The International Journal of Advanced Manufacturing Technology (2018) 97:105–116
https://doi.org/10.1007/s00170-018-1922-0

ORIGINAL ARTICLE

Input shaping-based corner rounding algorithm for machining short line segments

Chang Nho Cho[1] · Young Hun Song[1] · Chang Hyuk Lee[1] · Hong Ju Kim[1]

Received: 6 November 2017 / Accepted: 15 March 2018 / Published online: 27 March 2018
© Springer-Verlag London Ltd., part of Springer Nature 2018

Abstract
The smoothness and continuity of the tool path are crucial in high-quality machining. However, many tool paths are described using only line segments (G01), which inevitably cause discontinuities between blocks. Such discontinuity leads to vibration and feed rate fluctuation, which ultimately leads to a poor surface finish. This study proposes a novel input shaping-based corner rounding algorithm that ensure machining accuracy and vibration suppression. Input shaping is a model-based, robust vibration suppression solution, and it has been widely used in many applications. However, input shaping also distorts the original trajectory, which limits its usage in multiaxis systems. To ensure position accuracy, the corner rounding algorithm proposed in this study includes the position deviation regulation module and the distortion compensation module. The position deviation regulation module limits the position deviation due to corner rounding within the threshold while the distortion compensation scheme compensates for the distortion due to input shaping. The proposed algorithm has been verified via simulations and experiments using a two-degrees of freedom (DOF) Cartesian machine.

Keywords Input shaping · Corner rounding · Vibration reduction · Trajectory generation

1 Introduction

Computerized numerical control (CNC) machines are widely being used in many industrial applications, and extensive studies have been conducted to improve the efficiency and accuracy of CNC machines. Trajectory generation is one of the key elements that determine the performance of CNC machines, and many studies have been devoted on it. Parametric curves can be an attractive solution in terms of machining speed and data management [1, 2]. However, only a few commercial computer-aided design (CAD) or computer-aided manufacturing (CAM) software are capable of generating a tool path using parametric curves. In fact, many CNC tool paths are described only in line segments. With such tool paths, discontinuities between blocks are inevitable, which causes a sudden, excessive change in the velocity at the corner, resulting vibration; vibrations are undesired, and it leads to poor surface finish, inaccurate dimensions, and reduction of the overall machining quality. Thus, there is a strong demand in a trajectory generation algorithm which can cope with tool path discontinuities and vibrations.

Many studies have been devoted to modify the given tool path consists only of short line segments to have round corners to prevent abrupt change in velocity at corners. On the other hand, corner rounding causes position deviation at corners. Thus, an additional scheme is often used to keep the position deviation due to corner rounding within the pre-defined threshold. Many studies focuses on the fitting a spline at the junction of two line segments. The under-corner and over-corner strategies considering the bandwidth of the controllers are proposed [3]. Another algorithm using a curvature-continuous B-spline curve is proposed [4]. This research provides B-spline curve optimization algorithm along with jerk limited S-shape feed rate profile and real-time look-ahead scheme. Also, many studies uses Bezier [5], Pythagorean-hodograph quantic curve [6], quintic and septic [7–9], or two Bezier curves [10, 11] for corner rounding. In addition, several studies aim to include additional constraints, such as contour error [12] or expand the algorithm to a five-axis machine [7, 10, 11, 13]. On the other hand, several studies aim to directly plan jerk limited velocity profiles at corners instead of

✉ Hong Ju Kim
makeit.kim@gmail.com

[1] Korea Electrotechnology Research Institute, 12, Bulmosan-ro, 10beon-gil, Seongsan-gu, Changwon-si 51543, South Korea

relying on geometry-based corner rounding [14]. However, while these algorithms can provide a continuous tool path, they do not fully consider the dynamics of the system in reducing vibration.

Input shaping is a model-based vibration suppression technique that is widely used in many applications. The basic idea is to create two transient oscillations, which would lead to the constructive cancelation of vibrations [15]. Studies have shown that it shows a superior vibration suppression compensation performance over conventional filtering techniques [16]. Input shaping has been used in many applications, including gantry cranes [17], industrial robots with a compliant force sensor [18], or elastic robot arm [19]. Input shaping, however, adds a delay to individual axis; the delay in each axis would result in the distortion of the reference trajectory in Cartesian space. Due to this reason, the use of input shaping has been often limited to a single-axis system. Several algorithms have been proposed to cope with this problem. A contour error compensation algorithm is proposed [20]. It aims to predict and compensate the contour error due to the shaped trajectory. Another study imposes an additional constraint in input shaper generation so that the delay in the ramp response due to input shaping can be minimized [18]. Also, it has been reported that the distortion due to input shaping can be reduced by shaping the velocity profile and then integrating it to obtain the reference position profile [21].

This study aims to apply input shaping to the corner rounding problem. As mentioned above, this leads to two major hurdles: position deviation due to corner rounding and position distortion due to input shaping. Thus, in this study, a novel input shaping-based corner rounding algorithm with the position deviation regulation module and the distortion compensation module is proposed. A zero vibration derivative (ZVD) shaper can be used to create a robust, vibration-reduced tool path which reflects the dynamics of the system. The position deviation regulation module is used to limit the position deviation due to corner rounding within the threshold. On the other hand, to reduce the distortion due to input shaping, input shaping is applied to velocity command. Furthermore, the distortion compensation module compensates for the distortion caused by input shaping while suppressing vibration. These two modules are used with look-ahead algorithm to ensure dimensional accuracy and vibration. The performance of the proposed algorithm is verified through simulations and experiments using a two-degrees of freedom (DOF) Cartesian machine.

2 Input shaping

This section provides a brief deviation of a ZVD shaper. Consider a linear second-order system with input $u(s)$, output $y(s)$, natural frequency ω_0, and damping ratio of D. An input shaper is a sequence of impulse that yields constructive cancelation of vibrations. The response due to an input shaper can be computed by convolution as follows:

$$\begin{aligned} Y_{IS}(t) &= f(t)*y(t) = \sum_{i=0}^{n-1} A_i y(t-t_i) \\ &= A(\omega_0, D)\exp(-\omega_0 D t)\sin(\omega_d t + \varphi) \end{aligned} \tag{1}$$

where A_i is the amplitude of ith impulse and

$$A(\omega_0, D) = \omega_0 \sqrt{\frac{C^2(\omega_0, D) + S^2(\omega_0, D)}{1-D^2}} \tag{2}$$

$$C(\omega_0, D) = \sum_{i=0}^{n-1} A_i \exp(\omega_0, D t_i)\cos(\omega_d, t_i) \tag{3}$$

$$S(\omega_0, D) = \sum_{i=0}^{n-1} A_i \exp(\omega_0 D t_i)\sin(\omega_d t_i) \tag{4}$$

$$\varphi = a\cos(D) \tag{5}$$

It is important to note that Eq. (1) is the response of a system for $t \geq t_{n-1}$. Thus, Eq. (2) represents the magnitude of the residual vibration when $t \geq t_{n-1}$. In order to achieve zero vibration, Eq. (2) must be zero, which can be achieved by the following equations:

$$C(\omega_0, D) = 0 \tag{6}$$

$$S(\omega_0, D) = 0 \tag{7}$$

Additional constraints are also required. First, to achieve a unity static gain, the sum of the impulse amplitude is set to 1.

$$\sum_{i=0}^{n-1} A_i = 1 \tag{8}$$

Also, all amplitudes are constrained to be positive and finite, to avoid actuator saturation [22]. Furthermore, to improve the robustness of the shaper, the derivatives of Eqs. (3) and (4) with respect to ω_0 are set to 0. Equations (1)~(5) can be used to derive a ZVD shaper. For the detailed derivation of the ZVD shaper, refer to [16] or [18].

