# Efficient Time Optimal Feedrate Planning under Dynamic Constraints for High-order CNC Servo System[☆]

Jian-Xin Guo[a], Ke Zhang[a,b], Qiang Zhang[c], Xiao-Shan Gao[a,*]

[a] *KLMM, Academy of Mathematic and Systems Science, Chinese Academy of Sciences*
[b] *Department of Mechanical Engineering, University of British Columbia*
[c] *College of Information and Control Engineering, China University of Petroleum (East China)*

**Abstract**

In this paper, the time optimal feedrate planning problem under confined feedrate, axis velocity, axis acceleration, axis jerk, and axis tracking error for a high order CNC servo system is studied. The problem is useful in that the full ability of the CNC machine is used to enhance the machining productivity while keeping the machining precision under a given level. However, the problem is computationally challenging. The main contribution of this paper is to approximate the problem nicely by a finite state convex optimization problem which can be solved efficiently. The method consists of two key ingredients. First, a relationship between the tracking error and the input signal in a high-order CNC servo system is established. As a consequence, the tracking error constraint is reduced to a constraint about kinematic quantities. Second, a novel method is introduced to relax the nonlinear constraints about kinematic quantities to linear ones. Experimental results are used to validate the proposed method.

*Keywords:* Time optimal feedrate planning; tracking error; jerk; high order CNC servo system; convex optimization.

## 1. Introduction

The problem of time-optimal feedrate planning along a given parametric tool path has received a significant amount of attention in the CNC machining literature due to its ability to increase productivity of CNC machining by using the full ability of the machines [2, 3, 8, 9, 10, 17, 20, 21, 23, 24, 25, 26, 27]. The feedrate planning problem is usually formulated as a time-minimum optimal control problem under kinematic constraints such as confined feedrate, axis acceleration, jerk, even jounce, and efficient algorithms have been proposed to solve the problem. The acceleration bounds are introduced to reduce inertia and prevent

[☆]Partially supported by a National Key Basic Research Project of China (2011CB302400), by a grant from NSFC (60821002), and by a UPC grant (120501A)

[*]Corresponding author. Email: xgao@mmrc.iss.ac.cn, http://www.mmrc.iss.ac.cn/~xgao, tel: 86-10-62541834, fax: 86-10-62630706

mechanical shocks. The jerk and jounce bounds are used to generate smooth feedrate profiles aiming at improving machining quality.

Due to various reasons such as the inertia of the CNC axes and inaccurate modeling of the CNC dynamic system, the tracking error is not guarantied to reach the desired level even if the acceleration and jerk are bounded. A common method to reduce tracking errors is to use a closed-loop controller which calculates the difference between the desired signal and the feedback signal in real-time and generates a control signal to minimize the dynamic error. Many algorithms along this line were developed, such as the cross-coupled control strategy [14, 15], the model-referenced adaptive control [5], the predictive control [19], and the learning control [1]. In order to use these closed-loop methods, the users need to access the control system, which demands more from the end-users.

An alternative approach is to combine the "open-loop" feedrate planning with dynamic precision control by adding a tracking error bound as a new constraint in the feedrate planning phase. An advantage of this approach is that accessing the control system is not required and hence is more convenient for the end-users. Dong and Stori [6, 7] considered the dynamic error information in the feedrate planning phase by approximating the tracking error with the linear part of its Taylor expansion. Ernesto and Farouki [9] solved the problem of compensating for inertia and damping of the machine axes by a priori modifying the commanded tool path. In [18, 22], Lin, Tsai, et al used the critical point approaches to generate feedrate with confined contouring errors. In [12], a linear programming method is proposed to solve the feedrate planning problem under confined tracking error for CNC systems based on PD controllers.

In this paper, the time-minimum feedrate planning problem under confined feedrate, axis velocity, axis acceleration, axis jerk, and axis tracking error is studied. To be practical, the dynamic system with PID controllers are considered, where the tracking error satisfies a third order differential equation. The time-minimum feedrate planning problem in this situation is strongly nonlinear, and there exist no efficient algorithms for solving it before. The main contribution of this paper is to reduce the time-minimum feedrate planning problem into a finite state convex optimization problem whose global optimal solution can be computed efficiently. The work consists of two key ingredients. Firstly, a relationship between the tracking error and the input signal in high-order CNC servo systems is established. As a consequence, the tracking error constraint can be reduced to a constraint about a linear combination of kinematic quantities such as accelerations and jerks. Secondly, a novel method is introduced to relax the nonlinear constraints involving the jerk to linear constraints. Experimental results show that the new convex optimization problem gives nice approximation to the original problem and can be solved efficiently.

Comparing to the work [6, 9, 12], high-order CNC servo systems are considered which allows the usage of PID controllers, while the work [6, 9, 12] only considered second order systems for P or PD controllers. Furthermore, the relaxation of the tracking error in this paper is theoretically guaranteed to be valid, while the one given in [6] is an approximation. Also, our approach reduces the optimal feedrate planning problem into a convex programming problem which can be solved efficiently. The method proposed in [24], although more general than ours, is less efficient. Comparing to [18] and [22], our approach generates an

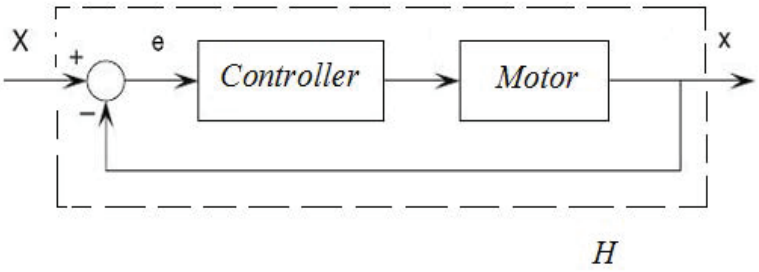


Figure 1: DC servo system

approximate time minimum feedrate, while the feedrate generated with methods in [18] and [22] is not time minimum. Comparing to [23] which initiates the powerful convex optimization approach, we consider a much more general and non-convex problem which is reduced to a convex optimization problem, while the problem considered in [23] is itself convex.

