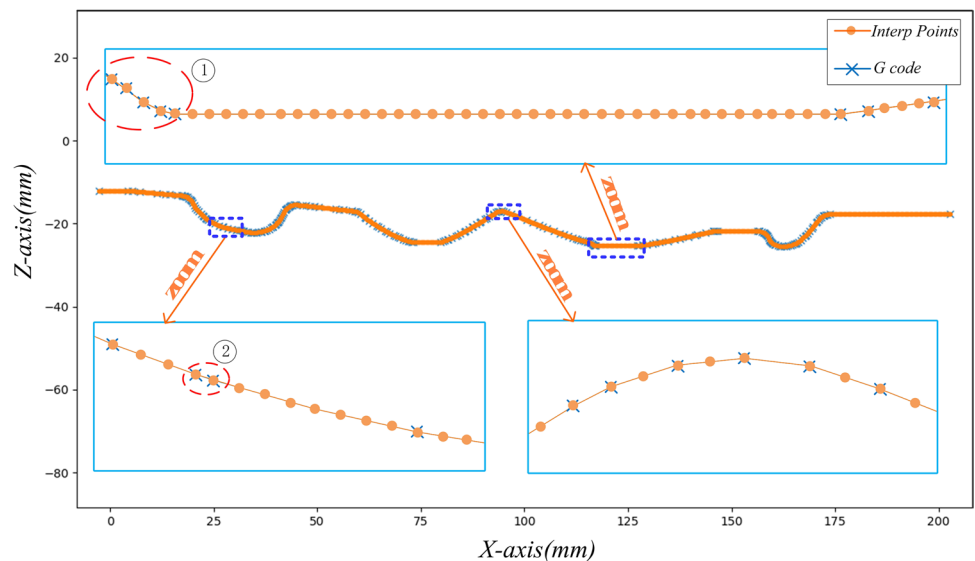
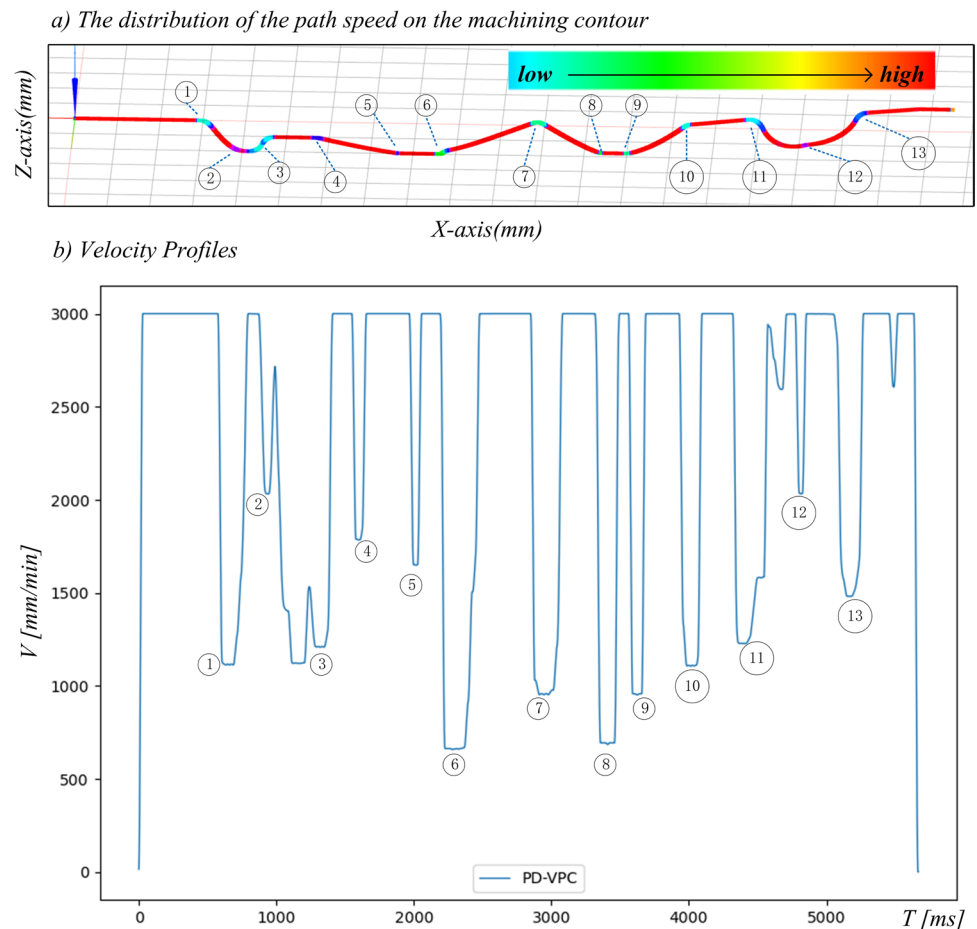


