

High speed interpolation for micro-line trajectory and adaptive real-time look-ahead scheme in CNC machining

ZHANG LiXian^{1,2*}, SUN RuiYong^{1,2}, GAO XiaoShan¹ & LI HongBo¹

¹Key Laboratory of Mathematics Mechanization, AMSS, Chinese Academy of Sciences, Beijing 100190, China;

²Graduate University of Chinese Academy of Sciences, Beijing 100049, China;

Received June 28, 2010; accepted January 14, 2011

CNC machining plays an important role in mechanical manufacturing. A key issue is to improve the machining feedrate while keeping the machining precision and satisfying the acceleration constraints of the CNC machine. For the consecutive micro-line segments interpolation, the velocities at the junction of two segments are the bottlenecks for the machining efficiency. This paper proposes a multi-period turning method to improve the feedrate at the junctions using the linear acceleration and deceleration mode, which utilizes the maximal acceleration capabilities of the NC machine while satisfying the machining precision. A new and more efficient look-ahead method and a feedrate override method are also proposed to boost the global machining speed. The proposed algorithm has been implemented on Blue Sky NC System, and experimented in real material manufacturing. Compared with several existing algorithms, the current algorithm can improve the manufacturing time ranging from 50% to 180%, depending on the machining parameters, and also results in better machining quality. In addition, the algorithm also satisfies the need of real-time interpolation.

CNC, micro-line segment, multi-period turning, look-ahead scheme, real-time interpolation, feedrate override

Citation: Zhang L X, Sun R Y, Gao X S, et al. High speed interpolation for micro-line trajectory and adaptive real-time look-ahead scheme in CNC machining. *Sci China Tech Sci*, 2011, doi: 10.1007/s11431-011-4329-9

1 Introduction

Sculptured surfaces are generally approximated by a sequence of micro-segments called G01 codes according to the given precision in CNC machining. However, in the micro-lines machining, the direction of machining velocity may change suddenly at the junction/corner between two adjacent micro-lines, which confines the machining velocity, or else serious vibration will occur, and is harmful for both the machine and the product quality. A simple solution is to set turning velocity to zero at corners, which, of course, costs efficiency. The key issue is how to improve the machining efficiency within the precision and acceleration

constraints to achieve high-speed and high-precision CNC machining.

The existing methods of micro-line segment interpolation include: the traditional method of zero turning velocity, the method of “turning internal bisector” [1], that is the values of ex-turning velocity and post-turning velocity are equal and are determined by the maximal acceleration of CNC machine. This method still fails to fully utilize the acceleration capacity of CNC machine. Other methods use fixed Ferguson splines or arc as the turning trace [2, 3]. The machining capacity is not fully used either.

In recent years, spline interpolation [4–7] has become the research hotspot in the field of CNC interpolation. In refs. [6–8], the authors considered the optimal interpolation control method under the CNC machine’s maximal acceleration and jerk constraints. However, for splines of degree three or

*Corresponding author (email: shadowfly12@126.com)

higher, the calculations are too complicated. In order to reduce the computational complexity as well as optimize the processing speed, refs. [4, 9] fitted the G01 micro-line segments into conics or NURBS curves, which reduced the amount of data transmission between CAD and CNC systems as well as the computational complexity, and improved the overall machining speed and precision. Ref. [10] considered the optimal interpolation algorithm for spline path under a given machining error. In general, spline interpolation has the advantage of less data and smooth paths, while the G01 codes have the advantage of computation simplicity. In the meantime, as the industry standard for many years, there exists a large amount of numbers of actual workpieces described by the G01 codes. Hence, research on the interpolation algorithm of the G01 codes is still very important.

Based on the above analysis, the turning velocity is the bottleneck of the overall speed of consecutive micro-line segments interpolation. In this paper, we present a multi-period turning algorithm (see Figure 1), in which the turning acceleration is chosen based on the principle of “full use of the maximal acceleration of CNC machine drivers” (see Section 2 for details). The turning interpolation time at each corner is determined by the machining precision. In addition, in this interpolation algorithm, the accelerations are limited within the feasible ranges of the drivers’ capacities, which effectively reduces the machining vibration and improves the quality of the machined surface. Other advantages of this method are the computational simplicity and explicit expressions which meet the real-time requirement in CNC machining. The look-ahead scheme guarantees the accessibility of the velocities on every line segment and the turning velocities are adjusted accordingly. The algorithm achieves optimization under the above constraints.

In this paper, the linear acceleration and deceleration approach is used for the optimal interpolation. However, there is sudden change of acceleration at a corner. Similar research based on S-shaped acceleration and deceleration with better servo instantaneous characteristic was also given in ref. [11], which will not be discussed in this paper due to the limitation of the length of the paper. On the other hand, the linear acceleration and deceleration approach has the advantage of computational simplicity and high efficiency in the machining with lower requirement of precision, which could be a choice for the balance between feedrate and precision in rough or finish machining. In addition, in CNC machining, we sometimes need to adjust the maximal machining feedrate (feedrate override) to meet the needs of the actual machining requirement, which may affect the results of previous look-ahead scheme. Our look-ahead scheme is adaptive to the feedrate override.

This paper is organized as follows: Section 2 introduces the multi-period turning interpolation algorithm taking 2-axis machining as an example, the results are also ex-

tended into 3-axis and 5-axis NC machining. In Section 3, the look-ahead scheme is presented. Section 4 describes how to update the previous results of the look-ahead scheme when feedrate override occurs. Section 5 introduces real-time interpolation methods. In Section 6, experimental results of machining test of our algorithm are presented which show the effectiveness of our algorithm.

2 Multi-period turning algorithm

In order to improve the turning velocity at a corner, we will use the maximal accelerations of CNC machine. This method is suitable for 2-axis–5-axis CNC machines. In this section we will take the 2-axis machining as an example. In Section 2.2 we will prove Theorems 1 and 2, which show that in the case of positive angle, the results we get are optimal. In the case of negative angle, the turning velocity is optimal under the constraint in Theorem 2. The turning interpolation time is determined according to the requirements of precision. Finally, we briefly show how to extend our algorithm to 3-axis and 5-axis machining.

2.1 Multi-period turning interpolation algorithm

The multi-period turning interpolation is shown in Figure 1, where the solid dots represent the actual interpolation trajectory in the corner. The ex-turning velocity and post-turning velocity are represented by (v_s, v_e) . S and E are the starting and ending points of the turning interpolation. We call the line segments SP and EP the ex-turning line and post-turning line segments. The error caused by deviation from the actual trajectory is ε .

We assume that the turning trajectory is determined by a constant acceleration \mathbf{a} for every corner respectively, whose rationality will be explained in Sections 2.2 and 2.3. According to the relation between the velocity and acceleration, we have

$$v_e \mathbf{e}_e - v_s \mathbf{e}_s = \mathbf{a} \cdot t_m, \quad (1)$$

where \mathbf{e}_s and \mathbf{e}_e represent the unit vectors of the ex-turning

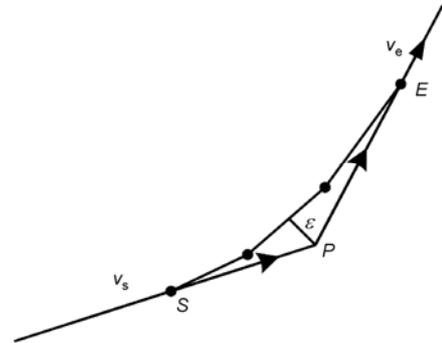


Figure 1 Multi-period turning interpolation at corner.

velocity and post-turning velocity respectively, t_m is the total turning interpolation time. According to eq. (1), the relation between ex-turning velocity and acceleration is

$$v_s = t_m \cdot (\mathbf{a} \times \mathbf{e}_e) / (\mathbf{e}_e \times \mathbf{e}_s), \quad (2)$$

In the same way, we get the post-turning velocity as follows:

$$v_e = t_m \cdot (\mathbf{a} \times \mathbf{e}_s) / (\mathbf{e}_e \times \mathbf{e}_s). \quad (3)$$

Since the first and the last turning interpolation points should be on the adjacent line segments respectively, and the coordinate origin is at the point P , as shown in Figure 1, the interpolation trajectory is a parabola which can be written as the following formula:

$$l_e \mathbf{e}_e = -l_s \mathbf{e}_s + v_s \mathbf{e}_s \cdot t_m + 0.5 \mathbf{a} t_m^2, \quad (4)$$

where $l_s = |SP|$, $l_e = |EP|$ are call the ex-turning distance and post-turning distance respectively. Combined with eqs (3) and (4), we have

$$l_e = (\mathbf{a} \times \mathbf{e}_s) t_m^2 / (2 \mathbf{e}_e \times \mathbf{e}_s) = v_e t_m / 2, \quad (5)$$

$$l_s = v_s t - (\mathbf{a} \times \mathbf{e}_e) t_m^2 / (2 \mathbf{e}_e \times \mathbf{e}_s) = v_s t_m / 2. \quad (6)$$

According to eqs. (2) and (3), v_s , v_e are in proportion to t_m when the acceleration \mathbf{a} is determined. Substituting eqs. (2) and (3) into eqs. (6) and (5) respectively, it is easy to see that l_s , l_e are in proportion to t_m^2 . Hence, the turning interpolation trajectory is determined by a single parameter t_m , which makes the following look-ahead adjustment easy.