The paper is organized as follows. Section 2 describes the high-order dynamic models and shows how to simplify the tracking error constraint. Section 3 presents an efficient method to solve the time-minimum feedrate planning problem by converting it to a convex optimization problem. In Section 4, experimental results are used to demonstrate the effectiveness of the approach. In Section 5, concluding remarks are given.

## 2. Tracking Error Simplification for High-order CNC Servo System

In this section, we will show that the tracking error constraint can be replaced by a constraint about kinematic quantities such as accelerations and jerks.

### *2.1. Tracking Error of High-order CNC Servo System*

Suppose that the CNC machine is controlled by $M$ axes and the subscript $\tau \in \{1, \ldots, M\}$ will represent these axes. Each axis is powered by a DC motor satisfying an $n$th-order linear system whose transfer function is [11, p. 70]

$$H_\tau(s) = \frac{x_\tau(s)}{X_\tau(s)} = \frac{b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0}{a_n s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}$$

where $X_\tau$ and $x_\tau$ are the commanded axis location and the actual axis location, respectively.

Using the closed form position control shown in Fig. 1, we can calculate the transfer function between the *tracking error* $e_\tau = X_\tau - x_\tau$ and the input signal $X_\tau$:

$$\begin{aligned} G(s) &= \frac{e_\tau(s)}{X_\tau(s)} \\ &= \frac{(a_n s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0) - (b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0)}{a_n s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0} \\ &= \frac{c_l s^l + c_{l-1}s^{l-1} + \cdots + c_1 s + c_0}{a_n s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}, \end{aligned}$$

where $l \leq \max(m, n)$. Using the inverse Laplace transformation to the above equation, if the initial values of $e_\tau(t)$ and its derivatives are zero, then the tracking error $e_\tau(t)$ satisfies the following differential equation:

$$a_n \frac{\mathrm{d}^n e_\tau}{\mathrm{d}t^n} + a_{n-1} \frac{\mathrm{d}^{n-1} e_\tau}{\mathrm{d}t^{n-1}} + \cdots + a_1 \frac{\mathrm{d}e_\tau}{\mathrm{d}t} + a_0 e_\tau = c_l \frac{\mathrm{d}^l X_\tau}{\mathrm{d}t^l} + c_{l-1} \frac{\mathrm{d}^{l-1} X_\tau}{\mathrm{d}t^{l-1}} + \cdots + c_1 \frac{\mathrm{d}X_\tau}{\mathrm{d}t} + c_0 X_\tau \quad (2.1)$$

where $t$ represents time. In order for the above system in $e_\tau$ to be stable, the real parts of the eigenvalues of the linear system in the left hand side of (2.1) are assumed to be negative [11, p. 76]. More precisely, the roots of the following the character equation of system (2.1)

$$a_n \lambda^n + a_{n-1} \lambda^{n-1} + \cdots + a_1 \lambda + a_0 = 0 \quad (2.2)$$

are assumed to have negative real parts.

Now, the problem of optimal trajectory planning along a given tool-path can be formulated as the following time minimum control problem under kinematic constraints and tracking error constraints:

$$\min T \quad s.t. \left| \frac{\mathrm{d}X_\tau}{\mathrm{d}t} \right| \leq X^1_{\max}, \left| \frac{\mathrm{d}^2 X_\tau}{\mathrm{d}t^2} \right| \leq X^2_{\max}, \cdots, \left| \frac{\mathrm{d}^m X_\tau}{\mathrm{d}t^m} \right| \leq X^m_{\max}, |e_\tau| \leq E_{\max} \quad (2.3)$$

where $\tau \in \{1, \ldots, M\}$ represents the axes, $m$ is a positive integer to be given by the user, $X^j_{\max}$ are postive real numbers, $e_\tau(t)$ satisfies equation (2.1).

Note that the number $m$ determines the smoothness of the velocity functions to be obtained. For instance, if $m = 2$, then the acceleration is bounded and the velocity function is continuous; if $m = 3$, then the acceleration and the jerk are bounded and the velocity function is differentiable; and if $m = 4$, then the acceleration, jerk, and jounce are bounded and the acceleration function is differentiable. Generally speaking, the more smooth the velocity function is, the less vibration occurs, as shown in [10].

*2.2. Simplification of the Tracking Error Constraint*

The last constraint of (2.3) is difficult to deal with since it will lead to complicated equations when solving the optimization problem via discrete methods. In this section, we will show that the tracking error constraint can be replaced with a constraint about kinematic quantities. More precisely, we have

**Theorem 2.1** *Let $e_\tau$ be the tracking error for the $\tau$ axis, which satisfies* (2.1) *and has the initial values $e_\tau(0) = \frac{\mathrm{d}e_\tau}{\mathrm{d}t}(0) = \cdots = \frac{\mathrm{d}^{n-1}e_\tau}{\mathrm{d}t^{n-1}}(0) = 0$. Then*

$$\left| c_l \frac{\mathrm{d}^l X_\tau}{\mathrm{d}t^l} + c_{l-1} \frac{\mathrm{d}^{l-1} X_\tau}{\mathrm{d}t^{l-1}} + \cdots + c_1 \frac{\mathrm{d}X_\tau}{\mathrm{d}t} + c_0 X_\tau \right| \leq \delta E_{\max} \quad (2.4)$$

*implies $|e_\tau| \leq E_{\max}$. The constant $\delta$ in* (2.4) *is $a_0$ in* (2.1) *if the roots of equation* (2.2) *are real numbers and is computed with* (2.7) *otherwise.*

The above theorem will be proved in the rest of this section. Two cases will be considered according to whether the roots of equation (2.2) are real numbers. Firstly, we deal with the easier case given in the following theorem.

**Theorem 2.2** *Let $y(t)$ be a function in time $t$, which satisfies the following $n$th-order ODE*

$$a_n \frac{\mathrm{d}^n y}{\mathrm{d}t^n} + a_{n-1} \frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n-1}} + \cdots + a_1 \frac{\mathrm{d}y}{\mathrm{d}t} + a_0 y = g(t), \tag{2.5}$$

*where $a_i$ are constants and $g(t)$ is a nonlinear function in $t$. Furthermore, assume $y(0) = \cdots = \frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n-1}}(0) = 0$ and the roots of equation (2.2) are negative real numbers. Then for a constant $M$, $|g(t)| \le |a_0| M$ implies $|y(t)| \le M$.*

In order to prove the above theorem, the following lemma is needed, which is from [12].