Fig. 13 Velocity distribution on the contour and profiles



the shortest, only 4.47 [s]. In P2P mode, given frequent acceleration and deceleration, the processing time is the longest, which is 11.43 [s]. Although the processing time of the PD-VPC algorithm is slightly longer than that of the NS-DCB algorithm, Fig. 16b and c show that the PD-VPC algorithm effectively reduces the acceleration and jerk amplitude of the system. The performance of acceleration on CNC machine tools is force. Decreasing the acceleration and jerk means that the amplitude and change rate of the force on the system become smaller. Therefore, system vibration will be reduced, and the surface quality will be improved. Given the starting and ending of the profiles correspond to the feed section and withdrawal section of the machining, it is not considered when comparing the acceleration and jerk amplitude.

Figure 17 shows the single-axis kinematics parameters for the X-axis and Z-axis of the system. Figure 17b and c show that the motion trajectory acceleration generated by the PD-VPC algorithm is continuous, the jerk is bounded, and the amplitude of acceleration and jerk is greatly reduced. Thus, the interpolation point trajectory

generated by PD-VPC algorithm is smooth and second-order continuous. The PD-VPC algorithm can effectively reduce the uniaxial vibration of the system. Given the trajectory characteristics of the Benz tool-path, the performance on each axis is also different. The acceleration amplitude of X-axis is reduced by 18%, and the jerk amplitude is reduced by 73%. The acceleration amplitude of Z-axis is reduced by 51%, and the jerk amplitude is reduced by 82%. Conclusively, the acceleration amplitude of each axis is reduced to a certain extent, and the jerk amplitude is greatly reduced.

5 Conclusions

This paper presents a universal algorithm for blending tool-paths machining of CNC system based on FIR filtering. The results show that the algorithm can control the contour error effectively, reduce system vibration, and improve machining quality. The existing algorithms mainly focus on smooth trajectory generation for long