2.2 Single-period optimized turning interpolation in two-axis CNC machining

This section takes two-axis CNC machining as an example to illustrate the optimized turning interpolation in the case of single period. Suppose that the absolute values of forward and backward maximal acceleration of the driving axis are equal, and denote A_x and A_y as the two axis' maximal accelerations. The feasible accelerations in machining are in a rectangular area called acceleration rectangle shown in Figure 2. A line connecting the origin and the vertex of the acceleration rectangle is called a semi-diagonal line. The ex-turning and post-turning velocities of a corner both lie in the acceleration rectangular plane. A turning acceleration is called feasible turning acceleration if the ex-turning and post-turning velocities satisfy the condition: $v_s \geq 0$ and $v_e \geq 0$. Proposition 1 gives the range of the feasible turning acceleration.

Proposition 1. The feasible turning accelerations are the intersection area of the acceleration rectangle and the area formed by ex-turning semi-line segment and post-turning semi-line segment with angles less than 180° , which is called feasible acceleration polygon.

Substituting the acceleration in feasible acceleration polygon, which is the shady part shown in Figure 2, into eqs. (2) and (3), the ex-turning and post-turning velocity of corner O are both greater than zero, as shown in Figure 2.

A turning angle at a corner is called positive angle if the signs of the x and y components of ex-turning and post-turning velocities are the same respectively. Otherwise, we call it negative angle. If one of the components of the ex-turning or post-turning velocity is zero, we set it to have the same sign with the other velocity's correspondence component. As shown in Figure 3, 1 represents the direction of ex-turning velocity, and 2, 3, 4 are the directions of post-turning velocities. The signs of the x and y components of 1, 2, 3, 4 are (+, -), (+, -), (+, +), (+, -) respectively. According to the definition, (1, 2) and (1, 4) are positive angles, and (1, 3) is a negative angle.

The best single period turning interpolation is decided by the optimized acceleration, because according to eqs. (2) and (3), when t_m is constant, the values of v_s , v_e are determined by the acceleration \mathbf{a} . In the coordinate shown in Figure 2, \mathbf{a} can be written by

$$\mathbf{a} = \lambda A_x \mathbf{e}_x + \mu A_y \mathbf{e}_y, \quad (7)$$

where \mathbf{e}_x , \mathbf{e}_y are the unit vectors of x and y directions. In the case of Figure 2, the range of the turning acceleration is the shady part, hence $\lambda \in [\lambda_1, 1]$, $\mu \in [\mu_1, 1]$. Substituting eq. (7) into eq. (2), we have

$$v_s = \frac{T \cdot (\lambda A_x \mathbf{e}_x \times \mathbf{e}_e)}{(\mathbf{e}_e \times \mathbf{e}_s)} + \frac{T \cdot (\mu A_y \mathbf{e}_y \times \mathbf{e}_e)}{(\mathbf{e}_e \times \mathbf{e}_s)}, \quad (8)$$

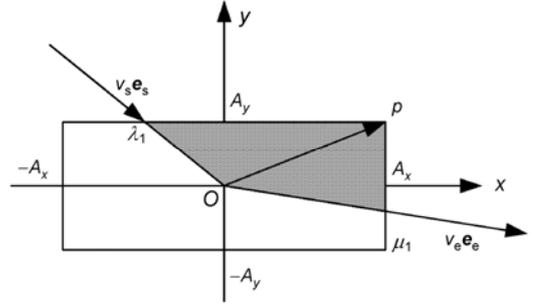


Figure 2 Feasible turning acceleration polygon.

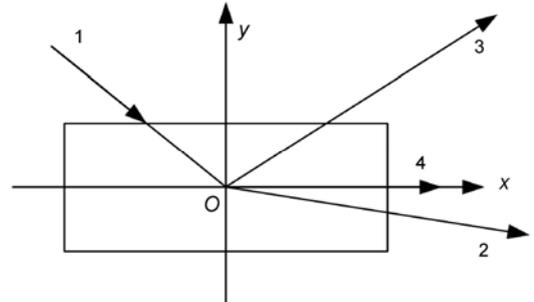


Figure 3 Positive angle and negative angle.

where T is one interpolation period. In eq. (8), the first and second terms are all greater than zero, so when $\lambda = 1$, $\mu = 1$, the value of ex-turning velocity is maximum. In a similar way, substituting eq. (7) into eq. (3), we have

$$v_e = \frac{T \cdot (\lambda A_x \mathbf{e}_x \times \mathbf{e}_s)}{(\mathbf{e}_e \times \mathbf{e}_s)} + \frac{T \cdot (\mu A_y \mathbf{e}_y \times \mathbf{e}_s)}{(\mathbf{e}_e \times \mathbf{e}_s)}. \quad (9)$$

So when $\lambda = 1$, $\mu = 1$, the value of post-turning velocity is maximum.

Theorem 1. For a positive angle when the turning interpolation time is constant, the best turning acceleration is the semi-diagonal line, which lies in the different quadrants with the ex-turning and post-turning lines. Using this acceleration the ex-turning and post-turning velocities are both maximum.

Proof. According to the definition of positive angle and eqs. (8) and (9), v_s , v_e have the same monotonicity with \mathbf{a} , that is, the right corresponding terms of eqs. (8) and (9) have the same sign. Hence, the ex-turning and post-turning velocities can reach the maximal value at the same time by suitable acceleration selection.

The geometric interpretation of positive angle is that the ex-turning line and post-turning line lie in different quadrants. Then there is always a semi-diagonal line in feasible turning acceleration polygon. According to eqs. (2) and (3), v_s , v_e are the linear functions of \mathbf{a} . Obviously, v_s , v_e are maximal when the turning acceleration is at the boundary of acceleration rectangular.

According to eqs. (2) and (3),

$$d_s = \mathbf{a} \times \mathbf{e}_e = |\mathbf{a}| \cdot |\mathbf{e}_e| \cdot \sin \theta_e = |\mathbf{a}| \cdot \sin \theta_e,$$

$$d_e = \mathbf{a} \times \mathbf{e}_s = |\mathbf{a}| \cdot |\mathbf{e}_s| \cdot \sin \theta_s = |\mathbf{a}| \cdot \sin \theta_s,$$

where θ_s represents the angle formed by the acceleration and the ex-turning velocity, θ_e is the angle formed by the acceleration and post-turning velocity, d_s , d_e are the distances from the acceleration vector to the ex-turning and post-turning velocities respectively. When $t_m = T$, v_s , v_e are in proportion to d_s , d_e . According to the geometric relation between the feasible turning acceleration polygon and the direction of ex-turning and post-turning velocities, v_s , v_e are both maximal when the turning acceleration is the semi-diagonal line lying in different quadrants with ex-turning line and post-turning line. We call it optimized turning acceleration. \square

We will compare this optimized turning method with the method of “turning internal bisector”. The ex-turning and post-turning velocities in “turning internal bisector” method is

$$v \cdot (\mathbf{e}_e - \mathbf{e}_s) = \mathbf{a} \cdot t_m. \quad (10)$$

The directions of turning acceleration and the vector $\mathbf{e}_e - \mathbf{e}_s$ are the same. Hence, the acceleration is the intersection point of the internal bisector of the turning angle and the

acceleration rectangle. This method is also called “turning internal bisector method”, as shown in Figure 4. According to Theorem 1, the ex-turning and post-turning velocities are both maximal when the turning acceleration is equal to \mathbf{a}' instead of \mathbf{a} . In conclusion, the ex-turning and post-turning velocities are optimal in the positive angle case.

The comparison about the ex-turning and post-turning velocities obtained by the above two methods is given by an example. The maximal accelerations of x and y axes of CNC machine are 2.9 m/s^2 and 1 m/s^2 . The interpolation period is 1 ms . The geometric relation between the corner and the acceleration rectangle is shown in Figure 5, where the angle formed by ex-turning line and x axis is $-\pi/6$, the angle of corner is $\pi - \theta$.

When the corner angle is positive, that is the angle formed by post-turning line and x axis belongs to $(-\pi/6, 0)$, the comparison of the two methods is shown in Figure 6,

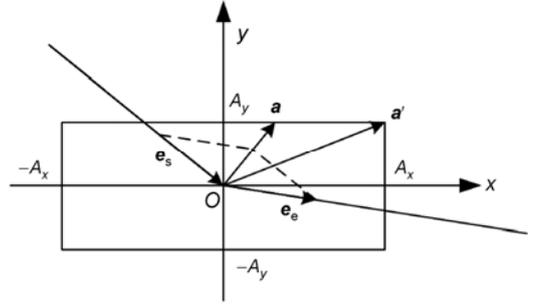


Figure 4 Acceleration of “turning internal bisector” method and optimized turning method.

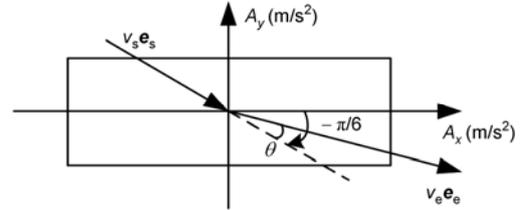


Figure 5 Geometric relation between corner and acceleration rectangle.

