

# On the development of a parametric interpolator with confined chord error, feedrate, acceleration and jerk

Jiing-Yih Lai · Kuan-Yuan Lin · Sheng-Jung Tseng · Wen-Der Ueng

Received: 20 September 2006 / Accepted: 17 January 2007 / Published online: 13 February 2007  
© Springer-Verlag London Limited 2007

**Abstract** A parametric interpolator for the trajectory planning of non-uniform rational basis spline (NURBS) curves is proposed in this study. The input constraints include the chord error, speed limit, acceleration limit and jerk limit. Since there is no unique representation to relate these constraints, appropriate models are developed to determine several feedrates in terms of different kinematic conditions, the minimum one of which is selected as the desired feedrate. A look-ahead stage is developed to plan a series of segment points from the curves, where a segment point represents the change in the acceleration across zero. The segment points, which enable the appropriate design of kinematic profiles, are used in real-time sampling for determining the sampling points. Several examples are presented to demonstrate the feasibility of the proposed algorithm.

**Keywords** Parametric interpolator · NURBS curve · Look-ahead · Real time sampling

## 1 Introduction

Modern computer-aided design/computer-aided manufacturing (CAD/CAM) systems apply parametric curves for

modelling freeform profiles. The parametric curves must be converted into linear and circular blocks based on pre-defined chord accuracy in CAM systems such that conventional computer numerically controlled (CNC) interpolators can be applied. The parametric interpolator has recently been accepted as an alternative to replace these conventional linear and circular interpolators because it is potentially more suitable for high-speed and high-accuracy machining. In addition to the chord accuracy that has been widely studied, the design of smooth kinematic profiles is especially important in high-speed machining, since acceleration and jerk fluctuations become a serious problem affecting the quality of the machined surface. The purpose of this study is to develop a parametric interpolator that emphasises the control of acceleration and jerk profiles in order to reduce the chattering or vibration due to high jerk.

Parametric interpolators may be classified into the following three types: uniform interpolator, speed-controlled interpolator and adaptive-feedrate interpolator. Bedi et al. [1] developed a uniform interpolator in which a constant parametric increment was applied. Although it was the simplest method, it was short of the chord error and speed control. Shpitalni et al. [2] applied arcs to approximate the parametric curve and utilised a constant feedrate to determine the sampling points. Yang and Kong [3] applied Taylor's expansion method for determining the sampling points and provided a parametric interpolator of constant feedrate. They concluded that the parametric interpolator was superior to a conventional linear interpolator in terms of its memory size requirement, speed accuracy, position tracking error and jerk magnitude.

Tsai et al. [4] applied the simplified Taylor's first-order and second-order approximation method for a non-uniform rational basis spline (NURBS) surface interpolator, emphasizing a constant cutter contact (CC) velocity along the CC

---

J.-Y. Lai (✉) · K.-Y. Lin · S.-J. Tseng  
Mechanical Engineering Department, National Central University,  
Taiwan, People's Republic of China  
e-mail: jylai@cc.ncu.edu.tw

W.-D. Ueng  
Mechanical Engineering Department,  
Tung Nan Institute of Technology,  
Taiwan, People's Republic of China

paths, instead of constant cutter location (CL) velocity. A real-time motion control network was developed to achieve multi-axis synchronous motion. Yeh and Hsu [5] included a compensatory value in the first-order approximation algorithm to increase the accuracy in both the position and the speed of a parametric interpolator. A constant-speed mode and a specified acceleration/deceleration mode were provided to meet different feedrate commands. Yeh and Hsu [6] further provided an adaptive-feedrate interpolator to change the curve speed adaptively, depending on the curvature during the interpolation process. However, the limits of acceleration and deceleration were not taken into account; it might result in excessive acceleration/deceleration that was beyond the machine's capability.

Xu et al. [7] developed a geometry-dependent feedrate and a time-varying feedrate for the determination of the sampling points. The feedrate was either varied according to the local ratios of curvature to ensure a nearly constant contour error or expressed as a linear relationship with a constant acceleration to reach a continuous feedrate profile, changing from one feedrate to another. Liu et al. [8] proposed an interpolator which met the requirements of not only the constant feedrate and chord accuracy, but also real-time integrated machine dynamics. Three aspects of machine dynamics were considered in this study: sharp corners, frequencies matching machine-natural ones and high jerk. Appropriate methods were also presented to eliminate each of them.

Yong et al. [9] proposed an offline predetermination of sharp corners and developed a speed-error-controlled interpolator to confine the chord error, the speed and the acceleration/deceleration of machining during the interpolation process. Using this algorithm, the program made a trial run searching for critical points and recording what action to be taken before real machining. Nam and Yang [10] presented a kinematics algorithm to determine an admissible path increment with real-time jerk-limited acceleration. The consecutive tool path was determined recursively at every sampling period by computing the kinematic profile and keeping track of when to start the final deceleration stage. This resulted in the exact feedrate trajectory generation for the parametric curve.

The machine's capability is an important issue affecting the efficiency of the CNC interpolator. The maximum feedrate, acceleration/deceleration and jerk that can be output by a servo motor are generally limited and should be specified in the design of the kinematic profiles. In particular, high jerk usually happens upon a sudden change in speed. This is usually the cause of machine vibration and should be avoided. In addition, there are several issues that may affect the performance of the parametric interpolator. First, the chord error and feedrate control are the fundamental requirements for most interpolators. Second, the

design of the kinematic profiles can affect the overall performance of the interpolator. Third, particular care to keep the balance of both accuracy and feedrate should be taken while crossing a sharp corner. Fourth, the velocity profiles of adjacent paths should also be kept as similar as possible in order to reduce the tool marks that might exist between the paths. Finally, multiple curves machining may cause additional problems, such as tiny curves and boundary continuity. It is necessary to develop appropriate algorithms to deal with each of these problems such that the proposed interpolator is able to achieve the goal in various conditions.

The purpose of this study is to propose a parametric interpolator that can process NURBS parametric curves. In addition to chord error and feedrate control, the proposed algorithm focusses on the following several issues: (1) it provides a look-ahead module to prevent high jerk and to keep the similarity of the velocity profile between adjacent paths; (2) acceleration/deceleration and jerk limits are provided as constraints to prevent saturation of the servo motor; (3) a strategy is taken to control the feedrate in sharp corners; (4) various continuity conditions for composite curves are considered in this algorithm. The proposed method is divided into a look-ahead stage and a real-time sampling stage. The look-ahead stage is primarily used to determine the segment points with significant variation in acceleration/deceleration, with all motion constraints specified. Once the segment points are determined, the real-time sampling stage is implemented to determine a jerk-limited trajectory between two segment points, while maintaining the necessary kinematics constraint conditions. Each of the algorithms is discussed in detail, with the above four issues being revolved. Several realistic examples are simulated to demonstrate the characteristics and feasibility of the proposed algorithm.

## 2 The fundamental sampling algorithm

Most parametric interpolators start with Taylor's first-order or second-order approximation method for sampling a parametric curve, in which the curve is sampled in a constant period of time in terms of a given feedrate and the first and second differential of the curve. The feedrate is represented as a variable which is a function of the allowable chord error and the curvature at a curve point. This sampling algorithm is the basis of the proposed interpolator, which is briefly discussed below.

Consider that the NURBS curve is  $\mathbf{C}(u) = (x(u)y(u)z(u))^T$ , where the parameter  $u$  is used to represent the position of a point on the curve. The increment in  $u$ , namely,  $\Delta u$ , can be determined based on the feedrate  $V$  and the first and second differentials of  $\mathbf{C}(u)$  at  $u_i$ . Taylor's

second-order approximation is applied for the determination of the next sampling point  $u_{i+1}$  [5] as follows:

$$u_{i+1} = u_i + \Delta u \tag{1}$$

$$\Delta u = \frac{V(u_i)T_s}{\left[\frac{dC(u_i)}{du}\right]} - \frac{V^2(u_i)T_s^2 \left(\frac{dC(u_i)}{du} \cdot \frac{d^2C(u_i)}{du^2}\right)}{2\left[\frac{dC(u_i)}{du}\right]^4} \tag{2}$$

where  $T_s$  is the sampling time and  $V$  is the feedrate. It is noted that Taylor’s first-order approximation method can also be applied. However, as long as the CPU time is allowed, the second-order algorithm is better.

The parameters affecting  $\Delta u$  in Eq. 2 are  $T_s$  and  $V(u_i)$ . The sampling time  $T_s$  is typically constant, which is determined in terms of the hardware capability. The design of the feedrate  $V(u_i)$  can affect the performance of the interpolator. A simple design of  $V(u_i)$  is to control the chord error between the chord of two adjacent sampling points and the arc corresponding to these two points, where the feedrate can be expressed as [6]:

$$V_e(u_i) = \frac{2}{T_s} \sqrt{r_i^2 - (r_i - ER)^2} \tag{3}$$

where the subscript e indicates that the feedrate is acquired in terms of the chord error,  $ER$  represents the allowable chord error and  $r_i$  is the radius of a circle which is tangential to the curve at  $u_i$ . Here,  $r_i$  is the reciprocal of the curvature at  $u_i$ . Equation 3 has nothing to do with the acceleration and jerk and, hence, the acceleration and jerk profiles may fluctuate frequently, with the maximum values beyond their limits.

### 3 The modified sampling algorithm

To yield smooth kinematic profiles for parametric curves, the feedrate should also be designed in terms of the acceleration and jerk limits. There is no unique representation to relate the feedrate, acceleration limit and jerk limit simultaneously. Appropriate models are provided to evaluate several feedrates in terms of different kinematic conditions. The minimum feedrate is then chosen for the determination of the next sampling point.

A local region of the curve at  $u_i$  is firstly considered as a circle and the feedrate in terms of the acceleration limit is determined. Here, the speed can be assumed to be constant along the circle, while the acceleration is primarily the centrifugal acceleration. Figure 1 depicts the variation of the speed  $\Delta V$  in a vector form between the sampling points

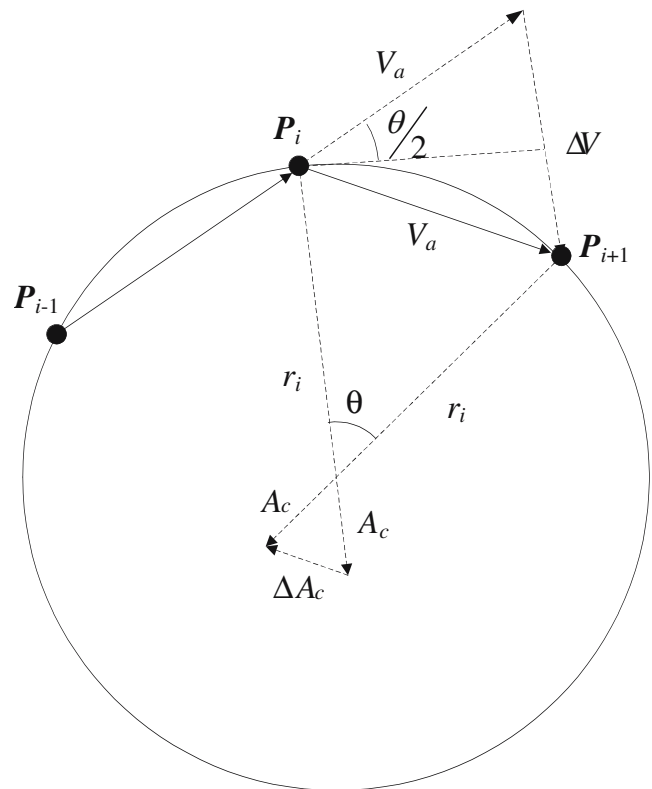


Fig. 1 Individual feedrate can be determined in terms of the allowable chord error, centrifugal acceleration and centrifugal jerk, respectively