**Lemma 2.3** *Let $y(t)$ be a differentiable function in the time domain $t \in [0, \infty)$ such that $y(0) = 0$, and $M$ and $r$ positive real numbers. If $y$ satisfies $\left| r \frac{\mathrm{d}y}{\mathrm{d}t} + y \right| \le M$, then $|y| \le M$.*

**Proof of Theorem 2.2:** By the assumption of the theorem, the roots of the character equation (2.2) are negative real numbers, which are denoted by $-\frac{1}{d_i}, i = 1, \ldots, n$, $d_i > 0$. Then, the character equation (2.2) of the left hand side of the system (2.5) can be expressed as:

$$f(\lambda)/a_0 = (d_1\lambda + 1)(d_2\lambda + 1) \cdots (d_n\lambda + 1) = 0$$

Define

$$\begin{cases} z_1 = d_1 \frac{\mathrm{d}y}{\mathrm{d}t} + y, \\ z_2 = d_2 \frac{\mathrm{d}z_1}{\mathrm{d}t} + z_1, \\ \cdots \\ z_n = d_n \frac{\mathrm{d}z_{n-1}}{\mathrm{d}t} + z_{n-1}. \end{cases}$$

It is easy to verify that the left hand side of (2.5) becomes:

$$(a_n \frac{\mathrm{d}^n y}{\mathrm{d}t^n} + a_{n-1} \frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n-1}} + \cdots + a_1 \frac{\mathrm{d}y}{\mathrm{d}t} + a_0 y)/a_0 = z_n = d_n \dot{z}_{n-1} + z_{n-1} = g(t)/a_0.$$

Thus by Lemma 2.3, from $|g(t)/a_0| \le M$, we have $|z_{n-1}| \le M$. Since $z_{n-1} = d_{n-1}\dot{z}_{n-2} + z_{n-2}$ and $d_{n-1} > 0$, using Lemma 2.3 again, we have $|z_{n-2}| \le M$. Repeatedly, we have $|y| \le M$, which proves the theorem. ∎

The second case, where equation (2.2) has complex roots, is more involved. Before giving the result, some notations and lemmas are needed.

Let $\mathbb{R}$ be the set of real numbers, $x \in \mathbb{R}^n$, and $A \in \mathbb{R}^{n \times n}$. Then it is well known that the following *vector 1-norm* and *matrix 1-norm*

$$\|x\|_1 = \sum_{j=1}^{n} |x_j|, \quad \|A\|_1 = \max_{1 \le j \le n} \sum_{i=1}^{n} |a_{ij}|$$

satisfy the following compatible condition

$$\|Ax\|_1 \leq \|A\|_1 \|x\|_1 .$$

Besides, the following lemma is needed.

**Lemma 2.4** *Let $x = (x_1(t), \ldots, x_n(t))$ and $x_i(t)$ integrable functions from $\mathbb{R}$ to $\mathbb{R}$. Then*

$$\left\| \int_a^b x(\tau) d\tau \right\|_1 \leq \int_a^b \|x(\tau)\|_1 \, d\tau.$$

*Proof.* $\left\| \int_a^b x(\tau) d\tau \right\|_1 = \sum_{j=1}^{n} \left| \int_a^b x_j(\tau) d\tau \right| \leq \sum_{j=1}^{n} \int_a^b |x_j(\tau)| \, d\tau = \int_a^b \{ \sum_{j=1}^{n} |x_j(\tau)| \} d\tau = \int_a^b \|x(\tau)\|_1 \, d\tau.$ ∎

The following theorem simplifies the tracking error in the second case.

**Theorem 2.5** *Let $y(t)$ be a function in time $t$, which satisfies the following $n$th-order ODE*

$$a_n \frac{\mathrm{d}^n y}{\mathrm{d}t^n} + a_{n-1} \frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n-1}} + \cdots + a_1 \frac{\mathrm{d}y}{\mathrm{d}t} + a_0 y = g(t), \tag{2.6}$$

*where $a_i$ are constants and $g(t)$ is a nonlinear function in $t$. Furthermore, assume $y(0) = \cdots = \frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n-1}}(0) = 0$. Then for a constant $M$, $|g(t)| \leq \delta M$ implies $|y| \leq M$, where the constant $\delta$ is calculated using:*

$$\delta = \frac{1}{\int_0^\infty \|e^{\tau A}\|_1 d\tau} \tag{2.7}$$

*for* $A = \begin{pmatrix} 0 & 1 & 0 & \cdots \\ \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & 0 & 1 \\ -\frac{a_0}{a_n} & -\frac{a_1}{a_n} & \cdots & -\frac{a_{n-1}}{a_n} \end{pmatrix}_{n \times n}$.

*Proof.* We rewrite the ODE (2.6) in matrix form:

$$\frac{\mathrm{d}\mathbf{y}}{\mathrm{d}t} = A\mathbf{y} + \mathbf{g}(t), \tag{2.8}$$

where $\mathbf{y} = (y, y_1, \ldots y_{n-1})^T$, $\mathbf{g}(t) = (0, 0, \ldots, 0, g(t))^T$, and matrix $A$ is given in the theorem to be proved. Obviously,

$$|\lambda I - A| = \det \begin{pmatrix} \lambda & -1 & 0 & \cdots \\ \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \lambda & -1 \\ \frac{a_0}{a_n} & \frac{a_1}{a_n} & \cdots & \lambda + \frac{a_{n-1}}{a_n} \end{pmatrix}_{n \times n}.$$

Expand this determinant along the last column:

$$|\lambda I - A| = (\lambda + \frac{a_{n-1}}{a_n})\lambda^{n-1} + \frac{a_{n-2}}{a_n}\lambda^{n-2} + \cdots + \frac{a_1}{a_n}\lambda + \frac{a_0}{a_n},$$

which is the character equation of (2.6). Hence the eigenvalues of $A$ are the same as the roots of equation (2.2).

