# A Greedy Algorithm for Feedrate Planning of CNC Machines along Curved Tool Paths with Confined Jerk*

Ke Zhang, Chun-Ming Yuan, Xiao-Shan Gao†, Hongbo Li
KLMM, Institute of Systems Science, Chinese Academy of Sciences

**Abstract**

In this paper, the problem of optimal feedrate planning along a curved tool path for 3-axis CNC machines with the acceleration and jerk limits for each axis and the tangential velocity bound is addressed. It is proved that the optimal feedrate planning must be "Bang-Bang" or "Bang-Bang-Singular" control, that is, at least one of the axes reaches its acceleration or jerk bound, or the tangential velocity reaches its bound throughout the motion. As a consequence, the optimal parametric velocity can be expressed as a piecewise analytic function of the curve parameter $u$. The explicit formula for the velocity function when a jerk reaches its bound is given by solving a second order differential equation. Under a "greedy rule", an algorithm for optimal jerk confined feedrate planning is presented. Experiment results show that the new algorithm can be used to reduce the machining vibration and improve the machining quality.

**Keywords.** Feedrate optimization, parametric tool path, confined jerk, velocity limit surface, analytical solutions for feedrate function.

## 1 Introduction

The feedrate optimization along curved tool paths is an important problem in CNC machining. In the feedrate planning, the acceleration on each axis of the machine must be constrained, because the torque (or force) capabilities of the axes drives are limited. Therefore, the problem is that how to identify the feedrate along a given path such that the machining time is minimal without exceeding the capabilities of the actuators.

Bobrow et al [1], Shiller and Lu [2] gave algorithms to determine the minimum-time motion for a robot manipulator along a specific path (at least a smooth curve) with actuator torque constraints. Farouki and Timar [3, 4] planned the feedrate for CNC machining with acceleration bounds on $x, y, z$ axes, and gave a piecewise-analytic expression of the optimal velocity planning function. Zhang et al simplified the method in [4] for quadratic B-splines and realized real-time manufacturing on industrial CNC machines [5]. Yuan et al [6] provided a time optimal feedrate planning method with tangential acceleration and chord error

*Partially supported by a National Key Basic Research Project of China (2011CB302400) and by a grant from NSFC (60821002).

†To whom correspondence should be addressed.

bounds. All of the methods mentioned above used the velocity limit curve and its switching points in the $u$-$\dot{u}$ phase plane to obtain an optimal solution which is a continuous time optimal velocity function along a specific path. Dong and Stori [7] gave a discrete greedy algorithm for the above problem with parametric velocity and acceleration constraints. These methods are all based on the idea of "Bang-Bang" control, that is, at least one of the axes reaches its acceleration bound (or torque limit) throughout the motion.

However, the acceleration profile obtained with the above methods has discontinuities, since the acceleration may change from the maximum $A$ to the minimum $-A$ instantly. These discontinuities correspond to step changes in the force output demanded of the drive, cause vibrations and then large contouring errors. One method to reduce vibrations is introducing jerk constraints along each axis to the original problem, which can generate a continuous acceleration profile.

When jerk constraints are added, the analysis must be performed in the $u$-$\dot{u}$-$\ddot{u}$ phase space instead of the $u$-$\dot{u}$ phase plane. The new optimization problem becomes more difficult. However, it is much easier when considering the constraints of the tangential acceleration and jerk. Such problems have received much attention in the robotics and manufacturing literature. Altintas and Erkorkmaz [8] presented a quintic spline trajectory generation algorithm that produces continuous position, velocity, and acceleration profiles with confined tangential acceleration and jerk. Macfarlane and Croft [9] developed and implemented an online method to obtain smooth, jerk-bounded trajectories with fifth-order polynomials for industrial robot applications. Their method is near time optimal with confined tangential jerk and acceleration. Nam and Yang [10] presented a recursive trajectory generation method that estimates an admissible path increment and determines the initiation of the final deceleration stage according to the distance left to travel estimated at every sampling time, resulting in exact feedrate trajectory generation through tangential jerk-confined acceleration profiles for the parametric curves. Lin et al [11] proposed a dynamics-based interpolator with real-time look-ahead algorithm to generate a smooth and tangential jerk-confined acceleration/deceleration feedrate profile. Emami and Arezoo [12] introduced a look-ahead trajectory generation method which determines the deceleration stage according to the fast estimated arc length and the reverse interpolation of each curve at every sampling time. They obtained a feedrate trajectory with tangential jerk-confined acceleration profiles for the NURBS curves. Lai et al [13] further proposed a method which can generate velocities with jerk limits as well as chord error, speed, and acceleration limits. The method uses a discrete model and satisfies all these constraints by backtracking at each step.

In order to make full use of the capabilities of the machine tool, it is desirable to solve the problem with jerk constraints on each axis, because the drivers of the axes of a CNC machine are controlled independently. Using a jerk limit on each axis will lead to a continuous acceleration curve for each axis. Dong et al [14] extended their discrete greedy algorithm [7] by adding parametric jerk constraints. However, none of these prior approaches have attempted to get an analytical solution for a continuous model with jerk constraints on each axis.

In this paper, the problem of optimal feedrate planning along a specific curved tool path $\vec{r}(u)$ with at least $C^2$ continuity under the acceleration and jerk limits for each axis and the tangential velocity bound for a 3-axis machine is considered. First, it is proved that the time-optimal feedrate planning must use “Bang-Bang” or “Bang-Bang-Singular” control, that is, at least one of the axes reaches its acceleration or jerk bound, or the tangential velocity reaches its bound throughout the motion. Then an optimal feedrate planning algorithm is given under a “greedy rule”: using the maximal jerk as much as possible.

This algorithm has two key components, which are also the main contribution of this paper. The first one is how to compute the parametric velocity function after the control axis and maximal (or minimal) jerk are given. To compute the parametric velocity function, it is necessary to solve a second-order differential equation, and the analytic solutions are given. The CASS (*control axis switching surface*) is also introduced in this paper. The control axis should be changed when the velocity integration trajectory passes through a CASS. The second key component is to introduce and use the VLS (*velocity limit surface*) for the feedrate planning. It is similar to the VLC (*velocity limit curve*) in the feedrate planning with acceleration constraints [1, 2, 3, 4]. The VLS is a surface in the $u$-$\dot{u}$-$\ddot{u}$ space which limits the parametric velocity and acceleration.

The general idea of this algorithm is to compute the integration trajectory forward from $(0, 0, 0)$ in the $u$-$\dot{u}$-$\ddot{u}$ space under the limit of VLS and a “greedy rule”; then to compute the integration trajectory backward from $(1, 0, 0)$ in a similar way; and finally to obtain a complete velocity integration trajectory with continuous acceleration by connecting the two integration trajectories.

Experiments are contacted to compare the algorithm with confined jerk with the similar algorithm with confined acceleration in a CNC machine. The results show that with confined jerk, the machining vibration can be reduced and the machining quality can be improved significantly.

The rest of this paper will be organized as follows. Section 2 gives the description and theoretical analysis of the feedrate optimization problem. Section 3 gives the feedrate planning algorithm. Section 4 gives the experimental results. Section 5 concludes the paper.

## 2 Problem description and theoretical analysis

### 2.1 Problem description

For brevity, the tool path is considered to be a plane piecewise parametric curve:

$$\vec{r}(u) = (x(u), y(u)), 0 \leq u \leq 1,$$

where $x(u), y(u) \in C^2([0, 1])$. Furthermore, each segment of the curve is assumed to be infinitely differentiable. For instance, a cubic B-spline curve and most NURBs curves satisfy the conditions. In this paper, the tangential velocity bound and the bounds on the $x$ and $y$ acceleration and jerk components are considered. The extension to spatial paths is relatively straightforward but more tedious. Denote the derivatives with respect to time $t$ and the parameter $u$ by dots and primes, respectively:

$$\dot{u} = du/dt, x' = dx/du.$$

Then, it is obvious that

$$\dot{u}' = \frac{\ddot{u}}{\dot{u}}, \tag{1}$$

$$\ddot{u}' = \frac{\dddot{u}}{\dot{u}}, \tag{2}$$

and

$$\dot{u}'' = (\frac{\ddot{u}}{\dot{u}})' = \frac{\dddot{u}}{\dot{u}^2} - \frac{\ddot{u}^2}{\dot{u}^3}. \tag{3}$$

The *tangential velocity* is:

$$v = |d\vec{r}/dt| = |\vec{r'}|\dot{u} = \sigma\dot{u}, \tag{4}$$

where $\sigma = \sqrt{x'^2 + y'^2}$. With the tangential velocity bound $V_{max}$, the constraint is

$$0 \leq \sigma\dot{u} \leq V_{max}. \tag{5}$$

The accelerations on the $x$ and $y$ axes are:

$$\begin{cases} a_x = \ddot{x} = (x'\dot{u})'\dot{u} = x''\dot{u}^2 + x'\dot{u}\dot{u}', \\ a_y = \ddot{y} = (y'\dot{u})'\dot{u} = y''\dot{u}^2 + y'\dot{u}\dot{u}'. \end{cases} \tag{6}$$

Substituting (1) into (6), $a_x, a_y$ can be expressed as

$$\begin{cases} a_x = x''\dot{u}^2 + x'\ddot{u}, \\ a_y = y''\dot{u}^2 + y'\ddot{u}. \end{cases} \tag{7}$$

The jerks on the $x$ and $y$ axes are:

$$\begin{cases} j_x = \dddot{x} = ((x'\dot{u})'\dot{u})'\dot{u} = x'''\dot{u}^3 + 3x''\dot{u}^2\dot{u}' + x'\dot{u}(\dot{u}')^2 + x'\dot{u}^2\dot{u}'', \\ j_y = \dddot{y} = ((y'\dot{u})'\dot{u})'\dot{u} = y'''\dot{u}^3 + 3y''\dot{u}^2\dot{u}' + y'\dot{u}(\dot{u}')^2 + y'\dot{u}^2\dot{u}''. \end{cases} \tag{8}$$

Similarly, substituting (1) and (3) into (8), $j_x, j_y$ can be expressed as

$$\begin{cases} j_x = x'''\dot{u}^3 + 3x''\dot{u}\ddot{u} + x'\dddot{u}, \\ j_y = y'''\dot{u}^3 + 3y''\dot{u}\ddot{u} + y'\dddot{u}. \end{cases} \tag{9}$$

In this paper, $\dot{u}, \ddot{u}$, and $\dddot{u}$ are called *parametric velocity, parametric acceleration*, and *parametric jerk*, respectively. Then the feedrate optimization problem becomes to plan the parametric velocity $\dot{u} \in C^1([0,1])$, such that the machining time is minimal:

$$\min t_f = \int_0^1 \frac{du}{\dot{u}} \tag{10}$$

under the following constraints:

$$\begin{cases} \dot{u}|_{u=0,1} = 0, \\ \ddot{u}|_{u=0,1} = 0, \end{cases} \tag{11}$$

$$\begin{cases} 0 \leq \dot{u} \leq V_{max}/\sigma, \\ |a_x| \leq A_x, |a_y| \leq A_y, \\ |j_x| \leq J_x, |j_y| \leq J_y, \end{cases} \tag{12}$$

where $A_x, A_y, J_x, J_y$ are positive constants, denoting maximal accelerations and jerks of $x, y$ axes, respectively.

