

Research Article

Efficient Planning and Solving Algorithm of S-Shape Acceleration and Deceleration

Zhijie Li ^{1,2,3,4}, Ligang Cai ^{2,4} and Zhifeng Liu ^{1,2}

¹Institute of Advanced Manufacturing and Intelligent Technology, Beijing University of Technology, Beijing 100124, China

²Beijing Key Laboratory of Advanced Manufacturing Technology, Beijing University of Technology, Beijing 100124, China

³College of Mechanical and Electronics Engineering, North China Institute of Aerospace Engineering, Langfang 065600, China

⁴Machinery Industry Key Laboratory of Heavy Machine Tool Digital Design and Testing Technology, Beijing 100124, China

Correspondence should be addressed to Zhifeng Liu; lzf@bjut.edu.cn

Received 17 September 2020; Revised 5 October 2020; Accepted 6 November 2020; Published 26 November 2020

Academic Editor: Junwu Zhu

Copyright © 2020 Zhijie Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

S-shape acceleration and deceleration are the most widely used flexible acceleration and deceleration method in the current CNC system, but its velocity solution equation contains irrational terms, which create a more complicated solution process. When analyzing the solution process of the S-shape acceleration and deceleration directly, using a traditional numerical solution method, the phenomenon of “solving the interval jump” arises, which is the main reason for low efficiency and poor stability of the solution. According to the S-curve profile and solution, the concept of separating the curve profile recognition from the velocity solution was proposed, and a method of quickly identifying the interval of the solution location was introduced. Through the method mentioned above, the complete acceleration and deceleration curve parameters can be obtained through a one-time plan and a one-time solution, and the solution efficiency and stability are guaranteed; solving the Newton problem depends too much on the initial value of Newton velocity, which not only retains the speed advantage of the Newton method but also uses the downhill factor to ensure its convergence. Through the simulation comparison and analysis, the efficiency, stability, and universality of the method are verified.

1. Introduction

The ability of acceleration and deceleration is an important index in evaluating the performance of CNC systems, which directly affects the processing efficiency and quality of CNC equipment. High-speed and high-quality machining demands higher requirements for the acceleration and deceleration control abilities of CNC systems. The CNC equipment must be able to move from the current position point to the next position point quickly and accurately, while also ensuring the smooth operation of the equipment and avoiding impact, out of step, over the range, or oscillation.

The commonly used acceleration and deceleration algorithms in medium and low-speed equipment include trapezoids [1, 2] and exponential acceleration and deceleration algorithms [3, 4]. The two algorithms are simple in programming, good in real-time performance, and high

in processing efficiency, but they all have the problems of abrupt change and discontinuity of the acceleration curve, which makes them not suitable for high-speed and high-accuracy machining [5].

In high-speed equipment, flexible acceleration and deceleration algorithms are often used, and their acceleration and deceleration function expressions are continuous, to reduce the impact on mechanical components [6]. S-shape acceleration and deceleration algorithm is the most widely used in flexible acc/dec algorithms at present. This method is based on the jerk constraint. The curves of displacement, velocity, and acceleration are composed of cubic, quadratic, and linear functions, respectively, and all of them are continuous without deviation [7, 8].

The mathematical programming and solution of the S-shape acc/dec method have attracted considerable attention. The complete S-shaped acc/dec curve is composed of 7 segments, with many parameters. In most cases, the

velocity solution equation is nonlinear, which makes its solution complex. Altintas [9] presented a detailed seven segment model of S-shape acc/dec, without elaborating its efficient solution. Barre [10] studied the influence of the acc/dec control in a high-speed machining process and analyzed the control algorithm of the seven-segment S-shape acc/dec. Shi [11] summarized eight possible cases of S-shape acc/dec in the planning, according to the characteristics of the curve segment. Nam [12] used the binary iteration to find the numerical solution, but the calculation amount was uncertain, and the solution efficiency was not ideal. Yang [13] adopted the Newton iteration, but the preconditions and convergence of Newton iteration were not clarified.

Many scholars simplified the S-shaped acc/dec curve in order to reduce the complexity of the solution. Wang and others [14, 15] introduced a look-ahead control algorithm based on the S-shaped acc/dec curve with assuming that the velocity curve is symmetrical. In reference [16], the uniform acceleration and deceleration segments were removed, and the model was simplified to 5 segments. Although the above method reduced the complexity of the solution, it limited the dynamic performance of the machine tool. Xia [17] used the S-shape acc/dec based on the reference table to plan the velocity, which reduced the calculation time while limiting the flexibility of S-shape acc/dec.

Presently, the planning of NURBS curve interpolation velocity based on S-shape acc/dec has become a hot research topic, and the solution mainly adopts the matching displacement by changing the velocity step by step [18, 19]. The main idea of this method is to use the assumed key velocity value to calculate the corresponding displacement, match it with the actual displacement, and then determine the curve profile according to the matching situation. For small line segments, the calculation needs to be repeated more than three times, which reduces the efficiency of the solution.

In order to reduce the complexity in S-shaped acc/dec curve planning and solutions, some scholars introduced the artificial intelligence algorithms such as genetic algorithms [20] and particle swarm optimization algorithms [21]. Although the above artificial intelligence algorithm has strong robustness, it needs to calculate the fitness of each iteration. Compared with the traditional dichotomy and Newton iteration, it has a large amount of calculation and low competitiveness in real-time.

For S-shaped acc/dec curves, the more ideal solution method should have smaller calculation complexity and higher solution accuracy. However, there is no general perfect solution at present.

In this paper, the characteristics of the S-shaped acc/dec curve were analyzed. Focusing on the disadvantages of low efficiency and poor stability of traditional algorithms, a more concise, efficient, and stable general solution was proposed. In this method, the curve profile planning and velocity solution were completely separated. On the basis of proving the continuity of displacement's segmented function about velocity, the interval boundary

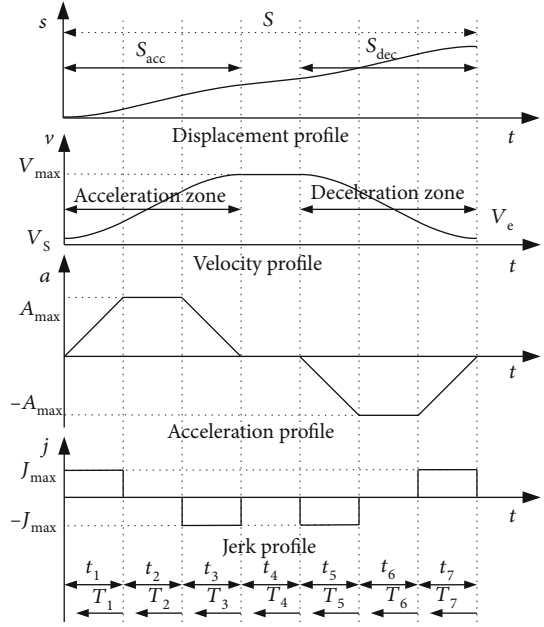


FIGURE 1: Profile of S-shape acceleration and deceleration.

points were used to analyze the interval of velocity, so as to complete the profile planning quickly. After obtaining the interval of the profile solution, the modified Newton method was introduced, which can make full use of the Newton method to accelerate up to the convergence speed, and at the same time, the downhill factor guarantees the convergence of the iteration.