According to the theory of differential equations, the solution of (2.8) can be calculated as:

$$\mathbf{y}(t) = e^{(t)A}\mathbf{y}(0) + \int_0^t e^{(t-\tau)A}\mathbf{g}(\tau)\mathrm{d}\tau,$$

where $\mathbf{y}(0)$ is the initial value. Since $\mathbf{y}(0) = \mathbf{0}$, according to the definition of vector norm and Lemma 2.4, we have

$$\|\mathbf{y}(t)\|_1 = \left\| \int_0^t e^{(t-\tau)A}\mathbf{g}(\tau)d\tau \right\|_1 = \left\| \int_0^t e^{\tau A}\mathbf{g}(t-\tau)d\tau \right\|_1 \leq \int_0^t \left\|e^{\tau A}\right\|_1 \|\mathbf{g}(t-\tau)\|_1 \mathrm{d}\tau.$$

Thus

$$|y| \leq \|\mathbf{y}(t)\|_1 \leq \delta M \int_0^\infty \left\|e^{\tau A}\right\|_1 \mathrm{d}\tau \leq M.$$

Since the roots of (2.2) have negative real parts, $\int_0^\infty \left\|e^{\tau A}\right\|_1 \mathrm{d}\tau$ exists and the theorem is proved. ∎

*2.3. PID Controller and Third Order System*

In most cases, error-based feedback controllers are based on PID controllers whose control parameters are the proportional, integral, derivative gains $k_P$, $k_I$, and $k_D$, respectively. An illustration is given in Fig. 2, where $e_\tau = X_\tau - x_\tau$ is the tracking error. The other parameters are explained below. The current amplifier $k_a$ converts the actuating signal $u$ into the current $i$ to control the motor, which produces a torque $T$ through the motor torque gain $k_t$. The torque $T$ determines the angular speed through the system inertia $J$ and damping $B$. The motor shaft angle $\theta$, obtained by integration of the motor shaft angle speed $w$, determines the axis linear position $x$ through the transmission ratio $g$. Besides, set $K = k_a k_t g$, since these three parameters often occur in this product form. Then, the tracking error and the input signal satisfy the following dynamic equation:

$$J\frac{\mathrm{d}^3 e_\tau}{\mathrm{d}^3 t} + (B + Kk_D)\frac{\mathrm{d}^2 e_\tau}{\mathrm{d}^2 t} + Kk_P\frac{\mathrm{d}e_\tau}{\mathrm{d}t} + Kk_I e_\tau = J\frac{\mathrm{d}^3 X_\tau}{\mathrm{d}^3 t} + B\frac{\mathrm{d}^2 X_\tau}{\mathrm{d}^2 t} = Jj_\tau + Ba_\tau, \quad (2.9)$$

where $a_\tau$ and $j_\tau$ are the acceleration and the jerk of the $\tau$-axis.

Notice that a third-order system with real coefficients either has one real eigenvalue or three real eigenvalues, and explicit criteria can be given for these cases. Thus, we have the following simplification criterion for system (2.9).

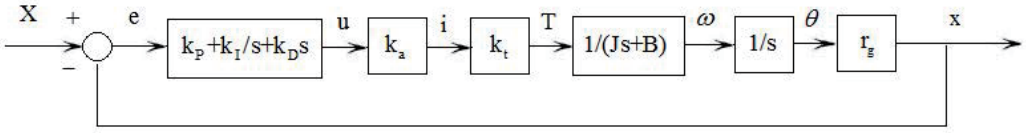


Figure 2: The PID Controller

**Theorem 2.6** *Let the tracking error $e_\tau(t)$ satisfy the initial values $e_\tau(0) = \frac{\mathrm{d}e_\tau}{\mathrm{d}t}(0) = \frac{\mathrm{d}^2e_\tau}{\mathrm{d}^2t}(0) = 0$. Then the tracking error constraint $|e_\tau| \le E_{\max}$ can be replaced by*

$$\left|\frac{J}{Kk_I}j_\tau + \frac{B}{Kk_I}a_\tau\right| \le \delta E_{\max} \tag{2.10}$$

*where $\delta$ is determined as follows. Let*

$$d = 4((B+Kk_D)^2 - 3JKk_P)^3 - (2(B+Kk_D)^3 - 9J(B+kk_D)Kk_P^2 + 27Kk_IJ^2).$$

*Then if $d > 0$, we can set $\delta = 1$. If $d < 0$, $\delta$ is calculated with* (2.7).

*Proof.* We need to know when the following cubic equation

$$J\lambda^3 + (B+Kk_D)\lambda^2 + Kk_P\lambda + Kk_I = 0 \tag{2.11}$$

has real or complex roots. A cubic equation has either one or three real roots. Following the criterion given on [13, p. 299], if $d > 0$, then equation (2.11) has three distinct real roots and $\delta = 1$ by Theorem 2.2; if $d < 0$, then equation (2.11) has one real root and two distinct complex roots and the theorem follows from Theorem 2.5. ∎

## 3. Time Minimum Feedrate Planning for CNC System with PID Controller

In this section, we will show how to convert the time optimal feedrate planning problem for CNC systems with PID controllers into a finite state convex optimization problem.

### *3.1. The Problem*

We consider a CNC system with three translational axes $x, y, z$ and assume that these axes have the same dynamic parameters. Also assume that the PID controller is used for each axis. Thus the tracking error satisfies equation (2.9). The tool-path of the three-axis CNC machine is given by a set of parametric functions with at least $C^3$ continuity:

$$r(u) = (x(u), y(u), z(u)), u \in [0,1].$$

The tool-path $r(u)$ could be NURBS, B-splines, or other parametric equations. We will see in Section 3.3 that only the evaluations of $r(u)$, $\frac{\mathrm{d}r(u)}{\mathrm{d}u}$, $\frac{\mathrm{d}^2r(u)}{\mathrm{d}^2u}$ at certain given parametric values are needed in the feedrate planning.