$P_i$  and  $P_{i+1}$ . The centrifugal acceleration  $A_c$  at  $u_i$  can be expressed as:

$$A_c = \frac{V_a^2(u_i)}{r_i} \tag{4}$$

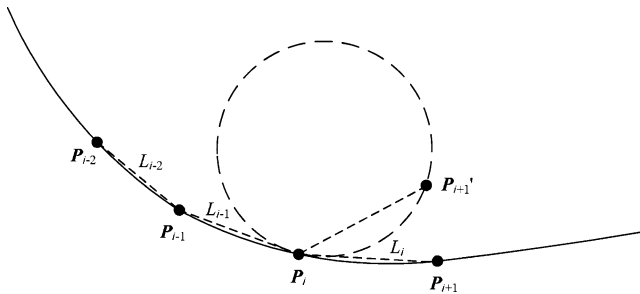
where  $V_a(u_i)$  denotes the feedrate at  $u_i$  and the subscript a indicates that the feedrate is due to the acceleration limit. Upon considering that the acceleration limit is  $A_{max}$ , then  $V_a(u_i)$  can be expressed as:

$$V_a(u_i) = \sqrt{r_i A_{max}} \tag{5}$$

Besides the variation in speed, the centrifugal acceleration will also vary, resulting in a jerk. As Fig. 1 depicts, the variation of the centrifugal acceleration  $\Delta A_c$  between the two sampling points  $P_i$  and  $P_{i+1}$  in a circle is primarily due to the directional change of the motion. A maximum feedrate can, therefore, be evaluated in terms of the jerk limit. In Fig. 1, the relationship between  $V_j$  and  $\theta$  is as follows:

$$V_j T_s = 2r_i \sin\left(\frac{\theta}{2}\right) \tag{6}$$

where the subscript j represents that the feedrate is due to the centrifugal jerk. Since  $\theta$  is generally small, the following



**Fig. 2** Terms used for the evaluation of the feedrate  $V_c$  along each coordinate direction

expression is obtained by approximating  $\sin(\frac{\theta}{2}) \approx \frac{\theta}{2}$ :

$$\theta = \frac{V_j T_s}{r_i} \tag{7}$$

Similarly, the following expression can be obtained from Fig. 1:

$$\Delta A_c = 2A_c \sin\left(\frac{\theta}{2}\right) = \frac{V_j^2}{r_i} \times \frac{V_j T_s}{r_i} \tag{8}$$

Since the jerk  $J_c$  can be expressed as:

$$J_c = \frac{\Delta A_c}{T_s} = \frac{V_j^3}{r_i^2} \tag{9}$$

thus, if the jerk limit is  $J_{max}$ , then Eq. 9 can be rewritten as:

$$V_j(u_i) = \sqrt[3]{r_i^2 J_{max}} \tag{10}$$

Now, there are three feedrates  $V_c(u_i)$ ,  $V_a(u_i)$  and  $V_j(u_i)$  in terms of the chord error, the centrifugal acceleration and the centrifugal jerk, respectively. The minimum value of these feedrates could be chosen as the desired feedrate for the computation of the next sampling point. However, all of these values are evaluated by approximating the curve as an arc. The actual sampling point on a curve is different from that on an arc. As Fig. 2 depicts, the actual sampling point on the curve is  $P_{i+1}$ , rather than  $P'_{i+1}$ , which may result in a directional change of motion and, hence, excessive acceleration and jerk along a coordinate direction. Therefore, a fourth algorithm is developed by limiting the feedrate in terms of the acceleration and jerk limits in each coordinate direction.

Consider that the point  $P_{i+1}$  as depicted in Fig. 2 is acquired by means of the minimum feedrate among  $V_c$ ,  $V_a$  and  $V_j$ . The points  $P_{i-2}$  to  $P_{i+1}$  are used to evaluate the acceleration and jerk in each coordinate direction. Hereafter, only the x component of the kinematic parameters is discussed. The other two components can also be evaluated in a similar manner. Since the time period between  $P_{i-2}$  and  $P_{i+1}$  is very short, the speed between  $P_{i-2}$  and  $P_{i+1}$  is

assumed to be constant and is denoted as  $V_c(u_i)$ . Let the lengths of the line segments as shown in Fig. 2 be  $L_{i-2}$ ,  $L_{i-1}$  and  $L_i$ . Therefore, the speed of each line segment along the x direction can be evaluated as:

$$V_{k,x} = V_c(u_i) \frac{P_{k+1,x} - P_{k,x}}{L_k} \tag{11}$$

$$k = i, i - 1 \text{ and } i - 2$$

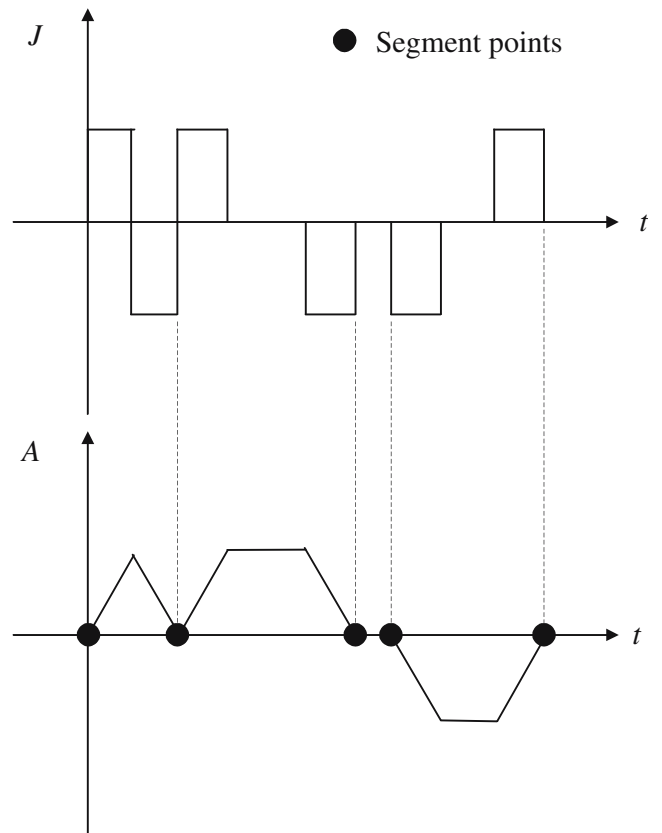
where  $P_{k,x}$  indicates the x component of point  $P_k$ . The jerk  $J_i$  at  $u_i$  can be evaluated as:

$$J_i = \frac{1}{T_s} \left( \frac{V_{i,x} - V_{i-1,x}}{T_s} - \frac{V_{i-1,x} - V_{i-2,x}}{T_s} \right) \tag{12}$$

We replace  $J_i$  with  $J_{max}$  and substitute Eq. 11 into Eq. 12 to yield:

$$V_c(u_i) = J_{max} T_s^2 \left( \frac{P_{i+1,x} - P_{i,x}}{L_i} - 2 \frac{P_{i,x} - P_{i-1,x}}{L_{i-1}} + \frac{P_{i-1,x} - P_{i-2,x}}{L_{i-2}} \right)^{-1} \tag{13}$$

The same procedure can be carried out along the y and z directions. Three values of  $V_c(u_i)$  can, thus, be obtained and the minimum one of these is chosen. The desired feedrate  $V_i$



**Fig. 3** Typical kinematic profiles for the proposed interpolator

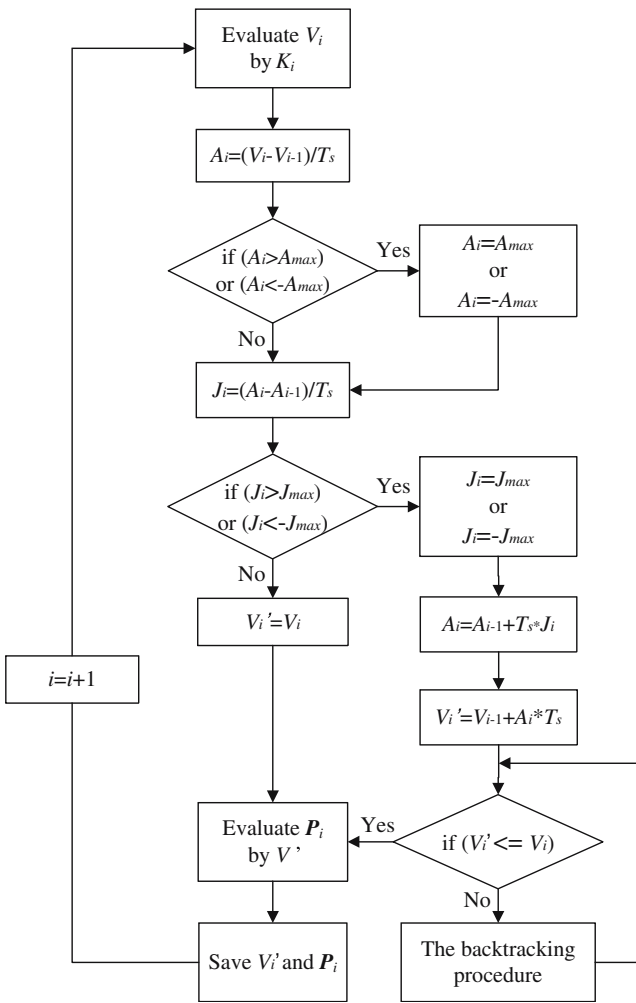


Fig. 4 The look-ahead procedure

at  $u_i$  is, thus, chosen as:

$$V_i = \min (V_e, V_a, V_j, V_c) \tag{14}$$

where the symbol  $u_i$  is neglected in  $V$ .  $V_i$  is then substituted into Eq. 2 for the determination of the next sampling point at  $u_{i+1}$ . Since  $V_i$  is the minimum value among the four values acquired, it is predictable that the chord error, acceleration limit and jerk limit are satisfied simultaneously.

4 The look-ahead stage

The  $V_i$  mentioned in Eq. 14 represents the speed limit allowed at  $u_i$  for the computation of the next sampling point. Appropriate kinematic profiles should be established to guarantee a smooth velocity profile along the parametric curve without violating the speed limit at each point. A two-stage process is proposed, namely, a look-ahead stage and a real-time sampling stage, for the design of the kinematic profiles. The look-ahead stage is a preview process, which determines a series of critical points on the