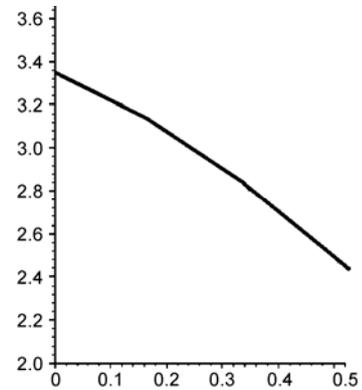


Figure 6 The diagram between $(v_s+v_e)/2v$ and θ for positive angles.

where the horizontal coordinate axis represents the value of θ , the vertical coordinate axis is the value of $(v_s+v_e)/2v$. From this diagram, the velocity of the optimal turning interpolation is 2.4–3.3 times of that of the “internal bisector method”.

According to the definition of negative angle and comparing each term on the right side of eqs. (8) and (9), the ex-turning and post-turning velocities cannot be maximal simultaneously when the acceleration is chosen in the feasible turning acceleration polygon. So, the turning acceleration is calculated according to the optimal function $f=v_s+v_e$, and the ex-turning and post-turning velocities are determined by eqs. (2) and (3) respectively.

The unit vector of ex-turning and post-turning velocities can be written as follows:

$$\mathbf{e}_s = \alpha_s \mathbf{e}_x + \beta_s \mathbf{e}_y, \quad (11)$$

$$\mathbf{e}_e = \alpha_e \mathbf{e}_x + \beta_e \mathbf{e}_y, \quad (12)$$

where (α_s, β_s) , (α_e, β_e) represent the x and y components of \mathbf{e}_s , \mathbf{e}_e . Substituting eqs. (11) and (12) into eqs. (9) and (8), the optimal function is simplified as follows:

$$f = (\mu(\alpha_e + \alpha_s)A_y - \lambda(\beta_e + \beta_s)A_x)T, \quad (13)$$

where $\lambda \in [\lambda_1, \lambda_2]$, $\mu \in [\mu_1, \mu_2]$, $|\lambda_i| \leq 1$, $|\mu_i| \leq 1$ ($i=1,2$). Eq. (13) is a linear programming problem, the maximal value is got when the turning acceleration is at the vertexes of feasible acceleration polygon. The optimal turning acceleration is received by substituting λ , μ , which are the solutions of optimization of eq. (13), into eq. (7).

The ex-turning and post-turning velocities of positive angle are both maximum, so the sum of them also satisfies the optimal function $f=v_s+v_e$. Hence,

Theorem 2. The turning velocity calculated by the method proposed in this section is optimal in the optimization function $f=v_s+v_e$.

2.3 Multi-period turning interpolation

The constant acceleration of single period turning interpolation is determined in the above section, by which the velocity of the positive angle is optimal and the velocity of the negative angle is optimal with respect to optimization function $f=v_s+v_e$. According to eqs. (2) and (3), v_s , v_e are in proportion to t_m when a is a constant. So the conclusion received in the above section is also suitable for multi-period turning interpolation. The turning velocity will be further increased by multi-period turning interpolation, with the turning acceleration determined by the method proposed in the above section. The turning interpolation time is determined by the maximal precision, maximal feedrate and maximal turning distance.

The shape precision can be guaranteed in the multi-period turning interpolation method due to the following

property:

Proposition 2. The interpolation trajectory of the multi-period turning interpolation at a corner is a parabola inside the triangle SPE , where S and E are the starting and ending points of the turning procedure shown in Figure 7.

Proof. The turning acceleration in the multi-period interpolation is determined by the method proposed in Section 2.2. The interpolation trajectory at the corner is

$$s(t) = -l_s \mathbf{e}_s + v_s \mathbf{e}_s \cdot t + 0.5at^2. \quad (14)$$

From eq. (14), we can see that the trajectory $s(t)$ is a quadratic function with respect to the variable t , which is a parabola, as shown in Figure 7. The parabola, whose starting and ending tangent lines are vectors \overline{SP} , \overline{PE} , can be represented by the Bezier curve:

$$s(u) = \mathbf{S} \cdot (1-u)^2 + \mathbf{P} \cdot 2u(1-u) + \mathbf{E} \cdot u^2 \quad (u = t/t_m). \quad (15)$$

Proposition 2 is proved by the convex property of Bezier curves. \square

We need to calculate the error caused by the multi-period turning interpolation and guarantee the error within the allowable range. The error direction is defined as the weighted slide direction \overline{PN} of the rational Bezier curve [12, p188], which is the direction of the line starting from the point on the parabola at $u = 0.5$ to the corner point, shown in Figure 7. For a parabola, the length of the line \overline{PM} decides the extent of the curve approximate to the control polygon. That is:

$$\varepsilon = |\overline{PM}|.$$

Obviously, the above error is greater than or equal to the practical error, so this error guarantees that the actual machining error is strictly confined in the required range. In the meantime, according to eq. (17), if $|PM|$ is taken as machining error, the error is in proportion to the square of turning interpolation time, which effectively improves the calculation velocity.

The maximal turning interpolation time is determined by the allowable maximal machining error. Substituting $u = 0.5$ into eq. (15), we have $\mathbf{M} = (\mathbf{S} + 2\mathbf{P} + \mathbf{E})/4$. Substituting it

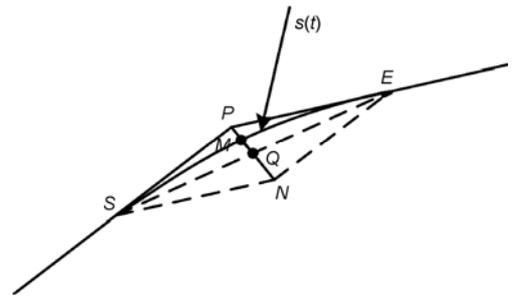


Figure 7 The error of multi-period turning interpolation.

into the above equation, we get the error expression:

$$\varepsilon = \left| \overline{PE} - \overline{SP} \right| / 4 = |l_e \mathbf{e}_e - l_s \mathbf{e}_s| / 4, \quad (16)$$

which means that the error caused by multi-period turning interpolation equals a quarter of the module of the difference of the ex-turning displacement vector and post-turning displacement vector. As shown in Figure 7, we have $|PM| = |PN| / 4$.

Substituting eqs. (5) and (6) into eq. (16) and combining with eq. (1), we thus have

$$\varepsilon = |\mathbf{a}| \cdot t_m^2 / 8. \quad (17)$$

Proposition 3. The maximal interpolation time within the error bound is determined by

$$t_m = \sqrt{8\varepsilon / |\mathbf{a}|}. \quad (18)$$

The ex-turning distance l_s and post-turning distance l_e should be within the half of the micro-line segments. The ex-turning distance or post-turning distance is calculated by substituting eq. (18) into eqs. (5) and (6) respectively. In order to guarantee that the actual line segment is greater than zero as shown in Figure 8, we have

$$\begin{cases} l_s \leq 0.5d_{i-1}, \\ l_e \leq 0.5d_i. \end{cases}$$

If the above equation is not satisfied, the turning interpolation time can be adjusted according to the above equation, eqs. (5) and (6), we have

$$t'_m = t_m \cdot \min(\sqrt{0.5d_{i-1}/l_s}, \sqrt{0.5d_i/l_e}). \quad (19)$$

In actual machining, the turning velocity should be within the given maximal feedrate v_m . We have the following proposition.

Proposition 4. The maximal velocity of the whole multi-period turning interpolation is either the ex-turning or post-turning velocity.

Proof. The velocity of the i th interpolation period is $v_i = |v_s \mathbf{e}_s + \mathbf{a}iT|$ in the multi-period turning interpolation, as shown in Figure 9. According to the triangle geometric relation, when the angle formed by ex-turning line segment and the acceleration direction satisfies $\phi \geq \pi/2$, the turning interpolation velocity monotonously increases with the interpolation time, so the post-turning velocity is the maximal

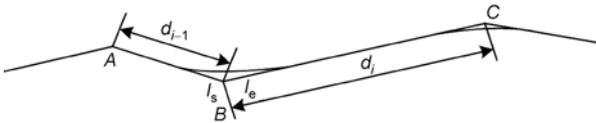


Figure 8 The ex-turning distance and post-turning distance are within half of the line segments

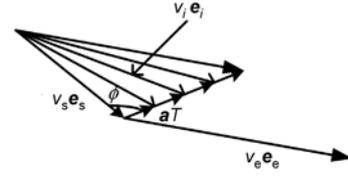


Figure 9 The turning velocity variation in multi-period turning interpolation ($\phi < \pi/2$).

value.

When the angle formed by ex-turning line segment and turning acceleration direction satisfies $\phi < \pi/2$, we consider two cases. For the negative angle, the interpolation velocity monotonously decreases first and increases subsequently with interpolation time; for positive angle, the interpolation velocity is either the same with the case of negative angle or monotonously decreases with interpolation time as shown in Figure 10. In conclusion, the maximal velocity is the ex-turning velocity or post-turning velocity in the whole turning interpolation \square

According to Proposition 4, if the value of ex-turning and post-turning velocities is not larger than the maximal machining feedrate v_m , the whole turning interpolation velocity is within the constrained condition of v_m . Otherwise, the turning interpolation time should be adjusted as follows:

$$t''_m = t'_m \cdot v_m / \max(v_s, v_e). \quad (20)$$

As stated above, the final turning interpolation time is finally decided by maximal error ε , the length of interpolated line, and the maximal machining feedrate v_m together, and is calculated by eqs. (18), (19), and (20) respectively.