2. Profile Characteristics of S-Shape Acceleration and Deceleration

The name of the S-acceleration and deceleration curves are derived from the fact that the acceleration curve is approximately S-shaped. This type of S-acceleration and deceleration processes includes the S-acceleration process, deceleration process, and the uniform connection between acceleration and deceleration. Acceleration processes includes the accelerated acceleration segment, uniform acceleration segment, and decelerated acceleration; the deceleration process includes the accelerated deceleration segment, uniform deceleration segment, and decelerated deceleration segment. There are 7 sections in total, as shown in Figure 1. The corresponding mathematical expression is as follows:

In the follow formula, $t_i (i = 1, 2, \dots, 7)$ is the running time of each segment. Command $T_1 = t_1$, $T_2 = t_1 + t_2$, and so on, $T_7 = t_1 + t_2 + t_3 + t_4 + t_5 + t_6 + t_7$, A is the maximum acceleration, J is the maximum jerk, V_s is the initial velocity, V_e is the final velocity, and V_{max} is the maximum velocity. According to the characteristics of the S-acceleration and deceleration curve, $t_1 = t_3$, $t_5 = t_7$.

$$J(t) = \begin{cases} J_1 = J & 0 \leq t < T_1 \\ J_2 = 0 & T_1 \leq t < T_2 \\ J_3 = -J & T_2 \leq t < T_3 \\ J_4 = 0 & T_3 \leq t < T_4 \\ J_5 = -J & T_4 \leq t < T_5 \\ J_6 = 0 & T_5 \leq t < T_6 \\ J_7 = J & T_6 \leq t < T_7 \end{cases},$$

$$a(t) = \begin{cases} a_1 = Jt & 0 \leq t < T_1 \\ a_2 = A & T_1 \leq t < T_2 \\ a_3 = A - J(t - T_2) & T_2 \leq t < T_3 \\ a_4 = 0 & T_3 \leq t < T_4 \\ a_5 = -J(t - T_4) & T_4 \leq t < T_5 \\ a_6 = -A & T_5 \leq t < T_6 \\ a_7 = -A + J(t - T_6) & T_6 \leq t < T_7 \end{cases},$$

$$v(t) = \begin{cases} v_1 = V_s + \frac{1}{2}Jt^2 \\ v_2 = V_1 + Jt_1(t - T_1) & V_1 = V_s + \frac{1}{2}JT_1^2 \\ v_3 = V_2 + Jt_1(t - T_2) - \frac{1}{2}J(t - T_2)^2 & V_2 = V_1 + Jt_1t_2 \\ v_4 = V_3 & V_3 = V_2 + \frac{1}{2}Jt_1^2 \\ v_5 = V_4 - \frac{1}{2}J(t - T_4)^2 & V_4 = V_3 \\ v_6 = V_5 - Jt_5(t - T_5) & V_5 = V_4 - \frac{1}{2}Jt_5^2 \\ v_7 = V_6 - Jt_5(t - T_6) + \frac{1}{2}J(t - T_6)^2 & V_6 = V_5 - Jt_5t_6 \end{cases}.$$

Displacement of the acceleration section:

$$S_{\text{acc}} = S_1 + S_2 + S_3 = (\nu + V_s)(2t_1 + t_2)/2. \quad (2)$$

Displacement of the deceleration section:

$$S_{\text{dec}} = S_5 + S_6 + S_7 = (\nu + V_e)(2t_5 + t_6)/2. \quad (3)$$

Total displacement:

$$S_T = S_{\text{acc}} + S_{\text{dec}} = (\nu + v_s)(2t_1 + t_2)/2 + (\nu + v_s)(2t_5 + t_6)/2. \quad (4)$$

If the actual displacement is $S > S_T$, then there is a uniform velocity segment and running time:

$$t_4 = \frac{S - S_T}{\nu}. \quad (5)$$

Whether there is a uniform acceleration section or a uniform deceleration section in the process of acceleration and deceleration is decided by V_s , V_e , ν , A , and J .

$$\nu - V_s = (t_1 + t_2)A, \nu - V_e = (t_5 + t_6)A, \quad (6)$$

$$t_1 = t_3 = \frac{A}{J}, t_2 = \frac{\nu - V_s}{A} - \frac{A}{J} \nu > V_s + \frac{A^2}{J}, \quad (7)$$

$$t_1 = t_3 = \sqrt{\frac{\nu - V_s}{J}}, t_2 = 0 \nu \leq V_s + \frac{A^2}{J}, \quad (8)$$

$$t_5 = t_7 = \frac{A}{J}, t_6 = \frac{\nu - V_e}{A} - \frac{A}{J} \nu > V_e + \frac{A^2}{J}, \quad (9)$$

$$t_5 = t_7 = \sqrt{\frac{\nu - V_e}{J}}, t_6 = 0 \nu \leq V_e + \frac{A^2}{J}. \quad (10)$$

Substituting the above time expression into the displacement formula:

- (1) While $\nu > V_s + (A^2/J)$ & $\nu > V_e + (A^2/J)$, then $S_T = (1/A)\nu^2 + (A/J)\nu - (V_s^2 + Ve^2/2A) + (V_s + V_e/2J)A$
- (2) While $\nu > V_s + (A^2/J)$ & $\nu \leq V_e + (A^2/J)$, then $S_T = (\nu + V_s/2)((A/J) + (\nu - V_s/A)) + (\nu + V_e)\sqrt{\nu - V_e/J}$
- (3) While $\nu \leq V_s + (A^2/J)$ & $\nu > V_e + (A^2/J)$, then $S_T = (\nu + V_e/2)((A/J) + (\nu - V_e/A)) + (\nu + V_s)\sqrt{\nu - V_s/J}$
- (4) While $\nu \leq V_s + (A^2/J)$ & $\nu \leq V_e + (A^2/J)$, then $S_T = (\nu + V_s)\sqrt{\nu - V_s/J} + (\nu + V_e)\sqrt{\nu - V_e/J}$

The expression (13) is a quadratic equation of one variable, and the exact solution can be obtained by using the root formula, while the expressions (14)-(16) contain the irrational part, so it is difficult to obtain the analytical solution.

3. Direct Numerical Solution

3.1. Dichotomy. Dichotomy is the most common method in solving nonlinear equations. Its main process in solving S-shape acceleration and deceleration is shown in Figure 2.

Step 1. input the initial velocity V_s , the final velocity V_e , maximum acceleration A , maximum jump A , required moving displacement S , and maximum limiting velocity V_{max} .

Step 2. compare the value of $V = V_{\text{max}}$, $V_s' = V_s + (A^2/J)$, and $V_e' = V_e + (A^2/J)$.

Step 3. call formulas (7)-(10) to calculate the time of each segment t_i and put them into formulas (13)-(16) to calculate the displacement S_T .

Step 4. if $S_T \leq S$, call formula (5) and obtain the running time of each segment and the overall curve profile. The calculation is finished.

If in step 4 $S_T > S$, the binary algorithm will be executed in step 5 to step 8:

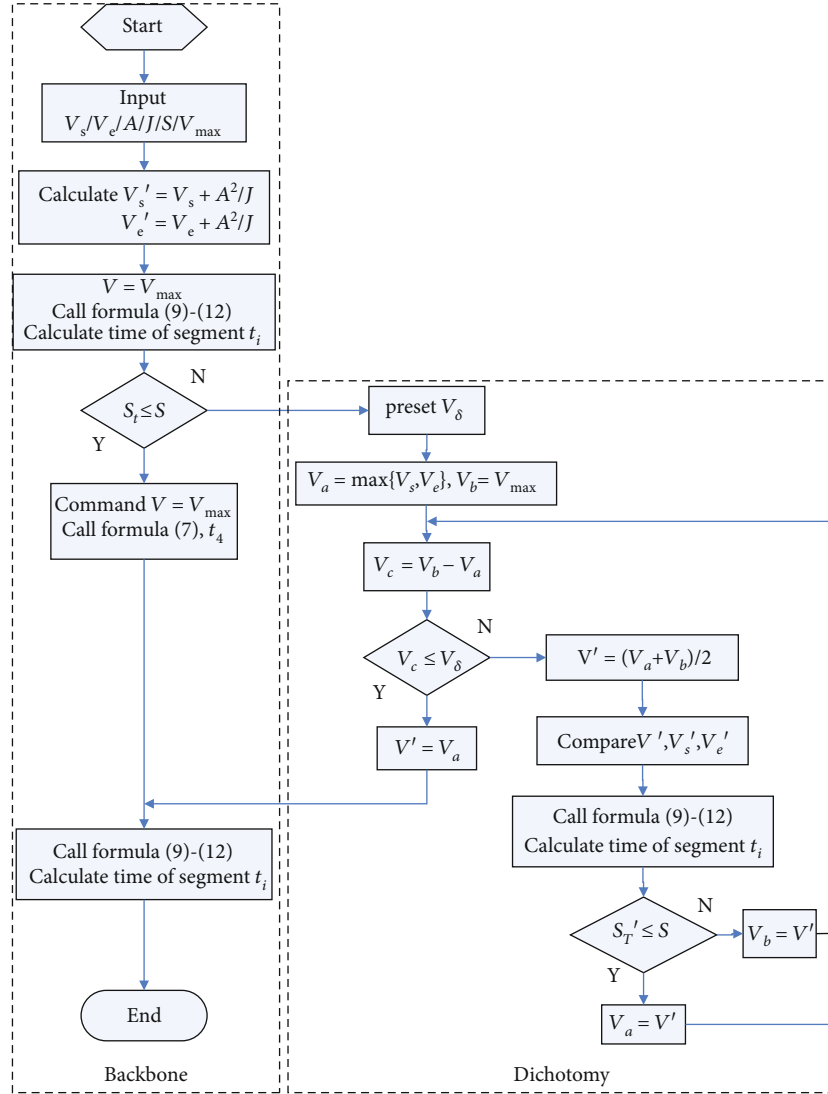


FIGURE 2: Solving the S-shape acceleration flow chart with dichotomy.