### 2.2 Optimal solution is "Bang-Bang" or "Bang-Bang-Singular" control

In the optimal problem (10), the control variables are $j_x, j_y$. When the solution satisfies $j_x = \pm J_x$ or $j_y = \pm J_y$ on an interval in $[0,1]$, it is "Bang-Bang" control on the interval. Otherwise, it is singular control on the interval. If the solution satisfies $j_x = \pm J_x$ or $j_y = \pm J_y$ on the whole interval $[0,1]$, the solution is called "Bang-Bang" control. If it satisfies $j_x = \pm J_x$ or $j_y = \pm J_y$ only in a proper subset of $[0,1]$ and satisfies $a_x = \pm A_x$, $a_y = \pm A_y$ or $v = V_{max}$ in its complementary set, the solution is called "Bang-Bang-Singular" control.

This section proves that the solution of the optimal problem must be "Bang-Bang" or "Bang-Bang-Singular" control, that is, at least one of the axes reaches its acceleration or jerk bound, or the tangential velocity reaches its bound throughout the motion. In other words, at least one of equalities $a_x = \pm A_x$, $a_y = \pm A_y$, $j_x = \pm J_x$, $j_y = \pm J_y$ and $v = V_{max}$ satisfies at every time. When there is an axis whose jerk reaches its bound, it is called the *control axis.*

The claim is proved by contradiction. Assume that the optimal parametric velocity function is $\dot{u}$, and there exists an interval $[u_1, u_2]$ in $[0,1]$, such that none of the $v$, $a_x$, $a_y$, $j_x$ and $j_y$ reaches its bound for $u \in [u_1, u_2]$, i.e., the inequalities in (12) are all strict. Then, for $u \in [u_1, u_2]$, there exists a positive constant $\varepsilon_0$, such that

$$\dot{u} + \varepsilon_0 \leq V_{max}/\sigma. \tag{13}$$

From (6) and (8), $a_x, a_y, j_x, j_y$ can be expressed as functions in $u, \dot{u}, \dot{u}', \dot{u}''$, denoted by $p_1, p_2, p_3, p_4$, respectively:

$$\begin{cases} a_x = p_1(u, \dot{u}, \dot{u}') = x''\dot{u}^2 + x'\dot{u}\dot{u}', \\ a_y = p_2(u, \dot{u}, \dot{u}') = y''\dot{u}^2 + y'\dot{u}\dot{u}', \\ j_x = p_3(u, \dot{u}, \dot{u}', \dot{u}'') = x'''\dot{u}^3 + 3x''\dot{u}^2\dot{u}' + x'\dot{u}(\dot{u}')^2 + x'\dot{u}^2\dot{u}'', \\ j_y = p_4(u, \dot{u}, \dot{u}', \dot{u}'') = y'''\dot{u}^3 + 3y''\dot{u}^2\dot{u}' + y'\dot{u}(\dot{u}')^2 + y'\dot{u}^2\dot{u}''. \end{cases}$$

So, for every $u \in [0,1]$, $p_1, p_2, p_3, p_4$ are polynomials in $\dot{u}, \dot{u}', \dot{u}''$. Using (12), there exist positive constants $D_1, D_2, D_3, D_4$, such that

$$\begin{cases} |p_1(u, \dot{u}, \dot{u}')| \leq D_1 < A_x, \\ |p_2(u, \dot{u}, \dot{u}')| \leq D_2 < A_y, \\ |p_3(u, \dot{u}, \dot{u}', \dot{u}'')| \leq D_3 < J_x, \\ |p_4(u, \dot{u}, \dot{u}', \dot{u}'')| \leq D_4 < J_y \end{cases} \tag{14}$$

is established for $u \in [u_1, u_2]$.

For every positive $\varepsilon$, construct a parametric velocity function as

$$\Delta\dot{u} = \begin{cases} \varepsilon(1 + \cos\frac{\pi(2u - u_1 - u_2)}{u_2 - u_1}) & u_1 \leq u \leq u_2; \\ 0 & \textit{otherwise}. \end{cases}$$

It is easy to show that

$$\begin{cases} \Delta\dot{u}|_{u_1,u_2} = 0, \\ (\Delta\dot{u})'|_{u_1,u_2} = 0, \end{cases} \tag{15}$$

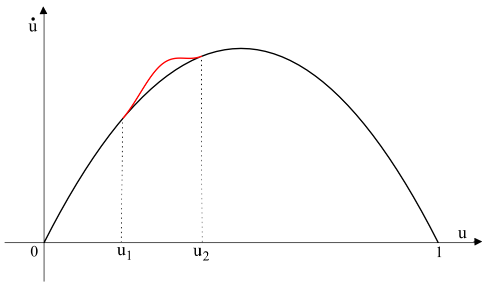



Figure 1: Black curve: original velocity function. Red curve: a better velocity function.

and

$$\begin{cases} 0 \le \Delta\dot{u} \le 2\varepsilon, \\ |\Delta\dot{u}'| \le B_1\varepsilon, \\ |\Delta\dot{u}''| \le B_2\varepsilon, \end{cases} \tag{16}$$

where $B_1, B_2$ are positive constants. In Fig. 1, $\Delta\dot{u}$ is represented by the red curve segment and $\dot{(u)}$ is represented by the black one.

Let $\dot{u}^* = \Delta\dot{u} + \dot{u}$, it is obvious that $\dot{u}^* \in C^1([0,1])$ from (15). As illustrated by Fig. 1, $\dot{u}^*$ has the same value as $\dot{u}$ outside $(u_1, u_2)$ and is strictly larger than $\dot{u}$ in $(u_1, u_2)$. We claim that when choosing the parameters properly, $\dot{u}^*$ also satisfies the constraints (11)(12) and as a consequence, a contraction will be obtained.

For every $u \in [u_1, u_2]$, use first-order Taylor expansion of $p_3$ to $\dot{u}, \dot{u}', \dot{u}''$ to obtain

$$\begin{aligned} p_3(u, \dot{u}^*, \dot{u}^{*'}, \dot{u}^{*''}) = \; & p_3(u, \dot{u}, \dot{u}', \dot{u}'') + \Delta\dot{u}\frac{\partial p_3}{\partial \dot{u}}(u, \xi(u), \eta(u), \tau(u)) \\ & + \Delta\dot{u}'\frac{\partial p_3}{\partial \dot{u}'}(u, \xi(u), \eta(u), \tau(u)) \\ & + \Delta\dot{u}''\frac{\partial p_3}{\partial \dot{u}''}(u, \xi(u), \eta(u), \tau(u)), \end{aligned} \tag{17}$$

where $\xi(u)$ is between $\dot{u}$ and $\dot{u}^*$, $\eta(u)$ is between $\dot{u}'$ and $\dot{u}^{*'}$, and $\tau(u)$ is between $\dot{u}''$ and $\dot{u}^{*''}$. So $\xi(u), \eta(u), \tau(u)$ are bounded for $u \in [u_1, u_2]$. Because the partial derivatives of $p_3$ in (17) are all polynomials in $\dot{u}, \dot{u}', \dot{u}''$, there exist constants $F_1$, $F_2$, $F_3$ such that $\forall u \in [u_1, u_2]$:

$$\begin{cases} |\frac{\partial p_3}{\partial \dot{u}}(u, \xi(u), \eta(u), \tau(u))| \le F_1, \\ |\frac{\partial p_3}{\partial \dot{u}'}(u, \xi(u), \eta(u), \tau(u))| \le F_2, \\ |\frac{\partial p_3}{\partial \dot{u}''}(u, \xi(u), \eta(u), \tau(u))| \le F_3. \end{cases} \tag{18}$$

Use (14) (16) (17) (18) to obtain

$$|p_3(u, \dot{u}^*, \dot{u}^{*'}, \dot{u}^{*''})| \le D_3 + C_3\varepsilon,$$

where $C_3 = 2F_1 + B_1F_2 + B_2F_3$. In a similar way, there exist $C_1, C_2, C_4$ such that:

$$|p_1(u, \dot{u}^*, \dot{u}^{*'})| \le D_1 + C_1\varepsilon.$$

$$|p_2(u, \dot{u}^*, \dot{u}^{*'})| \le D_2 + C_2\varepsilon.$$

$$|p_4(u, \dot{u}^*, \dot{u}^{*'}, \dot{u}^{*''})| \le D_4 + C_4\varepsilon.$$

Choosing

$$\varepsilon = \min\{\varepsilon_0/2, (A_x - D_1)/C_1, (A_y - D_2)/C_2, (J_x - D_3)/C_3, (J_y - D_4)/C_4\},$$

it can be shown that $\dot{u}^*$ also satisfies the constraints (11)(12) and the continuity condition. From (10) and $\dot{u}^* \ge \dot{u}$ for $u \in [0, 1]$, $\dot{u}^* > \dot{u}$ for $u \in (u_1, u_2)$, it is easy to show that $\dot{u}^*$ is a better solution, which contradicts the original claim of optimality of $\dot{u}$. So the optimal solution of the problem is "Bang-Bang" or "Bang-Bang-Singular" control.

# 3 Feedrate planning algorithm

## 3.1 Integration trajectory

Since the solution to the optimal problem uses "Bang-Bang" or "Bang-Bang-Singular" control, it is necessary to deduce the parametric velocity function $\dot{u}$ when any of $a_x = \pm A_x$, $a_y = \pm A_y$, $j_x = \pm J_x$, $j_y = \pm J_y$ and $v = V_{max}$ satisfies. Using (1), it is easy to show that once the parametric velocity function $\dot{u}$ in $u$ is known, the parametric acceleration function $\ddot{u}$ in $u$ is determined. Then the two functions $\dot{u}$ and $\ddot{u}$ in $u$ determine a curve in the $u - \dot{u} - \ddot{u}$ space, which is called *integration trajectory*. This subsection will discuss how to compute the parametric velocity function.