2.4 Multi-period turning interpolation in 3-axis and 5-axis CNC machining

The algorithm of multi-period turning interpolation in 3-axis and 5-axis CNC machining is similar to 2-axis machining. The turning interpolation parameters are calculated by eqs. (2), (3), (5), and (6). In the case of 3-axis machining, the range of feasible turning acceleration is the intersection of the plane where the corner lies in and the cube with the three maximal accelerations of the driving axis as edges, which is the hexagon or quadrilateral shown in Figure 11.

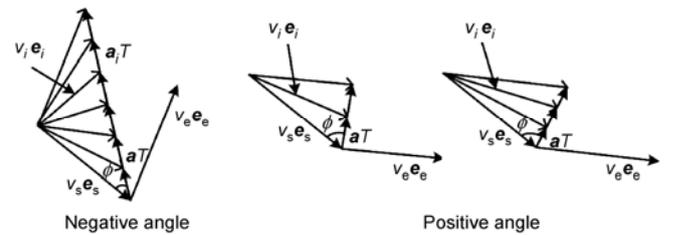


Figure 10 The turning velocity variation in multi-period turning interpolation ($\phi < \pi/2$).

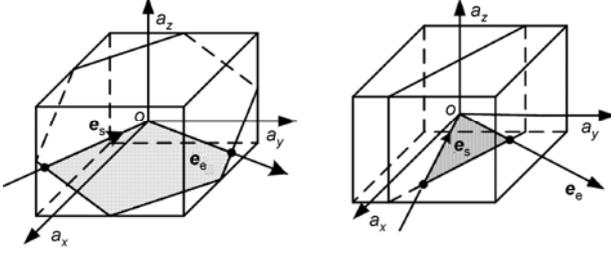


Figure 11 The range of turning acceleration of 3-axis machining.

When the ex-turning and post-turning velocity is above zero, the range of turning acceleration is in the shade area.

Similarly, the turning acceleration of 4-axis and 5-axis machines is the intersection of plane where the corner lies in and the 4D and 5D region formed by the maximal accelerations of 4 or 5 driving axis, which is a polygon area.

The optimal turning acceleration is determined by the optimization function of $f = v_s + v_e$ in 3-axis, 4-axis, or 5-axis machining. By decomposing the vectors e_s , e_e into two coordinate axes, the 3-axis, 4-axis and 5-axis machining is also a linear programming problem similar to the algorithm of the 2-axis machining. The maximal value of optimization function is obtained when the turning acceleration is at the vertex of polygon. In actual calculation, substituting all the vertexes of polygon into the optimization function, the optimal acceleration is the vertex corresponding to the maximal value of optimization function.

In the 4-axis and 5-axis CNC machining, the theoretical error is not equal to the actual error because the rotation axes are not equivalent to the translating axes. There is a nonlinear map between the two errors. We connect the machine coordinate system with workpiece coordinate system by coordinate transformation. The approximated map between the two coordinate systems is obtained by the first order of Taylor expansion of the derivative of the coordinate transformation. The specific algorithm is explained by two-rotary-table CNC machine, shown in Figure 12.

Let (x, y, z) and (X, Y, Z) be the translational coordinates

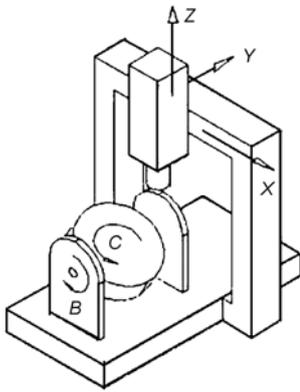


Figure 12 Machine and workpiece coordinate system (two-rotary-table machine).

in workpiece coordinate and in machine coordinate. The transformation between the two coordinates is as follows:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} \cos B \cos C & \sin C & -\sin B \cos C \\ -\cos B \sin C & \cos C & \sin B \sin C \\ \sin B & 0 & \cos B \end{bmatrix} \begin{pmatrix} X + x_0 \\ Y + y_0 \\ Z + z_0 \end{pmatrix}$$

and B, C are the rotation angles of rotary tables, $\mathbf{p} = (x_0, y_0, z_0)$ is the initial coordinates of the origin point of machine coordinate system in workpiece coordinate system.

The differential of the above equation about variables of X, Y, Z, B, C is

$$d\boldsymbol{\mu} = A d\boldsymbol{\eta},$$

where $d\boldsymbol{\mu} = (dx \ dy \ dz)^T$, $d\boldsymbol{\eta} = (dX \ dY \ dZ \ dB \ dC)^T$,

$$A = \begin{bmatrix} cB cC & sC & -sB cC & -z cC & y \\ -cB sC & cC & sB sC & z sC & -x \\ sB & 0 & cB & (x_0 + X)cB - (z + Z_0)sB & 0 \end{bmatrix},$$

where cB, sB, cC, sC represent $\cos B, \sin B, \cos C, \sin C$ respectively.

According to the above equation, the error in the workpiece coordinate system $\varepsilon^2 = (d\boldsymbol{\mu})^T (d\boldsymbol{\mu})$ and the error in the machine coordinate system $E^2 = (d\boldsymbol{\eta})^T (d\boldsymbol{\eta})$ have the following approximate relation:

$$\varepsilon^2 = (d\boldsymbol{\mu})^T (d\boldsymbol{\mu}) = (d\boldsymbol{\eta})^T A^T A (d\boldsymbol{\eta}).$$

The error produced by the rotation axis can be controlled by the above equation.

2.5 The flowchart of multi-period turning parameter calculation

The main contents of Section 2 are (1) calculation of turning acceleration which optimizes the $f = v_s + v_e$ within all feasible accelerations; (2) calculation of the turning interpolation time according to the allowable machining error; (3) adjustment of the turning interpolation time t_m rendering the ex-turning distance and post-turning distance smaller than a half of the line segments respectively; (4) adjustment of t_m according to the requirement that the turning interpolation velocity is not larger than the maximal machining federate; (5) calculation of the ex-turning and post-turning velocities, ex-turning distance and post-turning distance according to turning acceleration and turning interpolation time. The main flowchart is shown in Figure 13.

The multi-period turning interpolation method satisfies the maximal feedrate, precision, and acceleration constraints. The turning velocity is improved greatly by the multi-period transition with the acceleration of every driving axis fully used. The turning interpolation parameters are adjusted by the turning interpolation time uniformly. The calculation is simple and also suitable for 3-axis-5-axis CNC machines.

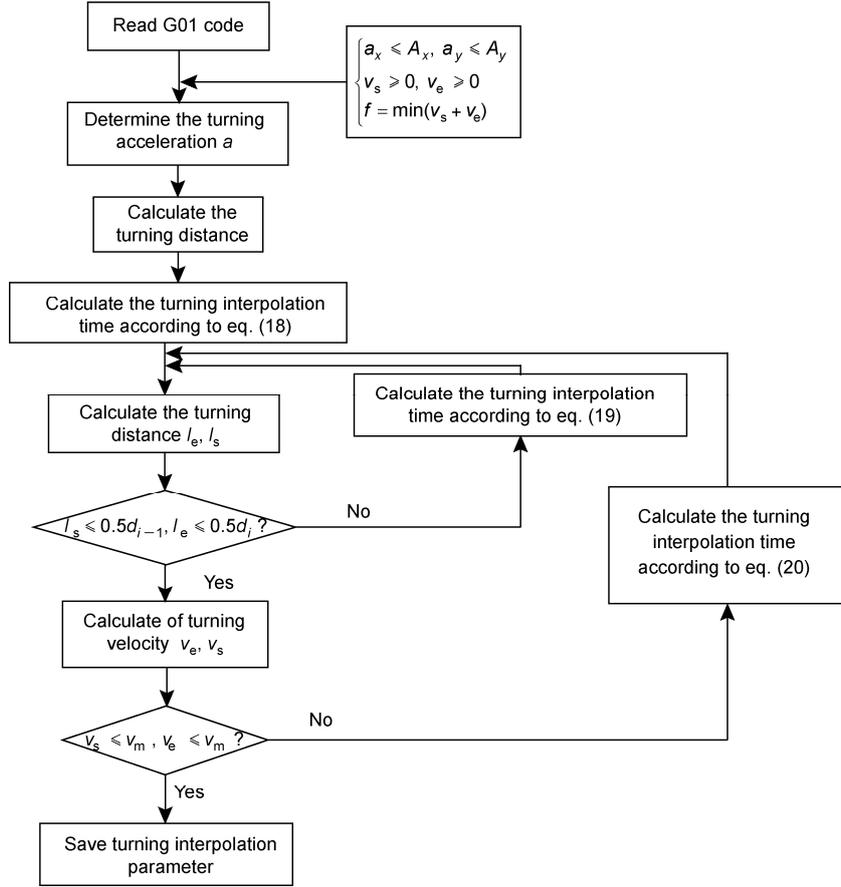


Figure 13 Flowchart of calculation of the optimal turning interpolation parameters.