Step 5. take the initial interval $[V_a, V_b]$, where $V_a = \max\{V_s, V_e\}$ and $V_b = V_{\max}$ and set the velocity accuracy V_ε .

Step 6. determination of termination conditions: if $(V_b - V_a) \leq V_\varepsilon$, the calculation will be stopped.

Step 7. for iterative calculation, take $V' = (V_b + V_a)/2$, compare V' , $V_s' = V_s + (A^2/J)$, and $V_e' = V_e + (A^2/J)$, call formulas (7)-(10) to calculate the time of each segment, and substitute them into formulas (13)-(16) for calculation S_T . If $S_T \leq S$, then set $V_a = V'$; otherwise, set $V_b = V'$.

Step 8. turn to step 6.

3.2. Newton Iteration

3.2.1. Principle. For nonlinear equation $f(v) = S_T - S = 0$, V_0 is assumed to be an approximate solution of V^* ; $f(v)$ is expanded in the Taylor form at V_0 , and the following formula is obtained:

$$f(v) \approx f(V_0) + f'(V_0)(v - V_0) = 0. \quad (11)$$

$V_1 = V_0 - f(V_0)/f'(V_0)$ $f'(V_0) \neq 0$ is obtained as the first approximation of V^* ; V_k is assumed as the current point, and $f(v)$ can be expanded in the Taylor form at V_k :

$$f(v) \approx f(V_k) + f'(V_k)(v - V_k) = 0. \quad (12)$$

Then,

$$V_{k+1} = V_k - f(V_k)/f'(V_k) \quad f'(V_k) \neq 0, k = 0, 1 \dots \quad (13)$$

3.2.2. Iteration Process. The first four steps are the same as the first four steps of dichotomy, so it will not be described here. Start with step 5:

Step 5. V_0 is taken as the initial value, N is the maximum number of iterations, and V_ε is taken as the accuracy requirements.

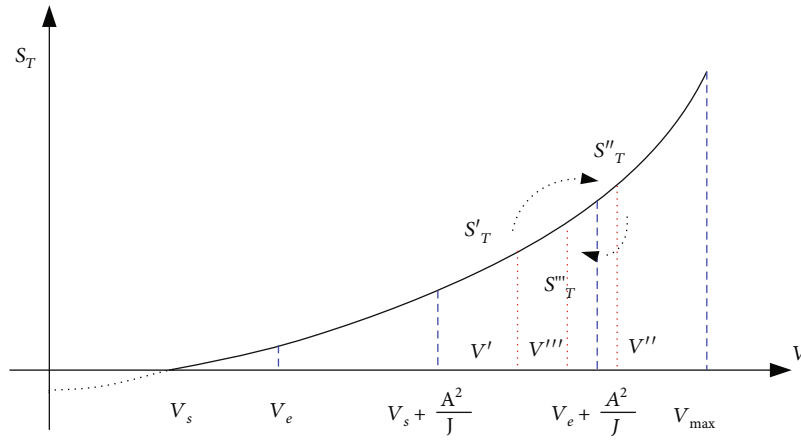


FIGURE 3: Schematic diagram of interval jump in the solving process.

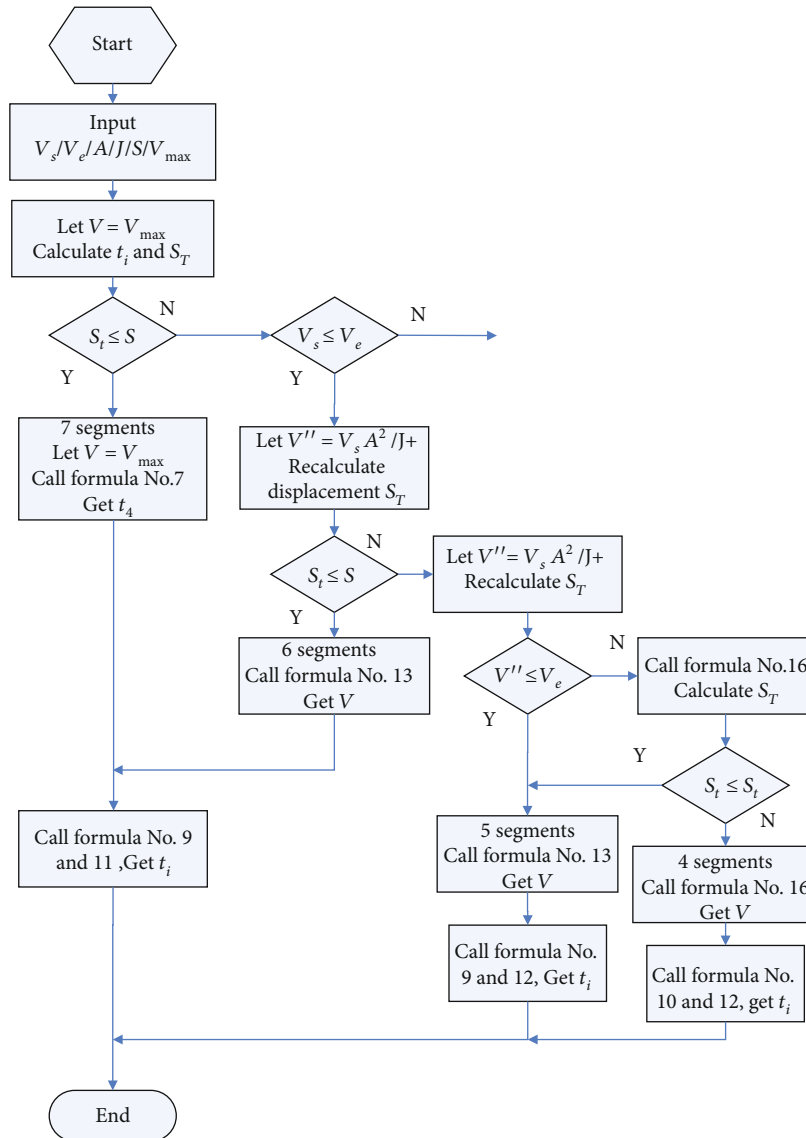


FIGURE 4: Flow chart of solution by matching with velocity step by step.

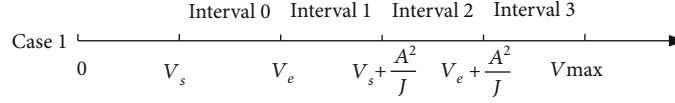


FIGURE 5: Case 1 with initial velocity less than the final velocity.

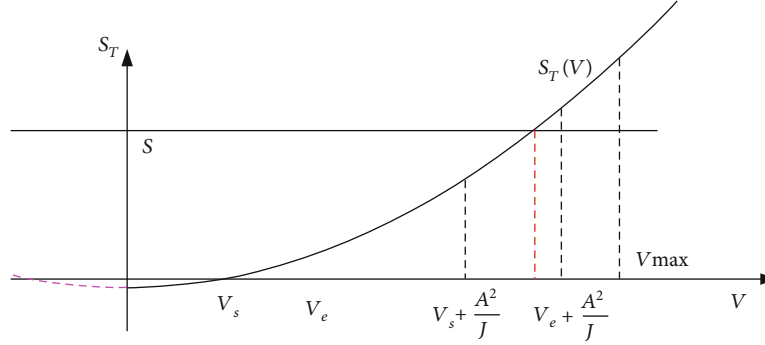
FIGURE 6: $S_T(v)$ Function curve.

TABLE 1: The summary of the curve profile planning of case 1.

Serial number	$F_{is} * F_{ie} \leq 0, f_{is} * f_{ie} \leq 0$ true or false			Expression for calculate t_i	Composition of curve profile
	$F_{1s} * F_{1e}$	$F_{2s} * F_{2e}$	$F_{3s} * F_{3e}$		
1	T	F	F	10/12	4 segments: $t_1/t_3/t_5/t_7$
2	F	T	F	9/12	5 segments: $t_1/t_2/t_3/t_5/t_7$
3	F	F	T	9/11	6 segments: $t_1/t_2/t_3/t_5/t_6/t_7$
4	F	F	F	9/11/7	7 segments: $t_1/t_2/t_3/t_4/t_5/t_6/t_7$

Step 6. calculate $V_{k+1} = V_k - f(V_k)/f'(V_k)$.