Firstly, the solution of parametric velocity function $\dot{u}$ when any axis reaches its jerk bound is considered. For example, if the $x$-axis reaches its jerk bound $J_x$, the following second-order differential equation need to be solved :

$$((x'\dot{u})'\dot{u})'\dot{u} = J_x. \tag{19}$$

Let $f = x'\dot{u}$. The differential equation becomes

$$\frac{d}{dx}(\frac{df}{dx}f)f = J_x.$$

Let $g = \frac{df}{dx}$. It becomes

$$g^2 f + g f^2 \frac{dg}{df} = J_x.$$

Let $h = g^2$. The equation above is

$$\frac{dh}{df} = \frac{2J_x}{f^2} - \frac{2h}{f}.$$

Solve the differential equation to obtain

$$h = \frac{2J_x}{f} - \frac{C_1}{f^2}, \tag{20}$$

where $C_1$ is an integration constant. The above equation can be rewritten as

$$\frac{df}{dx} = \pm \frac{\sqrt{2J_x f - C_1}}{f}.$$

Solve it to obtain

$$x - C_2 = \pm \int \frac{f df}{\sqrt{2J_x f - C_1}} = \pm (C_1 \sqrt{2J_x f - C_1} + \frac{1}{3} \sqrt{2J_x f - C_1}^3)/2J_x^2, \tag{21}$$

where $C_2$ is an integration constant. Solve the equation above to obtain

$$\dot{u} = \frac{1}{2J_x x'} [\omega(U + \sqrt{U^2 + C_1^3}\ )^{\frac{2}{3}} + \omega^2 (U - \sqrt{U^2 + C_1^3}\ )^{\frac{2}{3}} - C_1], \tag{22}$$

where $U = 3J_x^2(x - C_2)$, $\omega^3 = 1$.

Now the expressions of these integration constants $C_1, C_2$ in $u, \dot{u}, \ddot{u}$ will be deduced for the later algorithm. Substitute

$$h = (\frac{df}{dx})^2 = (\frac{x''\dot{u}^2 + x'\ddot{u}}{x'\dot{u}})^2.$$

into (20) to get

$$C_1 = 2J_x f - hf^2 = 2J_x x' \dot{u} - (x''\dot{u}^2 + x'\ddot{u})^2. \tag{23}$$

Use (21) (23) to obtain

$$C_2 = x \pm ((x''\dot{u}^2 + x'\ddot{u})^3 - 3J_x x' \dot{u}(x''\dot{u}^2 + x'\ddot{u}))/3J_x^2. \tag{24}$$

Then from (23) (24), the integration constants $C_1, C_2$ are determined by specifying a known point on the integration trajectory in the $u - \dot{u} - \ddot{u}$ space.

If the $y$-axis reaches its jerk bound $J_y$, solve the parametric velocity function in the same way to get

$$\dot{u} = \frac{1}{2J_y y'} [\omega(U + \sqrt{U^2 + C_1^3}\ )^{\frac{2}{3}} + \omega^2 (U - \sqrt{U^2 + C_1^3}\ )^{\frac{2}{3}} - C_1], \tag{25}$$

where $U = 3J_y^2(y - C_2)$, $\omega^3 = 1$. It is similar that

$$C_1 = 2J_y y' \dot{u} - (y''\dot{u}^2 + y'\ddot{u})^2, \tag{26}$$

$$C_2 = y \pm ((y''\dot{u}^2 + y'\ddot{u})^3 - 3J_y y' \dot{u}(y''\dot{u}^2 + y'\ddot{u}))/3J_y^2. \tag{27}$$

In (22) or (25), if $U^2 + C_1^3$ is negative in some interval of $u$, the expression of $\dot{u}$ should be converted, for the convenience of computation. Taking (25) for example, substitute $\omega$ by $e^{\frac{2}{3}ik\pi}$ $(k = 0, 1, 2)$ to obtain

$$\begin{aligned}
\dot{u} &= \frac{-C_1}{2J_y y'} [e^{\frac{2}{3}ik\pi} (\frac{U}{(-C_1)^{3/2}} + i\sqrt{1 - \frac{U^2}{(-C_1)^3}})^{2/3} + e^{-\frac{2}{3}ik\pi} (\frac{U}{(-C_1)^{3/2}} - i\sqrt{1 - \frac{U^2}{(-C_1)^3}})^{2/3} + 1] \\
&= \frac{-C_1}{2J_y y'} [e^{\frac{2}{3}ik\pi} e^{\frac{2}{3}i \arccos \frac{U}{(-C_1)^{3/2}}} + e^{-\frac{2}{3}ik\pi} e^{-\frac{2}{3}i \arccos \frac{U}{(-C_1)^{3/2}}} + 1] \\
&= \frac{-C_1}{2J_y y'} [2 \cos \frac{2}{3} (\arccos \frac{U}{(-C_1)^{3/2}} + k\pi) + 1].
\end{aligned}$$

If the $x$ (or $y$)-axis reaches its jerk bound $-J_x$ (or $-J_y$), it just need to replace $J_x$ (or $J_y$) by $-J_x$ (or $-J_y$) in the solutions above. In the $u$-$\dot{u}$-$\ddot{u}$ space, the integration trajectories determined by $j_x = \pm J_x$ or $j_y = \pm J_y$ are called *type one integration trajectory* (abbr. $\text{ITR}_1$).

Secondly, the situation when any axis reaches its acceleration bound is considered. For example, if the $x$-axis reaches its acceleration bound $A_x$, the following first-order differential equation need to be solved:

$$(x'\dot{u})'\dot{u} = A_x. \tag{28}$$

The above first-order ODE is solved following [1, 2, 3, 4]. Multiplying $x'$ to both sides of (28), it becomes

$$x'\dot{u}(x'\dot{u})' = A_x x'.$$

The above equation can be rewritten as

$$\frac{d}{du}(x'\dot{u})^2 = 2A_x x'.$$

Solve it to obtain

$$(x'\dot{u})^2 = 2A_x x + C_0.$$

Then the solution is

$$\dot{u} = \frac{\sqrt{2A_x x + C_0}}{|x'|}, \tag{29}$$

where the integration constant $C_0 = (x'\dot{u})^2 - 2A_x x$ at a known point $(u, \dot{u})$ on the trajectory. The solutions of $a_x = -A_x$ and $a_y = \pm A_y$ are similar. In the $u$-$\dot{u}$-$\ddot{u}$ space, the integration trajectories determined by $a_x = \pm A_x$ or $a_y = \pm A_y$ are called *type two integration trajectory* (abbr. $\text{ITR}_2$).

Finally, it is easy to know that when $v = V_{max}$ satisfies, the parametric velocity function is

$$\dot{u} = V_{max}/\sigma. \tag{30}$$

The trajectory determined by $v = V_{max}$ in the $u$-$\dot{u}$-$\ddot{u}$ space is denoted by $\text{ITR}_3$. Note that the $\text{ITR}_3$ determines a unique curve in the $u$-$\dot{u}$-$\ddot{u}$ space.

## 3.2 Velocity limit surface

Before proposing the algorithm for feedrate planning along curved tool paths, three kinds of velocity limit surfaces in the $u$-$\dot{u}$-$\ddot{u}$ space due to the velocity, acceleration, and jerk constraints need to be deduced. The velocity switching curves on the velocity limit surfaces and control axis switching surfaces will also be introduced in this subsection.

Use (9) to rewrite the jerk limits to be constraints of the parametric jerk $\dddot{u}$:

(a) When $x'y' \neq 0$, the jerk limits are equivalent to

$$\begin{cases} f_1(u, \dot{u}, \ddot{u}) \leq \dddot{u} \leq g_1(u, \dot{u}, \ddot{u}), \\ f_2(u, \dot{u}, \ddot{u}) \leq \dddot{u} \leq g_2(u, \dot{u}, \ddot{u}), \end{cases} \tag{31}$$

where

$$f_1(u,\dot u,\ddot u)=\begin{cases}(-J_x-x'''\dot u^3-3x''\dot u\ddot u)/x' & x'>0;\\ (J_x-x'''\dot u^3-3x''\dot u\ddot u)/x' & x'<0.\end{cases}$$

$$g_1(u,\dot u,\ddot u)=\begin{cases}(J_x-x'''\dot u^3-3x''\dot u\ddot u)/x' & x'>0;\\ (-J_x-x'''\dot u^3-3x''\dot u\ddot u)/x' & x'<0.\end{cases}$$

$$f_2(u,\dot u,\ddot u)=\begin{cases}(-J_y-y'''\dot u^3-3y''\dot u\ddot u)/y' & y'>0;\\ (J_y-y'''\dot u^3-3y''\dot u\ddot u)/y' & y'<0.\end{cases}$$

$$g_2(u,\dot u,\ddot u)=\begin{cases}(J_y-y'''\dot u^3-3y''\dot u\ddot u)/y' & y'>0;\\ (-J_y-y'''\dot u^3-3y''\dot u\ddot u)/y' & y'<0.\end{cases}$$

Let

$$\begin{cases}J_-(u,\dot u,\ddot u)=\max\{f_1,f_2\},\\ J_+(u,\dot u,\ddot u)=\min\{g_1,g_2\}.\end{cases}\tag{32}$$

Then constraints (31) become

$$J_-(u,\dot u,\ddot u)\le \dddot u\le J_+(u,\dot u,\ddot u).\tag{33}$$

It shows that in every point of the $u$-$\dot u$-$\ddot u$ space, $\dddot u$ has an upper bound $J_+$ and a lower bound $J_-$.

(b) When $x'=0$, the jerk limits become

$$\begin{cases}-J_x\le x'''\dot u^3+3x''\dot u\ddot u\le J_x,\\ f_2(u,\dot u,\ddot u)\le \dddot u\le g_2(u,\dot u,\ddot u).\end{cases}\tag{34}$$

The first equation of (34) indicates the range of $(\dot u,\ddot u)$ on the $u$ section where $u$ satisfies $x'=0$. The range is limited by two curves $x'''\dot u^3+3x''\dot u\ddot u=-J_x$ and $x'''\dot u^3+3x''\dot u\ddot u=J_x$ in the $u$-$\dot u$-$\ddot u$ space. These curves are called *type one velocity switching curve* (abbr. $\mathrm{VSC}_1$). The second equation of (34) still shows the upper and lower bounds of $\dddot u$, where now $J_+=g_2, J_-=f_2$.