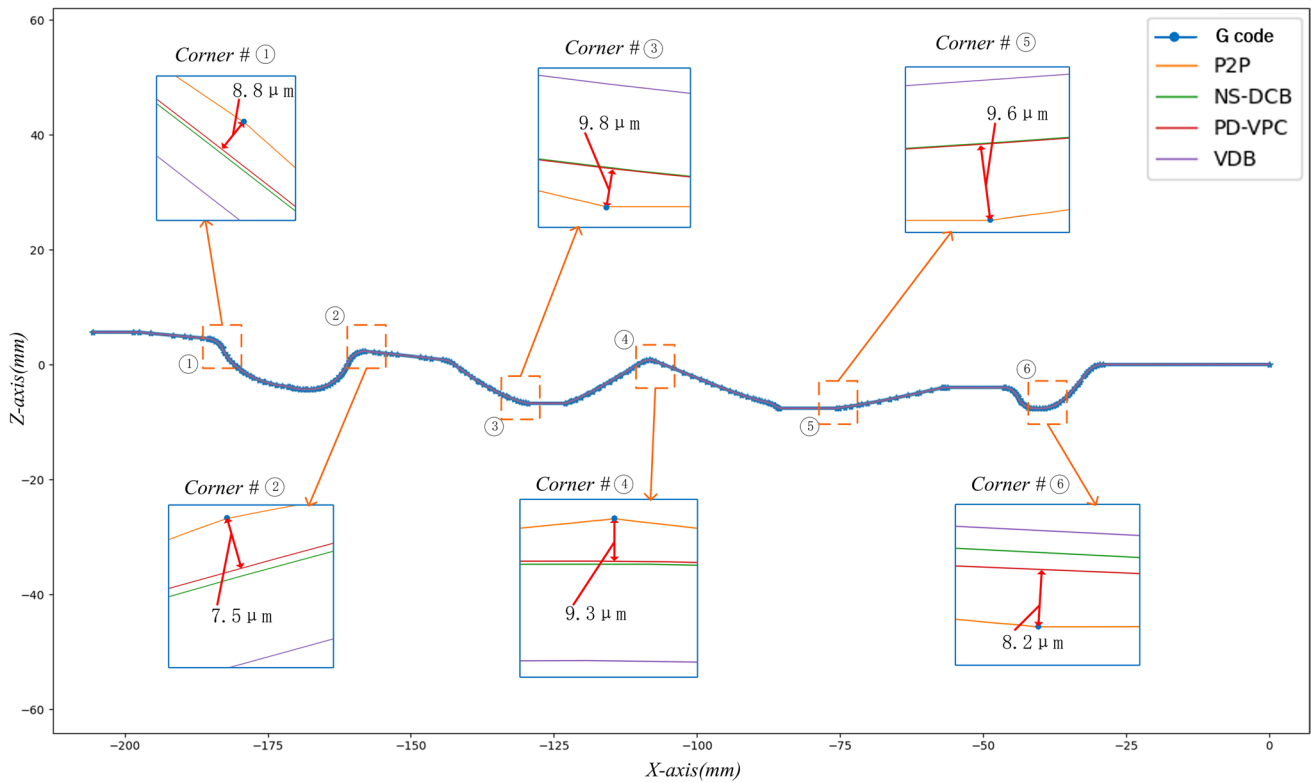
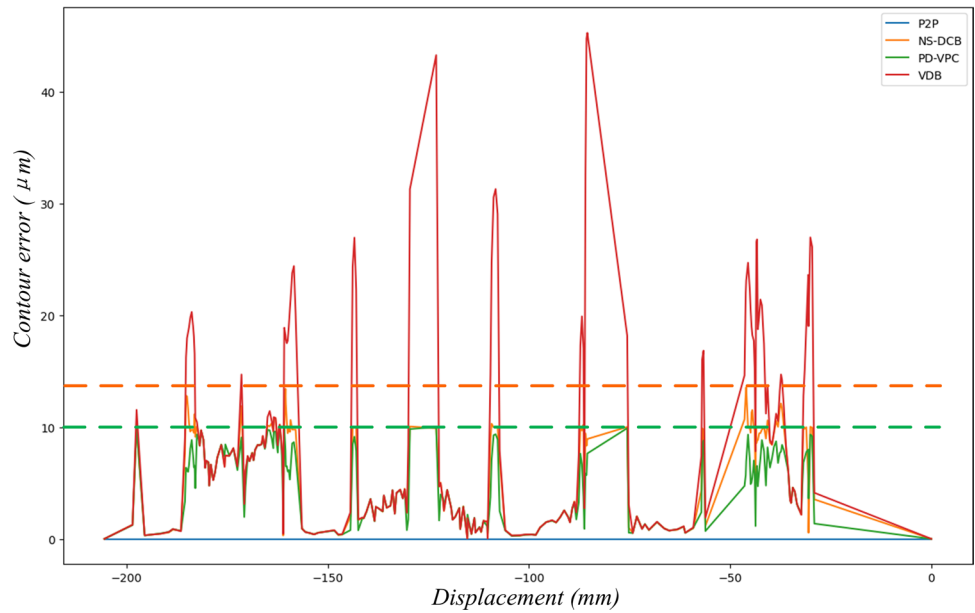