In order to obtain smooth velocity functions, bounds of axis velocity, acceleration, and jerk are introduced. Furthermore, high precision machining can be achieved by bounding the

axis tracking error. Therefore, the problem is to design a parametric velocity $\dot{u} = \frac{\mathrm{d}u}{\mathrm{d}t}$ such that the tool-path can be traversed with minimal time under the kinematic and precision constraints mentioned above:

$$\begin{aligned} &\min_{\dot{u}} \int_0^1 \frac{1}{\dot{u}}\,\mathrm{d}u \\ &s.t.\ \ |v_f| \le V_{f\max}, |v_\tau| \le V_{\max}, |a_\tau| \le A_{\max}, |j_\tau| \le J_{\max}, |e_\tau| \le E_{\max} \end{aligned} \tag{3.1}$$

where $\tau \in \{x, y, z\}$ is the axis, $v_f$ is the tangential feedrate, $v_\tau$ is the axis velocity, $a_\tau$ is the axis acceleration, $j_\tau$ is the axis jerk, $e_\tau$ is the axis dynamic error satisfying (2.9), and $V_{f\max}, V_{\max}, A_{\max}, J_{\max}, E_{\max}$ are the corresponding bounds, respectively. The following initial values are assumed $\dot{u}(0) = \dot{u}(1) = \ddot{u}(0) = \ddot{u}(1) = 0$.

Note that instead of the contour error, the tracking error is considered here for two reasons: the model based on tracking error is simpler and can be solved efficiently as shown by this paper and limited tracking errors lead to limited contour errors.

By Theorem 2.6, the tracking error constraint can be simplified and the optimal control problem (3.1) can be relaxed into the following problem whose constraints involves kinematic quantities only.

$$\min_{\dot{u}} \int_0^1 \frac{1}{\dot{u}}\,\mathrm{d}u \tag{3.2}$$

$$s.t. \begin{cases} |v_f| \le V_{f\max} \\ |v_\tau| \le V_{\max} \\ |a_\tau| \le A_{\max} \\ |j_\tau| \le J_{\max} \\ \left|\frac{J}{Kk_I} j_\tau + \frac{B}{Kk_I} a_\tau\right| \le \delta E_{\max} \end{cases} \tag{3.3}$$

where $\delta$ is given in Theorem 2.6 and $\dot{u}(0) = \dot{u}(1) = \ddot{u}(0) = \ddot{u}(1) = 0$.

In the rest of this section, we will rewrite problem (3.2) into a form which is more convenient for numerical solution. Denote "$'$" to be "$\frac{\mathrm{d}}{\mathrm{d}u}$" and "$\dot{}$" to be "$\frac{\mathrm{d}}{\mathrm{d}t}$". Introduce three new functions $a(u), b(u)$, and $c(u)$ in $u$:

$$\begin{aligned} a(u) &= \dot{u}(u)^2, \\ b(u) &= \ddot{u} = \frac{\mathrm{d}\dot{u}}{\mathrm{d}t} = \frac{1}{2}\left(\dot{u}^2\right)' = \frac{1}{2}a'(u), \\ c(u) &= b'(u). \end{aligned} \tag{3.4}$$

Then the kinematic quantities can be written as functions in $a, b$, and $c$:

$$\begin{aligned} v_\tau &= \tau'\dot{u} = \tau'\sqrt{a(u)}, \\ a_\tau &= \tau''\dot{u}^2 + \dot{\tau}\ddot{u} = \tau''a(u) + \tau' b(u), \\ j_\tau &= \tau'''\dot{u}^3 + 2\tau''\dot{u}\ddot{u} + \tau' = \sqrt{a(u)}(\tau'''a(u) + \tau''b(u) + \tau' c(u)), \\ v_f &= \sigma\sqrt{a(u)}, \end{aligned} \tag{3.5}$$

where $\sigma = \sqrt{x'^2 + y'^2 + z'^2}$ and $\tau \in \{x, y, z\}$.

Treat $c(u)$ as the control variable and $b, c$ as the state variables. Then problem (3.2) becomes a classic optimal control problem:

$$\min_{c(u)} \int_0^1 \frac{1}{\sqrt{a(u)}}\,\mathrm{d}u \tag{3.6}$$

$$s.t. \begin{cases} \sigma^2 a(u) \le V_{f\max}^2 \\ (\tau^{'})^2 a(u) \le V_{\max}^2 \\ \left|\tau^{''} a(u) + \tau^{'} b(u)\right| \le A_{\max} \\ \left|\sqrt{a(u)}(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u))\right| \le J_{\max} \\ \left|\frac{J\sqrt{a(u)}(\tau^{'''} a(u)+\tau^{''} b(u)+\tau^{'} c(u))+B(\tau^{''} a(u)+\tau^{'} b(u))}{Kk_I}\right| \le \delta E_{\max} \end{cases} \tag{3.7}$$

where $\tau \in \{x, y, z\}$ represents the axis. From (3.4), the initial values are $a(0) = a(1) = b(0) = b(1) = 0$.

*3.2. Simplification of the Optimal Control Problem*

Numerical solution to problem (3.6) is challenging due to the strong nonlinear part in the jerk and tracking error constraints, as shown in Section 4.2. In this section, we will show that problem (3.6) can be relaxed to a convex optimal control problem whose constraints are linear in the control and state variables.

Two ideas are used in the relaxation. Firstly, the unique optimal solution of the following time-minimum feedrate planning problem under confined acceleration

$$\min_{\dot{u}} \int_0^1 \frac{1}{\sqrt{\dot{u}}}\,\mathrm{d}u \ s.t.\ |v_f| \le V_{f\max}, |v_\tau| \le V_{\max}, |a_\tau| \le A_{\max} \tag{3.8}$$

is generated and used to simplify the last two constraints in (3.7). Secondly, new conditions about the dynamic parameters will be introduced to further simplify the last constraint in (3.7).

The following important property for problem (3.8) is proved in [3, 6]:

**Lemma 3.1** *The optimal solution of problem* (3.8) *achieves the maximum values among all feasible solutions at any parameter value.*

We now reduce problem (3.6) to a new form, where the constraints are linear in the control and state variables $a, b, c$.