(c) When $y'=0$, the analysis to the following equations are similar:

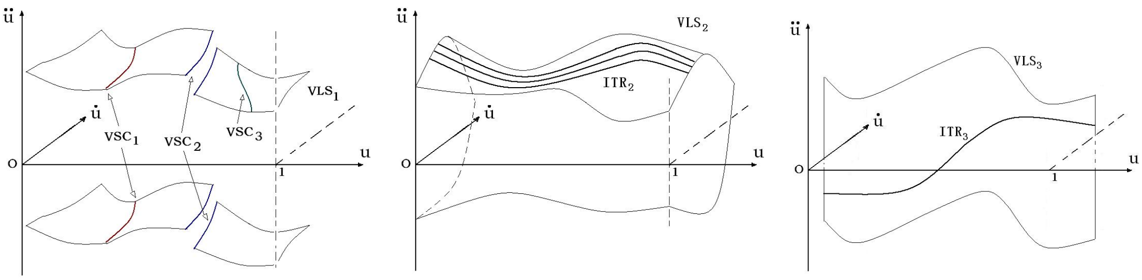
$$\begin{cases}f_1(u,\dot u,\ddot u)\le \dddot u\le g_1(u,\dot u,\ddot u),\\ -J_y\le y'''\dot u^3+3y''\dot u\ddot u\le J_y.\end{cases}\tag{35}$$

The first equation of (35) shows the upper and lower bounds of $\dddot u$, where $J_+=g_1, J_-=f_1$. The second equation of (35) indicates the range of $(\dot u,\ddot u)$ on the $u$ section where $u$ satisfies $y'=0$. It is limited by two curves $y'''\dot u^3+3y''\dot u\ddot u=-J_y$ and $y'''\dot u^3+3y''\dot u\ddot u=J_y$ in the $u$-$\dot u$-$\ddot u$ space. These curves are also $\mathrm{VSC}_1$.

Let $J_-(u,\dot u,\ddot u)$ and $J_+(u,\dot u,\ddot u)$ be the expressions defined in (32). The surface $J_-(u,\dot u,\ddot u)=J_+(u,\dot u,\ddot u)$ is called *type one velocity limit surface* (abbr. $\mathrm{VLS}_1$). Obviously, the integration trajectories cannot go beyond the $\mathrm{VLS}_1$.

Using (7), the acceleration limits are

$$\begin{cases}-A_x\le a_x(u,\dot u,\ddot u)\le A_x,\\ -A_y\le a_y(u,\dot u,\ddot u)\le A_y.\end{cases}\tag{36}$$




(a) $\mathrm{VLS}_1$ and VSC (b) $\mathrm{VLS}_2$ and $\mathrm{ITR}_2$ (c) $\mathrm{VLS}_3$ and $\mathrm{ITR}_3$

Figure 2: Three kinds of velocity limit surfaces.

The surfaces $a_x(u, \dot{u}, \ddot{u}) = \pm A_x$ and $a_y(u, \dot{u}, \ddot{u}) = \pm A_y$ are called *type two velocity limit surface* (abbr. $\mathrm{VLS}_2$). The integration trajectories also cannot go beyond the $\mathrm{VLS}_2$. However, it is easy to see that the $\mathrm{ITR}_2$ are on the $\mathrm{VLS}_2$. Actually, from any point of the $\mathrm{VLS}_2$, there exists an $\mathrm{ITR}_2$ on the $\mathrm{VLS}_2$.

The tangential velocity limit (5) induces the *type three velocity limit surface* (abbr. $\mathrm{VLS}_3$)

$$v = \sigma \dot{u} = V_{max}, \tag{37}$$

which is a cylinder in the $u$-$\dot{u}$-$\ddot{u}$ space. The integration trajectories also cannot go beyond the $\mathrm{VLS}_3$. Obviously, the unique $\mathrm{ITR}_3$ is on the $\mathrm{VLS}_3$.

Now there are three kinds of VLS, which are all algebraic surfaces in the following region of the $u$-$\dot{u}$-$\ddot{u}$ space:

$$D = \{(u, \dot{u}, \ddot{u}) | 0 \le u \le 1, \dot{u} \ge 0\}.$$

It is stated above that the integration trajectories cannot go beyond any of the three kinds of VLS, that is, the integration trajectories can only be planned in the region determined by

$$\begin{aligned}
&J_-(u, \dot{u}, \ddot{u}) \le J_+(u, \dot{u}, \ddot{u}), \\
&-A_x \le a_x(u, \dot{u}, \ddot{u}) \le A_x, -A_y \le a_y(u, \dot{u}, \ddot{u}) \le A_y, \\
&\sigma \dot{u} \le V_{max}.
\end{aligned}$$

Intuitively, this region is just a part of $D$ divided by the VLS, which contains $(0, 0, 0), (1, 0, 0)$ (see Fig. 2).

Besides $\mathrm{VSC}_1$ defined above, there are two kinds of velocity switching curves $\mathrm{VSC}_2$ and $\mathrm{VSC}_3$ on the $\mathrm{VLS}_1$. Since the tool path is considered to be a piecewise $C^2$ curve, there may exist discontinuities for $x'''$ or $y'''$. From (31) (32), they will cause discontinuities of the $\mathrm{VLS}_1$ along certain curves, which are called $\mathrm{VSC}_2$. Because each segment of the piecewise parametric curve is infinitely differentiable, the discontinuities for $x'''$ or $y'''$ can only occur in the nodes or connection points of the piecewise parametric curve.

Besides, the set of points (in fact, curves) where the $\mathrm{ITR}_1$ are tangent to the $\mathrm{VLS}_1$ are called $\mathrm{VSC}_3$. For $i = 1, j = 2$ or $i = 2, j = 1$, the $\mathrm{ITR}_1$ which are tangent to $f_i = g_j$ are just the solutions of $\dddot{u} = f_i$ or $\dddot{u} = g_j$. Differentiate $f_i - g_j = 0$ with respect to $u$, and use (1) (2) to obtain:

$$\frac{\partial}{\partial u}(f_i - g_j) + \frac{\ddot{u}}{\dot{u}}\frac{\partial}{\partial \dot{u}}(f_i - g_j) + \frac{\dddot{u}}{\dot{u}}\frac{\partial}{\partial \ddot{u}}(f_i - g_j) = 0.$$

Substitute $\dddot{u} = f_i$ into the equation above to obtain

$$\dot{u}\frac{\partial}{\partial u}(f_i - g_j) + \ddot{u}\frac{\partial}{\partial \dot{u}}(f_i - g_j) + f_i\frac{\partial}{\partial \ddot{u}}(f_i - g_j) = 0. \tag{38}$$

The intersection of (38) and the $\mathrm{VLS}_1$: $f_i = g_j$ is the $\mathrm{VSC}_3$.

It will be shown how to decide the control axis. There are two problems: determining control axis at the starting point and axis switching during the motion.

If integrate $\dddot{u} = J_+(u, \dot{u}, \ddot{u})$ forward from $(0, 0, 0)$ in the $u$-$\dot{u}$-$\ddot{u}$ space as the current integration trajectory, it is easy to determine the control axis from $(u, \dot{u}, \ddot{u}) = (0, 0, 0)$ and the three cases (a),(b),(c) in section 3.2. For example, when $x'(0) > 0, y'(0) > 0$, $x$-axis is the control axis if and only if $g_1(0, 0, 0) = J_x/x' < g_2(0, 0, 0) = J_y/y'$.

From (32), when integrate $\dddot{u} = J_+(u, \dot{u}, \ddot{u})$, the expression of the parametric velocity may change if the values of $g_1, g_2$ vary. It means that the control axis should be switched. So $g_1 = g_2$ is called the *control axis switching surface* (abbr. CASS). For example, if the integration trajectory passes through a CASS from the region $g_1 < g_2$ to the region $g_1 > g_2$, the control axis should be switched from $x$ to $y$, and vice versa.

The situation is similar when integrating $\dddot{u} = J_-(u, \dot{u}, \ddot{u})$, and the CASS is then $f_1 = f_2$. When the integration trajectory passes through the CASS from the region $f_1 > f_2$ to the region $f_1 < f_2$, the control axis should be switched from $x$ to $y$, and vice versa. It will not be mentioned about how to deal with the CASS when integrating $\dddot{u} = J_+(u, \dot{u}, \ddot{u})$ or $\dddot{u} = J_-(u, \dot{u}, \ddot{u})$ in the later algorithm.

### 3.3 Feedrate planning algorithm

The feedrate planning algorithm is designed under a "greedy rule": use the maximal parametric jerk $\dddot{u}$ as much as possible, that is, use the minimal parametric jerk and singular control only when it has to. The optimal feedrate planning problem with acceleration bounds [1, 2, 3, 4] uses a similar rule, but the difference is that it cannot be proved that the "greedy rule" generates a globally optimal solution for the problem. It will be discussed in the conclusion.

Firstly, the framework of the feedrate planning algorithm under confined jerk is presented. The specific computational methods in the algorithm will be given later.

**Algorithm FP_CJ.** Feedrate planning with velocity, acceleration and jerk constraints.
**Input**: $\vec{r}(u) = (x(u), y(u)), 0 \le u \le 1$; $V_{max}, A_x, A_y, J_x, J_y$.
**Output**: The integration trajectory for $u \in [0, 1]$.

Step0: Let $S = (0, 0, 0)$.

Step1: Generate a $J_+$ trajectory in the $u$-$\dot{u}$-$\ddot{u}$ space with the maximal jerk by integrating $\dddot{u} = J_+(u, \dot{u}, \ddot{u})$ forward from $S$, until the trajectory (if it passes through the CASS first, then change the control axis as previously mentioned) intersects the VLS. Denote the first intersection point of the $J_+$ trajectory and the VLS by $R$. If the $J_+$ trajectory does not intersect the VLS before $u = 1$, then a forward trajectory for $u \in [0, 1]$ is obtained . Denote its parametric velocity function by $\dot{u}_f$, and go to step4.