Step 7. if $|V_{k+1} - V_k| < V_e$, the calculation will be stopped.

Step 8. if $k = N$, stop; otherwise, $k = N + 1$ and turn to step 6.

4. Phenomenon of Solving Interval Jump

For the numerical solution, whether dichotomy or the Newton method, the solution interval is selected according to the initial value of the velocity in the first step, and then the displacement expression of this interval will be calculated. The solution obtained is not the final velocity solution, and the solution interval will be selected again according to the velocity value, and the iterative calculation will be carried out repeatedly until the termination condition is met. The process will cause the velocity value to jump at different intervals, and the displacement expression needs to be selected repeatedly, which will affect the solution efficiency, as shown in the following figure:

Figure 3 shows the case of $V_s < V_e < V_s + (A^2/J) < V_e + (A^2/J) < V_{max}$. The blue dotted line represents the segment point of the interval. From formula (13) to formula (16), it can be seen that the expression of displacement velocity is different in different segments. Taking dichotomy as an example, assuming that the required solution V^* is slightly less than $V_e + (A^2/J)$, the solution obtained after the first iter-

ation is $V' = V_a + V_b/2 = V_e + V_{max}/2(A^2/J)(A^2/J)$. Red indicates the corresponding position of the iterative solution. At this time, formula ((3)-(14)) is applicable, and the next iteration is carried out $V'' = V_a + V_b/2 = V' + V_{max}/2(A^2/J) \max]$. At this time, formula (16) is applicable, and the third iteration is carried out $V''' = V' + V''/2 \in [V_s + (A^2/J), V_e + (A^2/J)]$. In this case, (14) applies again.

This kind of iterative solution jumps between intervals, which causes different displacement expressions to be called back and forth, which leads to the instability of the solution process and affects the solution efficiency. In order to solve this kind of jump problem, the analytic programming method is introduced; that is, the interval of the expected solution is estimated first, and then the solution is calculated.

5. Solution by Matching with Velocity Step by Step

The main idea of this method is to use the assumed key velocity value to calculate the corresponding displacement, match it with the actual displacement, and then determine the curve profile according to the corresponding situation.

Take acceleration segment as example:

Step 1. assuming that the velocity can reach the required maximum velocity, the maximum velocity is substituted into

the displacement formula, and the calculated displacement is compared with the actual displacement. If the calculated displacement is smaller than the actual displacement, then there is a uniform velocity segment:

Step 2. if the calculated displacement is larger than the actual displacement, assume that there is just a uniform acceleration segment and substitute the assumed velocity value into the displacement formula and compare the calculated displacement with the actual displacement. If the calculated displacement is smaller than the actual displacement, it means that there is a uniform acceleration segment, then use the displacement formula including the uniform acceleration segment to solve the actual velocity value.

Step 3. if the calculated displacement is larger than the actual displacement, it means that there is no uniform acceleration section. At this time, the displacement formula without uniform acceleration section is used to solve for the actual velocity value.

The specific flow of the algorithm is shown in Figure 4.

The advantage of this method is that you can use the assumed velocity to plan first; then, you can solve for the actual velocity and avoid the jump in value ranges, increasing the stability of the solution, but there are also obvious disadvantages: the displacement is calculated and matched step by step with the assumed velocity. For the small line segment,

According to the definition of the function continuity, let

$$S_{Ti}(v) = \begin{cases} S_{T1} = (v + V_s)\sqrt{\frac{v - V_s}{J}} + (v + V_e)\sqrt{\frac{v - V_e}{J}} & v \leq V_s + \frac{A^2}{J} \leq V_e + \frac{A^2}{J} \leq V_{\max} \\ S_{T2} = \frac{v + V_s}{2} \left(\frac{A}{J} + \frac{v - V_s}{A} \right) + (v + V_e)\sqrt{\frac{v - V_e}{J}} & V_s + \frac{A^2}{J} \leq v \leq V_e + \frac{A^2}{J} \leq V_{\max} \\ S_{T3} = \frac{1}{A}v^2 + \frac{A}{J}v - \frac{V_s^2 + V_e^2}{2A} + \frac{V_s + V_e}{2J} & V_s + \frac{A^2}{J} \leq V_e + \frac{A^2}{J} \leq v \leq V_{\max} \end{cases} \quad (14)$$

$y = s(v)$ exist in a certain field of point v_0 ; when $v \rightarrow v_0$, if the limit of the function $s(v)$ exists and is equal to its function value at point v_0 , that is $\lim_{v \rightarrow v_0} s(v) = s(v_0)$, then function $y = s(v)$

is continuous at point v_0 . If $V_s' = V_s + (A^2/J)$ and $V_e' = V_e + (A^2/J)$, it can be proved that the function is continuous at the points V_s' and V_e' .

It can be seen from Figure 6 that the curve increases monotonously in the range of (V_s, V_{\max}) . Let $f_i(v) = S_{Ti}(v) - S$, ($i = 1, 2, 3$), S is the actual displacement, and there is a unique solution to $f_i(v) = S_{Ti}(v) - S = 0$ ($i = 1, 2, 3$), and with the difference of the displacement s value, the interval of solution of the piecewise function equation will change.

In the initial conditions, the initial velocity, the final velocity, the maximum acceleration, and the maximum jerk are given. The end values of three intervals are

the calculation needs to be repeated more than three times, which reduces the efficiency of the solution.

6. Interval Identification

To solve the problem of low efficiency of solution by matching with velocity, a new velocity planning method is proposed here.

From formula (5) to formula (10), it can be seen that when the initial velocity, final velocity, and maximum acceleration are given, the value of V directly affects the size and existence of each segment time t_i , and the actual s determines whether the actual velocity can reach the maximum allowable value.

When $V_s < V_e$, $V_e \leq V_s + (A^2/J)$, and $V_e + (A^2/J) < V_{\max}$, because of $\max\{V_s, V_e\} \leq V \leq V_{\max}$, interval 0 in Figure 5 is meaningless, and V may exist in the intervals 1-3.

The velocity value interval is divided into six sections and interval 0-5 by five key numerical points. The time expression (8) and formula (10) are suitable in interval 1, and the corresponding displacement expression is formula (16); the time expressions (7) and (10) are suitable in interval 2, and the corresponding displacement expression is formula (14); the time expressions (7) and (9) are suitable in interval 3, and the corresponding displacement expression is formula (15). Then, the piecewise function is obtained:

substituted into $f_i(v) = S_{Ti}(v) - S$, respectively. F_{is} and

F_{ie} are function values of the endpoints of the i -th interval.

$$\begin{aligned} F_{1s} &= f_1(v|v = V_e) = S_{T1}(v|v = V_e) - s \\ F_{1e} &= f_1\left(v\left|v = V_s + \frac{A}{J}\right.\right) = S_{T1}\left(v\left|v = V_s + \frac{A}{J}\right.\right) - s \\ F_{2s} &= f_2\left(v\left|v = V_s + \frac{A}{J}\right.\right) = S_{T2}\left(v\left|v = V_s + \frac{A}{J}\right.\right) - s \\ F_{2e} &= f_2\left(v\left|v = V_e + \frac{A}{J}\right.\right) = S_{T2}\left(v\left|v = V_e + \frac{A}{J}\right.\right) - s \\ F_{3s} &= f_3\left(v\left|v = V_e + \frac{A}{J}\right.\right) = S_{T3}\left(v\left|v = V_e + \frac{A}{J}\right.\right) - s \\ F_{3e} &= f_3(v|v = V_{\max}|_{T2}|v = V_{\max}|) \end{aligned} \quad (15)$$

From the properties of the continuity function, $F_{1e} = F_{2s}$, $F_{2e} = F_{3s}$.