3 Look-ahead scheme

The main task of look-ahead scheme is to ensure that all the processing speeds at both ends of line segments are accessible. The algorithm realizes the smooth speed variation and high machining speed automatically combining with geometric characteristics of machining path and the characteristics of the acceleration and deceleration control system. The efficiency of production has been improved under linear acceleration and deceleration control by look-ahead scheme in refs. [13, 14]. In ref. [15], the machining efficiency and quality are improved by look-ahead scheme based on linear and S-shaped acceleration and deceleration considering the error and dynamic performance confinement. Refs. [16, 17] introduced the double-linked list look-ahead scheme based on linear and cosine acceleration and deceleration. The speed variation chart is introduced based on the characteristics of acceleration and deceleration in ref. [18]. Refs. [19, 20] introduced the look-ahead scheme of NURBS and Bezier curve interpolation.

3.1 The algorithm of look-ahead scheme

The interpolation of G01 codes includes the interpolation of

line segments and interpolation of corners by multi-period turning interpolation, as shown in Figure 14. d_i and L_i represent the total and actual interpolation length of the line segment. $l_{i,e}$ and $l_{i+1,s}$ represent the post-turning and ex-turning distances of the i th line segment. We have the following equation:

$$L_i = d_i - l_{i,e} - l_{i+1,s}. \quad (21)$$

The maximal acceleration $a_{i,m}$ of every line segment interpolation is determined by the feasible accelerations of CNC machine and the direction of the line segment. For 3-axis CNC machine, the acceleration of line segment is

$$a_{i,m} = \min \left(\frac{A_x}{\cos \theta_x}, \frac{A_y}{\cos \theta_y}, \frac{A_z}{\cos \theta_z} \right),$$

where θ_x , θ_y , θ_z represent the angles formed by the interpo-

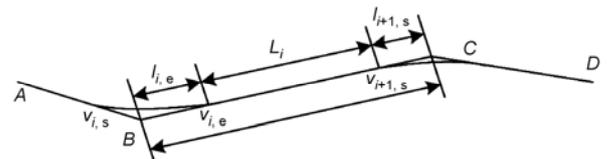


Figure 14 Interpolation on the line and the corner.

lated line segment and the direction of three linear driving axes. For 5-axis CNC machine including 3 linear driving axes and 2 rotation axes, we have

$$a_{i,m} = \min \left(\frac{A_x}{\cos \theta_x}, \frac{A_y}{\cos \theta_y}, \frac{A_z}{\cos \theta_z}, \frac{A_A}{\cos \theta_A}, \frac{A_B}{\cos \theta_B} \right),$$

where,

$$\begin{aligned} \cos \theta_x &= \Delta x / \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2 + \Delta A^2 + \Delta B^2}, \\ \cos \theta_y &= \Delta y / \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2 + \Delta A^2 + \Delta B^2}, \\ \cos \theta_z &= \Delta z / \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2 + \Delta A^2 + \Delta B^2}, \\ \cos \theta_A &= \Delta A / \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2 + \Delta A^2 + \Delta B^2}, \\ \cos \theta_B &= \Delta B / \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2 + \Delta A^2 + \Delta B^2}. \end{aligned}$$

$\Delta x, \Delta y, \Delta z, \Delta A, \Delta B$ represent the increment of components of adjacent G01 codes in the five directions of driving axes x, y, z, A, B .

The beginning interpolation velocity of a line segment is equal to the post-turning velocity of the starting point of the line, and the final interpolation velocity of a line segment is equal to the ex-turning velocity of the final point of the line segment. In order to guarantee the accessibility of the two ends' velocities of every line segment, the look-ahead scheme is executed as following:

Step 1. Data pre-read. Read m line segments before machining and the look-ahead segment number m is determined by the performance of CNC system. The beginning velocity of the first line segment and the final velocity of the last line segment are set to be zero. The other turning velocities are determined by the method stated in Section 2.

Step 2. Reverse accessibility check in acceleration mode. Starting with the last line segment, the accessibility of the velocities in acceleration mode at the endpoints of every line is checked in the reverse order until the first line segment or the termination condition given below is satisfied, and the look-ahead scheme is finished.

The accessibility in acceleration mode refers to that the ending velocity of the i th line segment can reach the starting velocity with the maximal acceleration on the line within the interpolation length of the line if the ending velocity is smaller than the starting velocity. If the ending velocity is larger than or equal to the starting velocity, the accessibility condition is automatically satisfied. We denote the $v_{i,rs}, v_{i,re}$ as the starting and ending velocities of the i th line segment which satisfy the accessibility in acceleration mode.

The accessibility test is stated as: if the velocity of endpoints satisfies the condition: $v_{i+1,rs} \geq v_{i,e}$ or $v_{i+1,rs} < v_{i,e}$ and $L_i \geq (v_{i,e}^2 - v_{i+1,rs}^2) / (2a_{i,m})$, the velocity is said to be reachable, and we set $v_{i,re} = v_{i,e}$. Otherwise, the velocity $v_{i,e}$ of the starting point should be reduced to $v_{i,re}$, and satisfies the following equation:

$$L_i = (v_{i,re}^2 - v_{i+1,rs}^2) / (2a_{i,m}). \quad (22)$$

According to eqs. (5), (6), and (22), we have

$$d_i - 0.5v_{i,re}t'_{i,m} - 0.5v_{i+1,rs}t_{i+1,m} = (v_{i,re}^2 - v_{i+1,rs}^2) / 2a_{i,m}, \quad (23)$$

where $t'_{i,m}, t_{i+1,m}$ represent the turning interpolation times of the starting and ending points of the line segment. The turning velocity is in proportion to the turning interpolation time, so we have

$$t'_{i,m} = t_{i,m} \cdot v_{i,re} / v_{i,e}, \quad (24)$$

where $t_{i,m}$ represents the turning interpolation time correspondent to the post-turning velocity $v_{i,e}$. Substituting eq. (24) into eq. (23), we obtain the reachable velocity

$$v_{i,re} = \sqrt{\frac{(v_{i+1,rs}^2 + 2a_{i,m}d_i - a_{i,m}v_{i+1,rs}t_{i+1,m})}{(1 + a_{i,m}t_{i,m} / v_{i,e})}}. \quad (25)$$

The ex-turning velocity and the correspondent turning interpolation time are adjusted accordingly:

$$\begin{cases} v_{i,rs} = v_{i,s} \cdot v_{i,re} / v_{i,e}, \\ t'_{i,m} = t_{i,m} \cdot v_{i,re} / v_{i,e}. \end{cases} \quad (26)$$

The ex-turning and post-turning distances are adjusted in the similar way according to eqs. (5) and (6).

Step 3. Real time interpolation. Take the first line segment in the look-ahead buffer queue. Calculate the interpolation point and implement the task of Step 4 once in an interpolation period time. Real time machining is carried on according to the interpolation points.

Step 4. Add a line segment into the end of the buffer-queue. The ending velocity of this line segment is set to be zero, and the starting velocity of this line is calculated by the multi-period turning method. The reverse acceleration accessibility is checked once again from the end of the queue according to Step 2.

We now give the termination condition for the j th reverse accessibility check in look-ahead scheme: the post-turning velocity $v'_{i,re}$ of the starting point in the j th look-ahead scheme is the same with the result $v'_{i,re}{}^{j-1}$ of the $(j-1)$ th look-ahead scheme. When the termination condition is satisfied, the j th reverse accessibility check will stop, and the rest of the turning velocities will stay unmodified.

When the interpolation of current line is finished, this line is deleted from the queue, then we take Steps 3 and 4 repeatedly.

The look-ahead scheme flowchart is shown in Figure 15, the dashed frame I is the process of data pre-read, where i is the sequence number of the waiting interpolation data. The dashed frame II represents the reverse accessibility check in Step 2. The real-time interpolation algorithm will be introduced in Section 5 in details.

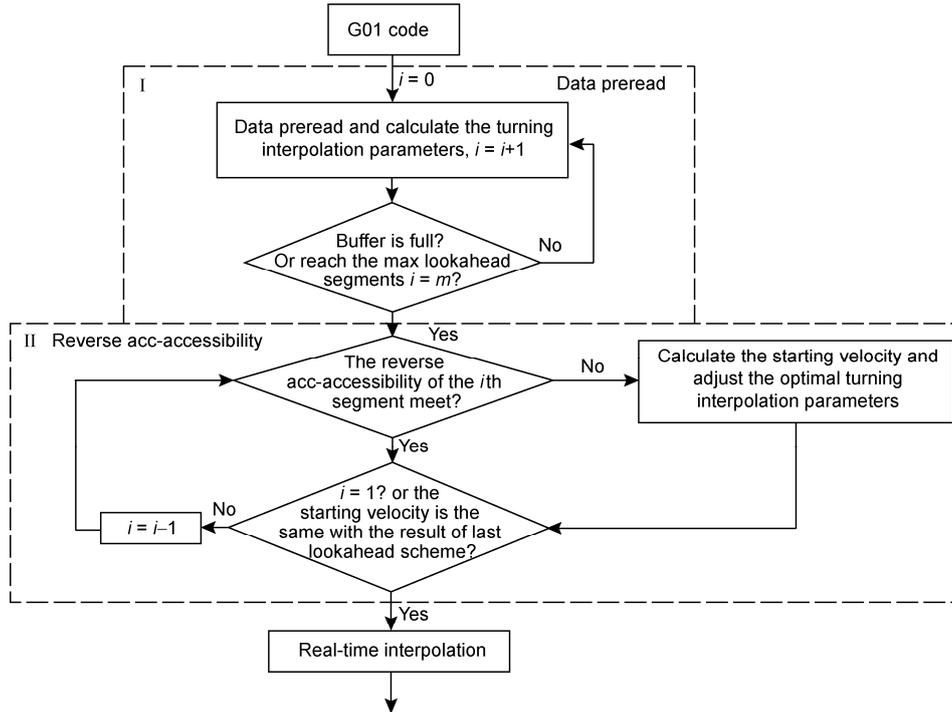


Figure 15 Flowchart of look-ahead scheme.