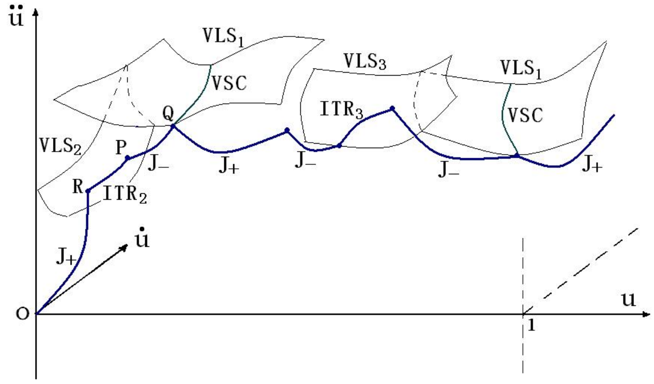



Figure 3: This illustrative forward integration trajectory consists of eight segments. The first segment $OR$ is a trajectory with maximal jerk. The second segment $RP$ is an $\mathrm{ITR}_2$ which meets a $\mathrm{VLS}_1$. The third segment $PQ$ is a trajectory with minimal jerk started from a VSC. The fourth segment is also a maximal trajectory which meets a $\mathrm{VLS}_3$ but not on $\mathrm{ITR}_3$. The fifth segment is a minimal trajectory started from the $\mathrm{ITR}_3$. The sixth segment is an $\mathrm{ITR}_3$ which meets $\mathrm{VLS}_1$. The seventh segment is also a minimal trajectory started from a VSC. The eighth segment is a maximal trajectory ending at $u = 1$.

Step2: Consider three cases (See Fig. 3 for an illustration):

**1)** If $R \in \mathrm{VLS}_2$, then generate an $\mathrm{ITR}_2$ from $R$ on this $\mathrm{VLS}_2$. There are two cases. If the $\mathrm{ITR}_2$ intersects the $\mathrm{VLS}_1$ or $\mathrm{VLS}_3$ at a point $T$, then add the $\mathrm{ITR}_2$ between $R$ and $T$ to the forward velocity function, set $R = T$, and goto step2. If the $\mathrm{ITR}_2$ terminates at $u = 1$, then add the $\mathrm{ITR}_2$ to the forward velocity function and go to step4.

**2)** If $R$ is on $\mathrm{VLS}_3$ and $\mathrm{ITR}_3$, then set the trajectory after $R$ to be the $\mathrm{ITR}_3$ until the $\mathrm{ITR}_3$ intersects the $\mathrm{VLS}_1$, $\mathrm{VLS}_2$ at a point $T$, or the $\mathrm{ITR}_3$ terminates at $u = 1$. In the first case, add the $\mathrm{VLS}_3$ to the velocity function, set $R = T$, and goto step 2. In the second case, add the $\mathrm{ITR}_3$ to the velocity function and go to step4.

**3)** Now $R \in \mathrm{VLS}_1$ or $R \in \mathrm{VLS}_3 \setminus \mathrm{ITR}_3$, which means it cannot continue to integrate with the maximal jerk. Generate a $J_-$ trajectory by integrating $\dddot{u} = J_-(u, \dot{u}, \ddot{u})$ backward from each point on VSC or the $\mathrm{ITR}_3$ after $R$. If the $J_-$ trajectory starting from point $Q$ on a VSC or the $\mathrm{ITR}_3$ intersects the previous trajectory at point $P$, where $P$ has the greatest parameter $u$, then update the trajectory between $P$ and $Q$ to be the $J_-$ trajectory from $P$ to $Q$.

Step3: Let $S = Q$. Iterate the process of steps 1-3 until $u = 1$. Denote the parametric velocity function of the whole forward trajectory by $\dot{u}_f$ (see Fig. 3).

Step4: Generate a backward trajectory in the $u$-$\dot{u}$-$\ddot{u}$ space starting from $(1, 0, 0)$ in a similar way as steps 1-3 until $u = 0$. Denote the parametric velocity function of the backward trajectory by $\dot{u}_b$.

Step5: Connect the two trajectories of $\dot{u}_f$ and $\dot{u}_b$ by $J_-$ trajectories. A complete integration trajectory for $u \in [0, 1]$ is obtained.

**Remark**. The "greedy" rule is used in step 2. When the trajectory meets the $VLS_1$ at a point, it will pass through the $VLS_1$ and violate the limits if it continues to use the same jerk control. In other words, it has to decelerate before this happens. According to the definition, the integration trajectory can meet the VSC at a point in a VSC. That is why the algorithm generates a $J_-$ integration trajectory starting from a point in the next VSC and try to use this trajectory to decelerate. When the trajectory meets the $VLS_2$ and $VLS_3$, it tries to be on the VLS since the $ITR_2$ are on the $VLS_2$ and the $ITR_3$ is on the $VLS_3$. In other words, the algorithm uses the maximal parametric jerk to accelerate as long as possible and then uses the minimal parametric jerk and singular control such that the velocity, acceleration and jerk limits (VLS) are not violated.

Concrete computational methods of step2 and step5 in the algorithm are given below. It will be shown how to connect two $J+$ trajectories by $J_-$ trajectories in step5 firstly.

For step5, because the control of the forward and backward trajectories may have been switched for several times, $\dot{u}_f$ and $\dot{u}_b$ are both piecewise-analytic functions. It is needed to traverse and choose each analytic segment of the forward and backward trajectories respectively, and to connect these two segments by a $J_-$ trajectory if there exists such a solution. After choosing one segment in $\dot{u}_f$ and $\dot{u}_b$ respectively, there are two cases:

1) The $J_-$ trajectory for connection does not pass through CASS. Assume the $J_-$ trajectory starts from point $(u_1, \dot{u}_f(u_1), \ddot{u}_f(u_1))$ on the forward trajectory to point $(u_2, \dot{u}_b(u_2), \ddot{u}_b(u_2))$ on the backward trajectory in the $u$-$\dot{u}$-$\ddot{u}$ space. From (23) (24) or (26) (27), the integration constants of the $J_-$ trajectory can be expressed as $C_1(u, \dot{u}, \ddot{u}), C_2(u, \dot{u}, \ddot{u})$. The following algebraic equation system

$$\begin{cases} C_1(u_1, \dot{u}_f(u_1), \ddot{u}_f(u_1)) = C_1(u_2, \dot{u}_b(u_2), \ddot{u}_b(u_2)), \\ C_2(u_1, \dot{u}_f(u_1), \ddot{u}_f(u_1)) = C_2(u_2, \dot{u}_b(u_2), \ddot{u}_b(u_2)) \end{cases} \tag{39}$$

need to be solved to obtain $u_1, u_2$. Then the integration constants of the $J_-$ connection trajectory are $C_1(\bar{u}_1,\, \dot{u}_f(\bar{u}_1), \ddot{u}_f(\bar{u}_1))$, $C_2(\bar{u}_1, \dot{u}_f(\bar{u}_1), \ddot{u}_f(\bar{u}_1))$, where $\bar{u}_1$ is a solution of equation (39). Then the $J_-$ trajectory for the connection in step5 is obtained.

2) The $J_-$ trajectory for connection passes through an CASS. Now the expressions of the $J_-$ trajectory and its integration constants are different in the two sides of the CASS. Suppose the left side is controlled by $j_x = -J_x$ and the right side is controlled by $j_y = -J_y$. Denote the integration constants of the $j_x = -J_x$ trajectory by $C_1^x, C_2^x$ and the integration constants of $j_y = -J_y$ trajectory by $C_1^y, C_2^y$. Assume the $J_-$ trajectory for connection passes through the CASS at the point $(u_c, \dot{u}_c, \ddot{u}_c)$, and it starts from the point $(u_l, \dot{u}_f(u_l), \ddot{u}_f(u_l))$

on the forward trajectory to the point $(u_r, \dot{u}_b(u_r), \ddot{u}_b(u_r))$ on the backward trajectory. Then, the following algebraic equation system

$$\begin{cases} f_1(u_c, \dot{u}_c, \ddot{u}_c) = f_2(u_c, \dot{u}_c, \ddot{u}_c), \\ C_1^x(u_l, \dot{u}_f(u_l), \ddot{u}_f(u_l)) = C_1^x(u_c, \dot{u}_c, \ddot{u}_c), \\ C_2^x(u_l, \dot{u}_f(u_l), \ddot{u}_f(u_l)) = C_2^x(u_c, \dot{u}_c, \ddot{u}_c), \\ C_1^y(u_r, \dot{u}_b(u_r), \ddot{u}_b(u_r)) = C_1^y(u_c, \dot{u}_c, \ddot{u}_c), \\ C_2^y(u_r, \dot{u}_b(u_r), \ddot{u}_b(u_r)) = C_2^y(u_c, \dot{u}_c, \ddot{u}_c) \end{cases} \tag{40}$$

need to be solved to obtain $u_l, u_c, u_r, \dot{u}_c, \ddot{u}_c$ and the two sets of integration constants of the $J_-$ trajectory for the connection: $C_1^x(u_c, \dot{u}_c, \ddot{u}_c), C_2^x(u_c, \dot{u}_c, \ddot{u}_c)$ and $C_1^y(u_c, \dot{u}_c, \ddot{u}_c), C_2^y(u_c, \dot{u}_c, \ddot{u}_c)$. It is similar to deal with the case when the $J_-$ trajectory passes through the CASS more than once.

In general, the solutions of the above equation systems are finite. It just need to compare these solutions to get an optimal one according to the machining time in (10).

For step2, denote the parametric velocity function of the previous trajectory by $\dot{u}_1$. There are two cases:

1) Point $Q$ is on a VSC$_1$ or a VSC$_2$. If $u = u_0$ at $Q$, assume the coordinate of $Q$ is $(u, \dot{u}, \ddot{u}) = (u_0, b, c)$ and denote the expression of VSC$_1$ or VSC$_2$ on the $u_0$ section by $h_1(\dot{u}, \ddot{u}) = 0$ as previously mentioned. Assume $u = a$ at $P$. Then the coordinate of $P$ is $(a, \dot{u}_1(a), \ddot{u}_1(a))$. The integration constants of the $J_-$ trajectory are $C_1(u, \dot{u}, \ddot{u}), C_2(u, \dot{u}, \ddot{u})$ as above. It just need to solve the following algebraic equation system

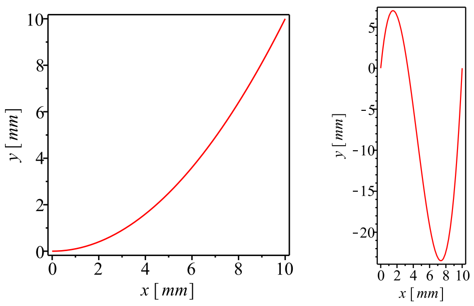
$$\begin{cases} C_1(u_0, b, c) = C_1(a, \dot{u}_1(a), \ddot{u}_1(a)), \\ C_2(u_0, b, c) = C_2(a, \dot{u}_1(a), \ddot{u}_1(a)), \\ h_1(b, c) = 0 \end{cases} \tag{41}$$

to obtain $a, b, c$. If the equation system has more than one solutions or there are more VSCs, the solution with maximal parametric $a$ should be chosen according to the "greedy rule". The equation systems occurring later will be dealt with in the same way. The integration constants of the $J_-$ trajectory can be computed by $C_1(u_0, b, c), C_2(u_0, b, c)$. Then the $J_-$ trajectory in step2 is obtained.