**Theorem 3.2** *Let* $a^*(u) = \dot{u}(u)^2$ *be the optimal solution of problem* (3.8) *and assume that the parameters satisfy the following condition*

$$E_{max} K k_I - B A_{max} > 0. \tag{3.9}$$

*Then problem* (3.6) *can be relaxed to the following optimal control problem*

$$\min_{c(u)} \int_0^1 \frac{1}{\sqrt{a(u)}}\,\mathrm{d}u \tag{3.10}$$

$$s.t. \begin{cases} \sigma^2 a(u) \le V_{f\max}^2 \\ (\tau^{'})^2 a(u) \le V_{\max}^2 \\ \left|\tau^{''} a(u) + \tau^{'} b(u)\right| \le A_{\max} \\ \left|\sqrt{a^*(u)}(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u))\right| \le J_{\max} \\ \left|\frac{J\sqrt{a^*(u)}(\tau^{'''} a(u)+\tau^{''} b(u)+\tau^{'} c(u))+B(\tau^{''} a(u)+\tau^{'} b(u))}{Kk_I}\right| \le E_{\max} \end{cases} \tag{3.11}$$

*with initial values* $a(0) = a(1) = b(0) = b(1) = 0$. *More precisely, if* $a, b, c$ *satisfy constraints* (3.11)*, then they must satisfy constraints* (3.7).

*Proof.* Since the first three constrains in (3.7) and (3.11) are the same, we just need to show that the fourth and fifth constraints of (3.11) imply the fourth and fifth constraints of (3.7).

Let $a(u), b(u), c(u)$ be feasible solutions to problem (3.10). Since the first three constraints of (3.11) are the same as that of (3.8), $a(u)$ is also a feasible solution to problem (3.8). Then by Lemma 3.1, $a(u) \le a^*(u)$ for $u \in [0, 1]$. We thus have

$$\begin{aligned} \left|\sqrt{a(u)}(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u))\right| &= \left|\sqrt{a(u)}\right| \left|(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u))\right| \\ &\le \left|\sqrt{a^*(u)}\right| \left|(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u))\right|. \end{aligned}$$

and hence, the fourth constraint in (3.11) implies the fourth constraint in (3.7).

For the fifth constraint, two cases are considered. In the first case, the fifth constraint in (3.11) is assumed to be bang-bang, that is

$$\left|\frac{J\sqrt{a^*(u)}(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u)) + B(\tau^{''} a(u) + \tau^{'} b(u))}{Kk_I}\right| = E_{\max},$$

which means either

$$J\sqrt{a^*(u)}(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u)) + B(\tau^{''} a(u) + \tau^{'} b(u)) = E_{\max} K k_I, \tag{3.12}$$

or

$$J\sqrt{a^*(u)}(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u)) + B(\tau^{''} a(u) + \tau^{'} b(u)) = -E_{\max} K k_I. \tag{3.13}$$

Suppose (3.12) is valid. Due to condition (3.9),

$$\begin{aligned} J\sqrt{a^*(u)}(\tau^{'''} a(u) + \tau^{''} b(u) + \tau^{'} c(u)) &= E_{\max} K k_I - B(\tau^{''} a(u) + \tau^{'} b(u)) \\ &= E_{\max} K k_I - B a_\tau \ge E_{\max} K k_I - B\left|a_\tau\right| \ge E_{\max} K k_I - B A_{max} \ge 0. \end{aligned}$$

Hence, $(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) \geq 0$ and

$$J\sqrt{a^*(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) \geq J\sqrt{a(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)).$$

Therefore, the fifth constraint in (3.11) implies the fifth constraint in (3.7). Likewise, if (3.13) is valid,

$$\begin{aligned} & J\sqrt{a^*(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) = -E_{\max}Kk_I - B(\tau^{''}a(u) + \tau^{'}b(u)) \\ = \; & -E_{\max}Kk_I - Ba_\tau \leq -E_{\max}Kk_I + BA_{max} \leq -(E_{\max}Kk_I - BA_{max}) \leq 0. \end{aligned}$$

Then, in this case,

$$J\sqrt{a^*(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) \leq J\sqrt{a(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)).$$

Thus, in the first case, a solution of (3.10) is also a feasible solution of (3.6).

Now consider the second case, that is, the fifth constraint in (3.11) is not bang-bang:

$$\left| \frac{J\sqrt{a^*(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) + B(\tau^{''}a(u) + \tau^{'}b(u))}{Kk_I} \right| < E_{\max}. \tag{3.14}$$

We will show that the fifth constraint in (3.7) is also valid. Suppose the contrary, that is,

$$\left| \frac{J\sqrt{a(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) + B(\tau^{''}a(u) + \tau^{'}b(u))}{Kk_I} \right| > E_{\max}.$$

Consider one case as an example:

$$\frac{J\sqrt{a(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) + B(\tau^{''}a(u) + \tau^{'}b(u))}{Kk_I} > E_{\max}. \tag{3.15}$$

From (3.9) and (3.15), we have

$$J\sqrt{a(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) > Kk_IE_{\max} - B(\tau^{''}a(u) + \tau^{'}b(u)) > 0.$$

Hence, $(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) > 0$ and

$$\begin{aligned} J\sqrt{a^*(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) \; & \geq \; J\sqrt{a(u)}(\tau^{'''}a(u) + \tau^{''}b(u) + \tau^{'}c(u)) \\ & \geq \; Kk_IE_{\max} - B(\tau^{''}a(u) + \tau^{'}b(u)) \end{aligned}$$

which contradicts (3.14). The theorem is proved. ∎

Note that the constraints in (3.11) are linear in the control and state variables and the objective function is a convex function [23]. Then, problem (3.10) is a convex optimal control problem.

*3.3. Convert the Problem to a Convex Optimization Problem*

In this section, the direct transcription method is used to solve problem (3.10). It is shown that problem (3.10) can be reduced to a finite state convex optimization problem which admits efficient numerical solution.

To convert the infinite state optimization problem into a finite state optimization problem, the parametric interval $[0,1]$ is divided into $N$ equal parts with knots $u_i = \frac{i}{N}$, $i = 0,1,\ldots,N$. The length of each sub-interval is $\Delta = \frac{1}{N}$. Since $\Delta$ is very small, the constraints can be approximately transformed into discrete inequalities at points $u_i$.

Problem (3.10) is solved in two steps.

Firstly, efficient numerical methods from [27] or [23] are used to compute a discrete optimal solution $a_i^* = a(u_i), i = 0,\ldots,N$ for problem (3.8).