3 Position deviation estimation

Position accuracy is essential for high-performance CNC machines. Corner rounding aims to smooth sharp corners, and thus, position deviation at corners is inevitable. Figure 1 shows the original and corner rounded tool paths, where area δ corresponds to the position deviation. To ensure accuracy, it is desirable to keep the position deviation due to corner rounding below the user-defined tolerance.

Many studies have been devoted on jerk limited trajectory generation, in order to create a smooth, continuous trajectory [23–25]. However, these trajectories are complex and often consist of multiple phases, which make analytic analysis difficult. In this study, a simple trapezoidal velocity profile is used [26]. Such a profile is known to include infinite jerk, but the resulting vibration can be suppressed using input shaping. Also, its simple structure enables one to analytically analyze the position deviations. Note that the position deviation analysis provided in this section is conducted for each axis, based on the input given to the axis.

The velocity input at the start of a block can be regarded as a sum of step and ramp inputs. The ramp input is due to the trapezoidal velocity profile while the step input is due to the discontinuity at the junction. The step and ramp response of a typical second-order system can be expressed as follows:

$$s(t) = 1 - \frac{\exp(-\omega_0 D t)}{\sqrt{1-D^2}} \sin(\omega_d t + \varphi) \tag{9}$$

$$r(t) = t + \frac{\exp(-\omega_0 D t)}{\omega_d} \sin(\omega_d t + 2\varphi) - \frac{2D}{\omega_0} \tag{10}$$

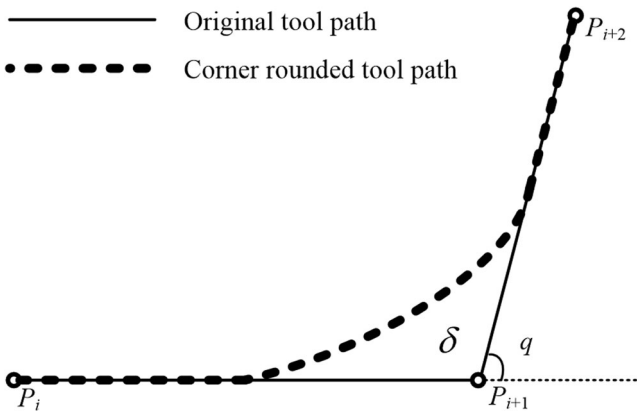

Fig. 1 Junction between two consecutive blocks and position deviation due to corner rounding

Note that unity gain is used for both responses. The responses of the system due to shaped step and ramp inputs can be computed using Eqs. (1), (9), and (10).

$$\begin{aligned} s_{shaped}(t) &= \sum_{i=0}^{n-1} A_i s(t-t_i) \\ &= \sum_{i=0}^{n-1} A_i + \sum_{i=0}^{n-1} A_i \left(\frac{\exp(-\omega_0 D(t-t_i))}{\sqrt{1-D^2}} \sin(\omega_d (t-t_i) + \varphi) \right) \end{aligned} \tag{11}$$

$$\begin{aligned} r_{shaped}(t) &= \sum_{i=0}^{n-1} A_i r(t-t_i) \\ &= \sum_{i=0}^{n-1} A_i \left(t - \frac{2D}{\omega_0} \right) - \sum_{i=0}^{n-1} A_i t_i \\ &\quad + \sum_{i=0}^{n-1} A_i \left(\frac{\exp(-\omega_0 D(t-t_i))}{\omega_d} \sin(\omega_d (t-t_i) + 2\varphi) \right) \end{aligned} \tag{12}$$

Using Eq. (11), the expression for the error due to shaped step input can be expressed as follows:

$$e_{step}(t) = S - S \cdot \sum_{i=0}^{n-1} A_i \left(1 + \frac{\exp(-\omega_0 D(t-t_i))}{\sqrt{1-D^2}} \sin(\omega_d (t-t_i) + \varphi) \right) \tag{13}$$

where S is the magnitude of the step input. Noting Eqs. (6), (7), and (8), the error at $t \geq t_{n-1}$ can be expressed as follows:

$$e_{step}(t) = 0 \tag{14}$$

Similarly, the error due to shaped ramp input can be derived as follows:

$$\begin{aligned} e_{ramp}(t) &= Acc \cdot t - Acc \left(\sum_{i=0}^{n-1} A_i \left(t - \frac{2D}{\omega_0} \right) - \sum_{i=0}^{n-1} A_i t_i \right) \\ &\quad + Acc \sum_{i=0}^{n-1} A_i \left(\frac{\exp(-\omega_0 D(t-t_i))}{\omega_d} \sin(\omega_d (t-t_i) + 2\varphi) \right) \end{aligned} \tag{15}$$

and

$$e_{ramp}(t) = Acc \left(\sum_{i=0}^{n-1} A_i t_i + \frac{2D}{\omega_0} \right) \tag{16}$$

where Acc is the acceleration of the corresponding axis, and Eq. (16) describes the error for $t \geq t_{n-1}$. From Eqs. (13, 14, 15, 16), note that for $t \geq t_{n-1}$, only the shaped ramp input contributes to the error, which is a constant. Thus, it can be concluded that corner rounding is performed for $t < t_{n-1}$. Then, the error

due to corner rounding at the beginning of each block can be approximated as follows:

$$\hat{\delta} = \int_0^{t_{n-1}} \left(e_{ramp}(t) + e_{step}(t)\right) dt \tag{17}$$

where $\hat{\delta}$ is the estimated value of δ, the position deviation. Equation (17) expresses the error induced by corner rounding as the area. It can be computed for each axis, and the error at each axis can be combined to obtain a scalar representation of the error due to corner rounding. Note from Eqs. (13) and (15) that the size of corner rounding depends on the length of the shaper, magnitude of the step input, and acceleration. This implies that the position deviation due to corner rounding can be controlled by adjusting the length of the shaper, acceleration, or junction velocity. However, the length of the shaper and acceleration are fixed as the shaper length is determined by the damping ratio and natural frequency of the system whereas acceleration is chosen considering the capability of the system. Thus, in this study, the magnitude of the step input, which is the velocity at corners, is controlled to regulate the position deviation. The velocity at a corner is determined by the look-ahead algorithm, and the magnitude of step input and acceleration can be computed using the junction velocity and unit vector for corresponding blocks. Thus, by modifying the velocity at the junction between blocks, the size of corner rounding can be controlled; a faster velocity would lead to larger corner rounding.

4 Proposed corner rounding scheme

4.1 Position deviation regulation module

As discussed in previous section, the size of corner rounding depends on the velocity at each junction. Figure 2 shows the junction between two blocks. It is desired to have the velocity at the junction inversely proportional to the angle between blocks, q. Also, if q is 180 degree, the machine should stop at the junction. Thus, in this study, the following dimensionless scalar is used to adjust the junction velocity.