Fig. 14 Interpolation point track of line-cutting

Fig. 15 Experimental contouring errors

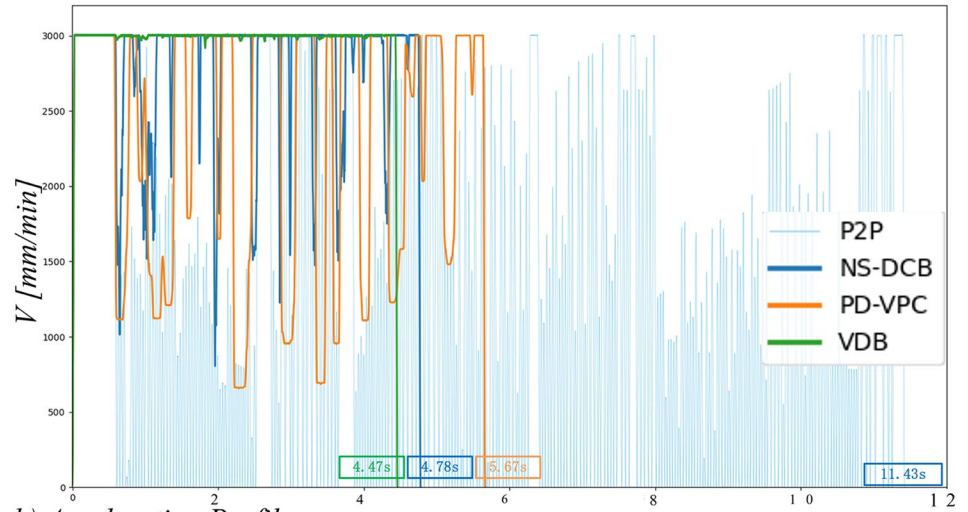


segments. Few studies for continuous micro-segments adopt the trapezoidal shaped velocity scheme when calculating the corner contour error. This study proposes a general algorithm for generating continuous acceleration

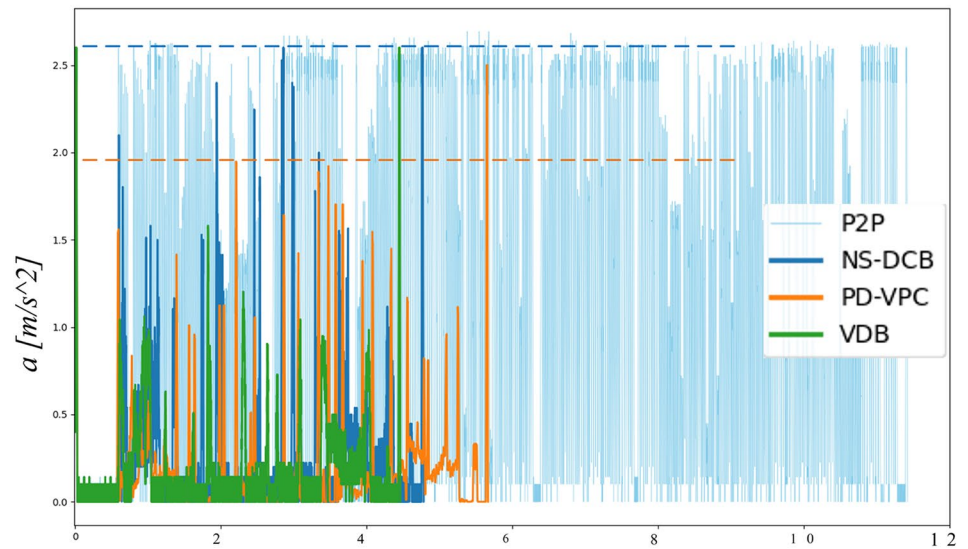
trajectory based on FIR filtering, which can be applied to the blending tool-paths with long segments and micro-segments and effectively control the contour error. Experimental results show that the overall processing