In the next step, the function values of $f_i(v) = S_{T_i}(v) - S$ at the end points of each group of intervals are multiplied by each other to determine whether the product result is less than zero. There are two situations for the result:

(1) If $F_{is} * F_{ie} > 0$, the S_T and S curves do not intersect in this interval, and there is no solution in this interval. If there is no solution in the three intervals, the velocity is taken as V_{\max} , which indicates that the system has a uniform segment, the time is t_4 , and the curve is composed of seven segments. The t_i can be obtained by substituting the velocity value into the solutions of formulas (4) and (5)

(2) If $F_{is} * F_{ie} \leq 0$, it means that the S_T curve and S line intersect in this interval, and the equation has a solution, which is the actual value of velocity V that the system can achieve. At this time, substituting V into equation (7)–(10) can calculate the running time of each segment

Table 1 is the summary of the curve profile planning of case 1. The first three lines show the three cases of $F_{is} * F_{ie} \leq 0$ when i equals 1,2,3, respectively. The fourth line shows that $i = 1, 2, 3$ cannot meet the situation of $F_{is} * F_{ie} \leq 0$.

Formulas (14)-(16) can be solved by the dichotomy and Newton method listed above to get the velocity value. Then, the value is brought into the corresponding time calculation formula to get the running time of each segments.

As shown in Table 2, there are 12 possibilities in the comparison of five key values of the initial velocity V_s , the final velocity V_e , $V_s + (A^2/J)$, $V_e + (A^2/J)$, and V_{\max} .

Next, case 2 and case 3 in Table 2 will be discussed, and the interval distribution is shown in Figure 7.

Case 2. Interval 1 and interval 2 make sense.

The time expression (8) and equation (10) in interval 1 are suitable, and the displacement expression is the same as equation (17).

The time expression (7) and equation (10) in interval 2 are suitable, and the displacement expression is the same as equation (18).

Case 3. Only interval 1 makes sense. The time expression (8) and equation (10) in interval 1 are suitable, and the displacement expression is the same as equation (17).

Case 4 and case 5 express the situation when $V_e > V_s + (A^2/J)$, as shown in Figure 8.

Case 4. interval 1s and 2 make sense.

The time expression (7) formula (10) in interval 1 is applicable, and the displacement expression is the same as formula (14).

Time expression (7) and equation (9) in interval 2 are applicable, and displacement expression is the same as equation (13).

TABLE 2: Possibility of key point value comparison.

Case no.	Description of value
1	$V_s < V_e < V_s + \frac{A^2}{J} < V_e + \frac{A^2}{J} < V_{\max}$
2	$V_s < V_e < V_s + \frac{A^2}{J} < V_{\frac{A^2}{J_{\max}}}$
3	$V_s < V_e < V_{\frac{A^2 A^2}{J J_{\max}}}$
4	$V_s < V_s + \frac{A^2}{J} < V_e < V_e + \frac{A^2}{J} < V_{\max}$
5	$V_s < V_s + \frac{A^2}{J} < V_e < V_{\frac{A^2}{J_{\max}}}$
6	$V_s = V_e < V_s + \frac{A^2}{J} < V_{\max}$
7	$V_s = V_e < V_{\frac{A^2}{J_{\max}}}$
8	$V_e < V_s < V_e + \frac{A^2}{J} < V_s + \frac{A^2}{J} < V_{\max}$
9	$V_e < V_s < V_e + \frac{A^2}{J} < V_{\frac{A^2}{J_{\max}}}$
10	$V_e < V_s < V_{\frac{A^2 A^2}{J J_{\max}}}$
11	$V_e < V_e + \frac{A^2}{J} < V_s < V_s + \frac{A^2}{J} < V_{\max}$
12	$V_e < V_e + \frac{A^2}{J} < V_s < V_{\frac{A^2}{J_{\max}}}$

Case 5. Interval 1 makes sense.

The time expression (7) and formula (10) in interval 1 are suitable, and the displacement expression is the same as formula (15).

(1) When $V_s = V_e$, there are only two cases, case 6 and 7, as shown in Figure 9

Case 6. Interval 1 and 2 make sense.

The time expression (8) and equation (10) in interval 1 are suitable, and the displacement expression is the same as equation (17).

The time expression (7) and equation (9) in interval 2 are suitable, and the displacement expression is the same as equation (24).

(2) When $V_s > V_e$, the possible value range of velocity is from case 8 to case 12. The actual solution process is the same as that of $V_s < V_e$, and only 1 and 2 need to be interchanged. In consideration of space, it will not be expanded here

(3) There are four other possibilities, which are not listed in the bid, as shown in Table 3

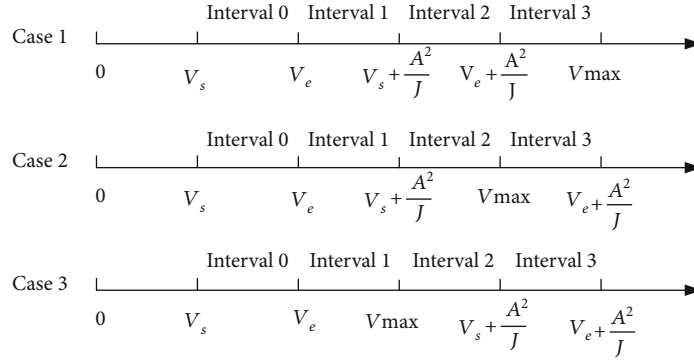


FIGURE 7: Interval partition graph of cases 1-3.

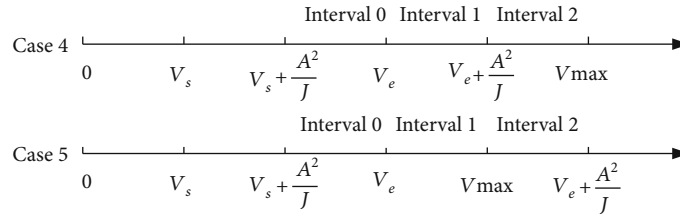


FIGURE 8: Interval partition graph of cases 4 and 5.

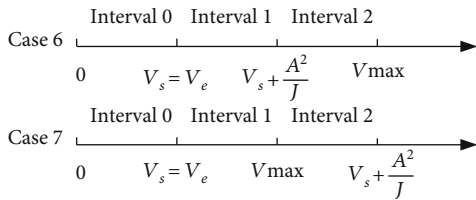


FIGURE 9: Interval partition graph of case 6 and 7.

TABLE 3: Only acceleration section or deceleration section.

Case no.	Description of value
13	$V_s < V_s + \frac{A^2}{J} < V_{\max}$
14	$V_s < V_{\frac{A^2}{J_{\max}}}$
15	$V_s < V_e + \frac{A^2}{J} < V_{\max}$
16	$V_e < V_{\frac{A^2}{J_{\max}}}$

Case 7. Only interval 1 makes sense. The time expression (8) and equation (10) in interval 1 are suitable, and the displacement expression is the same as equation (17).

The so-called S-shape means that the shape of acceleration is similar to the S-shape, case 13—case listed in Table 3. There is only acceleration section or deceleration section, and acceleration is similar to half circle. According to the definition of the S-shaped acceleration and deceleration curve, these four cases do not meet the definition, but these four cases do exist in small displacement, especially in

the speed sensitive area, so they are listed here. The calculation method is the same as above and will not be described here.

7. Comparison of Solution Methods

7.1. Match with Velocity VS Interval Identification. The first step when using the velocity method is to assume that the curve has seven complete segments, and that the time and velocity values are substituted into the formula to calculate the displacement. If the calculated displacement is less than the actual displacement, the one with the larger value in $V_s + A^2/J$ and $V_e + A^2/J$ will be substituted into the equation again to calculate the displacement, and the calculated displacement and the actual displacement will be compared again. If the calculated displacement is greater than or equal to the actual displacement, then the actual velocity and running time can be obtained by solving the equation.

If the calculated displacement is smaller than the actual displacement, it is necessary to replace the smaller value of $V_s + A^2/J$ and $V_e + A^2/J$ into the displacement equation again, calculate and compare the displacement for the third time, and then plan out the profile of the curve. Finally, the actual velocity can be obtained by solving the equation.

The specific steps of the interval identification method are as follows: ① input the initial value, compare 5 key speed values, and find the corresponding case No.; ② in the corresponding case No., substitute the value to judge $F_{is} * F_{ie} \leq 0$; ③ solve the equation in the interval of $F_{is} * F_{ie} \leq 0$ and get the velocity; and ④ replace the speed V with the time calculation formula and get the whole curve profile.