Using the idea in refs. [7, 10], we can show that the velocity obtained with our method is globally time-optimal in the case of single period turning.

3.2 Experimental results of look-ahead scheme

The look-ahead scheme checks the reverse accessibility in acceleration mode from the last line segment whose ending velocity is set zero in the buffer in every interpolation period. If the starting velocity of the interpolated line segment is greater than the ending velocity, look-ahead scheme guarantees the accessibility in deceleration mode in real-time interpolation. The test of forward acceleration accessibility will be introduced in the next section.

The turning interpolation parameters of m corners in the look-ahead buffer are calculated in step 1, so we need only to calculate the turning interpolation parameters of the last corner in Step 4 of the look-ahead scheme. In the 5-axis machining of a blade in Figure 25(a), the average calculation time of turning interpolation parameters is 0.00558 ms which satisfies the real-time look-ahead needs for 2–5-axis CNC machines.

This look-ahead scheme needs no backtracking and the processing time is shorter. In the meantime, the look-ahead scheme will be terminated when the termination condition is satisfied, hence the number of look-ahead segments can be adjusted dynamically according to the current computational time when the termination condition is satisfied. The processing time and look-ahead segments are improved accordingly. By experimental tests, the capacity of this

look-ahead scheme can reach 2000–5000 segments in common CNC machine. The look-ahead scheme is real-time interactive and online response to the feedrate override, which will be introduced in Section 4.

The actual machining result of Figure 16 employing the look-ahead scheme is shown as Figure 16(b). The comparison of computational time between our look-ahead scheme and the traditional method is shown in Table 1, where the data in the first row of our algorithm represent the time of the look-ahead scheme for Steps 1 and 2, and the data in the second row for Step 4. Since there is no backtracking and the adoption of the termination condition, the computational time of our algorithm is less than the available algorithm's. The average number of actually needed look-ahead segments in Step 4 is about 200 when the termination condition is satisfied.



Figure 16 The original graphic and the machining result.

Table 1 The comparison of computational time

Look-ahead segment	Computational time (ms)		
	our method		available method
	preread	add a new G01 code	
1000	0.14	0.027	0.25
2000	0.24	0.025	0.46
3000	0.38	0.031	0.70
4000	0.49	0.032	0.96

4 Feedrate override

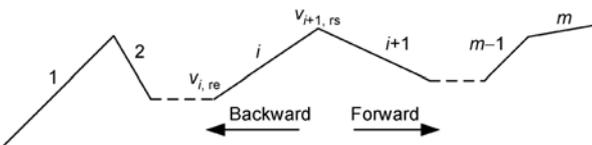
In practical CNC machining, the result of the look-ahead scheme would not be suitable when there is a feedrate override that is the changes of machining parameters such as the maximal programming feedrate. In order to keep fast response to the feedrate override and improve the machining velocity at the confinement of maximal accelerations of every axis, the look-ahead result should be adjusted according to feedrate override, which will be introduced in this section.

4.1 Maximal feedrate rise

The ex-turning and post-turning velocities are determined by the maximal machining error ε , ex-turning and post-turning distances, and the maximal feedrate v_m . The ex-turning and post-turning velocities determined by maximal error and turning distance are denoted by (v_s, v_e) , based on which, the turning velocity determined by maximal feedrate is denoted by (v'_s, v'_e) . The velocities of (v'_s, v'_e) will be adjusted as follows when the maximal feedrate rises to v_m .

Step 1. When $\max(v_s, v_e) \leq v_m$, the turning velocity will not be affected by the new v_m , otherwise, the turning velocities need to be recalculated by v'_m and ε .

Step 2. Carry out the reverse acceleration accessibility check again from the last modified turning velocity when the maximal feedrate is increased. As shown in Figure 17, the reverse acceleration accessibility is carried out when the starting velocity of the i th line segment increases according to the maximal feedrate, which is the last modified velocity in the look-ahead buffer. If the reverse acceleration accessibility is not satisfied, the turning velocity is modified according to eqs. (25) and (26). The procedure is carried out until the accessibility is satisfied, then jump to the previous modified turning velocity and carry on the same procedure

**Figure 17** The look-ahead segments.

until all the modified velocity is checked by the reverse acceleration accessibility.

4.2 Maximal feedrate decrease

When the maximal feedrate falls, all the turning velocities which are greater than the new maximal feedrate will be calculated again according to eqs. (20), (2), and (3). Compared with the case that the maximal feedrate increases, the look-ahead result will not be affected in this case.

However, there is another problem when the maximal feedrate falls: the forward deceleration is inaccessible for the current interpolating segment if its ending velocity is modified according to the new maximal feedrate and the two conditions are satisfied simultaneously: $v_{1, re} > v'_{2, rs}$ and $L_1 < (v_{1, re}^2 - v_{2, rs}^2) / 2a_{1, m}$, because the starting velocity cannot be modified in the implementation of interpolation. In order to reduce the speed to the new maximal feedrate within the maximal deceleration capacity of every axis, the adopted approach is introduced as follows:

(1) Calculate the accessible minimum ending velocity of the current interpolating line segment from the starting velocity through deceleration mode in the length L_1 range of the line.

The new ending velocity $v'_{2, rs}$ satisfies the following condition:

$$(d_1 - 0.5v_{1, re}t_{1, m} - 0.5v'_{2, rs}t'_{2, m}) \geq \frac{v_{1, re}^2 - v_{2, rs}^2}{2a_{1, m}}, \quad (27)$$

where $t_{1, m}, t'_{2, m}$ are the turning interpolation time of the starting and ending corners of the line segment, and we have

$$v'_{2, rs} / v_{2, rs} = t'_{2, m} / t_{2, m},$$

$v_{2, rs}$ is the ex-turning velocity of the starting corner of the current interpolating line before feedrate override. Substituting the above equation into eq. (27), we have

$$v_{2, rs}^2 \left(\frac{t_{2, m}}{2v_{2, rs}} - \frac{1}{2a_{1, m}} \right) \leq d_1 - \frac{v_{1, re}t_{1, m}}{2} - \frac{v_{1, re}^2}{2a_{1, m}}. \quad (28)$$

According to eq. (28), the following cases are discussed:

(a) If $t_{2, m}a_{1, m} > v_{2, rs}$ and $d_1 - \frac{v_{1, re}t_{1, m}}{2} - \frac{v_{1, re}^2}{2a_{1, m}} > 0$, then

the range of ending velocity $v'_{2, rs}$ is

$$0 \leq v'_{2, rs} \leq \sqrt{\frac{2a_{1, m}d_1 - v_{1, re}t_{1, m}a_{1, m} - v_{1, re}^2}{(a_{1, m}t_{2, m}/v_{2, rs}) - 1}}. \quad (29)$$

(b) If $t_{2, m}a_{1, m} < v_{2, rs}$ and $d_1 - \frac{v_{1, re}t_{1, m}}{2} - \frac{v_{1, re}^2}{2a_{1, m}} < 0$, then

the range of ending velocity $v'_{2,rs}$ is

$$v'_{2,r1} \geq \sqrt{\frac{2a_{1,m}d_1 - v_{1,re}t_{1,m}a_{1,m} - v_{1,re}^2}{(a_{1,m}t_{2,m}/v_{2,rs}) - 1}}. \quad (30)$$

(c) If $t_{2,m}a_{1,m} < v_{2,rs}$ and $d_1 - \frac{v_{1,r2}t_1}{2} - \frac{v_{1,r2}^2}{2a_m} > 0$, the

equation is always valid.