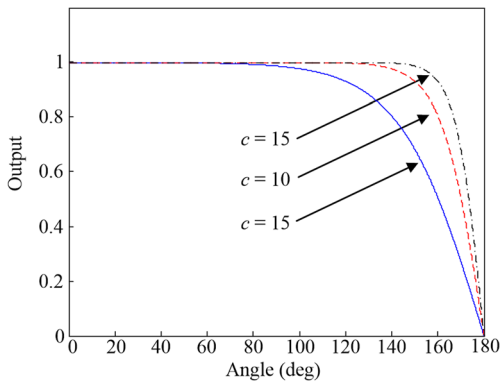

Fig. 2 Proposed junction velocity regulation function

$$\eta(q) = \frac{-1 + \exp(c_1|q| + C_2)}{1 + \exp(c_1|q| + C_2)} \tag{18}$$

where

$$c_1 = \frac{2c}{\pi} \tag{19}$$

$$c_2 = 2c \tag{20}$$

and c is a constant. The proposed function at different c values is shown in Fig. 2.

As can be seen from Fig. 2, the scalar $\eta(q)$ becomes zero as q approaches 180°. The junction velocity is determined by a look-ahead algorithm, which will be discussed later. A junction velocity candidate can be computed as follows:

$$V_{i,exit} = \eta(q) k V_{\boldsymbol{max}} \tag{21}$$

where $V_{i,exit}$ is the exit velocity of block i, V_{max} is the maximum feed rate of the machine, and k is a constant. To keep the deviation due to corner rounding below the tolerance, $\delta_{tolerance}$, the following scheme is proposed:

1. Set k to 1.
2. Perform look-ahead to select junction velocity and compute the position deviation at the corner using Eq. (17).
3. If the deviation is smaller than the tolerance, terminate.
4. If the deviation is greater than the tolerance, decrease k by a certain amount and repeat step 2~4.

The proposed scheme iteratively decreases k and thus the junction velocity until $\delta_{tolerance}$ is satisfied and is also depicted in Fig. 6.

4.2 Distortion compensation module

Input shaping distorts the reference trajectory. For precise machining, it is desirable to compensate the distortion. Also, it must be ensured that the compensation signal does not induce vibration. To directly compensate for the distortion in the position command, the position commands obtained by integrating the velocity commands of each axis are used for the compensation. Note from Eq. (16) that $e_{ramp}(t)$ is the delay induced by input shaping. This can be resolved by adding a zero

padding to the integrated position command and shift them according to the delay expressed in Eq. (16). Then, between the shifted original position profile and the shaped position profile, the distortion due to input shaping can be estimated. Distortion compensation can be done using the estimated distortion.

However, to prevent the compensation signal from causing vibrations, the compensation signal must be low pass filtered. In this study, a fourth-order low-pass Butterworth filter with the cutoff frequency one fifth of the system natural frequency is used to filter the vibrational components from the estimated distortion. Then, the delay due to the filtering is again compensated, and the resulting signal is added to the shaped velocity signal to compensate for the distortion due to shaping.

The proposed distortion compensation algorithm is experimentally verified using a two-DOF Cartesian machine shown in Fig. 3.

The Cartesian machine (ADR-SX441-V1, Alpha Robot, Korea) consists of three joints, but it is controlled as a two-DOF system by only controlling its *X* and *Y* axes. A 200 W AC servo motor (MSMD022G1S, Panasonic, Japan), 400 W AC servo motor (MSMD042G1S, Panasonic, Japan), and a 200 W AC servo motor (MSMD022G1S, Panasonic, Japan) are installed in its *X*, *Y*, and *Z* axes, respectively.

To design input shapers, the natural frequency and damping ratio of each axis are required. Thus, an accelerometer (MP6050, InvenSense, USA) is attached to the axis output and the acceleration data is recorded by an embedded board running real-time Linux (BeagleBone Black, Ti, USA). Each axis is given an aggressive trapezoidal velocity profile with acceleration of 1000 mm/s^2 and fast Fourier transform is performed on the resulting acceleration to estimate the natural frequency for each axis. The damping ratio can be estimated from the encoder data, by observing the decay of the error. It is found that the damping ratio and natural frequency for *X* and *Y* axes are 0.0641, 23.9 Hz, 0.081, 39.74 Hz, respectively. Then, the previous experiment is repeated for three times, with unshaped position profile, shaped position profile, and distortion-compensated profile. The results are presented in Fig. 4 (*X* axis) and 5 (*Y* axis), respectively (Fig. 5).

It should be noted that the original reference trajectory caused vibration at 23.9 and 39.74 Hz for *X* and *Y* axes, respectively. Using the ZVD shaper, the vibration is successfully suppressed, and the distortion compensated command yields a result similar to the ZVD. Thus, it can be concluded that the proposed compensation scheme does not induce vibration. Note that another peak at 7 Hz can be observed; this is caused by the dynamics of the controllers.

4.3 Look-ahead

Look-ahead is an important algorithm in CNC machine control [27, 28]. The overall flow of the proposed algorithm, including look-ahead, the position deviation regulation module, and the distortion compensation module, is depicted in Fig. 6. It consists of a backward pass and forward pass, and it computes three junction velocity candidates V_1, V_2, and V_3. For a trapezoidal velocity profile, V_1 (backward pass) and V_2 (forward pass) can be computed using Eq. (22):

$$V_{entry} = \sqrt{V_{exit}^2 + 2 \cdot Acc \cdot \Delta X} \tag{22}$$

where ΔX, V_{entry}, and V_{exit} are the length of the block, entry velocity, and exit velocity, respectively. On the other hand, V_3

Fig. 3 Two-DOF Cartesian machine used for experiments

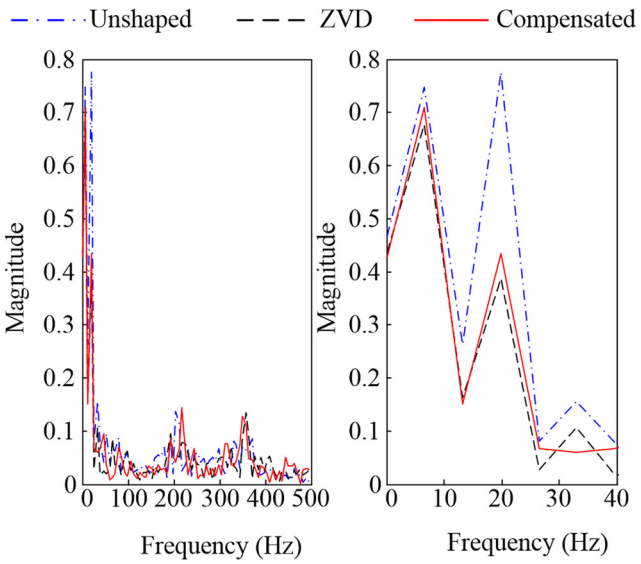

Fig. 4 Fast Fourier transform plot for *X* axis for unshaped, shaped, and distortion compensated command

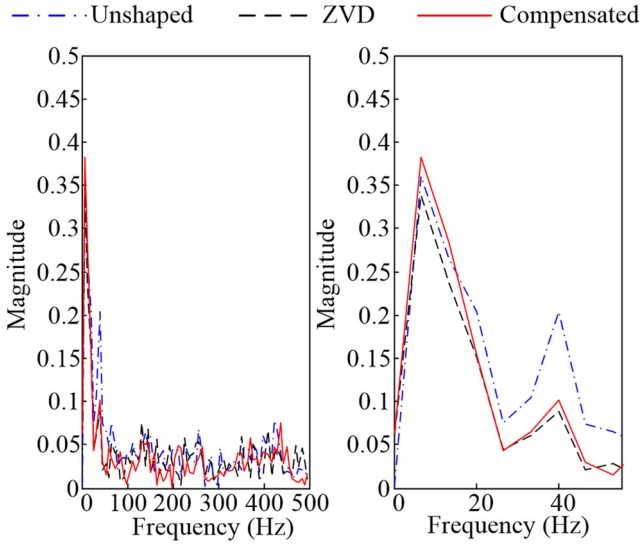

Fig. 5 Fast Fourier transform plot for Y axis for unshaped, shaped, and distortion compensated command

is calculated using Eq. (21). The minimum of three candidates is used to ensure that the generated trajectory is realizable and position deviation is kept under the threshold.