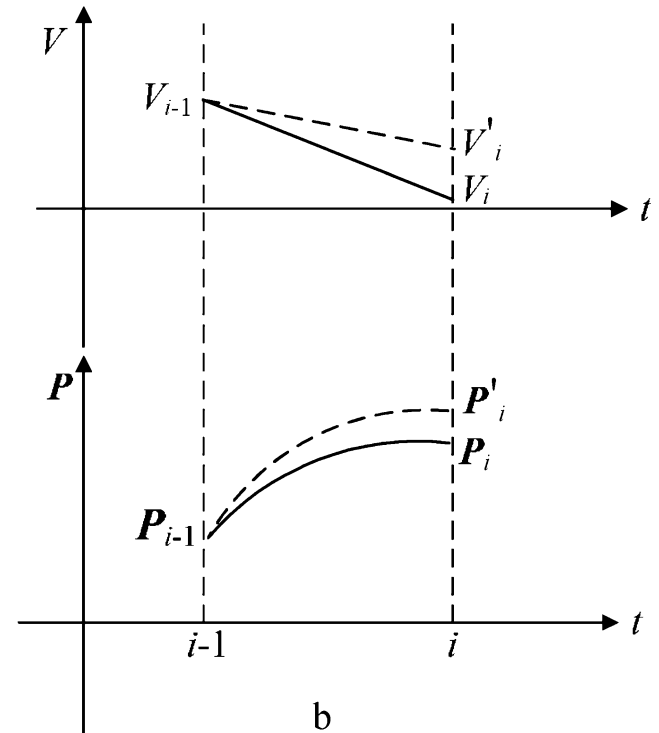
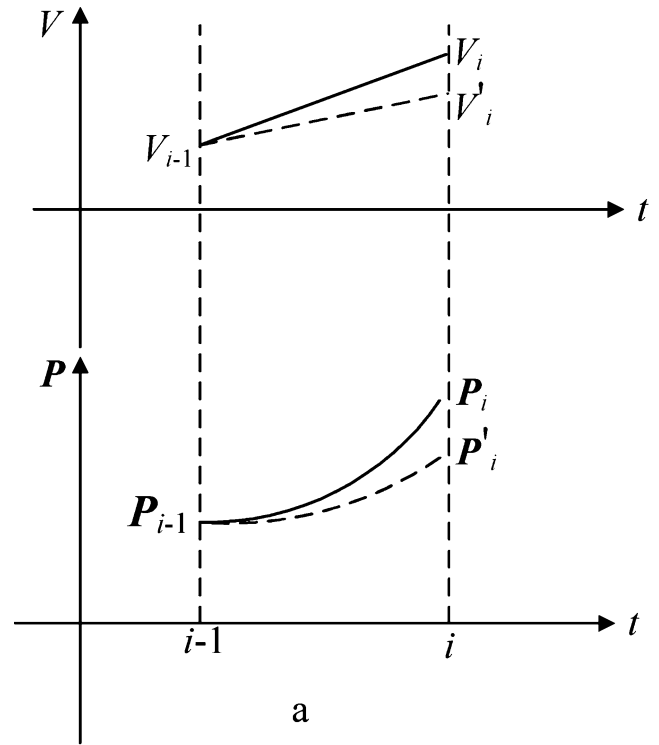


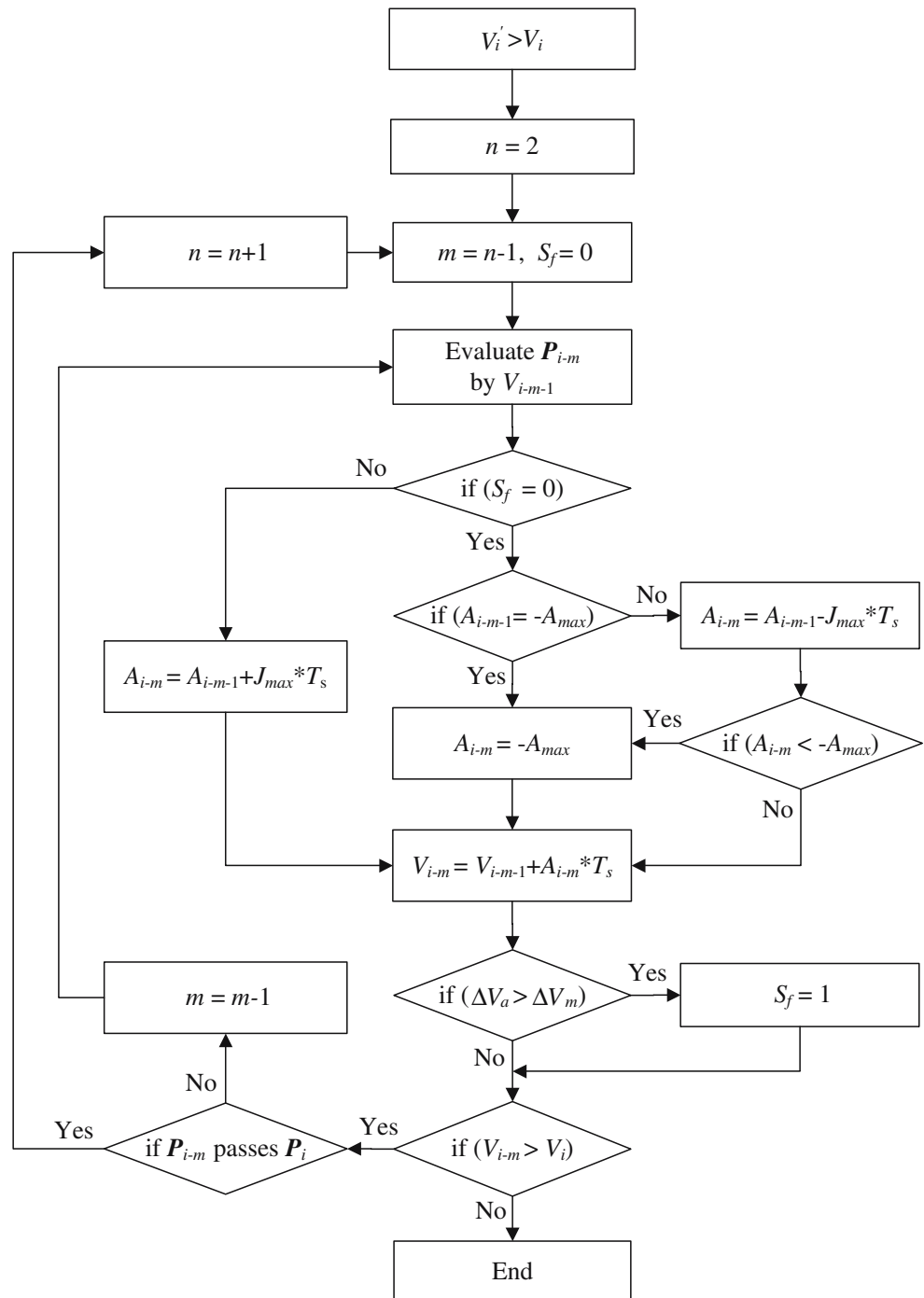
Fig. 5 Modified position  $P_i$  and speed  $V_i'$  after the check of the acceleration and jerk. a Positive acceleration  $A_i$ . b Negative acceleration  $A_i$

parametric curve for the design of the acceleration and jerk profiles. The real-time sampling stage is implemented on-line, which computes a motion command and sends this to the servo controller periodically.

Figure 3 depicts typical kinematic profiles designed in the proposed interpolator, where the jerk profile is primarily composed of pulse functions, while the acceleration profile is composed of triangular and trapezoidal functions. The design of the critical points is to provide an appropriate algorithm to command the system to adjust the motion in

the real-time sampling. If there is no prior information provided, the system would not know when to start to accelerate, decelerate or keep zero acceleration. Therefore, the objective of the look-ahead stage is to go through the entire motion design first and then identify all of the critical points which can be used for the design of the motion in the real-time sampling. The critical points, hereafter, denoted “segments points,” are indicated as spots on the acceleration plot of Fig. 3. Each segment point represents the transition of the acceleration across zero. Each interval

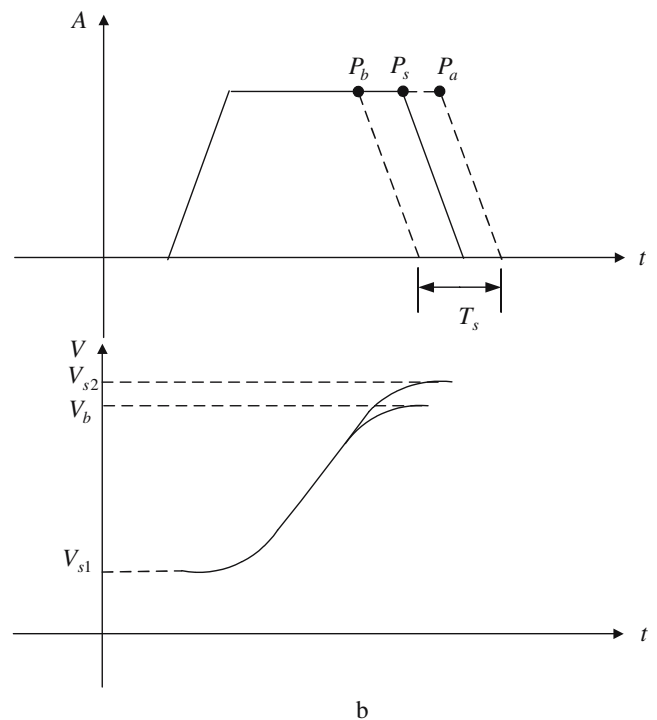
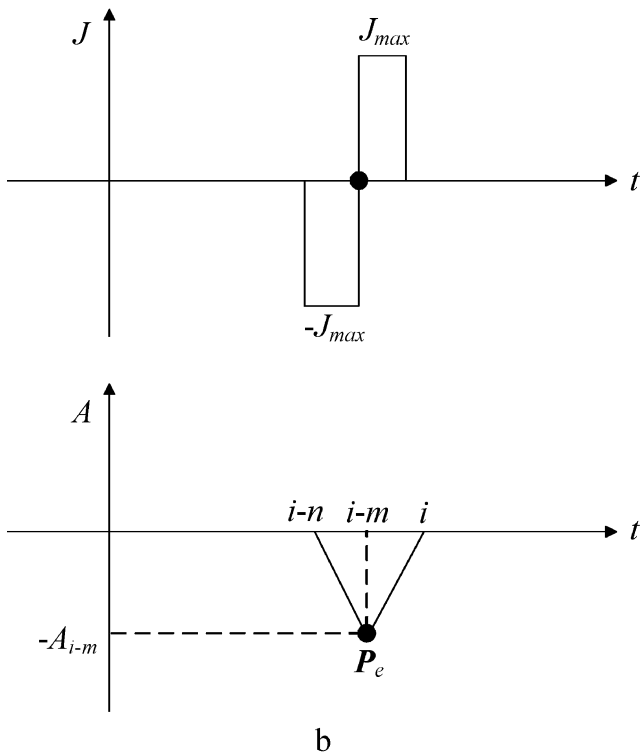
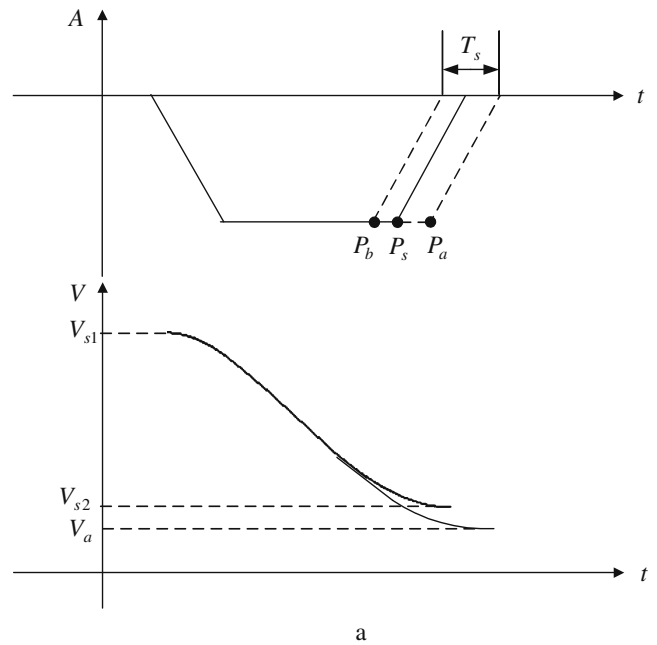
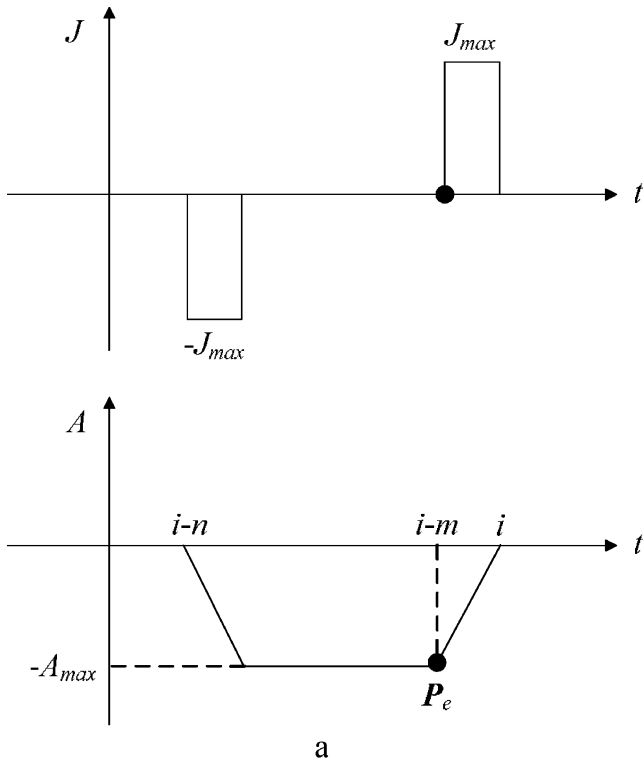
Fig. 6 The backtracking procedure



between two adjacent segment points in the acceleration profile may be null, triangular or trapezoidal. Once all of the segment points are identified and applied in the real-

time sampling stage, it is possible to design an appropriate acceleration profile for each interval divided by two segment points. The design of the acceleration profile in the real-time sampling will be discussed later.