Fig. 16 Interpolated kinematic profiles along line-cutting

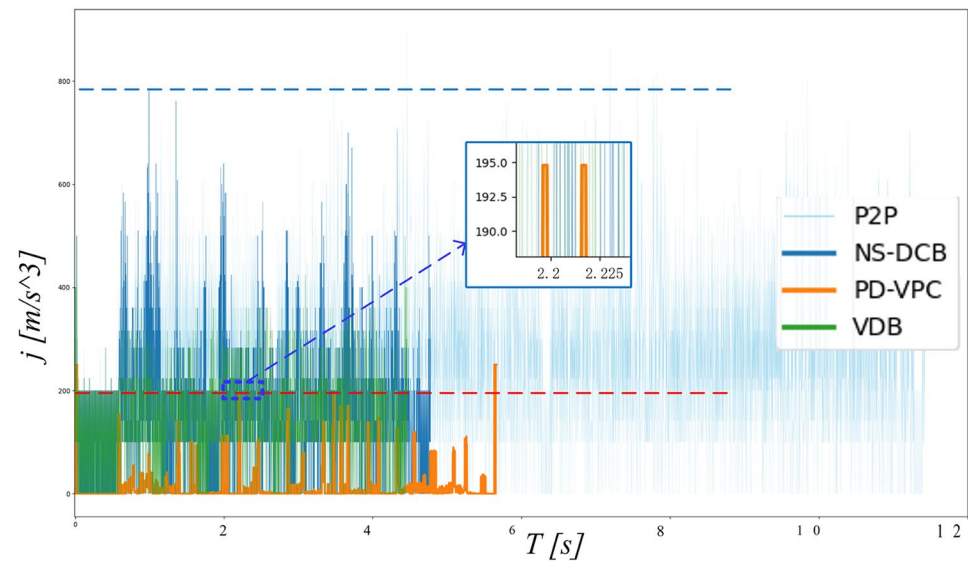
a) Velocity Profiles



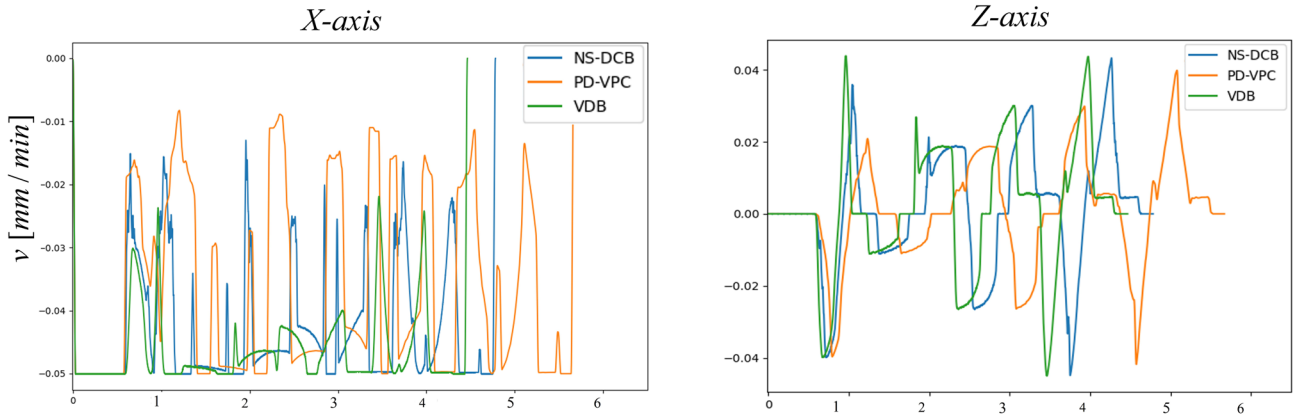
b) Acceleration Profiles



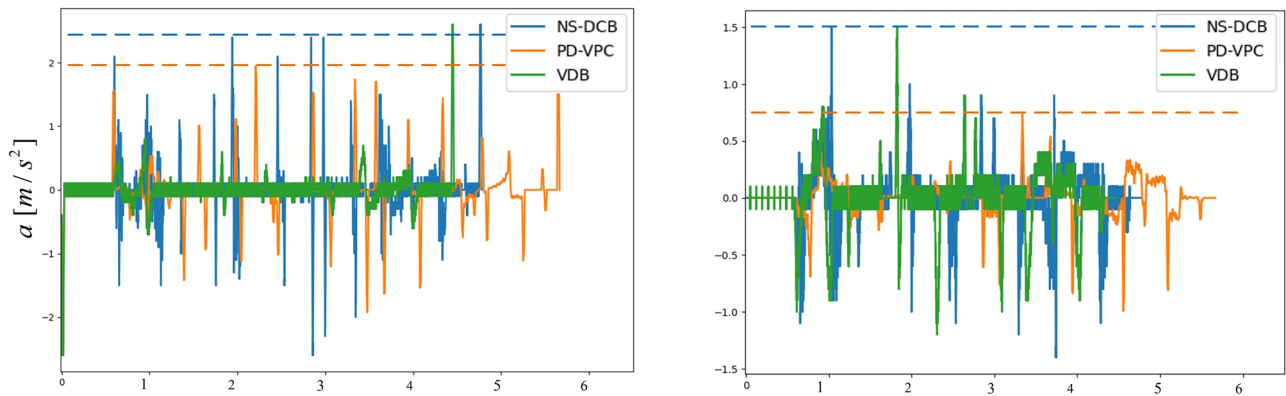
c) Jerk Profiles



a) Velocity Profiles



b) Acceleration Profiles



c) Jerk Profiles

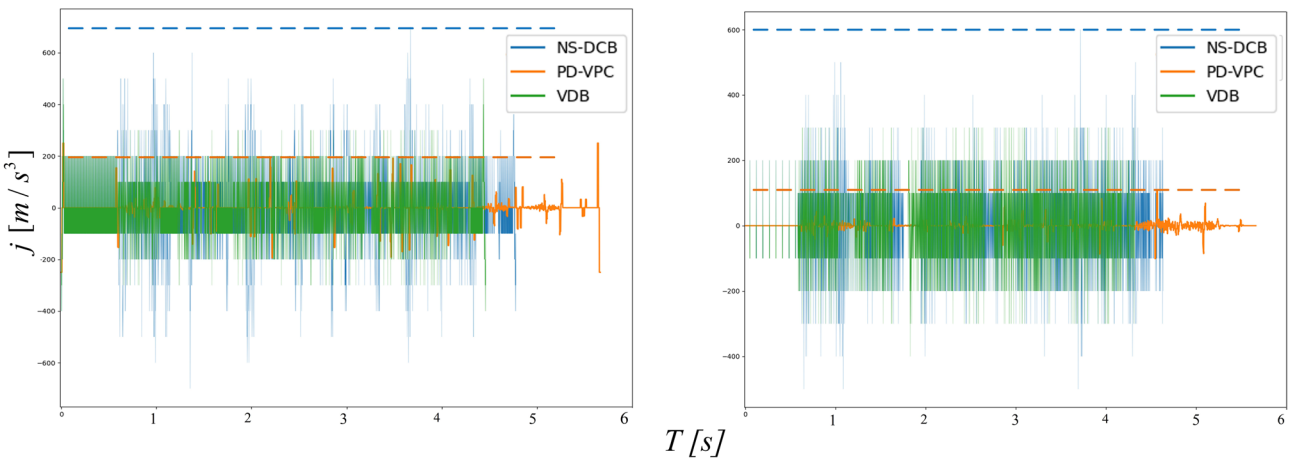


Fig. 17 X-axis and Z-axis interpolated kinematic profiles

time can be reduced up to 50% compared to the P2P mode. On the other hand, the acceleration amplitudes of the single axis can be reduced up to 10–50%, and the jerk amplitude can be reduced up to 70–80%. Thus, the algorithm can effectively suppress the vibration in

the processing process and improve machining surface quality. In the future, the PD-VPC algorithm will also be applied to the research of 5-axis trajectory planning, and the core problem is to solve the model of tool orientation vector contour error.

Author contribution All authors contributed to the study conception and design; material preparation; data collection and analysis; draft of the manuscript and the final manuscript.

Data availability The data sets supporting the results of this article are available from the corresponding author on reasonable request.

Code availability The code used during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

Competing interests The authors declare no competing interests.