Once look-ahead is complete, the algorithm performs velocity trajectory generation, input shaping, integration, and distortion compensation. This yields the shaped position profile with corner rounding error within the tolerance and distortion compensated, which is then sent to the CNC machine.

5 Simulation

Simulations have been conducted on Matlab (Natick, USA) to illustrate the effectiveness of the algorithm. A five-point star is used as the target trajectory, and the natural frequency and damping ratio of each axis are the same as that of the two-DOF Cartesian machine. As X and Y axes have different damping ratio and natural frequency, the shaper created by convolving the shapers for X and Y axes is used to shape the trajectories. To demonstrate the regulation of the position deviation due to corner rounding, the simulation is repeated twice, with $\delta_{tolerance}$ set to 10 and 3 mm^2.

The overall trajectories are shown in Figs. 7a and 8a, and the close-up view of a corner is given in Figs. 7b and 8b. As can be seen from the results, the corners have been successfully rounded and the position deviation at the corner is regulated by modifying the value of tolerance. The corresponding feed rates are plotted in Fig. 9, where Fig. 9a, b corresponds to $\delta_{tolerance}$ set to 10 and 3 mm^2, respectively.

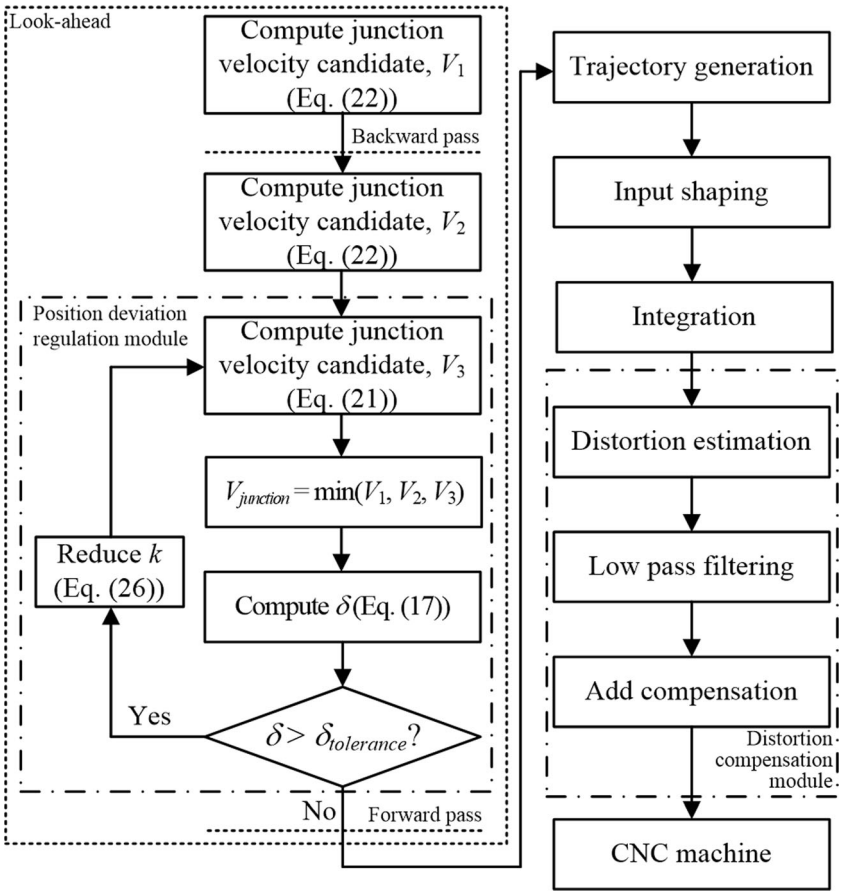

Fig. 6 Proposed corner rounding scheme, including distortion compensation module and position deviation regulation module

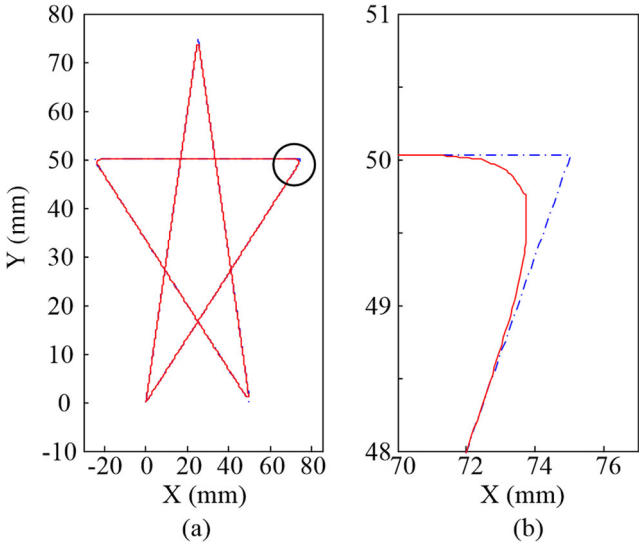

Fig. 7 Simulation results with $\delta_{tolerance} = 10$ mm^2. **a** Overall trajectory. **b** Close-up at corner

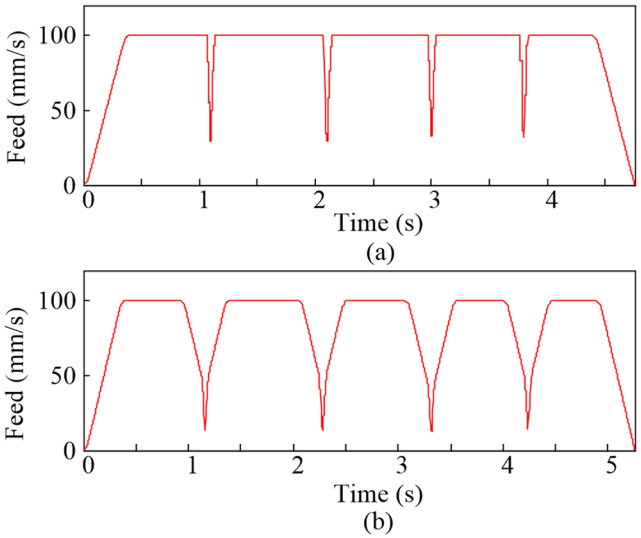

Fig. 9 Resulting feed rate. **a** $\delta_{tolerance} = 10$ mm^2. **b** $\delta_{tolerance} = 3$ mm^2

It should be noted that when a smaller value of $\delta_{tolerance}$ is used, the feed rate at the junctions is reduced to induce a smaller position deviation.