2) Point $Q$ is on a VSC$_3$ (or the ITR$_3$, the case is similar). Assume the coordinate of $Q$ to be $(d_0, b_0, c_0)$. From (38), denote the VSC$_3$ by $\{(u, \dot{u}, \ddot{u}) | h_2(u, \dot{u}, \ddot{u}) = 0, h_3(u, \dot{u}, \ddot{u}) = 0\}$. Assume $u = a_0$ at $P$. Then the coordinate of $P$ is $(a_0, \dot{u}_1(a_0), \ddot{u}_1(a_0))$. The integration constants of the $J_-$ trajectory can also be expressed as $C_1(u, \dot{u}, \ddot{u})$ and $C_2(u, \dot{u}, \ddot{u})$. It just need to solve the following algebraic equation system

$$\begin{cases} C_1(d_0, b_0, c_0) = C_1(a_0, \dot{u}_1(a_0), \ddot{u}_1(a_0)), \\ C_2(d_0, b_0, c_0) = C_2(a_0, \dot{u}_1(a_0), \ddot{u}_1(a_0)), \\ h_2(d_0, b_0, c_0) = 0, \\ h_3(d_0, b_0, c_0) = 0 \end{cases} \tag{42}$$

to obtain $a_0, d_0, b_0, c_0$. The integration constants of the $J_-$ trajectory are $C_1(d_0, b_0, c_0)$ and $C_2(d_0, b_0, c_0)$. If the $J_-$ trajectory passes through a CASS between $P$ and $Q$, use the method mentioned above for case 2) of step5 to deal with this situation.




(a) A parabola (b) A cubic curve

Figure 4: The tool paths of the examples.

So far, a complete integration trajectory is obtained. Its parametric velocity function satisfies (11)(12) and the "greedy rule".

The algorithm generates a unique and optimal feedrate planning along specific tool paths with velocity, acceleration and jerk constraints under the "greedy rule".

# 4 Experimental results

In this section, experimental results are presented to compare the machining results with confined jerk and confined acceleration.

## 4.1 Computing the feedrate integration trajectory

In this section, the following two examples are used to illustrate the feedrate planning algorithm. First, a simple tool path (see Fig. 4(a)) with only jerk limits is given to show how the algorithm works.

**Example 1.**

$$\vec{r}(u) = (10u, 10u^2), 0 \le u \le 1,$$

$$J_x = J_y = 10^4 \ mm/s^3.$$

The algorithm has the following steps:

1) Firstly, compute the $\mathrm{VLS}_1$ and the CASS :

Fig. 5(a): $\mathrm{VLS}_1$ $\quad J_-(u,\dot{u},\ddot{u}) = J_+(u,\dot{u},\ddot{u})$;

Fig. 5(b): CASS of maximal parametric jerk $g_1(u,\dot{u},\ddot{u}) = g_2(u,\dot{u},\ddot{u})$;

Fig. 5(c): CASS of minimal parametric jerk $f_1(u,\dot{u},\ddot{u}) = f_2(u,\dot{u},\ddot{u})$.

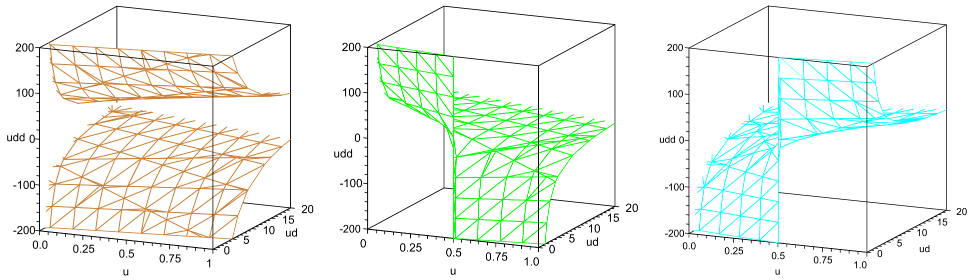
Then, compute the three kinds of VSC :

$\mathrm{VSC}_1$: $\{(0,\dot{u},\ddot{u}) \mid 10^4 - 60\dot{u}\ddot{u} = 0\}$ and $\{(0,\dot{u},\ddot{u}) \mid 10^4 + 60\dot{u}\ddot{u} = 0\}$;

$\mathrm{VSC}_3$: $\{(u,\dot{u},\ddot{u}) \mid 10^4 - 60\dot{u}\ddot{u} + 2\cdot 10^4 u = 0, 4\cdot 10^4\dot{u} - 30\ddot{u}^2 = 0\}$ and $\{(u,\dot{u},\ddot{u}) \mid 10^4 + 60\dot{u}\ddot{u} + 2\cdot 10^4 u = 0, 4\cdot 10^4\dot{u} + 30\ddot{u}^2 = 0\}$;

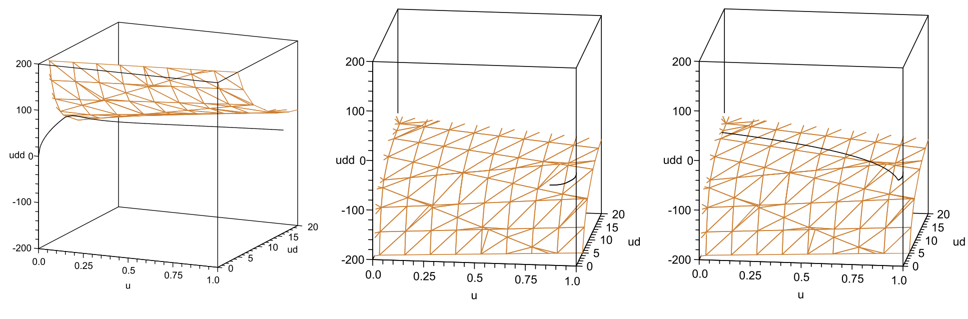
$\mathrm{VSC}_2$ does not exist here.

2) Generate a $J_+$ trajectory forward from $(0,0,0)$. The trajectory is controlled by $j_x = J_x$ in the beginning, then intersects the CASS $g_1 = g_2$ at $u = 0.05$ and switches to the control




(a) $VLS_1$ (b) CASS (c) CASS

Figure 5: The $VLS_1$ and the CASS, where $ud = \dot{u}, udd = \ddot{u}$. The unit for $\dot{u}$ is $s^{-1}$. The unit for $\ddot{u}$ is $s^{-2}$.




(a) (b) (c)

Figure 6: Forward integration trajectory: (a); backward integration trajectory: (b) and (c). In the figure, $ud = \dot{u}, udd = \ddot{u}$. The unit for $\dot{u}$ is $s^{-1}$. The unit for $\ddot{u}$ is $s^{-2}$.

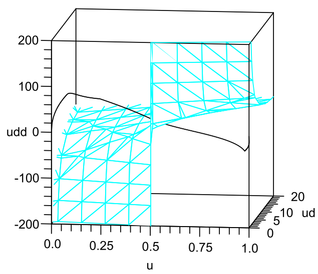
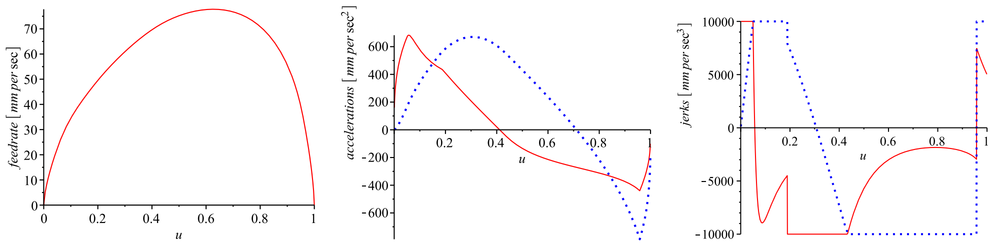



Figure 7: Connect the forward and backward trajectories, where $ud = \dot{u}, udd = \ddot{u}$. The unit for $\dot{u}$ is $s^{-1}$. The unit for $\ddot{u}$ is $s^{-2}$.




(a) Feedrate (b) Solid: $a_x$; Dotted: $a_y$ (c) Solid: $j_x$; Dotted: $j_y$.

Figure 8: Tangential velocity, $x, y$ accelerations and jerks in $u$. The horizontal axis is the parameter $u \in [0, 1]$.