The look-ahead procedure is depicted in Fig. 4. Consider that the  $(i-1)$ th sampling is known and the  $i$ th sampling



**Fig. 7** Acceleration and jerk profiles for backtracking. **a** Three-stage profile. **b** Two-stage profile

**Fig. 8** Inexact final position due to the constant sampling period. **a** Speed reducing. **b** Speed increasing

needs to be evaluated. First, a feedrate  $V_i$  is determined by Eq. 14. Since the variation in feedrate between two adjacent sampling points may be too large, it is necessary to check the acceleration and jerk in order to avoid the occurrence of saturation. The saturation of the acceleration is checked first. If the acceleration exceeds the saturation value, the feedrate in the  $i$ th sampling must be reduced. The acceleration  $A_i$  in the  $i$ th sampling can be expressed as  $A_i = \frac{V_i - V_{i-1}}{T_s}$ , where  $V_i$  and  $V_{i-1}$  are the feedrates in the  $i$ th and  $(i-1)$ th samplings, respectively. The following two rules are applied:

1. if  $A_i > A_{max}$ ,  $V'_i = V_{i-1} + A_{max} * T_s$
2. if  $A_i < -A_{max}$ ,  $V'_i = V_{i-1} - A_{max} * T_s$

where  $V'_i$  indicates the modified feedrate that should be specified in order to restrict the maximum value of the acceleration. It is noted, however, that  $V'_i$  may not always be smaller than  $V_i$ . In particular, when  $A_i$  is positive,  $V'_i$  is smaller than  $V_i$ , whereas  $V'_i$  is larger than  $V_i$  when  $A_i$  is negative.

The saturation of the jerk is checked next. If the jerk exceeds the saturation value, the acceleration in the  $i$ th sampling must be reduced. The jerk  $J_i$  in the  $i$ th sampling

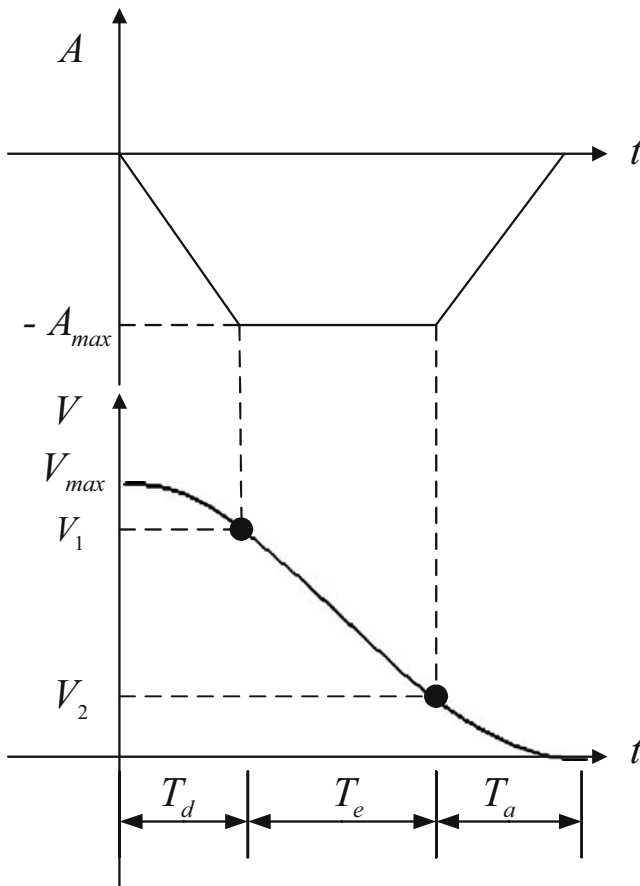


Fig. 9 Kinematic profiles for reducing the feedrate from  $V_{max}$  to 0

can be expressed as  $J_i = \frac{A_i - A_{i-1}}{T_s}$ , where  $A_i$  and  $A_{i-1}$  are the accelerations in the  $i$ th and  $(i-1)$ th samplings, respectively. The following two rules are applied:

1. if  $J_i > J_{max}$ ,  $A'_i = A_{i-1} + J_{max} * T_s$  and  $V'_i = V_{i-1} + A'_i * T_s$
2. if  $J_i < -J_{max}$ ,  $A'_i = A_{i-1} - J_{max} * T_s$  and  $V'_i = V_{i-1} - A'_i * T_s$

It is noted that, after the modification of the feedrate, due to the check of either the acceleration or the jerk, the modified feedrate  $V'_i$  may not be equal to the original feedrate  $V_i$ . If  $A_i$  is positive,  $V'_i < V_i$ , whereas if  $A_i$  is negative,  $V'_i > V_i$ . For  $V'_i < V_i$  as shown in Fig. 5a, the next sampling point will be located at  $P'_i$ , rather than  $P_i$ . The actual chord error is, thus, reduced because  $P'_i$  moves closer to  $P_{i-1}$ . On the contrary, for  $V'_i > V_i$  as shown in Fig. 5b, the actual chord error is increased because the next sampling point  $P'_i$  moves further away from  $P_{i-1}$ .

To overcome such a problem, it is necessary to apply a backtracking procedure, as shown in Fig. 6, to repeatedly check the feedrate in each sampling during the deceleration stage. It goes back to the  $(i-n)$ th point to re-compute the feedrate with a negative jerk by starting with  $n=2$ . When the deceleration reaches the minimum value  $-A_{max}$ , the deceleration must be kept constant. Therefore, there are two kinds of deceleration profiles: trapezoidal and triangular, as explained below (see Fig. 7):

1. When there is a significant difference between  $V_{i-n}$  and  $V_i$ , the acceleration and jerk profiles as shown in Fig. 7a are applied. It starts with a negative jerk, then zero jerk and ends in a positive jerk.
2. When the difference between  $V_{i-n}$  and  $V_i$  is small, the acceleration and jerk profiles as shown in Fig. 7b are

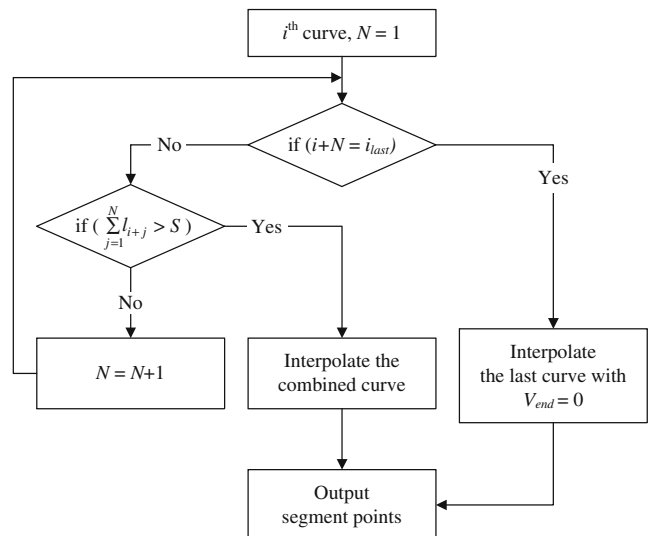


Fig. 10 The preview strategy of curves



applied. It starts with a negative jerk and changes to a positive jerk before the deceleration reaches its minimum value.

For both cases, it is necessary to determine a point  $P_e$  corresponding to the  $(i-m)$ th point in Fig. 7, which represents the starting point of the final step in the acceleration profile. The known parameters at this point are the current feedrate  $V_{i-m}$ , the current deceleration  $A_{i-m}$  and the desired feedrate  $V_i$ . If a maximum jerk and the transition in the acceleration from  $-A_{i-m}$  to zero are applied, the variation in feedrate will be:

$$\Delta V_a = \frac{-A_{i-m}^2}{2J_{max}} \tag{15}$$

Since it is in the decelerating stage,  $\Delta V_a$  should be smaller than zero. If  $V_i - V_{i-m} < \Delta V_a$ , the feedrate is still less than  $V_i$  when the acceleration reaches zero. Therefore, the  $(i-m)$ th point can be considered as the starting point of the final step in the acceleration profile. If the acceleration reaches zero at  $P_i$  and the new speed at  $u_i$  is less than  $V_i$ , the backtracking procedure is stopped. On the contrary, if the distance for the decelerating is not long enough, the feedrate at  $P_i$  will still be larger than  $V_i$ . Therefore, it is necessary to move backwards one more point by setting  $n=n+1$  and recompute the kinematic profiles again. Such a procedure is repeated until the speed at  $P_i$  is equal to or less than  $V_i$ .

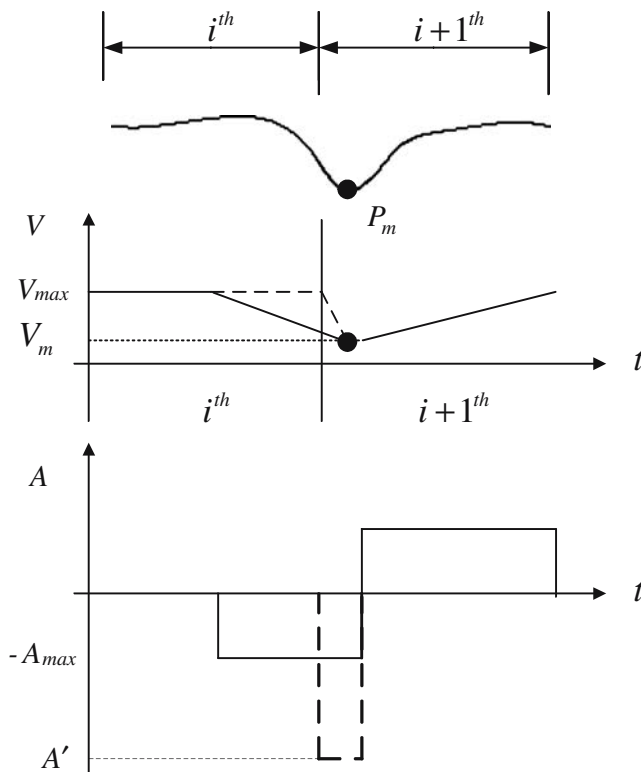


Fig. 11 Kinematic profiles for the preview of two curves

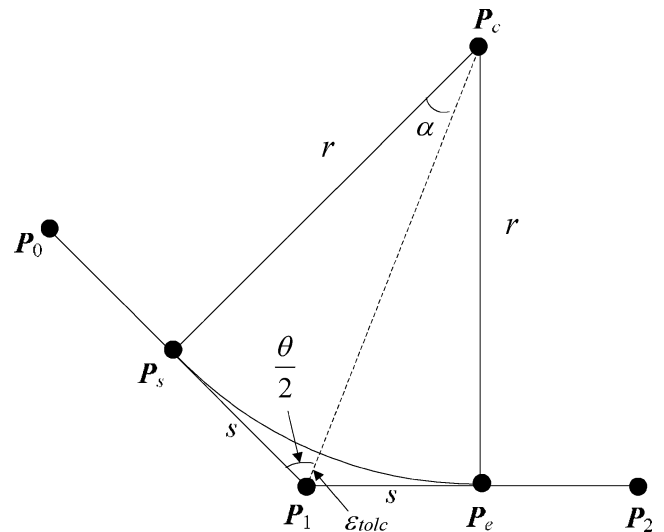


Fig. 12 An arc is generated to blend two curves of G0 continuity

### 5 The real-time sampling

Once the look-ahead stage is implemented, the segment points and the corresponding feedrate for each of them in the acceleration profile can be evaluated. In the real-time sampling, the kinematic profiles in each interval of two adjacent segment points are designed individually. Consider the interval between the segment points  $P_{s1}$  and  $P_{s2}$  as an example. Since there are two types of acceleration profiles, triangular and trapezoidal, it is necessary to decide which one should be applied. Assume that the difference of the feedrates between  $P_{s1}$  and  $P_{s2}$  is  $\Delta V_s = V_{s2} - V_{s1}$ , where  $V_{s1}$  and  $V_{s2}$  are the desired feedrates of  $P_{s1}$  and  $P_{s2}$ , respectively. When accelerating from 0 to  $A_{max}$  with maximum jerk  $J_{max}$ , the maximum variation of the feedrate  $\Delta V_{max}$  is:

$$\Delta V_{max} = \frac{A_{max}^2}{2J_{max}} \tag{16}$$

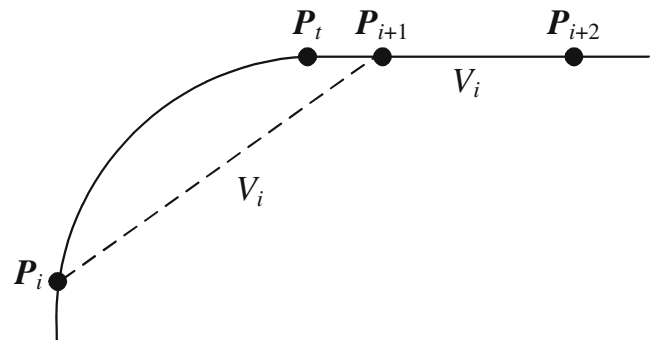


Fig. 13 Discontinuity in curvature between two curves

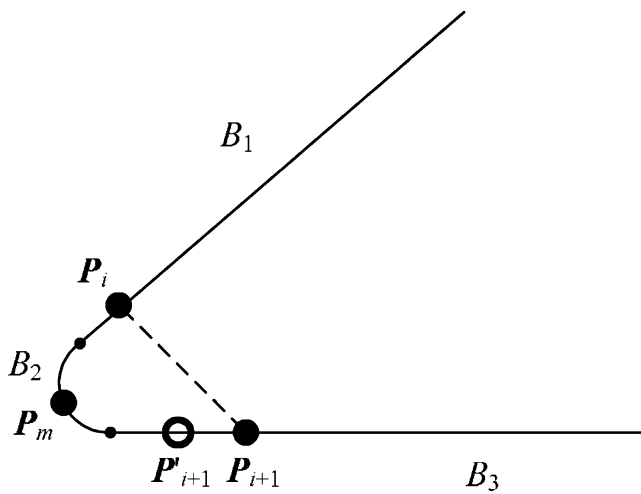


Fig. 14 A point  $P_m$  is generated in a short span

Thus, the acceleration profile based on the difference between  $\Delta V_s$  and  $2\Delta V_{max}$  can be determined as follows:

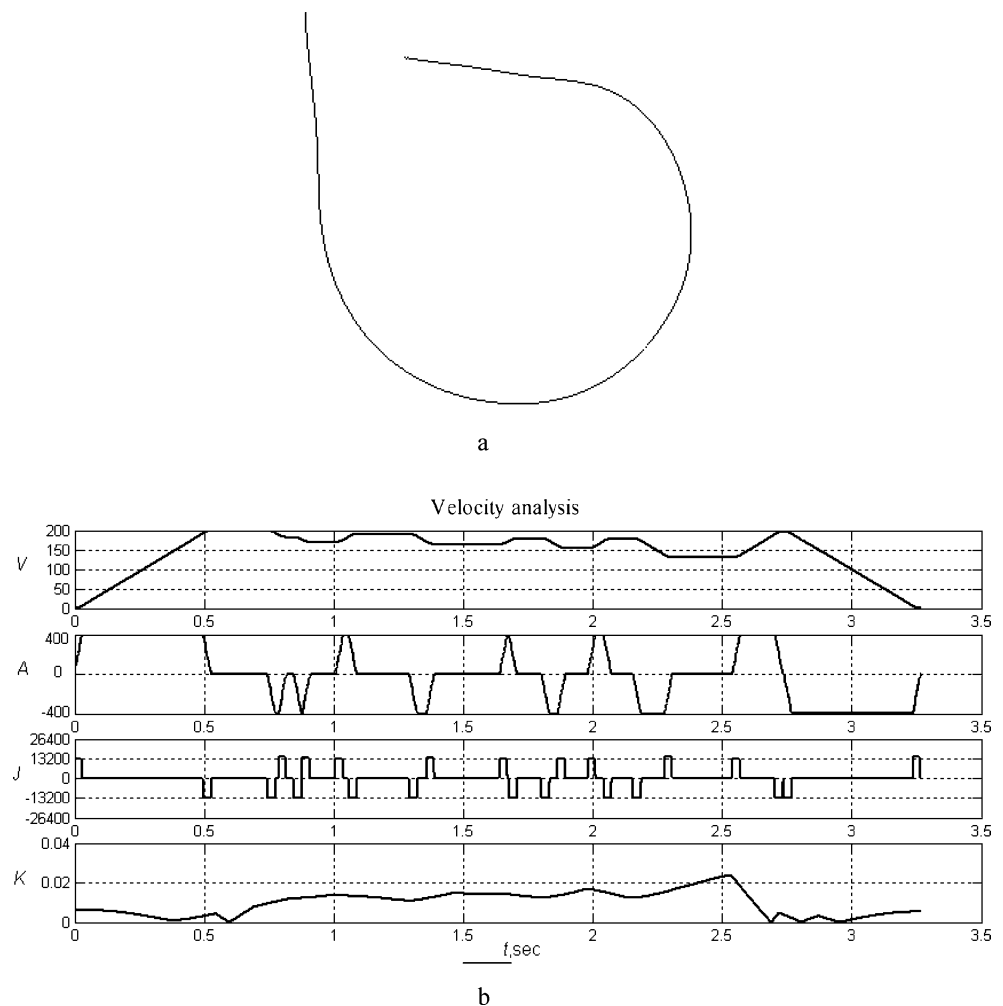
1. If  $\Delta V_s > 2\Delta V_{max}$ , then the variation of the feedrate in this block is large enough so that a three-stage jerk

profile can be applied, as shown in Fig. 7a, for the motion design. When an increase in feedrate is necessary, a positive jerk  $J_{max}$  is applied to accelerate from 0 to  $A_{max}$ , a zero jerk is then applied to retain the acceleration at  $A_{max}$  and finally a negative jerk  $-J_{max}$  is applied to decelerate from  $A_{max}$  to 0. When a decrease in speed is necessary, a similar procedure can be applied to decelerate from 0 to  $-A_{max}$ , retain the deceleration at  $-A_{max}$  for a period and finally accelerate from  $-A_{max}$  to 0. For both cases, the acceleration plot is, essentially, a trapezoid.

2. If  $\Delta V_s < 2\Delta V_{max}$ , the variation of the feedrate in this block is very small and a two-stage jerk profile is applied, as shown in Fig. 7b, for the motion design. It is noted that the extreme value of the acceleration in both cases will not attain the maximum value  $A_{max}$  or the minimum value  $-A_{max}$ .

The detailed procedure in the real-time sampling is discussed below. Consider that it starts from a point  $P_i, i=0$ , where  $P_0$  is the final point of the previous interval (between two segment points). The feedrate of  $P_0$  is denoted as  $V_0$ .

Fig. 15 An example of a single curve. a The curve. b Kinematic profiles



The acceleration profile as shown in Fig. 7 based on the difference between  $\Delta V_s$  and  $2\Delta V_{\max}$  is selected first. The next sampling point  $\mathbf{P}_{i, i=1}$  is then determined by Eq. 2. Since  $\mathbf{P}_{s1}$  and  $\mathbf{P}_{s2}$  have been determined in terms of the chord error, the acceleration limit and the jerk limit in the look-ahead stage, it is unnecessary to apply Eq. 14 because the maximum acceleration and jerk are already within the limits. The index  $i$  is then increased by 1 in each sampling to compute a new sampling point until the termination condition is reached.

The termination condition is discussed in the following. Figure 8a shows the sampling conditions near  $\mathbf{P}_{s2}$  when the speed near the final point is decreasing. Assume that if the acceleration is increased from  $\mathbf{P}_s$  with a jerk  $J_{\max}$ , it can reach  $\mathbf{P}_{s2}$  with the desired speed  $V_{s2}$ . However, due to the limitation of the constant sampling period, the acceleration can be increased either at  $\mathbf{P}_a$  or  $\mathbf{P}_b$  only. Here,  $\mathbf{P}_a$  is chosen because it can make the final speed  $V_a$  lower than the desired speed  $V_{s2}$ , which is better in terms of the machine dynamics. On the contrary, Fig. 8b depicts the final conditions when the speed near the final point is increasing. In this case,  $\mathbf{P}_b$  is a better position for starting to decelerate, as it can yield a speed  $V_b$  lower than the desired speed  $V_{s2}$ . For both cases, when the final speed is reached, the actual distance of the motion is slightly shorter than that of the look-ahead stage and, hence, the final position  $\mathbf{P}_{s2}$  is still not attained. Thus, it is necessary to retain a constant speed  $V_a$  (or  $V_b$ ) until the actual position exceeds the final point

$\mathbf{P}_{s2}$ . This is the initial position for the sampling of the next interval.

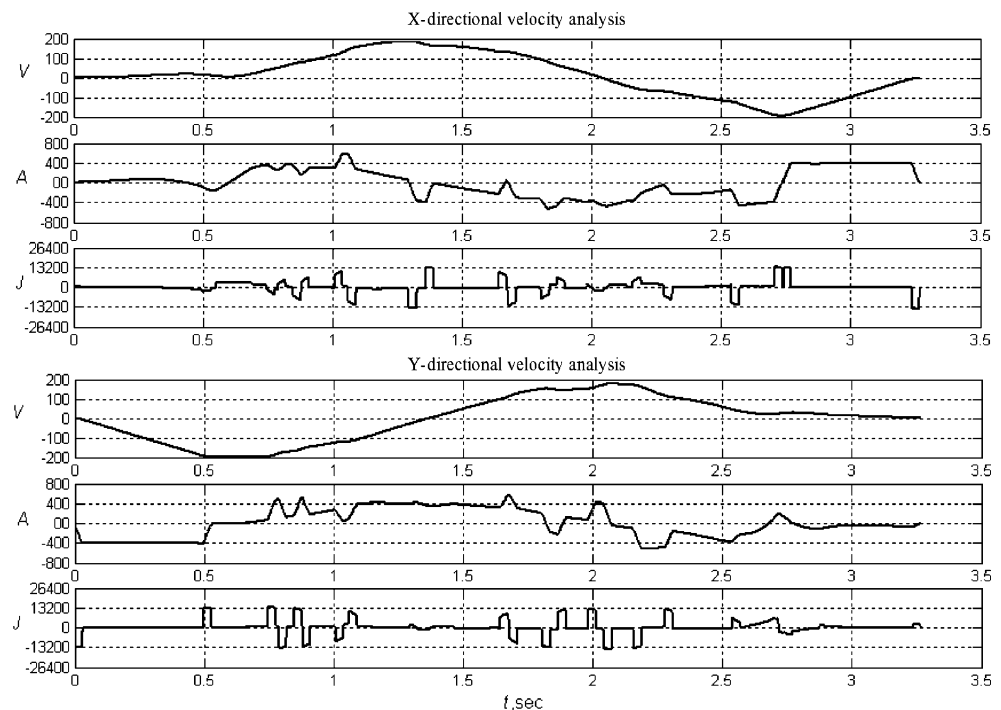
## 6 Multiple curves sampling

Additional issues must be addressed when multiple curves are sampled. They are classified as the following four types: (1) determining the number of curves for preview; (2) discontinuity in slope at the connecting point; (3) discontinuity in curvature; (4) effect of short knot spans. Each of the problems and the associated algorithm are discussed below.