6 Experiments and results

Experiments are performed using the two-DOF Cartesian platform introduced in previous section to further demonstrate the feasibility of the proposed algorithm. The proposed algorithm is implemented in an offline manner, such that the trajectory generation is done before executing the motion. Considering the capability of the Cartesian platform, acceleration and maximum feed are chosen to be 300 mm/s^2 and 100 mm/s, respectively. Two tool paths are considered, each has the shape of Greek letter π and μ. The target trajectory and close-up view at a corner are illustrated in Figs. 10 and 11.

For both cases, the tolerance for the position deviation is set to 10 mm^2, and it can be seen from Figs. 10b and 11b that the corner rounding is successfully achieved at corners for a smooth motion.

Figures 12a and 13a show the composition of line segments, and Figs. 12b and 13b depict the resulting motion of the Cartesian platform, reconstructed from the recorded encoder data. Note that the target tool path consists of short line segments, and the Cartesian platform successfully executed the given tool path using the proposed algorithm.

The planned feed rate when $\delta_{tolerance}$ is set to 10 mm^2 and 3 mm^2 for drawing π are shown in Figs. 14a and 15a. The reduction in the velocity at junctions can be noted when a

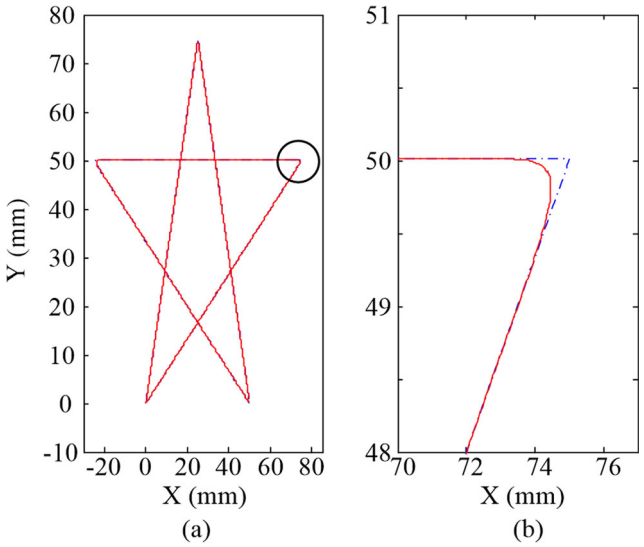

Fig. 8 Simulation results with $\delta_{tolerance} = 3$ mm^2. **a** Overview of trajectory. **b** Close-up at corner

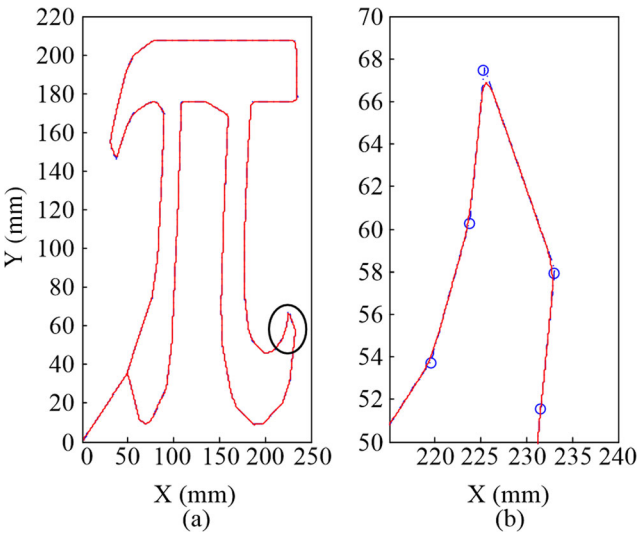

Fig. 10 Tool path for experiment. **a** Overview of trajectory. **b** Close-up at corner

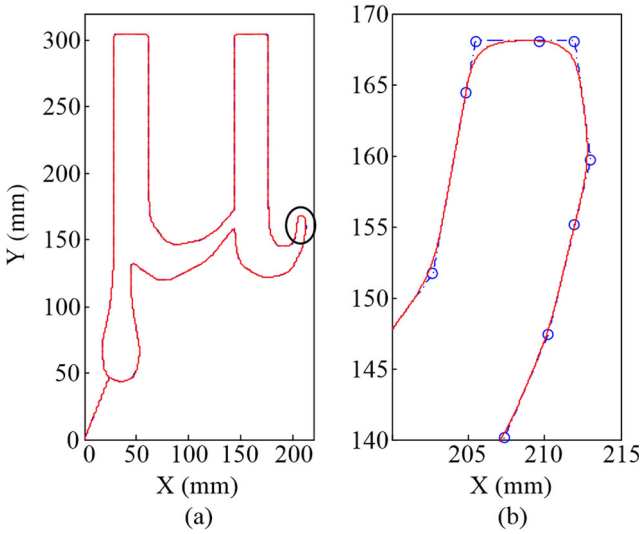

Fig. 11 Tool path for experiment. **a** Overview of trajectory. **b** Close-up at corner

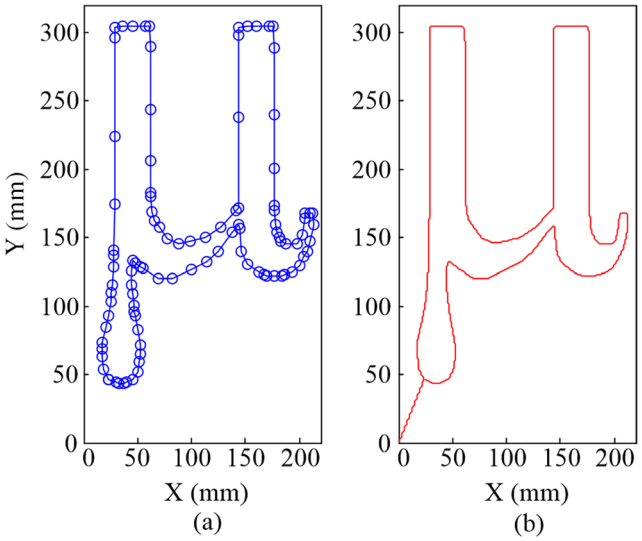

Fig. 13 Target and resulting motion. **a** Composition of line segments. **b** Actual platform motion

smaller value of $\delta_{tolerance}$ is used. The estimated deviations are presented in Figs. 14b and 15b. The results show that the proposed algorithm successfully kept the deviation below the pre-defined threshold. The data for μ are presented in Figs. 16 and 17, which show similar results.

The performance of the distortion compensation module is also evaluated. To analyze the distortion of the tool path due to input shaping, the differences between the shifted, original position command, and shaped position command are computed. Figures 18a and 19a show the difference for drawing π for X and Y axes, respectively. This difference represents the distortion caused by input shaping at each axis. Figures 18b and 19b show the difference between the shifted original position command and shaped position command with compensation. The same results for drawing μ are presented in Figs. 20 and 21. As can be seen from the results, the distortion is greatly reduced by using the proposed distortion compensation module. The difference should not be zero; then the trajectory can no longer guarantee vibration suppression. The shaped trajectory and shaped trajectory with distortion compensation can both suppress the vibration, as can be seen from the results presented in Figs. 4 and 5. However, with the distortion compensation module, vibration suppression can be achieved while minimizing the distortion of the trajectory due to input shaping. Thus, precise trajectory tracking can be realized while suppressing the vibration.