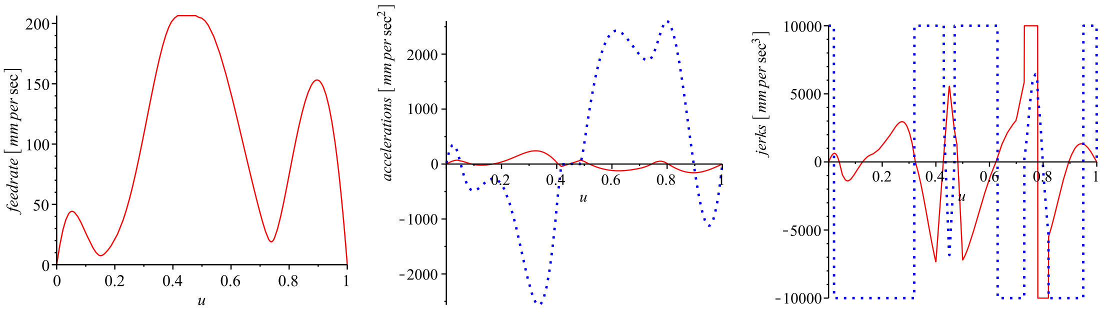
of $j_y = J_y$. It will not intersect the VLS$_1$ or the CASS before reaching $u = 1$ (see Fig. 6(a)). The parametric velocity function of the forward integration trajectory is:

$$\dot{u}_f = \begin{cases} 5(6u)^{\frac{2}{3}}, & 0 \le u \le 0.05; \\ \frac{5}{2u}((\sqrt{9u^4 - 0.5625 \cdot 10^{-2}u^2 + 0.5625 \cdot 10^{-5}} + 3u^2 - 0.9375 \cdot 10^{-3})^{\frac{2}{3}} & \\ +(\sqrt{9u^4 - 0.5625 \cdot 10^{-2}u^2 + 0.5625 \cdot 10^{-5}} - 3u^2 + 0.9375 \cdot 10^{-3})^{\frac{2}{3}} & \\ -0.0168), & 0.05 \le u \le 1. \end{cases}$$

3) Generate a $J_+$ trajectory backward from $(1, 0, 0)$. The trajectory $\dot{u}_b$ is controlled by $j_y = J_y$ in the beginning. It intersects the the CASS $g_1 = g_2$ at $u = 0.9253$, then switches to the control of $j_x = J_x$. Then it intersects the VLS$_1$ at $u = 0.8104$ (see Fig. 6(b)). Now, execute step2 of the algorithm by solving the equation system (41). The only solution is a $j_y = -J_y$ trajectory from the point $(u, \dot{u}, \ddot{u}) = (0, 7.528, -22.14)$ on the VSC$_1$ at $u = 0$ to the point $(0.9561, 1.679, -44.86)$ on trajectory $\dot{u}_b$ (see Fig. 6(c)). Then the the backward integration trajectory is:

$$\dot{u}_b = \begin{cases} \frac{5}{2u}(6(1-u^2))^{\frac{2}{3}}, & 0.9561 \le u \le 1; \\ \frac{3.211}{u}(2\sin(\frac{2}{3}\arccos(2.0607u^2 - 1) + \frac{1}{6}\pi) - 1), & 0 \le u \le 0.9561. \end{cases}$$

4) Connect the integration trajectories of $\dot{u}_f$ and $\dot{u}_b$ by $J_-$ trajectories. Solving the previous equation system (40), the only solution is that the $J_-$ trajectory connects the




(a) Feedrate (b) Solid: $a_x$; Dotted: $a_y$ (c) Solid: $j_x$; Dotted: $j_y$

Figure 9: Tangential velocity, $x, y$ accelerations and jerks in $u$. The horizontal axis is the parameter $u \in [0, 1]$

integration trajectories of the second segment of $\dot{u}_f$ and the first segment of $\dot{u}_b$, and it intersects the CASS at $u = 0.4336$ (see Fig. 7). It is controlled by $j_x = -J_x$ for $u \in [0.1893, 0.4336]$ and by $j_y = -J_y$ for $u \in [0.4336, 0.9580]$. Then the parametric velocity function of the complete integration trajectory is:

$$\dot{u} = \begin{cases} 5(6u)^{\frac{2}{3}}, & 0 \le u \le 0.05; \\ \frac{5}{2u}((\sqrt{9u^4 - 0.5625 \cdot 10^{-2}u^2 + 0.5625 \cdot 10^{-5}} + 3u^2 - 0.9375 \cdot 10^{-3})^{\frac{2}{3}} & \\ +(\sqrt{9u^4 - 0.5625 \cdot 10^{-2}u^2 + 0.5625 \cdot 10^{-5}} - 3u^2 + 0.9375 \cdot 10^{-3})^{\frac{2}{3}} & \\ -0.0168), & 0.05 \le u \le 0.1893; \\ 5.44(2\sin(\frac{2}{3}\arccos(2.6437u - 1.0877) + \frac{1}{6}\pi) - 1), & 0.1893 \le u \le 0.4336; \\ \frac{3.124}{u}(2\sin(\frac{2}{3}\arccos(2.1476u^2 - 1.0869) + \frac{1}{6}\pi) - 1), & 0.4336 \le u \le 0.9580; \\ \frac{5}{2u}(6(1-u^2))^{\frac{2}{3}}, & 0.9580 \le u \le 1. \end{cases}$$

The smooth feedrate is shown in Fig. 8(a). From Fig. 8(b), the accelerations are continuous and from Fig. 8(c), the solution is "Bang-Bang" control. The five segments of the integration trajectory are respectively controlled by $J_x$, $J_y$, $-J_x$, $-J_y$, and $J_y$ in the $u_+$ direction.

A more complex tool path with sharp turns (see Fig. 4(b)) is also shown. The tangential velocity bound is added to this example such that the solution is "Bang-Bang-Singular" control.

**Example 2.**

$$\vec{r}(u) = (10u, 100(3u^3 - 4u^2 + u)), 0 \le u \le 1,$$

$$J_x = J_y = 10^4 \ mm/s^3, V_{max} = 200 \ mm/s.$$

The smooth feedrate, $x, y$ accelerations and jerks in $u$ are shown in Fig. 9. The solution is "Bang-Bang-Singular" control. The segments of the trajectory are respectively controlled by $J_y$, $-J_y$, $J_y$, $V_{max}$, $J_y$, $-J_y$, $J_x$, $-J_x$, $-J_y$, and $J_y$ in the $u_+$ direction.

### 4.2 Experimental results

In this section, real CNC machining experiments are contacted to compare the interpolation algorithm with confined jerk presented in this paper and the optimal interpolation algorithm with confined acceleration presented in [3].

The experiment consists of three steps. Firstly, the feedrate integration trajectory $v(u)$ is computed with Algorithm **FP_CJ** for a given tool path $C(u), u \in [0, 1]$. Secondly, the interpolation points of the tool path are computed with the feedrate $v(u)$ and a given sampling period $T$. Finally, the interpolation points are used to manufacture the tool path on a CNC machine. The first and second steps are performed off-line.

Note that the above procedure is only used for the convenience of testing the algorithm. In order to use the algorithm in real CNC controllers, the following approach could be adopted. The feedrate trajectory $v(u)$ is computed off-line with Algorithm **FP_CJ**. Then new G codes are generated to include information about the feedrate trajectory. Finally, the CNC controller is modified to accept the new G codes. If the expression for $v(u)$ is too complicated, a simpler function, such as the quadratic B-spline, will be used to fit $v(u)$ and then used as the new feedrate function. This strategy is adopted by many existing work such as [5, 6, 13, 15]. In particular, industrial CNC machining is realized in [5, 6, 15] using this approach.

Before describing the experiment, a procedure is given to compute the interpolation points of the tool path when the feedrate $v(u)$ and a sampling period $T$ are given. If $u_k$ is the parameter value for the $k$-th interpolation point to be determined and $u_{k-1}$ is already known, then the equation that determines $u_k$ is:

$$T = \int_{u_{k-1}}^{u_k} \frac{du}{\dot{u}}.$$

From (19), it admits a closed-form integration for jerk limits, which is similar to the acceleration case in [3]. Then the above equation becomes

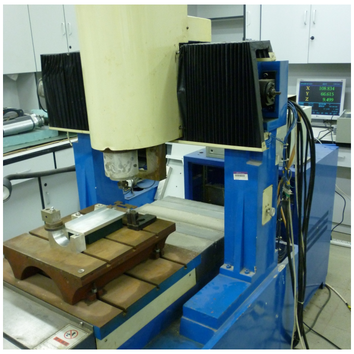
$$T = \frac{(x'\dot{u})'\dot{u}}{J_x}\Big|_{u_{k-1}}^{u_k}.$$

The parameter value $u_k$ is computed by numerical method. Since $u$ is a monotonously increasing function in $t$, this equation has only one real root.

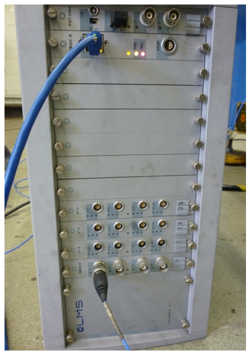
The experiment is contacted on a 3-axis CNC milling machine (see Fig. 10(a)). A fast signal acquisition and analysis system LMS SCADAS3 (see Fig. 10(b)) is used to measure the vibration during the machining.

The test tool path shown in Fig. 12(a) is the curve segment in Fig. 4(a) copied five times and the feedrate will decrease to zero at each connection point. Two experiments are contacted to compare the feedrate planning algorithm in [3] with confined acceleration (abbr. **FP_CA**) and the algorithm **FP_CJ** proposed in this paper. The tangential velocity bound is $V_{max} = 80\ mm/s$ and the acceleration bounds are $A_x = A_y = 800\ mm/s^2$ for both algorithms. The jerk bounds for **FP_CJ** are set to be $J_x = J_y = 10^4\ mm/s^3$. The sampling period $T$ is 1 ms.

Since the CNC machine available to us cannot manufacture metal, wax machining is used in the experiments. The machining results are shown in Fig. 11, where Fig. 11(a) is the

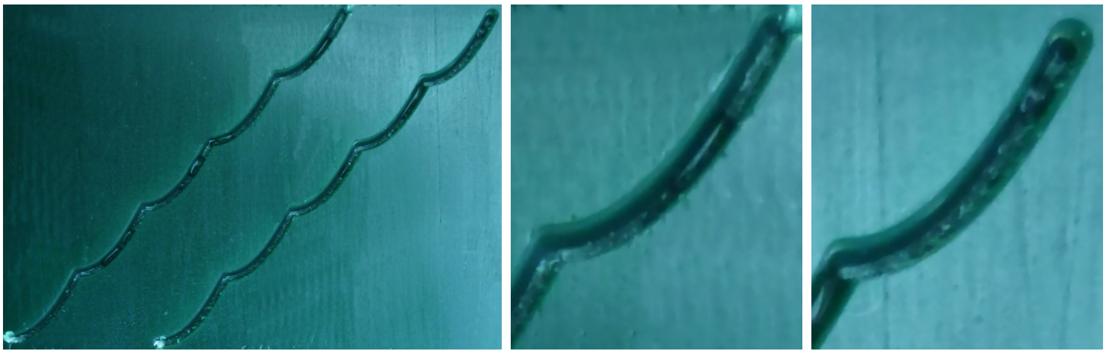


(a) The 3-axis CNC machine



(b) The vibration test equipment

Figure 10: Experiment setup.