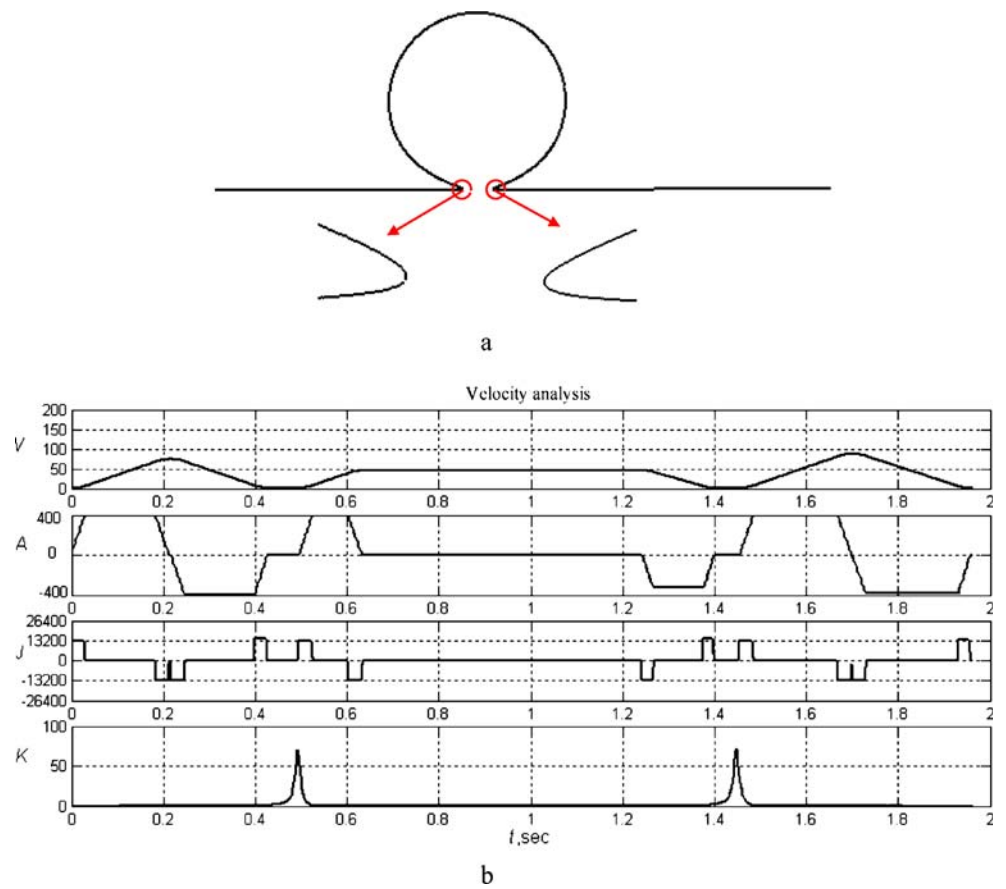
### 6.1 Determining the number of curves for preview

A preview of the curves is necessary for multiple curves sampling. When many short curves are connected to each other or when a long and flat curve is followed by a short one, the feedrate must be reduced in advance. The length of the curves for preview must be long enough such that the speed profile can be predicted accurately. Figure 9 depicts the kinematic profiles for the worst case of speed reduction, where the feedrate is reduced from  $V_{\max}$  to 0. The area covered by the feedrate profile represents the total length of the curves that should be previewed in advance; the total length  $S$  is composed of three lengths  $S_d$ ,  $S_c$  and  $S_a$ , representing the decelerating stage, the constant accelera-

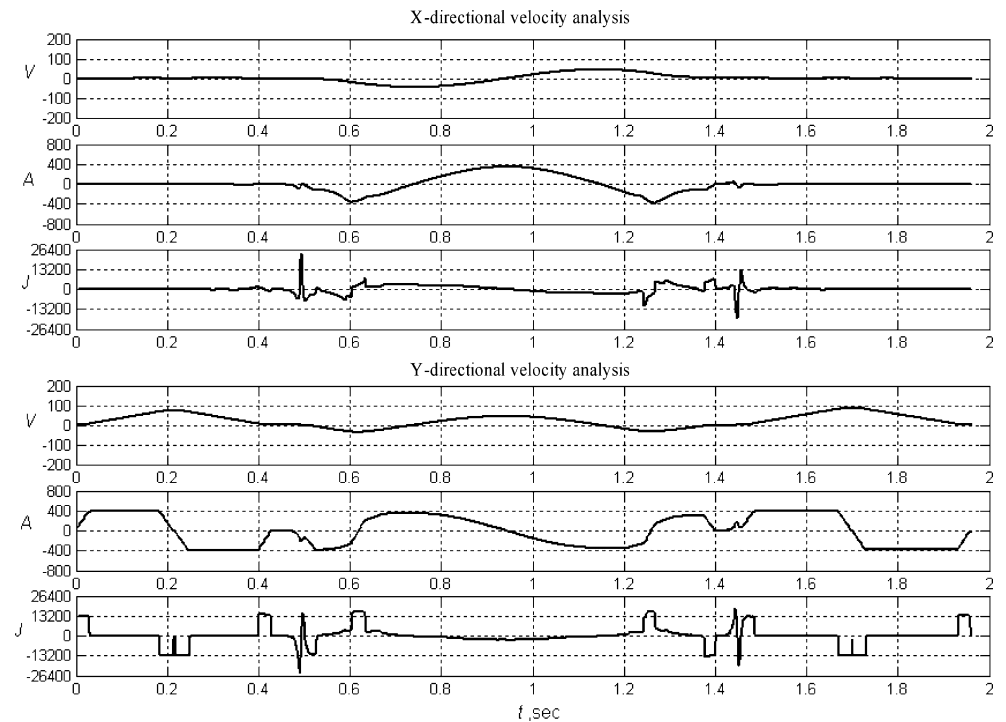
**Fig. 16** Speed, acceleration and jerk profiles along the  $x$  and  $y$  directions for the example in Fig. 15



**Fig. 17** An example of a single curve with substantial changes in curvature. **a** The curve. **b** Kinematic profiles



**Fig. 18** Speed, acceleration and jerk profiles along the  $x$  and  $y$  directions for the example in Fig. 17



tion stage and the accelerating stage, respectively. Here,  $S$  can be expressed as follows:

$$S = S_d + S_e + S_a \tag{17}$$

$$S_d = V_{\max} T_d - \frac{1}{6} J_{\max} T_d^3 \tag{18}$$

$$S_e = V_1 T_e - \frac{1}{2} A_{\max} T_e^2 \tag{19}$$

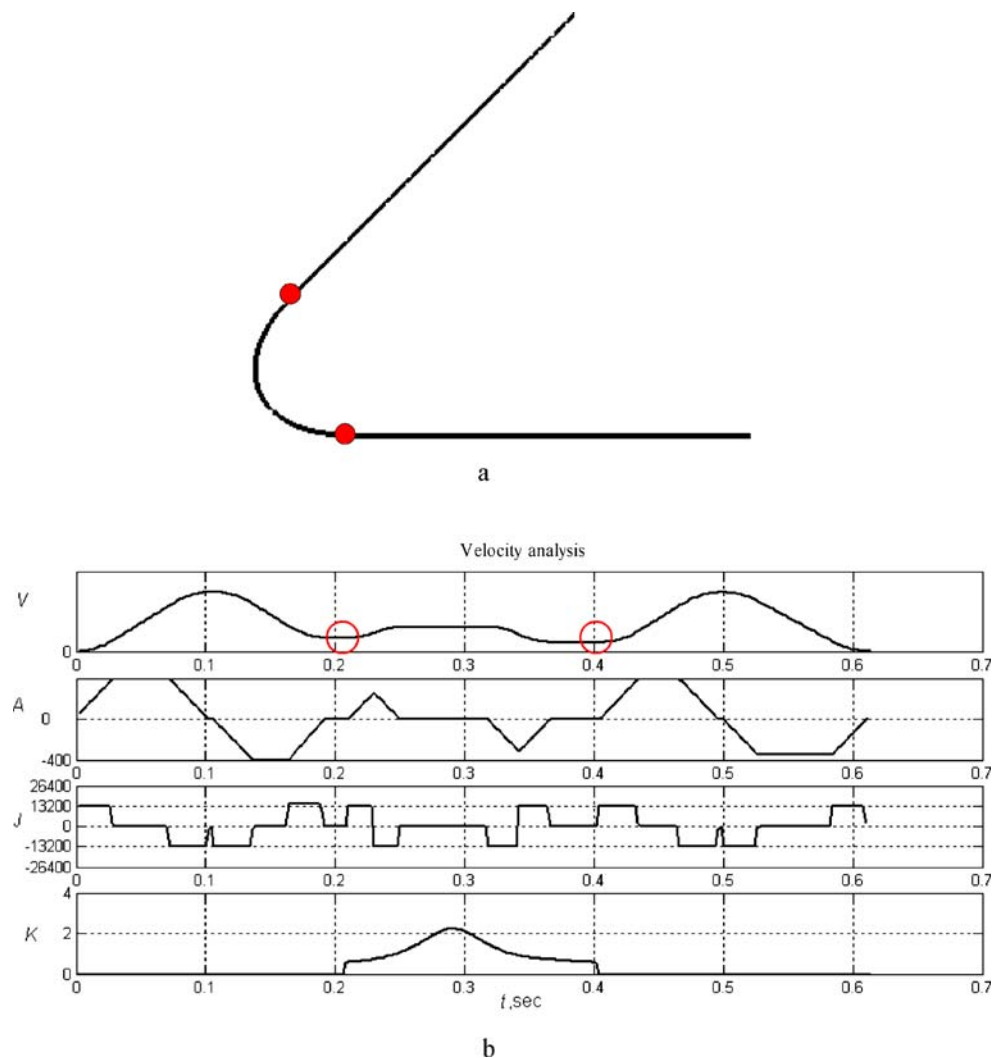
$$S_a = \frac{1}{6} J_{\max} T_a^3 \tag{20}$$

A procedure for the detection of the number of curves for preview is depicted in Fig. 10. Starting from  $N=1$ , it computes the length,  $l_{i+N}$ , of the  $(i+N)$ th curve upon