Lastly, to further validate the feasibility of the proposed algorithm, an acetal block has been machined using the Cartesian platform. A spindle is attached at the Z axis of the Cartesian machine, and utilizing the full three-DOF of

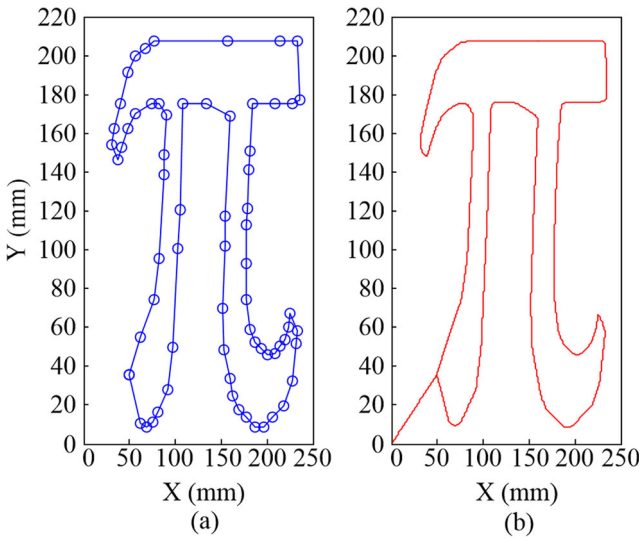

Fig. 12 Target and resulting motion. **a** Composition of line segments. **b** Actual platform motion

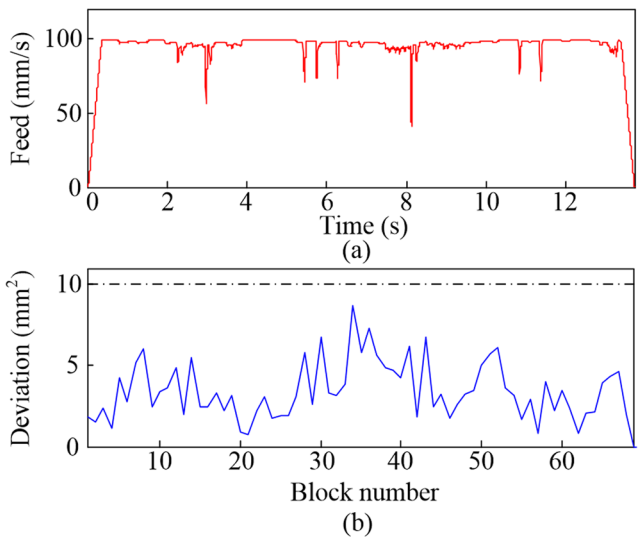

Fig. 14 Scheduled feed rate and position deviation due to corner rounding for π when $\delta_{tolerance} = 10$ mm^2. **a** Feed rate. **b** Position deviation

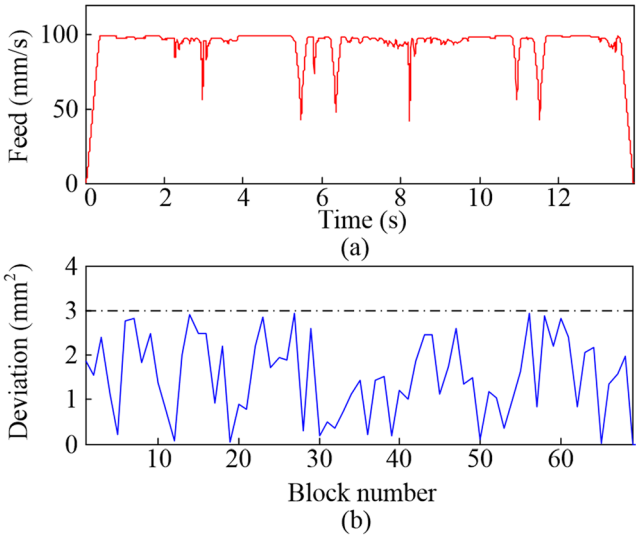

Fig. 15 Scheduled feed rate and position deviation due to corner rounding for π when $\delta_{tolerance} = 3$ mm^2. **a** Feed rate. **b** Position deviation

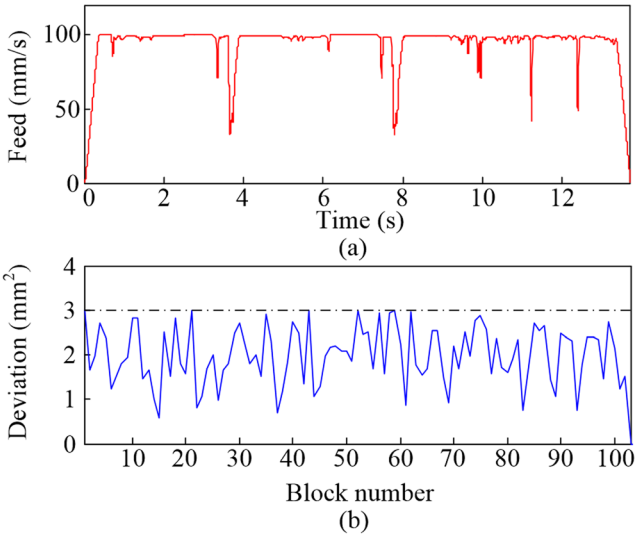

Fig. 17 Scheduled feed rate and position deviation due to corner rounding for μ when $\delta_{tolerance} = 3$ mm^2. **a** Feed rate. **b** Position deviation

the Cartesian machine, the tool path given in Fig. 12a is executed. Figure 22 shows the machined acetal block, and it can be seen that the desired shape is successfully machined using the proposed algorithm. Note that input shaping is not performed in Z axis; Z axis is controlled using a simple position controller.

7 Discussion

Many studies aimed to place a parametric curve between two line segments to obtain a smooth transition. However, while these algorithms are capable of creating a smooth, continuous trajectory, most of these algorithms do not consider the dynamics of the system to be controlled. Thus, these algorithms cannot guarantee the suppression of vibration. Input shaping, on the other hand, is a model-based method. It can be viewed as a more direct way to reduce the vibration of the given reference trajectory. Also, by adding derivative terms, its robustness against model uncertainties can be increased. Thus, input shaping-based corner rounding offers an attractive way of generating a smooth tool path from line segments.

Conventional corner rounding algorithms, which are based on geometric or kinematics smoothing of the trajectory, require complex mathematical models. Thus, while effective, these algorithms have limited flexibility and often ill-suited to be implemented to conventional systems. On the other hand, the proposed corner rounding scheme is highly flexible and can be used for any conventional trajectory generation or

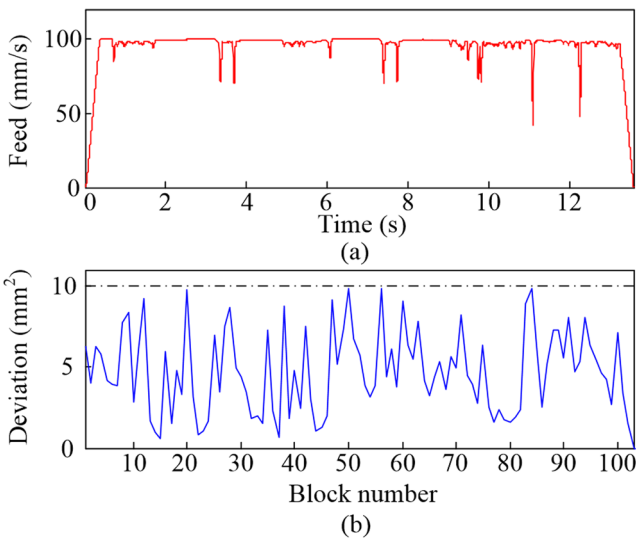

Fig. 16 Scheduled feed rate and position deviation due to corner rounding for μ when $\delta_{tolerance} = 10$ mm^2. **a** Feed rate. **b** Position deviation