Secondly, to solve problem (3.10), we use $c_i = c(u_i), i = 0,\ldots,N$ as the control variables and the state variables $a_i = a(u_i), i = 0,\ldots,N$ and $b_i = b(u_i), i = 0,\ldots,N$ can be calculated with finite differences due to (3.4):

$$b_i \approx \frac{a_{i+1} - a_i}{2\Delta}, c_i \approx \frac{b_{i+1} - b_i}{\Delta}, i = 0,1,\ldots,N-1.$$

Since $a(0) = b(0) = a(1) = b(1) = 0$, we have $a_0 = a_1 = b_0 = a_N = b_N = 0$ and

$$b_{i+1} = \Delta \sum_{k=0}^{i} c_k + b_0 = \Delta \sum_{k=0}^{i} c_k, i = 0,\ldots,N-1. \tag{3.16}$$

$$a_{i+1} = 2\Delta \sum_{k=0}^{i} b_k = 2\Delta^2 \sum_{l=0}^{i-1} (i-l)c_l, i = 1,\ldots,N-1. \tag{3.17}$$

Following the two steps described above, problem (3.10) can be approximated by the following finite state nonlinear programming problem:

$$\min_{c_i} \sum_{j=2}^{N-1} \frac{1}{\sqrt{a_j}} \; s.t. \begin{cases} \sigma_i^2 a_i \le V_{f\max}^2 \\ (\tau_i^{'})^2 a_i \le V_{\max}^2 \\ \left|\tau_i^{''} a_i + \tau_i^{'} b_i\right| \le A_{\max} \\ \left|\sqrt{a_i^*}(\tau_i^{'''} a_i + \tau_i^{''} b_i + \tau_i^{'} c_i)\right| \le J_{\max} \\ \left|\frac{J(\sqrt{a_i^*}(\tau_i^{'''} a_i + \tau_i^{''} b_i + \tau_i^{'} c_i)) + B(\tau_i^{''} a_i + \tau_i^{'} b_i)}{K k_I}\right| \le E_{\max} \\ \Xi \in \Omega, \end{cases} \tag{3.18}$$

where $\tau \in \{x,y,z\}$, $i = 0,\ldots,N$, $\tau_i^{'} = \tau^{'}(u_i)$, $\tau_i^{''} = \tau^{''}(u_i)$, $a_i^*$ are the solution of (3.8) obtained in the first step, $a_i$ and $b_i$ can be derived from (3.17) and (3.16), $\Xi$ is the set of mechanism and bound parameters, $\Omega$ is the parameter domain determined by (3.9). Also, the initial values $a_N = b_N = 0$ lead to the following constraints on $c_i$

$$\sum_{i=0}^{N-1} c_i = 0 \text{ and } \sum_{i=0}^{N-2} (N-1-i)c_i = 0.$$

The constraints of problem (3.18) are linear in variables $c_i$ and the objective function is a convex function in $c_i$ [23]. As a consequence, problem (3.18) is a convex optimization problem which has the nice property that a local optimal solution of the problem is its unique global optimal solution. This allows us to use any gradient based method to solve problem (3.18).

In this paper, the Sequential Quadratic Programming (SQP) method [4] is used to solve problem (3.18). The SQP method reduces the problem to a sequence of optimization subproblems, each of which optimizes a quadratic model of the objective function to a linearization of the constraints. Since the constraints of problem (3.18) are already linear, the SQP method particularly suits for it.

In order to use the SQP method to solve problem (3.18), the number $N$ of discretization need to be given. The selection of $N$ is closely related with the lengths $\Delta s_i = ||r(u_i) - r(u_{i-1})||$ of the line segments $r(u_i)r(u_{i-1})$. According to the theory of finite element method [20], these lengths need to satisfy $\Delta s_i \leq 0.1$mm in order to achieve high accurate computation for most CNC machining. So, a lower bound $N_l$ for $N$ is the least integer such that $\Delta s_i \leq 0.1$mm for $i = 1, \ldots, N$. Furthermore, $N$ should be large enough for the tool-path to be sufficiently subdivided. Combine these two factors, an experimental lower bound for $N$ is $\max\{100, N_l\}$.

## 4. Experimental Result

In this section, the proposed method is tested in a CNC tool to validate its effectiveness.

### *4.1. Experiment Setup*

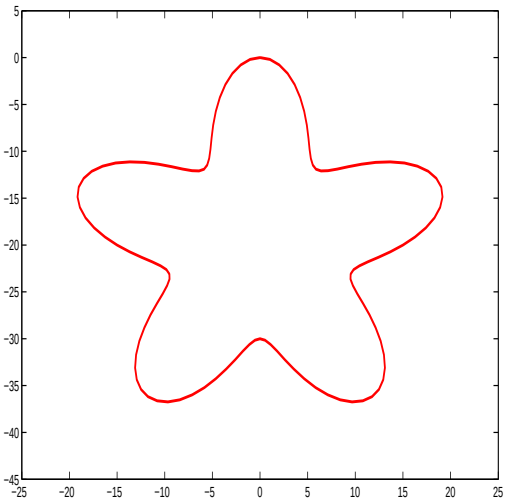


Figure 3: Star Curve

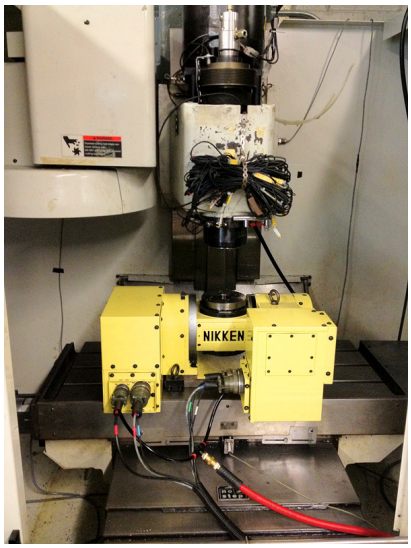


Figure 4: Fadal machine

| Parameters | X-axis | Y-axis |
|---|---|---|
| $k_a$ [A/V] | 6.5723 | 6.2274 |
| $k_t$ [Nm/A] | 0.4769 | 0.4769 |
| $r_g$ [mm/rad] | 1.5915 | 1.5915 |
| $J$ [kgm$^2$] | 0.0070028 | 0.0081904 |
| $B$ [kgm$^2$/s] | 0.023569 | 0.043009 |

Table 1: Parameters used in the experiment

The plane curve "star" shown in Fig. 3 is used as the experimental tool-path whose parametric equation is

$$\begin{cases} x = (15 + 5 \cdot \cos(10\pi u)) \cdot \cos(2\pi u + 0.5\pi) \\ y = (15 + 5 \cdot \cos(10\pi u)) \cdot \sin(2\pi u + 0.5\pi), u \in [0, 1]. \end{cases}$$

Moreover, the curve will be initialized from the original point $(0, 0)$ in order to satisfy the inverse Laplace transformation.