Figure 10 shows the comparison of the processes of the two methods. In summation, if the curve has 7 or 6 segments, the calculation steps and calculation amounts of the two

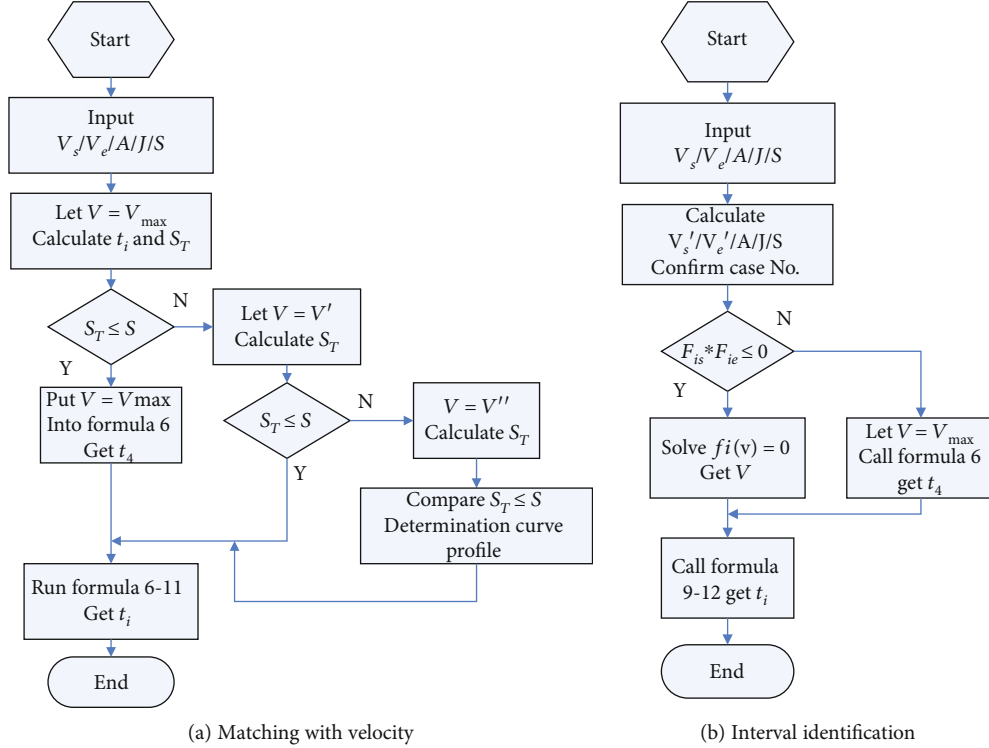


FIGURE 10: Comparison of matching with velocity and interval identification.

methods are basically the same. If the curve has 4 or 5 segments, the calculation steps and calculation amounts of the interval identification method are less and therefore, the efficiency is higher.

7.2. Direct Numerical Solution vs Interval Identification. In order to verify the efficiency of the solution, various methods and algorithms are programmed with matlab2018a, and the displacement is taken as a variable between 0.0001 m and 0.018 m, and the interval between the value points is 0 and 1 mm. Computer configuration: processor Intel (R) core (TM) i5-4200u memory 8 g. The solution time of each algorithm under different displacement conditions is calculated, and the solution efficiency of each method is compared. The number of Newton iteration and dichotomy iteration termination is 10000. The specific parameter settings are shown in Table 4, and the solution efficiency is shown in Figure 11.

It can be seen from Figure 11 that the abscissa of either direct Newton iteration or interval identification+Newton iteration does not start from zero; that is, the red dotted area in the figure which indicates that Newton iteration does not return the velocity solution and directly jumps out of the iteration cycle after reaching the limit of iteration rounds. As mentioned in the previous paper, the convergence of the Newton iteration needs to be further verified. When the displacement value is about 1-3 mm, the number of Newton iterations is also significantly higher than the number after it is stable. When the displacement is more than 4 mm, all algorithms tend to be more stable as displacement increases.

TABLE 4: Parameters used in the simulation.

Symbol	Item	Value
V_s	Initial velocity	20 mm/s
V_e	Final velocity	30 mm/s
V_{\max}	Maximum velocity	100 mm/s
A_{\max}	Maximum acceleration	600 mm/s ²
J_{\max}	Maximum jerk	3×10^4 mm/s ³
V_e	Iteration termination precision	1×10^{-6} mm/s
T_s	Interpolation period	1 ms
S	Displacement	0.01-18 mm
V_o	Initial velocity of iteration	15 mm/s

In order to see the solution efficiency of the four methods more clearly and intuitively, the abscissa in Figure 11 is intercepted through 7-12 mm part, resulting in Figure 12.

Regarding solving efficiency, the efficiency of the interval identification method is obviously better than that of the direct iterative method when the solving process of various algorithms is stable. The efficiency of interval identification plus Newton iteration is the highest, and then the efficiency of direct Newton iteration is about one third of that of interval identification plus Newton iteration. The efficiency of dichotomy is obviously lower than that of Newton iteration. The efficiency of direct dichotomy is about one seventh of interval identification plus the Newton iteration. From the point of view of solving stability, the interval identification method is also better than the direct

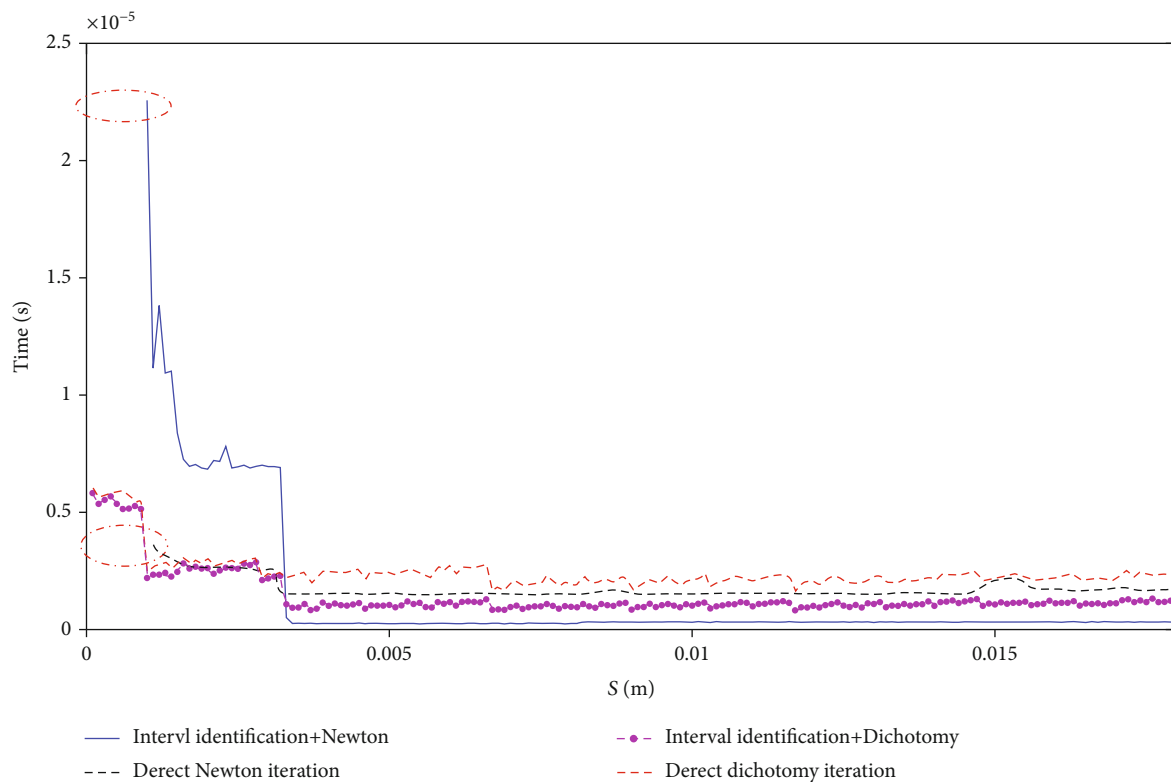


FIGURE 11: Solution efficiency comparison.