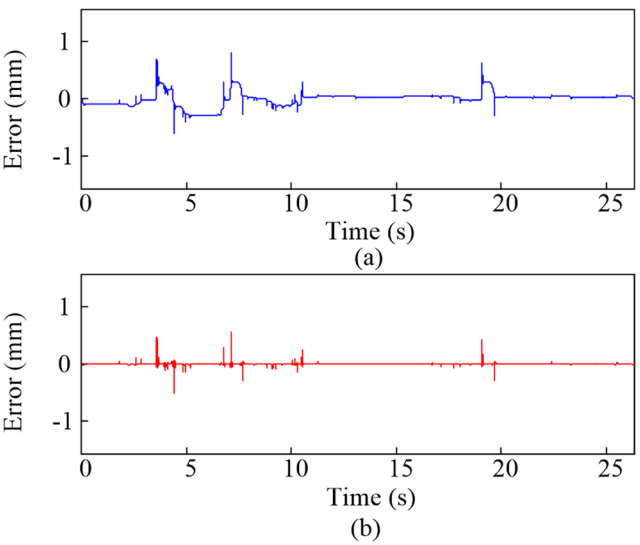

Fig. 18 Distortion at X axis due to input shaping for π. **a** Without compensation. **b** With compensation

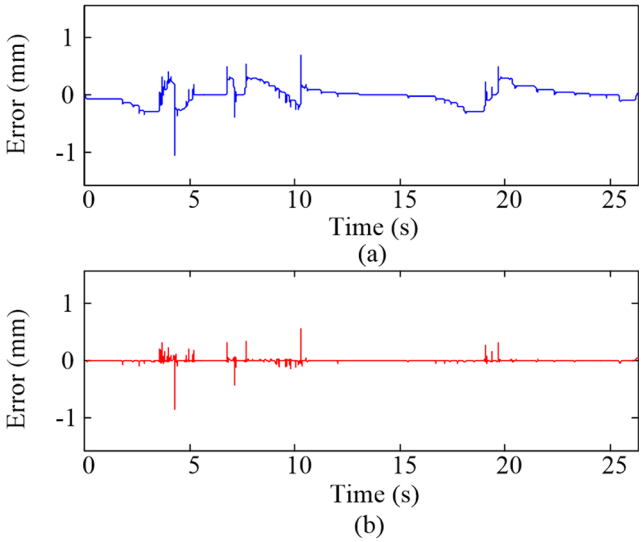

Fig. 19 Distortion at Y axis due to input shaping for π. **a** Without compensation. **b** With compensation

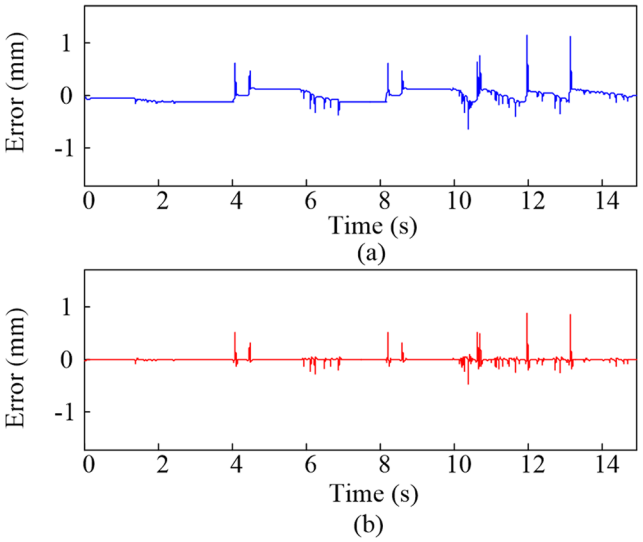

Fig. 21 Distortion at Y axis due to input shaping for μ. **a** Without compensation. **b** With compensation

CNC control algorithms. Input shaping would guarantee vibration suppression of the generated trajectory, and the position deviation regulation module and distortion compensation module ensure precise trajectory tracking.

In this study, the position deviation due to corner rounding is estimated using Eq. (17). However, it should be noted that it is an approximation. The current algorithm simply assumes that the input for every block consists of step and ramp inputs; however, it is not always the case. For example, if the next block is cruise-only, there would be no ramp input. As can be seen from the simulation and experimental results, the approximated value is sufficient to control the position deviation. However, if needed, more accurate position estimation is possible at the increased computational load.

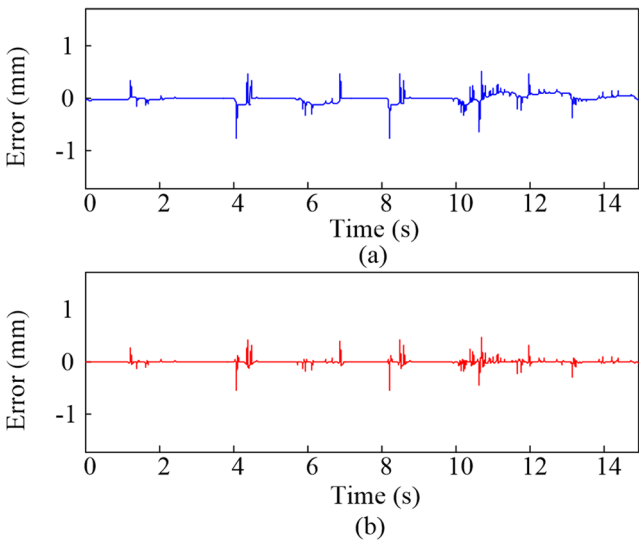

Fig. 20 Distortion at X axis due to input shaping for μ. **a** Without compensation. **b** With compensation

Input shaping may cause errors when the length of the velocity profile is shorter than that of the shaper. This is an unlikely case, as often, the length of the shaper is kept short in order to reduce the distortion. However, in such a case, an additional step can be added in look-ahead routine to prevent errors. The shortest possible length of a block would be $\Delta X/(V_{max} \cdot T)$, where ΔX is the length of the block and T is the sampling period. Thus, by computing the shortest possible length of the block and disable input shaping of the block it if its length is shorter than that of the shaper, such errors can be avoided.

In this study, input shaping is used for the corner rounding problem. However, input shaping requires considerable

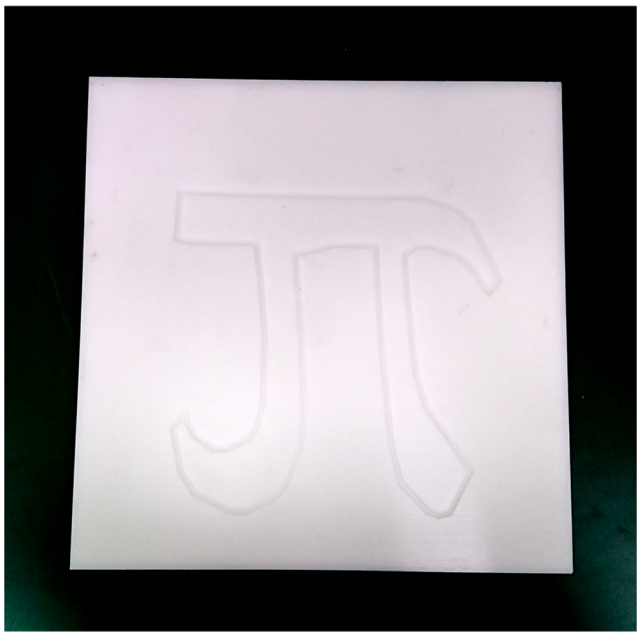

Fig. 22 Machining of Greek letter π on acetal block

computational load and memory. To efficiently convolve a long signal, overlap-add method can be employed [29]. This method divides the signal to be convolved into m segments of length L and convolve them one by one. Convolution tends to lengthen the original signal by the length of the kernel. The overlap-add method overlaps the lengthened part of the first convolved segment and adds it to the first part of the second convolved segment, allowing a piece-by-piece convolution, which is much easier to implement.