Hence, forward deceleration inaccessibility would not occur for cases (a) and (c) in the feedrate override, and the ending velocity of the current interpolating line is modified as

$$v'_{2,rs} = \sqrt{\frac{2a_{1,m}d_1 - v_{1,re}t_{1,m}a_{1,m} - v_{1,re}^2}{(a_{1,m}t_{2,m}/v_{2,rs}) - 1}}.$$

The corresponding post-turning velocity is

$$v'_{2,re} = v_{2,re} \cdot v'_{2,rs} / v_{2,rs}.$$

If $v'_{2,re} \leq v'_m$, the deceleration accessibility must be satisfied for the next line segment, the following look-ahead result will not be influenced by the feedrate override. If $v'_{2,re} > v'_m$ and the starting velocity and ending velocity of the next line are deceleration inaccessible, we adjust the turning acceleration to keep the $v'_{2,rs}$ invariable and the $v'_{2,re}$ as close as possible to the new maximal feedrate v'_m . Take the 3-axis CNC machine as an example, the main calculation is as follows:

(2) The range of turning acceleration a is the shady part in Figures 18 and 19, and the optimal turning acceleration is the point b in Figure 18 and point a in Figure 19. $v'_{2,rs}$ and $v'_{2,re}$ are known, according to eq. (2), we have

$$|a \times e_e| = |v'_{2,rs}(e_e \times e_s) / t_{2,m}| = u_2, \quad (31)$$

$$|a \times e_e| = |v'_{2,re}(e_e \times e_s) / t_{2,m}| = u_1. \quad (32)$$

The turning acceleration a corresponding to the ex-turning velocity $v'_{2,rs}$ is on the line ab in Figure 18 and point a in Figure 19, where the line l_2 is parallel to the direction of post-turning velocity e_e with distance u_2 . Similarly, the turning acceleration a corresponding to $v'_{2,re}$ is the point b in Figure 18 and on the line ab in Figure 19, where the line l_1 is parallel to the direction of e_s with distance u_1 . If $v'_{2,re} = v'_m$, then a is on the line cd (Figures 18 and 19), and the line l'_1 is parallel to e_s with distance u'_1 . Because of $v'_m < v'_{2,re}$ and according to eq. (32), we have $u'_1 < u_1$. In Figure 18, the intersection point g of cd and ab is in the available range of acceleration, so acceleration a_1 on point g corresponds to $v'_{2,rs}$ and v'_m . But the acceleration of every

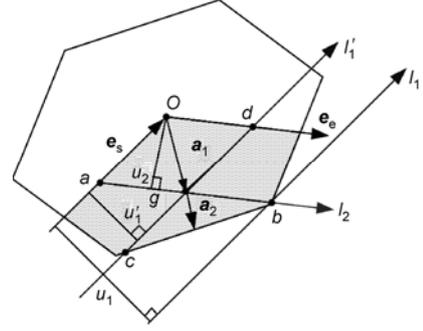


Figure 18 The available range I of the turning acceleration.

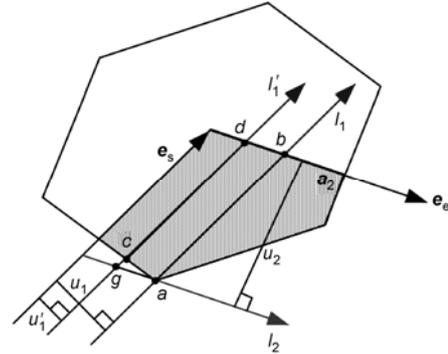


Figure 19 The available range II of the turning acceleration.

axis is not fully used, so the turning interpolation time is adjusted by

$$t''_{2,m} = t'_{2,m} |a_1| / |a_2|. \quad (33)$$

When the intersection point g is not in the available range of acceleration in Figure 19, the post-turning velocity cannot be modified to v'_m by the change of turning acceleration at a time. Steps (1) and (2) are carried on once again in the next line segment until the maximal feedrate is satisfied.

5 Real-time interpolation algorithm

The overall interpolation algorithm is divided into the corner interpolation and the line segment interpolation. In corner interpolation algorithm, the interpolation point sequence can be calculated by eq. (4) in every interpolation period and the optimal turning interpolation parameters are obtained in Section 3. The line segment interpolation algorithm is the calculation of the interpolation point sequence on the line segment according to the optimal turning interpolation parameters after look-ahead scheme in linear acc/dec mode.

The forward deceleration accessibility on a line segment is always satisfied because of the processing of reverse acceleration accessibility in the look-ahead scheme. Before

interpolation, we need to check the forward acceleration accessibility of the starting velocity and ending velocity of the current interpolating line segment:

$$L_1 \geq (v_{2,rs}^2 - v_{1,re}^2) / 2a_{1,m},$$

where the subscripts 1 and 2 represent the starting corner and ending corner of the current interpolating segment. If the above inequality is satisfied, then we can carry on the line segment and corner interpolation; otherwise, the ex-turning velocity of the ending corner of the line segment is adjusted according to the method proposed in Section 3:

$$v'_{2,rs} = \sqrt{\frac{(v_{1,re}^2 + 2a_{1,m}d_1 - a_{1,m}v_{1,re}t_{1,m})}{(1 + a_{1,m}t_{2,m}/v_{2,rs})}} \quad (34)$$

and other turning interpolation parameters are adjusted according to $v'_{2,rs}$. The interpolation on line segment and corner is implemented according to $v_{1,re}$ and $v'_{2,rs}$.

We use linear acc/dec mode, which consists of three or two stages as shown in Figures 20 and 21. The acceleration time, constant velocity time, and the deceleration time t_a , t_c , t_d are calculated by the length L_1 of the interpolated line segment and $v_{1,re}$, $v_{2,rs}$. We have

$$L_1 = d_1 - 0.5v_{1,re}t_{1,m} - 0.5v_{2,rs}t_{2,m}.$$

The maximal velocity in the line segment interpolation is $v_{1,m}$. If $(2v_m^2 - v_{1,re}^2 - v_{2,rs}^2) / a_{1,m} \leq L_1$, then

$$v_{1,m} = v_m. \quad (35)$$

Otherwise,

$$v_{1,m} = \sqrt{(2a_{1,m}L_1 + v_{1,re}^2 + v_{2,rs}^2) / 2}, \quad (36)$$

where $a_{1,m}$ is the maximal acceleration in the line segment interpolation, v_m is the maximal feedrate velocity. After v_m is obtained, we can compute t_a , t_c , t_d as follows:

$$\begin{cases} t_a = (v_{1,m} - v_{1,re}) / a_{1,m}, \\ t_c = \max((L_1 - (2v_{1,m}^2 - v_{1,re}^2 - v_{2,rs}^2) / a_{1,m}) / v_m, 0), \\ t_d = (v_{1,m} - v_{2,rs}) / a_{1,m}. \end{cases}$$

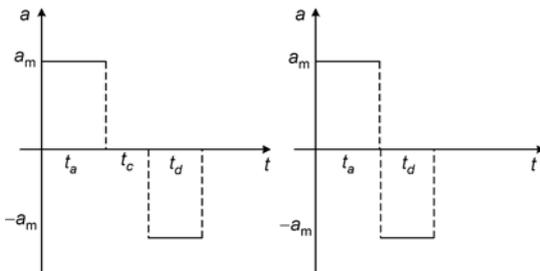


Figure 20 Acceleration-time graphs for the line segment interpolation.

The velocity-time graph is shown in Figure 21. The interpolation point sequence is calculated according to the sample period.

The starting position of corner interpolation is the last point on the line segment interpolation. The following interpolation point sequence is calculated by eq. (14) shown as the solid dots (A, B, C, \dots) in Figure 22. In practical machining, the interpolation time of the join point between line segment and corner might not be integer times of the interpolation period, that is the interpolation time t' satisfies: $(n-1)T < t' < nT$, as shown in Figure 22. The actual interpolation trajectory is the hollow points sequence (A', B', \dots). The calculation formula of new interpolation point sequence is

$$\frac{A'A}{AB} = \frac{B'B}{BC} = \frac{C'C}{CD} = \dots = \tau, \quad (37)$$

where A' is calculated by the algorithm for line segment interpolation. The new interpolation point sequence calculated by eq. (37) satisfies the acceleration requirement because of the convex property.

The main flowchart of the real-time interpolation is shown in Figure 23.

6 Experimental results

The multi-period turning interpolation algorithm shown in Figure 23 for consecutive micro-line segments is implemented in 3-axis and 5-axis Sky-Blue CNC systems. The machining velocity and quality are both improved. The vibration of the CNC machine is weakened because the acceleration is strictly within the maximal value of every

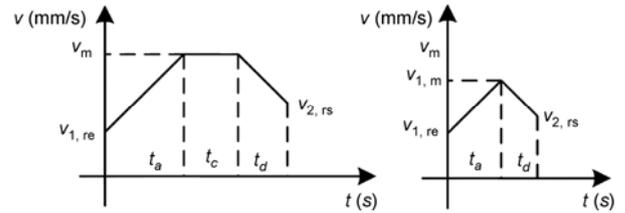


Figure 21 Velocity-time graphs for line segment interpolation.

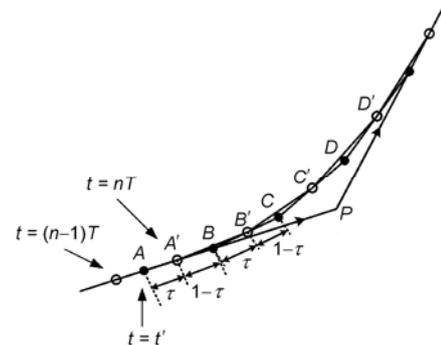


Figure 22 Joint of line segment and corner.