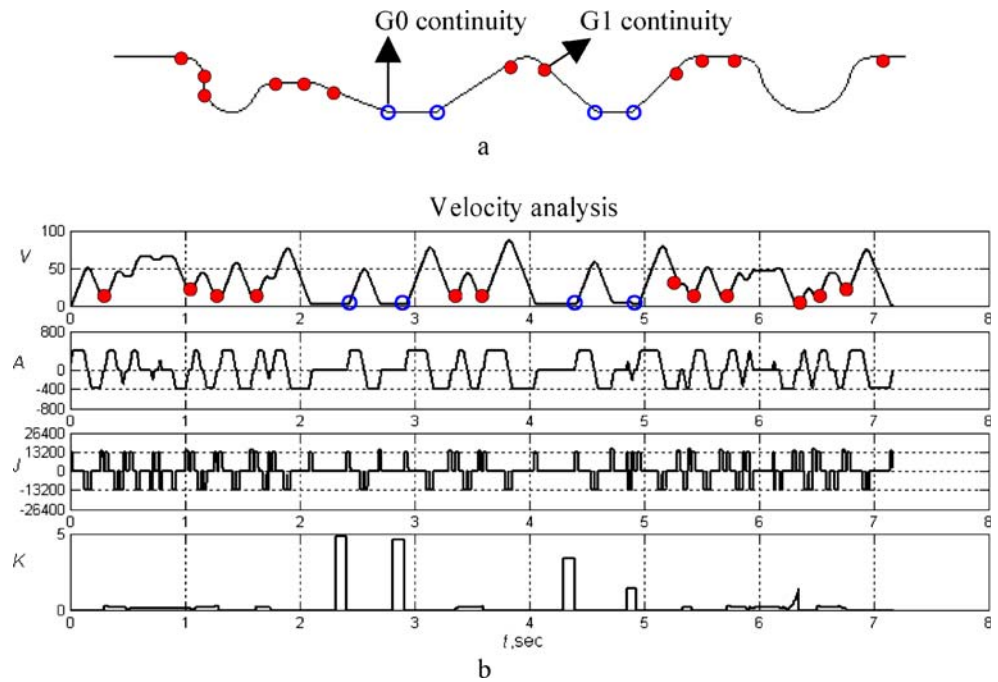
considering the  $i$ th curve. If  $\sum_{j=1}^N l_{i+j} > S$ , then the total distance is long enough for the design of the feedrate profile. Hence, the  $i$ th and  $(i+N)$ th curves are merged into one curve. The pre-sampling process can then be implemented to evaluate the segment points for the  $i$ th curve.

If  $\sum_{j=1}^N l_{i+j} < S$ , the parameter  $N$  is increased by 1 to include one more curve and computes the total length of  $N$  curves ahead. It repeats the same process of comparing the total length  $\sum_{j=1}^N l_{i+j}$  with  $S$ . The above procedure is carried out repeatedly until the  $(i+N)$  curve becomes the final curve. When the final curve is reached, its final speed will be set as 0. Figure 11 depicts an example with two curves to illustrate the above idea, where the curvature of the second curve is larger than that of the first. Here, the maximum feedrate  $V_{\max}$  is applied for the first curve as it is smooth. Since the curvature at  $\mathbf{P}_m$  is increased, the feedrate must be

**Fig. 19** A curve of G0 continuity in two points. **a** The curve. **b** Kinematic profiles



**Fig. 20** A group of curves with G0 and G1 continuity. **a** The curves. **b** Kinematic profiles



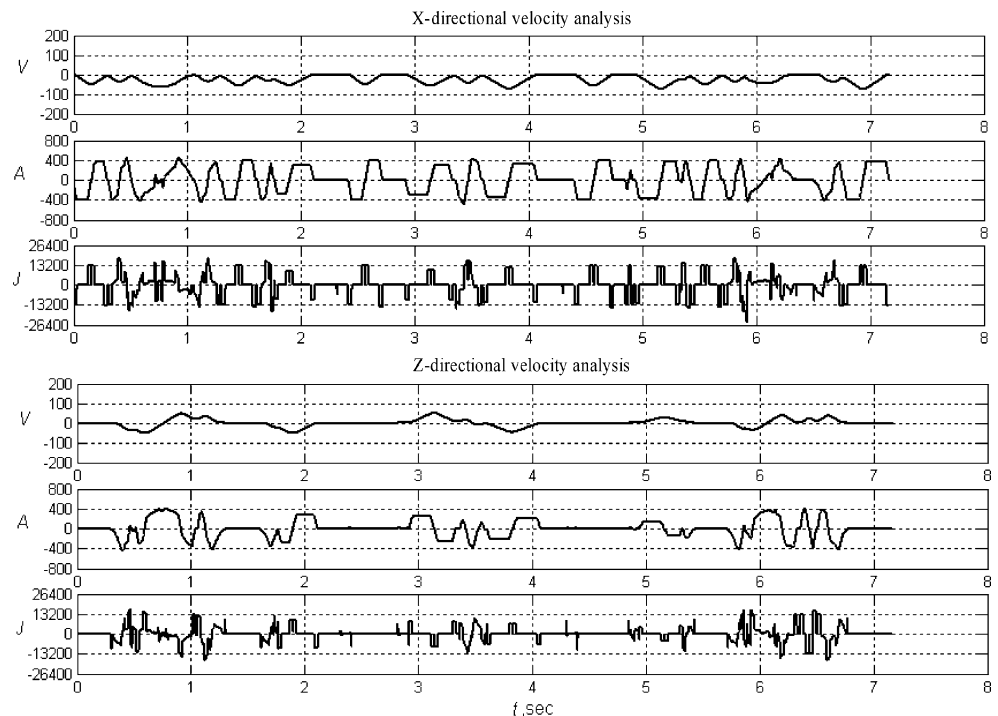
reduced before it reaches  $P_m$ . If there is no such a preview process, the maximum acceleration will be  $A'$ , as in Fig. 11, rather than  $A_{max}$ . Here  $A'$  is obviously larger than  $A_{max}$  and it is not accepted.

### 6.2 Discontinuity in slope at the connecting point

When the slope at the connecting point of two curves is discontinuous, the acceleration in each axis becomes

discontinuous also. To reach the continuity in acceleration at the corner of the connecting point, the feedrate in one of the axes may approach zero, which is practically inefficient. In addition, the instantaneous acceleration in one of the axes may increase drastically, which can excite substantial machine vibration. For such a case, it is difficult to follow the sharp corner precisely because the feedrate at the corner must be reduced to zero. Instead of doing this, a round corner is proposed to approximate the original shape, which

**Fig. 21** Speed, acceleration and jerk profiles along the x and z directions for the example in Fig. 20



can hold a minimum feedrate at the corner. It applies a blending curve to smoothly connect both lines, while maintaining the maximum chord error within the allowable one. There are several kinds of blending curves which can achieve the above task. Here, an arc is used for the blending curve, as depicted in Fig. 12, where the arc is tangential to both lines. The problem here is to compute a distance  $s$  on both sides of the corner point  $\mathbf{P}_1$ , and, hence, the points  $\mathbf{P}_s$  and  $\mathbf{P}_e$  can be evaluated, which act as the marginal points of the arc. Assume that the angle between both lines is  $\theta$ . Here,  $s$  can be expressed as:

$$s = r \tan \alpha \quad \alpha = \frac{\pi - \theta}{2} \tag{21}$$

$$r = \frac{\varepsilon_{\text{tolc}}}{\left(\frac{1}{\cos \alpha}\right) - 1} \tag{22}$$

where  $\varepsilon_{\text{tolc}}$  is the allowable chord error at the corner. It is noted that the chord error at the corner should be larger than that at other places. If the allowable chord error at the corner is set as large as that at other places, the length of the arc would become too small to be machined correctly. Once  $\mathbf{P}_s$  and  $\mathbf{P}_e$  are obtained,  $\mathbf{P}_s$ ,  $\mathbf{P}_1$  and  $\mathbf{P}_e$  can be used to generate an arc, which is a B-spline curve and is tangential to both lines. Thus, the two original lines  $\mathbf{P}_0\mathbf{P}_1$  and  $\mathbf{P}_1\mathbf{P}_2$  become lines  $\mathbf{P}_0\mathbf{P}_s$  and  $\mathbf{P}_e\mathbf{P}_2$ , and an arc  $\mathbf{P}_s\mathbf{P}_1\mathbf{P}_e$ . Then, the composition principle of B-spline curves [11] can be applied to combine all three curves into a composite curve. The composite curve is still a NURBS curve. Hence, the proposed parametric interpolator can be applied.

### 6.3 Discontinuity in curvature

When there is a discontinuity in the curvature, the centrifugal acceleration will become discontinuous at the discontinuous point, which may result in a high jerk that is beyond the saturated value. This situation may happen at the connecting point of two curves or within a curve. As Fig. 13 shows,  $\mathbf{P}_t$  is the transition point of a circle and a line of the discontinuous curvature. The centrifugal acceleration exists before  $\mathbf{P}_t$  but it disappears after  $\mathbf{P}_t$ . Such a variation will result in a sudden change in jerk at  $\mathbf{P}_t$ . The speed at  $\mathbf{P}_t$  must be reduced in order to yield a maximum jerk that is within the saturation value.

To determine the allowable feedrate near  $\mathbf{P}_t$ , it is necessary to choose the larger curvature from both sides of  $\mathbf{P}_t$ , because the larger the curvature is, the smaller the allowable feedrate should be. For example, in Fig. 13, the curvature at  $\mathbf{P}_t$  is determined based on the circle, instead of the line. Let the radius of the circle be  $r_i$  and the allowable feedrate be  $V_i$ . The maximum centrifugal acceleration at this point is  $J_{\text{max}}T_s$ . In order to reduce the centrifugal accel-

eration from  $J_{\text{max}}T_s$  to 0 within one sampling period, the allowable feedrate  $V_i$  at  $u_i$  can be designed as:

$$V_i = \sqrt{J_{\text{max}} \cdot T_s \cdot r_i} \tag{23}$$

### 6.4 Effect of short knot spans

Each knot span in a NURBS curve essentially represents a curve segment in a polynomial form. When a knot span is too small and the speed is too fast, no point in such a span could be sampled. When such a knot span occurs near a sharp region of the curve (the curve is still G2 continuity), it may result in shape distortion near such a region. As Fig. 14 shows, a small knot span  $B_2$  is between the two long spans  $B_1$  and  $B_3$ . Since  $B_1$  is a line, the speed in this region is very fast. However, since  $B_2$  is a small segment, the next sampling point  $\mathbf{P}_{i+1}$  is located on  $B_3$ , rather than  $B_2$ . Thus, the actual cutting path would pass  $\mathbf{P}_i$  and  $\mathbf{P}_{i+1}$ ,