In this study, the proposed algorithm has been implemented offline due to the difficulties associated with performing Butterworth filtering in real-time. However, previous studies have shown that a similar control algorithm can be implemented in real-time using a more advanced controller [20]. Thus, it is expected that with a more sophisticated hardware and optimized control algorithm, real-time implementation of the proposed algorithm can be done.

8 Conclusion

Along with increasing interest and demands for high performance machine tools, many studies have been devoted to improve the efficiency and quality of CNC machines. Tool path continuity is one of the important factors to achieve good machining quality and several corner rounding algorithms have been proposed. In this study, a novel input shaping-based corner rounding algorithm is proposed for smoothing a tool path consists of line segments. Additional modules have been included to regulate the position deviation due to corner rounding and distortion caused by input shaping. The results from the simulations and experiments verify the feasibility of the proposed algorithm.

Funding information This research was supported by Korea Electrotechnology Research Institute (KERI) Primary research program through the National Research Council of Science & Technology (NST) funded by the Ministry of Science, ICT and Future Planning (MSIP) (No. 17-12-N0101-22).

References

1. Heng M, Erkorkmaz K (2010) Design of a NURBS interpolator with minimal feed fluctuation and continuous feed modulation capability. Int J Mach Tools Manuf 50(3):281–293
2. Lee A-C, Lin M-T, Pan Y-R, Lin W-Y (2011) The feedrate scheduling of NURBS interpolator for CNC machine tools. Comput Aided Des 43(6):612–628
3. Erkorkmaz K, Yeung C-H, Altintas Y (2006) Virtual CNC system. Part II. High speed contouring application. Int J Mach Tools Manuf 46(10):1124–1138
4. Zhao H, Zhu L, Ding H (2013) A real-time look-ahead interpolation methodology with curvature-continuous B-spline transition. Int J Mach Tools Manuf 65:88–98
5. Walton D, Meek D (2009) G2 blends of linear segments with cubics and Pythagorean-hodograph quintics. Int J Comput Math 86:1498–1511
6. Bi Q, Yuhan W, Zhu L, Ding H (2011) A practical continuous-curvature Bezier transition algorithm for high speed machining of linear tool path. The paper presented at the Proceedings of the 4th International Conference on Intelligent Robotics and Applications, Aachen, Germany, Dec, 6–8
7. Tulsyan S, Altintas Y (2015) Local toolpath smoothing for five-axis machine tools. Int J Mach Tools Manuf 96:15–26
8. Sencer B, Ishizaki K, Shamoto E (2015) A curvature optimal sharp corner smoothing algorithm for high-speed feed motion generation of NC systems along linear tool paths. Int J Adv Manuf Technol 76: 1977–1992
9. Yang J, Yuen A (2017) An analytical local corner smoothing algorithm for five-axis CNC machining. Int J Mach Tools Manuf 123: 22–35
10. Bi Q, Shi J, Wang Y, Zhu L, Ding H (2015) Analytical curvature-continuous dual-Bezier corner transition for five-axis tool path. Int J Mach Tools Manuf 91:96–108
11. Xu F, Sun Y (2018) A circumscribed corner rounding method based on double cubic B-splines for a five-axis linear tool path. Int J Adv Manuf Technol 94:451–462
12. Dong J, Wang T, Li B, Ding Y (2014) Smooth feedrate planning for continuous short line tool path with contour error constraints. Int J Mach Tools Manuf 76:1–12
13. Beudaert X, Lavernhe S, Tournier C (2013) 5-Axis local corner rounding of linear tool path discontinuities. Int J Mach Tools Manuf 73:9–16
14. Tajima S, Sencer B (2016) Kinematic corner smoothing for high speed machine tools. Int J Mach Tools Manuf 108:27–43
15. Singer N, Seering W (1990) Preshaping command inputs to reduce system vibrations. J Dyn Syst Meas Control 112(1):76–82
16. Singer N, Singhose W, Seering W (1999) Comparison of filtering methods for reducing residual vibrations. Eur J Control 5(2–4): 208–218
17. Singhose W, Porter L, Kenison M, Kriikku E (2000) Effects of hoisting on the input shaping control of gantry cranes. Control Eng Pract 8(10):1159–1165
18. Kamel A, Lange F, Hirzinger G (2008) New aspects of input shaping control to damp oscillations of a compliant force sensor. Paper presented at the Proceedings of IEEE International Conference on Robotics and Automation, Pasadena, May, 19–23
19. Malzahn J, Ruderman M, Phung A S, Hoffmann F, Bertram T (2010) Input shaping and strain gauge feedback vibration control of an elastic robotic arm. The paper presented at the Conference on Control and Fault Tolerant Systems, Nice, Oct, 6–8
20. Khoshdarregi MR, Tappe S, Altintas Y (2014) Integrated five-axis trajectory shaping and contour error compensation for high-speed CNC machine tools. IEEE ASME Trans Mechatron 19(6): 1859–1871
21. Palaez G, Palaez G, Perez JM, Vizan A, Bautista E (2005) Input shaping reference commands for trajectory following Cartesian machine. Control Eng Pract 13(8):941–958
22. Singhose WE, Seering WP, Singer N (1997) Time-optimal negative input shapers. J Dyn Syst Meas Control 119(2):198–205
23. Erkormaz K, Altintas Y (2001) High speed CNC system design part I: jerk limited trajectory generation and quantic spline interpolation. Int J Mach Tools Manuf 41(9):1323–1345
24. Zhang K, Yuan C-M, Gao X-S, Li H (2012) A greedy algorithm for feedrate planning of CNC machines along curved tool path with confined jerk. Robot Comput Integr Manuf 28(4):472–483
25. Macfarlane SE, Croft EA (2003) Jerk-bounded manipulator trajectory planning: design for real-time applications. IEEE Trans Robot Autom 19(1):42–52

26. Altintas Y, (2012) Manufacturing automation: metal cutting mechanics, machine tool vibrations and CNC design. Cambridge University Press, New York
27. Han G-C, Kim D-I, Kim H-G, Nam K, Choi B-K, Kim S-K (1999) A high speed machining algorithm for CNC machine tools. The paper presented at the Proceedings of IEEE the 25th Annual Conference of the Industrial Electronics Society, San Jose, CA, USA, Nov, 29-Dec, 3
28. Shiller Z, Lu H-H (1999) Robust computation of path constrained time optimal motions. The paper presented at the Proceedings of IEEE International Conference on Robotics and Automation, Cincinnati, May, 13–18
29. Madisetti V (2009) The digital signal processing handbook. CRS Press, Boca Raton

Publisher's Note

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