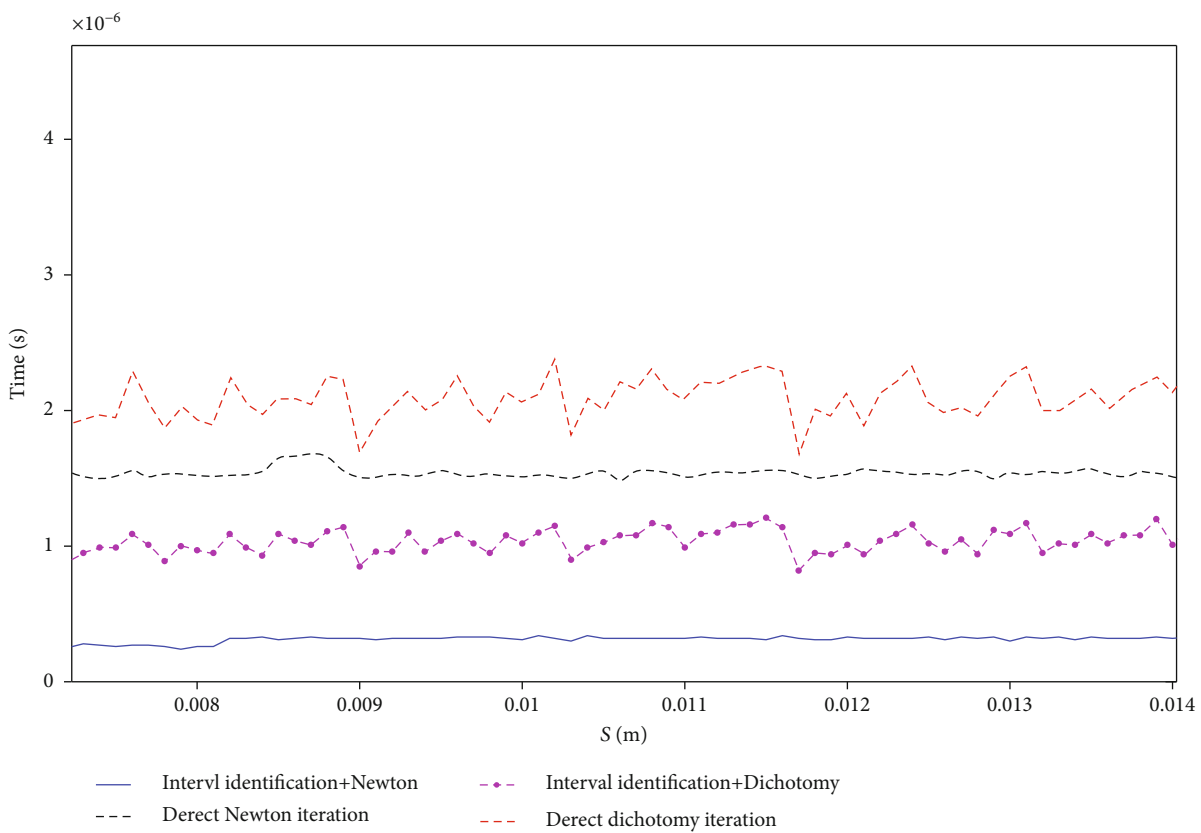


FIGURE 12: Local amplification of efficiency comparison graph after stabilization.

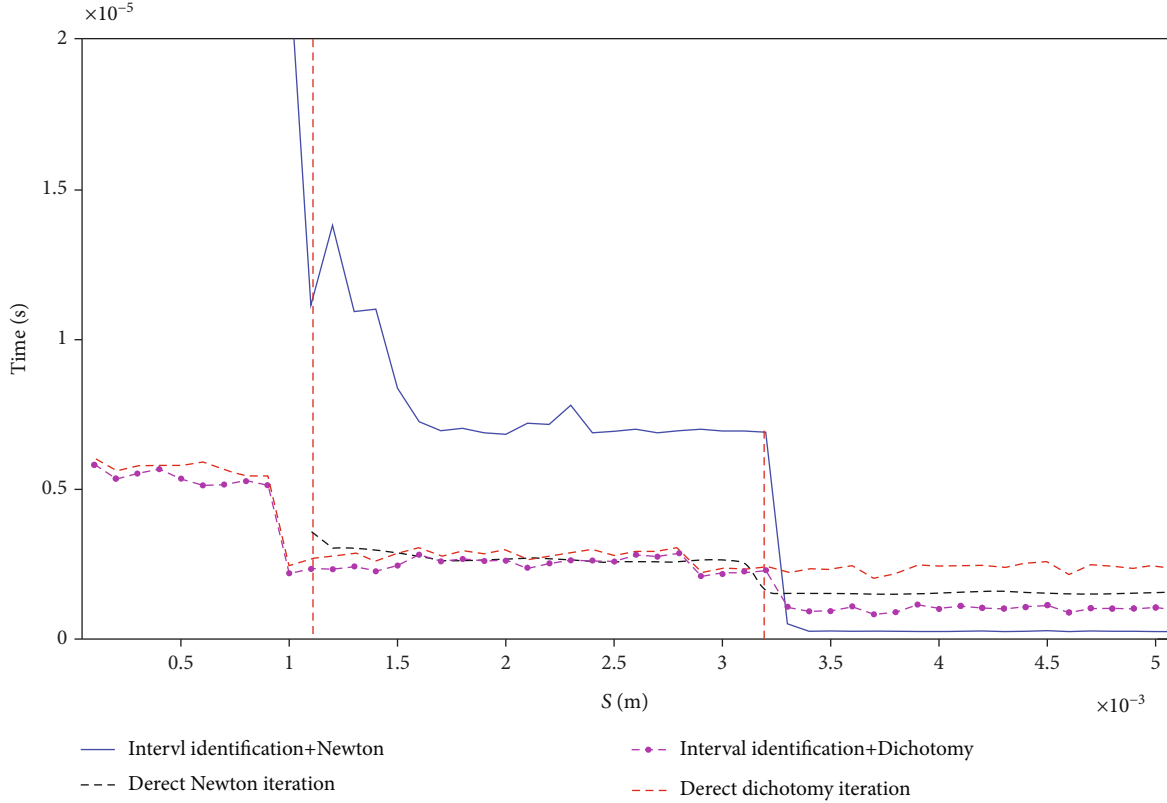


FIGURE 13: Comparison of solution efficiency when small displacement.

iteration method. As mentioned earlier in the paper, the interval identification avoids the phenomenon of iteration jumping back and forth in different value intervals, thus improving the stability.

To sum up, the interval identification+Newton iteration method is the best both in solving efficiency and solving stability.

Figure 11 is the abscissa which is truncated at 0-5 mm and enlarged locally to get Figure 13. It can be seen that the efficiency of all solutions fluctuates within a certain boundary, and the Newton iteration is the obvious. In the figure, the red dot dash appears near two key points, which are S_{T1} and S_{T2} in formula (17), corresponding to the interval boundary points in the interval identification method. This shows that the expression change caused by the change of displacement length has an obvious effect on the iteration efficiency. The smaller the displacement, the more time-consuming the solution.

8. Interval Identification plus Modified Newton Iteration

In order to prevent the divergence of Newton iteration, a requirement is attached to the iteration process to keep its monotonicity.

$$|f(V_{k+1})| < |f(V_k)|. \quad (16)$$

In order to meet this requirement, Newton iteration is modified.

$$\begin{cases} V_{k+1} = V_k - \frac{f(V_k)}{f'(V_k)} \\ V_{k+1} = \lambda V_{k+1} + (1 - \lambda)V_k \end{cases}, \quad (17)$$

Where $\lambda(0 < \lambda \leq 1)$ is the downhill factor, and formula (18) is obtained after the transformation of the above formula:

$$V_{k+1} = V_k - \lambda \frac{f(V_k)}{f'(V_k)} \quad k = 0, 1, \dots \quad (18)$$

In the iteration, it starts from $\lambda = 1$ and reduces by half until it meets the requirements (16).

After the modified Newton iteration is introduced, the iteration parameters and initial conditions are set according to Table 4. Under the same conditions, the efficiency curve Figure 14 is obtained.

Only Newton iteration can be seen in Figure 14, because the iteration diverges until the number of iterations jumps out of the cycle. The record time of 0.01 seconds in the figure is the time taken to jump out of the cycle. Figure 13 is partially enlarged to obtain Figure 15.