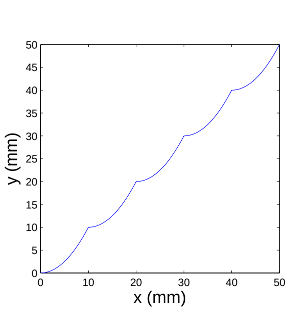


(a) Left: **FP_CA**; Right: **FP_CJ** (b) **FP_CA** (c) **FP_CJ**

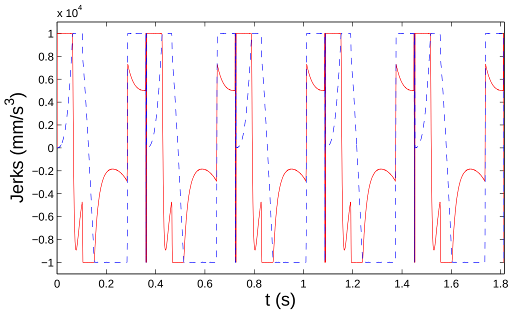
Figure 11: The machining results.

whole machined tool path, Fig. 11(b) and Fig. 11(c) are the last segments of the machined tool pathes for the cases of confined acceleration and confined jerk respectively. Note that the surface of the machined path is covered with small wax particles from the machining, which are not easy to remove. Therefore, we check the machining quality from the edges of the machined path. From Fig. 11(b) and Fig. 11(c), it can be seen that the machining quality with confined jerk is better than that of confined acceleration.

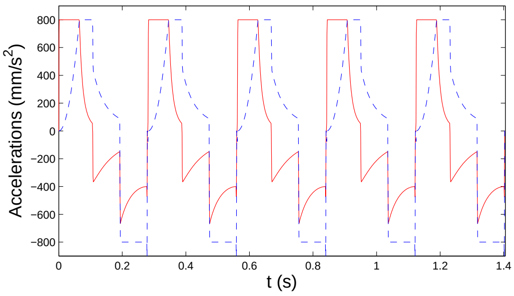
The above comparison is not quantitative. In Fig. 12, more precise comparisons are given. Fig. 12(b)-(f) give the theoretical jerk, acceleration, and feedrate in both cases. From Fig. 12 (e) and (f), the feedrate of the jerk limited trajectory is smoother than that of the acceleration limited trajectory. In Fig. 12 (g) and (h), vibration frequency spectrum diagrams of the two tests are given, where the vertical axis is the vibration intensity whose unit is the gravitational acceleration $g$, and the unit of horizontal axis is Hz. The spectrum diagram gives the distribution of the intensity of the vibration at difference frequencies. For instance, from Fig. 12 (g) and Fig. 12 (h), the strongest vibrations in the cases of **FP_CA** and **FP_CJ** have intensities 8g and 6.5g respectively and occur when the machine tool vibrates at frequency 100Hz. From Fig. 12 (g) and (h), vibrations for algorithm **FP_CA** are significantly stronger than that of **FP_CJ** except at three isolated frequencies: 600HZ,



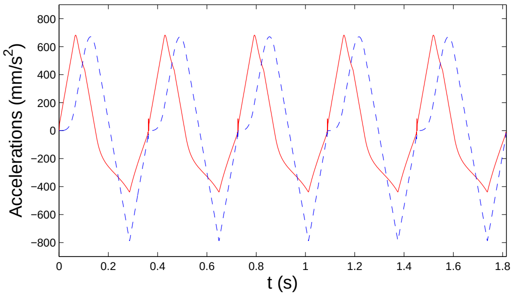

(a) The test tool path



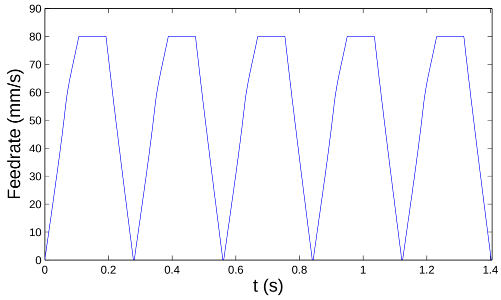

(b) $j_x$ (solid) and $j_y$ (dotted) of **FP_CJ**



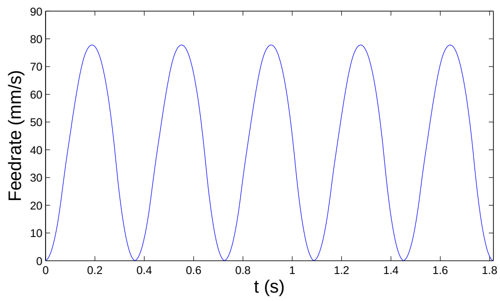

(c) $a_x$ (solid) and $a_y$ (dotted) of **FP_CA**



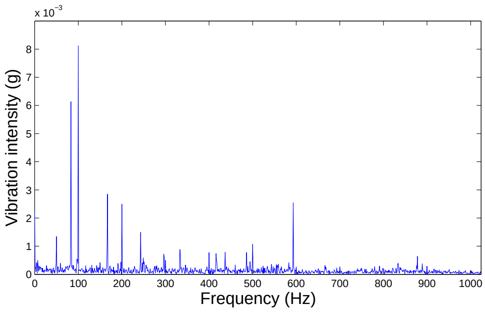

(d) $a_x$ (solid) and $a_y$ (dotted) of **FP_CJ**



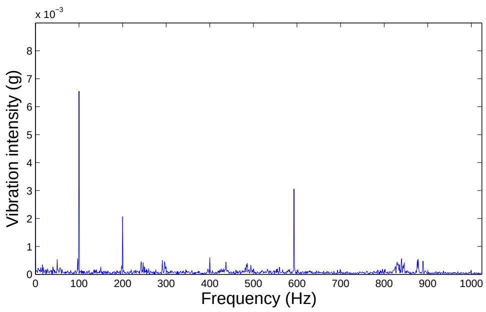

(e) Feedrate of **FP_CA**




(f) Feedrate of **FP_CJ**




(g) Vibration intensity of **FP_CA**




(h) Vibration intensity of **FP_CJ**

Figure 12: Feedrate, acceleration, and jerk of the two algorithms. The horizontal axes for (b)-(f) are time with unit second.

850Hz, and 890Hz, where the vibrations in the two cases are comparable. The machining times of using algorithms **FP_CA** and **FP_CA** are 1.405s and 1.815s respectively. Since vibration of the machining tool is one of the important factors affecting machining quality, it can be concluded that machining quality can be improved significantly by introducing jerk limits with the costs of reducing a reasonable amount of machining time.

# 5 Conclusion

High speed and high quality machining requires feedrate planning algorithms which provide continuous position, velocity, and acceleration profiles. This paper presents an optimal jerk confined feedrate planning algorithm under a "greedy rule", which generates a smooth and analytical feedrate function. Experimental results show that the new algorithm can be used to reduce the machine vibration and improve the machining quality.

It is a significant open problem to show that the algorithm is globally optimal without the "greedy rule". The main difficulty is that, for second-order differential equations, there exist no results similar to the "comparison theorem" for first-order differential equations (p.25, [16]), and as a consequence, it is not possible to prove that the time-optimal velocity function will achieve the maximal possible value at any place as in the confined acceleration case.

**Acknowledgement.** The authors are grateful to Prof. Peiqing Ye from Tsinghua University, who provided the equipments for the experimental results reported in Section 4.2.

# References

[1] Bobrow JE, Dubowsky S, Gibson JS. Time-optimal control of robotic manipulators along specified paths. *Int J Robotics Res* 1985; 4(3):3-17.

[2] Shiller Z, Lu HH. Robust computation of path constrained time optimal motions. *IEEE Inter. Conf. on Robotics and Automation*, Cincinnati, OH, 144-149, 1990.

[3] Timar SD, Farouki RT, Smith TS, Boyadjieff CL. Algorithms for time-optimal control of CNC machines along curved tool paths. *Robotics and Computer-Integrated Manufacturing* 2005; 21(1):37-53.

[4] Timar SD, Farouki RT. Time-optimal traversal of curved paths by Cartesian CNC machines under both constant and speed-dependent axis acceleration bounds. *Robotics and Computer-Integrated Manufacturing* 2007; 23(2):563-579.

[5] Zhang M, Yan W, Yuan CM, Wang D, Gao, XS. Curve fitting and optimal interpolation on CNC machines based on quadratic B-splines. *Science China - Information Sciences* 2011; 54(7):1407-1418.

[6] Yuan CM, Zhang K, Fan W, Gao, XS. Time-optimal interpolation for CNC machining along curved tool paths with confined chord error. *MM Research Preprints* 2011; 30:57-89.

[7] Dong J, Stori JA. A generalized time-optimal bi-directional scan algorithm for constrained feedrate optimization. *ASME Journal of Dynamic Systems, Measurement, and Control* 2006; 128:379-390.

[8] Erkorkmaz K, Altintas Y. High speed CNC system design Part I: jerk limited trajectory generation and quintic spline interpolation. *International Journal of Machine Tools and Manufacture* 2001; 41:1323-1345.

[9] Macfarlane S, Croft EA. Jerk-bounded manipulator trajectory planning: design for real-time applications. *IEEE Transactions on Robotics and Automation* 2003; 19:42-52.

[10] Nam SH, Yang MY. A study on a generalized parametric interpolator with real-time jerklimited acceleration. *Computer-Aided Design* 2004; 36:27-36.

[11] Lin MT, Tsai MS, Yau HT. Development of a dynamics-based NURBS interpolator with real-time look-ahead algorithm. *International Journal of Machine Tools and Manufacture* 2007; 47(15):2246-2262.

[12] Emami MM, Arezoo B. A look-ahead command generator with control over trajectory and chord error for NURBS curve with unknown arc length. *Computer-Aided Design* 2010; 42:625-632.

[13] Lai JY, Lin KY, Tseng SJ, Ueng WD. On the development of a parametric interpolator with confined chord error, feedrate, acceleration and jerk. *Int J Adv Manuf Technol* 2008; 37(1-2):104-121.

[14] Dong J, Ferreiraa PM, Stori JA. Feed-rate optimization with jerk constraints for generating minimum-time trajectories. *International Journal of Machine Tools and Manufacture* 2007; 47:1941-1955.

[15] Zhang LX, Sun RY, Gao XS, Li HB. High speed interpolation for micro-line trajectory and adaptive real-time look-ahead in CNC machining. *Science China - Technological Sciences* 2011; 54(6):1481-1495.

[16] Birkhoff G, Rota GC. *Ordinary differential equations.* John Wiley, New York, 1969.