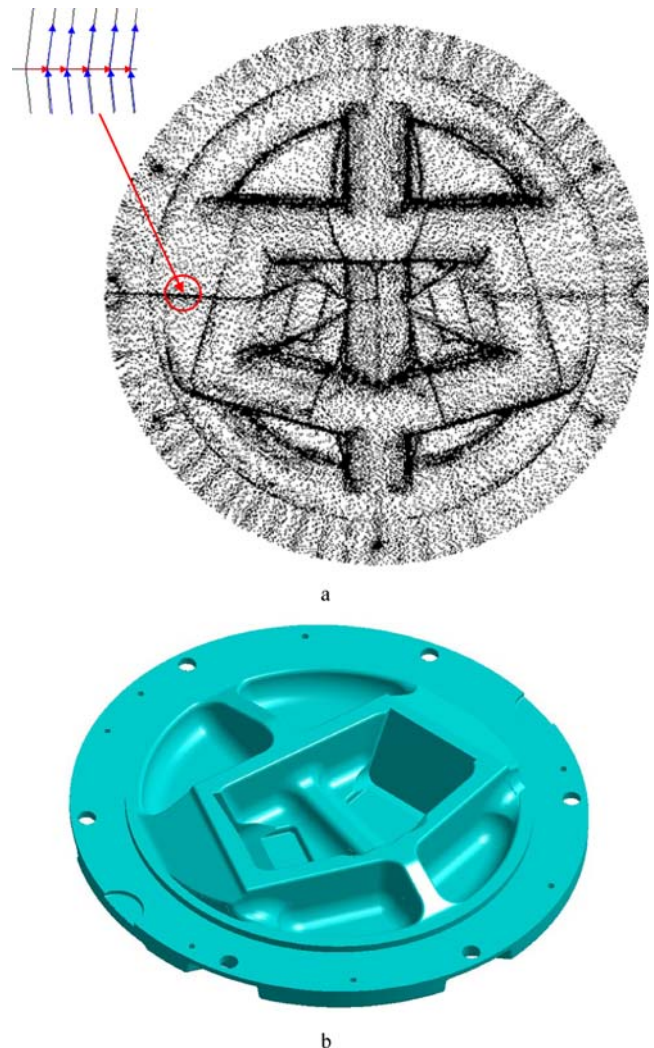
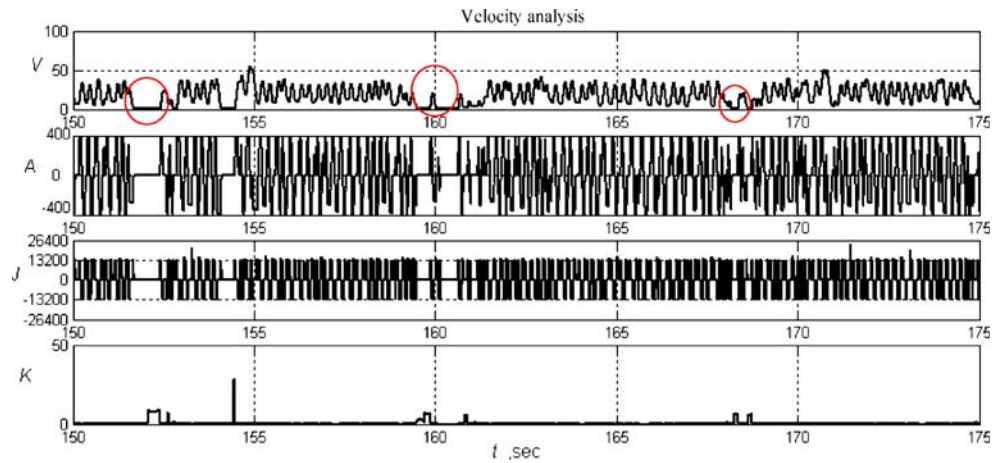


Fig. 22 Actual workpiece. a The cloud of the machining path. b The surface model

**Fig. 23** Speed, acceleration, jerk and curvature profiles for the example in Fig. 22



neglecting  $P_m$ , which is on  $B_2$ . To eliminate such a problem, the feedrate at  $P_i$  must be reduced such that the next sampling point would be located in the short knot span.


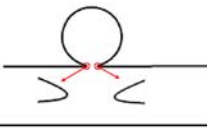



To avoid the problem mentioned above, it is necessary to detect if a knot span is neglected during the sampling process. A check is implemented to detect if any knot span is completely within  $(u_i, u_{i+1})$ . If yes, the short knot span exists. It is required to re-compute the feedrate at  $u_i$  and, hence,  $P_{i+1}$ . It first computes an intermediate point  $P_m$  corresponding to the mean parametric value of the short knot span. The feedrate  $V_m$  at this point is computed in terms of the curvature at  $P_m$ . If  $V_i > V_m$ , it means that the

feedrate at  $u_i$  is too fast and must be reduced to  $V_m$ . With such a reduced feedrate, the next sampling point would be  $P'_{i+1}$ , as shown in Fig. 14, instead of  $P_{i+1}$ .

### 7 Examples and discussion

Several examples are presented to demonstrate the feasibility of the proposed method. They are mainly classified into two types: single curve and multiple curves. For multiple curves, the following issues must further be considered: the

**Table 1** The CPU time and predicted machining time for the examples discussed in this paper

Profile	Total length (mm)	Number of sampling points	Predicted machining time (s)	CPU time (s)	
				Look-ahead stage	Real-time sampling
	480	1,636	3.272	0.235	0.093
	72.9	999	1.998	0.859	0.094
	8.5	308	0.616	0.11	0.016
	228.2	3,587	7.174	0.719	0.218
	59,929.6	2,808,691	5,617	1,105.83	184.65



continuity between the curves, the length of the curves for preview and the composition of the curves. The chord error, speed, acceleration, jerk and curvature for each example are displayed to illustrate the kinematic properties of the motion. The parameters used in these examples are: the order of the curves is 4, the sampling time is 0.002 s, the maximum chord error is 0.002 mm, the speed limit is 200 mm/s, the acceleration limit is 800 mm/s<sup>2</sup> and the jerk limit is 26,400 mm/s<sup>3</sup>. The computation is performed on a personal computer with an AMD-1.7G CPU with 512 MB RAM.

Figure 15a depicts the first example, which is a single curve with a length of 480 mm. The kinematic profiles of this example, as shown in Fig. 15b, indicate that the speed is almost maintained at its maximum value, namely 200 mm/s, except near the beginning and the end of the curve. Since the curve is smooth and long enough, the cutter can accelerate to the maximum speed and go through the curve at this speed. The small variation in speed is due to the variation of the curvature. Comparing the speed plot and the curvature plot, it shows that the speed decreases for large curvatures, whereas it increases for small curvatures, as expected. The kinematic profiles along the  $x$  and  $y$  axes are depicted in Fig. 16. The acceleration and jerk plots indicate that the maximum values of the acceleration and jerk are within the designed limits, namely 800 mm/s<sup>2</sup> and 26,400 mm/s<sup>3</sup>, respectively.

Figure 17a shows the second curve, with substantial changes in curvature for the two regions. The kinematic profiles, as shown in Fig. 17b, indicate that the speed near the two such regions is kept very slow, so as to maintain the chord error within the allowable error limit. The curvature on the two such regions is very large, while the lengths of the curve before and after them are either too short or too curved. Therefore, the speed does not reach its maximum. The total length of this curve is 72.89 mm, with 50 control points. The kinematic profiles as shown in Fig. 17b indicate that the profiles of the acceleration and jerk are reasonable. The kinematic profiles along the  $x$  and  $y$  axes are depicted in Fig. 18. Although there are sudden changes in jerk near the two such regions, the maximum jerk is still within the saturated value.

The third example indicates a curve with G0 continuity in two points, as Fig. 19a depicts, where the length of the curve is 8.49 mm. The discontinuity in the curvature may result in a sudden change in acceleration and jerk. Therefore, the speed must be reduced near those two discontinuous points, as shown on the speed plot in Fig. 19b, so as to control the maximum values of acceleration and jerk. The entire curve is not long enough, so the maximum speed is about 35 mm/s only. The kinematic profiles indicate that the acceleration and jerk profiles are acceptable, with maximum values within the saturated values.

Figure 20a depicts the fourth example composed of 17 curves, with a total length 228.2 mm, where the spots indicate the G1 continuities and the circles indicate the G0 continuities. A circle of 0.01 mm in radius is inserted between the curves of G0 continuity. Figure 20b shows that the curvature of the circle inserted is basically the largest, resulting in the slowest speed to pass such a region. The speed near G1 continuities must also be reduced due to the variation of the centrifugal acceleration. The kinematic profiles along the  $x$  and  $z$  axes of this example are shown in Fig. 21, where the path is located on the  $z$ - $x$  plane. Since the speed in each axis is affected not only by the variation of the curvature but also by the directional change, the maximum acceleration and jerk, therefore, may not occur at the points of G0 or G1 continuity. Figure 21 indicates that all of the maximum accelerations and jerks are within the saturated values.

Figure 22 depicts the numerical control (NC) paths of a real example, which has a size of 91.5 mm×91.5 mm×15 mm and contains 8,168 curves. It takes the program 52.75 s to convert all G0 continuities into G1 continuities. The computational time required, including the look-ahead stage and the real-time sampling, is 1,290 s. The number of sampling points generated is 2,808,691. The predicted machining time is 5,617 s. Therefore, the total computational time is still less than the machining time. Figure 23 depicts part of the kinematic profiles from 150 to 175 s. The circles marked on the speed plot represent the changes of the machining paths, where the speed is kept slow to attain the final position accurately.

The speed is, essentially, determined based on the curvature of the curve and the continuity conditions between the curves. When the continuity conditions between the curves is better, a higher speed can be applied to cross the common boundary of the curves. For the same shape of curves, a G2 continuity can offer a speed faster than a G1 continuity and a G1 continuity can offer a speed faster than a G0 continuity. Therefore, the pre-processing of the curves before parametric interpolators can affect the machining speed significantly. Table 1 lists the predicted machining time, the CPU time required in the look-ahead stage and that in the real-time sampling stage for all of the examples. The results indicate that the total time required for the look-ahead stage and the real-time sampling is always less than the predicted machining time. Thus, the proposed two-stage parametric interpolator is practically feasible for real-time implementation.

## 8 Conclusion

In this paper, a parametric interpolator has been proposed to control the speed, acceleration and jerk limits at the same

time, while maintaining the allowable chord error in the paths. It applied a look-ahead algorithm for the preview of the curves, in which several models were presented to determine the minimum feedrate in terms of different kinematic conditions. It demonstrated that both the vector form of the feedrate and its component along each coordinate direction must be considered simultaneously in order to yield maximum accelerations and jerks that are within the limits. A set of segment points was computed based on the acceleration profile, which was used in the real-time sampling to aid the determination of the kinematic profiles. With the segment points determined beforehand, the real-time sampling can focus on the generation of the sampling points in the interval of two segment points, without worrying about the conditions of the curves ahead. When the look-ahead stage was implemented on-line, a multi-tasking module must be developed, where the look-ahead algorithm was implemented backwards, while the real-time sampling was implemented forwards. The look-ahead stage must be implemented one block ahead, such that it can always output an enough amount of segment points for the use in the real-time sampling. Since the computational time required in analysing a curve in the look-ahead stage was less than that required in machining a curve, the proposed two-stage algorithm was practically feasible.

## References

1. Bedi S, Ali I, Quan N (1993) Advanced interpolation techniques for NC machines. *ASME J Eng Ind* 115(3):329–336
2. Shpitalni M, Koren Y, Lo CC (1994) Realtime curve interpolators. *CAD* 26(11):832–838
3. Yang DCH, Kong T (1994) Parametric interpolator versus linear interpolator for precision CNC machining. *CAD* 26(3):225–233
4. Tsai M-C, Cheng C-W, Cheng M-Y (2003) A real-time NURBS surface interpolator for precision three-axis CNC machining. *Int J Mach Tools Manuf* 43(12):1217–1227
5. Yeh S-S, Hsu P-L (1999) The speed-controlled interpolator for machining parametric curves. *CAD* 31(5):349–357
6. Yeh S-S, Hsu P-L (2002) Adaptive-feedrate interpolation for parametric curves with a confined chord error. *CAD* 34(3):229–237
7. Xu H-Y, Tam H-Y, Zhou Z, Tse PW (2001) Variable feedrate CNC interpolation for planar implicit curves. *Int J Adv Manuf Technol* 18(11):794–800
8. Liu A, Ahmad F, Yamazaki K, Mori M (2005) Adaptive interpolation scheme for NURBS curves with the integration of machining dynamics. *Int J Mach Tools Manuf* 45(4–5):433–444
9. Yong T, Narayanaswami R (2003) A parametric interpolator with confined chord errors, acceleration and deceleration for NC machining. *CAD* 35(13):1249–1259
10. Nam S-H, Yang M-Y (2004) A study on a generalized parametric interpolator with real-time jerk-limited acceleration. *CAD* 36(1): 27–36
11. Lee K (1999) *Principles of CAD/CAM/CAE systems*. Addison-Wesley, Reading, Massachusetts