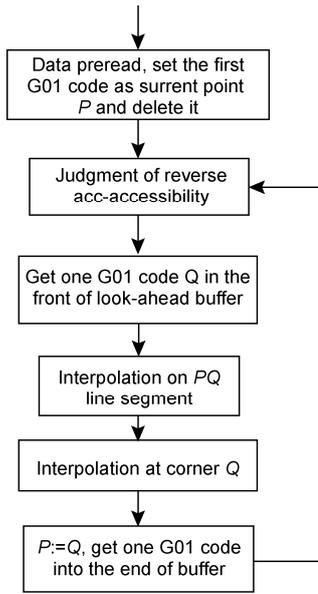


Figure 23 Flowchart of real-time interpolation.

driving axis. As a consequence, the quality of products is also improved.

The machining test of our algorithm is implemented on the “vase” design. Figure 16(b) is the actual machined product. The CNC machine used is shown in Figure 24(a). The total G01 codes of this design have 116000 line segments with total length 20.25 m, in which the shortest line segment is 0.09 mm. The machined material is wood. The machining parameters are: the maximal acceleration of both the x -axis and y -axis is 1000 mm/s^2 , the maximal feedrate is 200 mm/s , the allowable machining error is 0.01 mm . The total machining time is 22.78 min by the multi-period turning interpolation algorithm proposed in this paper.

The comparison is shown in Table 2 between our algorithm and two existing methods: “turning internal bisector” method and the traditional method of zero turning velocity with different machining parameters.

Table 3 is the machining time in the process of a small blade and a large blade. Figure 25 is the comparison of the actual machining result: blade 1 is manufactured by the ex-



Figure 24 The 3-axis and 5-axis CNC machines used in our experiments.

Table 2 Interpolation time comparison between different interpolation algorithms for “vase machining”

Acc limitation of x, y -axes (mm/s^2)	Overall interpolation time (min)			Velocity improvement (%)	
	our algorithm	bisector	stop	our algorithm vs. bisectors	our algorithm vs. stop
1000/1000	22.78	40.52	57.13	77.88	150.79
3000/1000	18.62	31.59	44.16	69.66	137.16
3000/3000	13.15	21.22	34.92	61.37	165.55
6000/6000	9.30	14.33	26.15	54.09	181.18

Table 3 Interpolation time comparison between different interpolation algorithms for “blade machining”

Operation	Maximal acceleration of every axis ($\text{mm/s}^2, \text{rad/s}^2$)	Program feedrate (mm/min)	Spindle rotation speed (round/min)	Overall interpolation time (min)		Velocity improvement (%)
				our algorithm	bisectors	
Channel rough machining		4000	12000	10.3	17.5	69.90
Rough milling large blade		1500	8000	26.5	40	50.94
Rough milling small blade	x, y, z -axes: $200 / 200 / 200$ A, B -axes: $120 / 120$	1000	8000	8.7	14.2	63.22
Finish milling large blade		4000	12000	6.9	12.1	75.36
Finish milling small blade		4000	12000	4.9	10.6	116.32

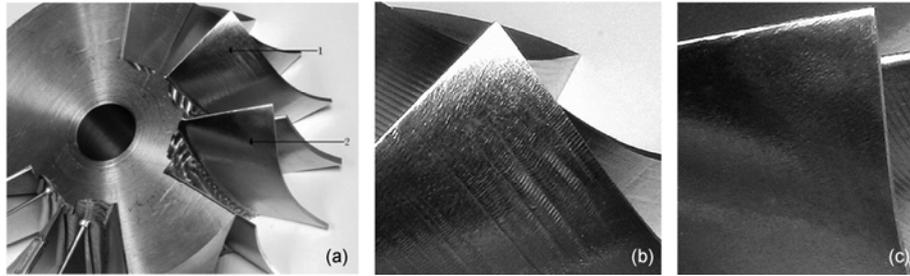


Figure 25 The overall and local comparison of manufacturing results. (a) Manufacturing result; (b) Blade 1; (c) Blade 2.

isting algorithm and blade 2 is by our algorithm. The CNC machine used is shown in Figure 24 (b). The machined material is aluminum. From the local magnified pictures, we can see that the multi-period turning interpolation algorithm also improves the manufacturing quality, due to the turning interpolation parameters determined by maximal error allowance, the maximal feedrate and acceleration of the used CNC machine. From Table 3, we can see that our new method is also faster than the existing methods.

7 Conclusion

The key issue of high speed and high precision CNC machining of micro-line interpolation is the turning velocity improvement at the corners. We propose the multi-period turning interpolation algorithm with a constant acceleration at the corner, which satisfies the machining error allowance, the maximal acceleration bound and feedrate bound. In certain sense, the turning velocity is optimal. The look-ahead scheme effectively predetermines the global accessibility efficiently. As a result, the vibration caused by the sudden velocity change is weakened. Our algorithm improves the machining speed significantly compared with the existing algorithms. In the meantime, the calculations are simple which meets the requirements of the real-time interpolation and online feedrate override. This algorithm also has the advantage of broad applications suitable for 2-axis to 5-axis CNC machining. At last, the practical CNC machining results of this algorithm show the advantages of the method.

This work was supported by the National Key Basic Research Project of China (Grant Nos. 2011CB302400) and the National Natural Science Foundation of China (Grant Nos. 60821002, 10871195, 10925105), Major National S&T Project "Advanced CNC Systems" and CAS Project "MM Methods for Advanced CNC Systems".

- 1 Ye P Q, Zhao S L. Study on control algorithm for micro-line continuous interpolation (in Chinese). *China Mech Eng*, 2004, 15: 1354–1356
- 2 He J, You Y P, Wang H M. A micro-line transition algorithm based on ferguson spline for high speed machining. (in Chinese). *China Mech Eng*, 2008, 19: 2085–2089
- 3 Lv Q, Zhang H, Yang K M, et al. Study on the method of increasing turning velocity during CNC continuous machining (in Chinese). *Technol Test*. 2008, 7: 79–83
- 4 Zhang M, Yan W, Yuan C M, et al. Curve fitting and optimal interpolation on CNC machines based on quadratic B-splines. *Sci China Infor Sci*, 2011, doi: 10.1007/s11432-011-4237-4
- 5 Zhang X H, Yu D, Hu Y, et al. Research on NUEBS curve interpolation algorithm base on filtering. (in Chinese). *China Mech Eng*, 2009, 20: 1695–1699
- 6 Sebastian D T, Farouki R T, Smith T S, et al. Algorithms for time-optimal control of CNC machines along curved tool paths. *Robot CIM-Int Manuf*, 2005, 21(1): 37–53
- 7 Timar S D, Farouki R T. Time-optimal traversal of curved paths by Cartesian CNC machines under both constant and speed-dependent axis acceleration bounds. *Robot CIM-Int Manuf*, 2007, 23(2): 563–579
- 8 Zhang K, Gao X S, Li H, et al. A greedy algorithm for feed-rate planning of CNC machines along curved tool paths with confined jerk for each axis. *MM-Preprints, KLMM, CAS, Beijing*, 2010, 29: 189–205
- 9 Wang J B, Yau H T. Real-time NURBS interpolator: application to short linear segments. *Int J Adv Manuf Technol*, 2009, 41: 1169–1185
- 10 Yuan C M, Gao X S. Time-optimal interpolation of CNC machines along parametric path with chord error bounds. *MM-Preprints, KLMM, CAS, Beijing*, 2010, 29: 165–188
- 11 Zhang L X, Li H B, Gao X S, et al. High speed micro-line interpolation algorithm under jerk limited control in CNC machining (in Chinese). *MM-Preprints, KLMM, CAS, Beijing*, 2010, 29: 228–238
- 12 Farin G, Hoschek J, Kim M S. *Handbook of computer aided geometric design*. Amsterdam: Elsevier Science B.V, 2002. 188
- 13 Hu J, Xiao L, Wang Y, et al. An optimal feedrate model and solution algorithm for a high-speed machine of small line blocks with look-ahead. *Int J Adv Manuf Technol*, 2006, 27: 930–935
- 14 Xu H F, Wang Y H. Research on feedrate model of look-ahead and solution algorithm for high-speed machining of small line segments. *Mech Eng*, 2005, 9: 9–13
- 15 Peng F, Chen X, Zhou Y, et al. Look-ahead scheme for high speed consecutive micro line interpolation based on dynamics of machine tool. *Heidelberg: Springer*, 2008. 743–753
- 16 Luo F Y, Zhou Y F, Yin J. A universal velocity profile generation approach for high-speed machining of small line segments with look-ahead. *Int J Adv Manuf Technol*, 2007, 35: 505–518
- 17 Luo F Y, Zhou Y F, Yin J. A generalized acceleration and deceleration approach for continuous small line blocks with look-ahead. *J Wuhan Univ Technol*, 2006, 27: 1052–1058
- 18 Jeon J W, Ha Y Y. A generalized approach for the acceleration and deceleration of industrial robots and CNC machine tools. *IEEE Trans Ind Electron*, 2000, 47: 132–139
- 19 Tsai M S, Nien H W, Yau H T. Development of an integrated look-ahead dynamics-based NURBS interpolator for high precision machinery. *Computer-Aided Design*, 2008, 40: 554–566
- 20 Yau H T, Wang J B. Fast Bezier interpolator with real-time look-ahead function for high-accuracy machining. *Int J Mach Tools Manuf*, 2007, 47: 1518–1529