It can be seen from Figure 15 that, when the displacement is large, the solution efficiency of interval identification plus

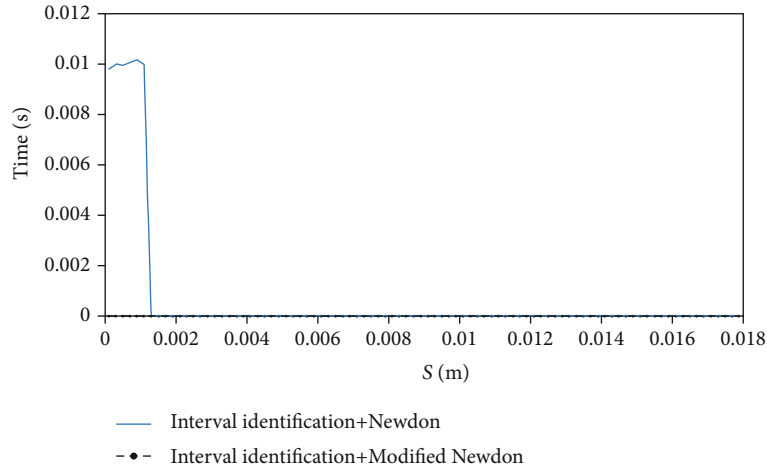


FIGURE 14: Comparison between Newton iteration and Newton downhill method.

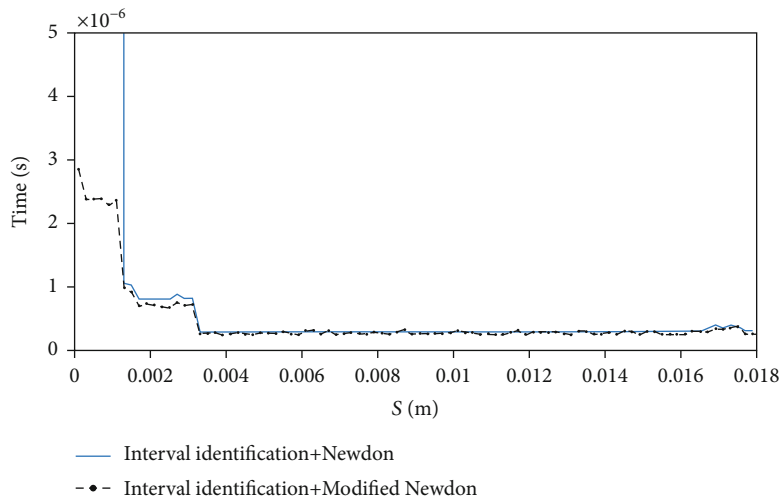


FIGURE 15: Local enlarged map of comparison between Newton and Newton downhill.

modified Newton method is very close to that of interval identification plus Newton iteration, which retains the characteristics of fast convergence of Newton iteration; when the displacement is small, the downhill factor ensures the convergence of iteration.

9. Conclusion

- (1) When analyzing the process of solving S-shape acceleration and deceleration directly by the traditional numerical solution method, the phenomenon of “solving interval jump” was concerned, which is a main reason for the low efficiency and poor stability of the solution
- (2) According to the S-curve profile and solution, the concept of separating the curve profile recognition from the velocity solution was introduced, and a plan of quickly identifying the interval of the solution location was put forth. With this method, the com-

plete acceleration and deceleration curve parameters can be obtained through one-time planning and one-time solutions, and the solution efficiency and stability are certain

- (3) The modified Newton method was introduced to solve the problem that the Newton iteration depends too much on the initial value of velocity, which not only retains the speed advantage of the Newton method but also uses the downhill factor to ensure its convergence
- (4) Finally, through the simulation comparison and analysis, the efficiency and stability of the interval identification and modified Newton method were verified

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author(s) declare(s) that they have no conflicts of interest.

Acknowledgments

This study was funded by the National Natural Science Foundation of China (No. 51975019) and the Science and Technology Project of Hebei Education Department (No. QN2019193).

References

- [1] J. W. Jeon and Y. Y. Ha, "A generalized approach for the acceleration and deceleration of industrial robots and CNC machine tools," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 1, pp. 133–139, 2000.
- [2] J. Hu, L. J. Xiao, Y. H. Wang, and Z. Wu, "An optimal feedrate model and solution algorithm for a high-speed machine of small line blocks with look-ahead," *International Journal of Advanced Manufacturing Technology*, vol. 28, no. 9-10, pp. 930–935, 2006.
- [3] Y. P. Li and S. N. Wang, "Improved exponential acceleration and deceleration algorithm," *Machine Tool and Hydraulics*, vol. 1, pp. 39–40, 2006.
- [4] H. Li, H. Li, L. Song, Y. Yin, L. Huang, and W. Li, "Design of global sliding-mode controlled AC servo controller based on exponential acceleration/deceleration algorithm," in *2010 IEEE International Conference on Mechatronics and Automation*, pp. 1507–1511, Xi'an, China, 2010.
- [5] H. Jun, Y. Youpeng, C. Hao, and W. Huaming, "A fast nested look-ahead algorithm with S-shape acceleration and deceleration," *Acta Aeronautica Et Astronautica Sinica*, vol. 31, no. 4, pp. 842–851, 2010.
- [6] K. Erkorkmaz and Y. Altintas, "High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation," *International Journal of Machine Tools & Manufacture*, vol. 41, no. 9, pp. 1323–1345, 2001.
- [7] J. H. Chen, S. S. Yeh, and J. T. Sun, "An S-curve acceleration/-deceleration design for CNC machine tools using Quintic Feedrate function," *Computer-Aided Design and Applications*, vol. 8, no. 4, pp. 583–592, 2011.
- [8] K. Z. Sun, X. F. Zhou, G. Zhang, and X. S. Chen, "An interpolation algorithm based on S-curve Feedrate profile and look-ahead function," *Applied Mechanics & Materials*, vol. 300-301, pp. 1389–1396, 2013.
- [9] Y. Altintas, *Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design*, Cambridge University Press, London, 2000.
- [10] P. J. Barre, R. Bearee, P. Borne, and E. Dumetz, "Influence of a jerk controlled movement law on the vibratory behaviour of high-dynamics systems," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 42, no. 3, pp. 275–293, 2005.
- [11] S. Chuan, Z. Tong, Y. Peiqing, and L. Qiang, "Study on S-shape curve acceleration and deceleration control on NC system," *China Mechanical Engineering*, vol. 18, no. 12, pp. 1421–1425, 2007.
- [12] S. H. Nam and M. Y. Yang, "A study on a generalized parametric interpolator with real-time jerk-limited acceleration," *Computer Aided Design*, vol. 36, no. 1, pp. 27–36, 2004.
- [13] Y. Liangliang, X. Shoujin, S. Weimin, and G. Hongwei, "Research on S type acceleration and deceleration time planning algorithm with beginning and end speed non-zero," *Journal of Mechanical Engineering*, vol. 52, no. 23, pp. 199–206, 2016.
- [14] Y. Chen, X. Ji, Y. Tao, and H. Wei, "Look-ahead algorithm with whole S-curve acceleration and deceleration," *Advances in Mechanical Engineering*, vol. 5, no. 10, Article ID 974152, 2015.
- [15] L. Wang and J. Cao, "A look-ahead and adaptive speed control algorithm for high-speed CNC equipment," *International Journal of Advanced Manufacturing Technology*, vol. 63, no. 5-8, pp. 705–717, 2012.
- [16] J. A. Tao, S. L. Gao, Q. N. You, and Q. Shi, "Look ahead and acceleration deceleration algorithms for micro-line blocks machining," *Journal of Computer-Aided Design & Computer Graphics*, vol. 22, no. 9, pp. 1570–1577, 2010.
- [17] Y. B. Xia and W. Wang, "S-curve acceleration/deceleration algorithm based on reference table," *Research Journal of Applied Sciences Engineering & Technology*, vol. 4, no. 2, pp. 131–134, 2012.
- [18] H. Ni, J. Yuan, S. Ji, C. Zhang, and T. Hu, "Feedrate Scheduling of NURBS Interpolation Based on A Novel Jerk-continuous ACC/DEC Algorithm," *IEEE Access*, vol. 6, pp. 66403–66417, 2018.
- [19] D. Li, J. Wu, J. Wan, S. Wang, S. Li, and C. Liu, "The implementation and experimental research on an S-curve acceleration and deceleration control algorithm with the characteristics of end-point and target speed modification on the fly," *The International Journal of Advanced Manufacturing Technology*, vol. 91, no. 1-4, pp. 1145–1169, 2017.
- [20] T.-C. Lu and S.-L. Chen, "Genetic algorithm-based S-curve acceleration and deceleration for five-axis machine tools," *International Journal of Advanced Manufacturing Technology*, vol. 87, no. 1-4, pp. 219–232, 2016.
- [21] F. Xu, H. Chen, X. Gong, and Q. Mei, "Fast nonlinear model predictive control on FPGA using particle swarm optimization," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 1, pp. 310–321, 2016.