References

- Altintas Y, Verl A, Brecher C, Uriarte L, Pritschow G (2011) Machine tool feed drives. *CIRP Ann - Manuf Technol* 60:779–796
- Li BR, Zhang H, Ye PQ, Wang JS (2020) Trajectory smoothing method using reinforcement learning for computer numerical control machine tools. *Robot Comput Integr Manuf* 61:101847
- Sencer B, Altintas Y, Croft E (2008) Feed optimization for five-axis CNC machine tools with drive constraints. *Int J Mach Tool Manufact* 48:733–745
- Beudaert X, Lavernhe S, Tournier C (2013) 5-Axis local corner rounding of linear tool path discontinuities. *Int J Mach Tool Manufact* 73:9–16
- Zhang Y, Ye PQ, Wu JQ, Zhang H (2018) An optimal curvature-smooth transition algorithm with axis jerk limitations along linear segments. *Int J Adv Manuf Technol* 95:875–888
- Zhao H, Zhu LM, Ding H (2013) A real-time look-ahead interpolation methodology with curvature-continuous B-spline transition scheme for CNC machining of short line segments. *Int J Mach Tool Manufact* 65:88–98
- Yau HT, Wang JB (2007) Fast Bezier interpolator with real-time lookahead function for high-accuracy machining. *Int J Mach Tool Manufact* 47:1518–1529
- Park H (2011) B-spline surface fitting based on adaptive knot placement using dominant columns. *Comput-Aided Des* 43:258–264
- Tsai MS, Nien HW, Yau HT (2010) Development of a real-time look-ahead interpolation methodology with spline-fitting technique for high-speed machining. *Int J Adv Manuf Technol* 47:621–638
- Erkorkmaz K, Altintas Y (2001) High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation. *Int J Mach Tool Manufact* 41:1323–1345
- Ha CW, Rew KH, Kim KS (2008) A complete solution to asymmetric S-curve motion profile: Theory & experiments. *Int Conf Control Autom Syst* 2845–2849
- Nam SH, Yang MY (2004) A study on a generalized parametric interpolator with real-time jerk-limited acceleration. *Comput-Aided Des* 36:27–36
- Lee AH, Choi Y (2015) Smooth trajectory planning methods using physical limits. *Proc Inst Mech Eng Part C* 229:2127–2143
- Huang J, Zhu LM (2016) Feedrate scheduling for interpolation of parametric tool path using the sine series representation of jerk profile. *Proc Inst Mech Eng Part B* 231:2359–2371
- Wang YS, Yang DS, Gai RL, Wang SH, Sun SJ (2015) Design of trigonometric velocity scheduling algorithm based on pre-interpolation and look-ahead interpolation. *Int J Mach Tool Manufact* 96:94–105
- Song F, Hao SH, Hao MH, Yang ZM (2008) Research on acceleration and deceleration control algorithm of NC instruction interpretations with high-order smooth. *Intell Robot Appl* 5315
- Sencer B, Ishizaki K, Shamoto E (2015) High speed cornering strategy with confined contour error and vibration suppression for CNC machine tools. *CIRP Ann - Manuf Technol* 64:369–372
- Liu Y, Wan M, Qin XB, Xiao QB, Zhang WH (2020) FIR filter-based continuous interpolation of G01 commands with bounded axial and tangential kinematics in industrial five-axis machine tools. *Int J Mech Sci* 169
- Besset P, Béarée R (2017) FIR filter-based online jerk-constrained trajectory generation. *Control Eng Pract* 66:169–180
- Li DD, Zhang LQ, Yang L, Mao J (2020) Accurate interpolation and contouring control algorithm based on FIR filters for the corner transition. *Int J Adv Manuf Technol* 109:1775–1788
- Tajima J, Sencer B (2019) Accurate real-time interpolation of 5-axis tool-paths with local corner smoothing. *Int J Mach Tool Manufact* 142:1–15
- Hayasaka T, Minoura K, Ishizaki K, Shamoto E, Sencer B (2019) A lightweight interpolation algorithm for short-segmented machining tool paths to realize vibration avoidance, high accuracy, and short machining time. *Precis Eng* 59:1–17
- Tajima S, Sencer B (2020) Real-time trajectory generation for 5-axis machine tools with singularity avoidance. *CIRP Ann - Manuf Technol* 69:349–352
- Tajima S, Sencer B (2020) Online interpolation of 5-axis machining toolpaths with global blending. *Int J Mach Tool Manufact* 175
- Tajima S, Sencer B, Shamoto E (2018) Accurate interpolation of machining tool-paths based on FIR filtering. *Precis Eng* 52:332–344

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

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