A Fadal CNC machine tool shown Fig. 4 is used for the experiment. The kinematic limits are set to be $V_{f\max} = 150$ mm/s, $\boldsymbol{V}_{\max} = (250, 250)$ mm/s, $\boldsymbol{A}_{\max} = (1500, 1500)$ mm/s$^2$, $\boldsymbol{J}_{\max} = (18000, 18000)$ mm/s$^3$, which are for the tangent velocity, axis velocity, axis acceleration, and axis jerk, respectively. The initial and terminal axis velocities and axis accelerations are all zero. The number $N$ of discretization is set to be 120. The sampling period is 1 ms. The drive parameters for $x$ and $y$ axes are listed in Table 1. The actual positions on $x$ axes and $y$ axes are obtained from their encoders mounted on the CNC machine tool.

*4.2. Experimental Result*

Two tests are conducted to illustrate the real and complex eigenvalue cases.

**Test 1: Real eigenvalue case.**

The proportional, integral, derivative gains of the PID controller are chosen as $k_P = 30$ V/mm, $k_I = 650$ V/(mm·s), and $k_D = 0.4$ V/(mm/s) for both axes. It is easy to show that, the eigenvalues of equation (2.11) are negative real numbers.

We first run the test without considering the tracking error constraints, that is, problem (3.18) is solved without the last constraint. Fig. 5-Fig. 8 illustrate the feedrate, acceleration, pseudo jerk, and the real-time tracking error obtained from the encoder mounted on the CNC machine tool. By the *pseudo-jerk*, we mean $j^* = \sqrt{a_i^*}(\tau_i^{'''} a_i + \tau_i^{''} b_i + \tau_i^{'} c_i)$ which is an approximation of the real jerk (See (3.18)). Obviously, the pseudo-jerk is in bang-bang structure, that is, one of the constraints reaches its bound for any parametric value. From Fig. 8, the tracking error is fluctuating between $-0.044$mm to 0.044mm for both $x$ and $y$ axes.

To make a comparison, nominal bounds $\boldsymbol{E}_{\max} = (0.022mm, 0.022mm)$ on the tracking error are added and the method proposed in this paper is used to compute the optimal feedrate. Results are shown in Fig. 9-Fig. 13. In Fig. 12, the *pseudo tracking error* is the

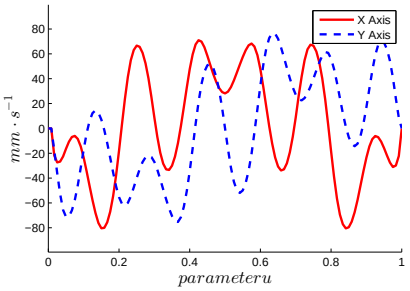


Figure 5: Velocity without tracking error constraint

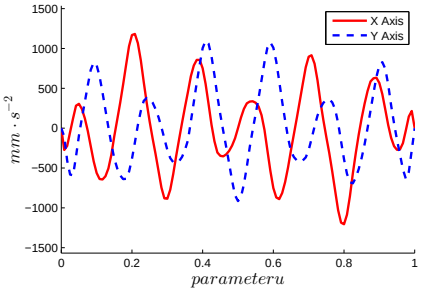


Figure 6: Acceleration without tracking error constraint

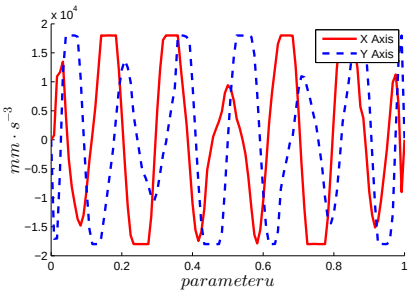


Figure 7: Pseudo jerk without tracking error constraint

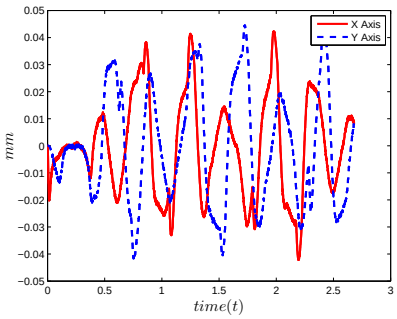


Figure 8: Real time tracking error without tracking error constraint

expression $|\frac{J}{Kk_I}j_\tau + \frac{B}{Kk_I}a_\tau|$ in (3.3), which is used to approximate the tracking error. It can be seen from Fig. 13 that the real time tracking error is confined with the desired bound 0.022mm. The fact that the real-time tracking error bound is approximately the same as the theoretical bound shows that the simplification method proposed in this paper performs nicely. The machining time of the test in Fig. 13 is increased to 3.7s from 2.7s which is the machining time of the test in Fig. 8. This is a reasonable price to pay for reducing the tracking error by half.

Comparing with the test without tracking error bounds, the control is still bang-bang. However, in this case, the pseudo tracking error constraints are bang-bang in most places, as shown by Fig. 12. The reason for that is obvious. From Fig. 8, it can be seen that the tracking errors exceed the bound $0.022mm$ in most places. After adding the tracking error bound, the pseudo tracking error bound will be the control variable in these places, and the velocity and acceleration have to coordinate themselves to satisfy the tracking error constraints.

Finally, a set of data is used to show the computational scalability of problem (3.18). $N$ is set to be different values and the executing times are listed in Table 2. The computation times are taken for 50 iterations of the algorithm, where the algorithm reaches about 99% of the optimum value. The computation times are collected from Matlab on a PC with a 3.0GHz CPU. On the other hand, the direct solution to the discrete form of problem (3.6) using the same parameters fails for all $N$.

| $N$ | 100 | 150 | 200 | 300 | 500 |
|---|---|---|---|---|---|
| Time | 22.34s | 57.24s | 98.68s | 275.15s | 421.64s |

Table 2: Times to solve problem (3.18) for different $N$

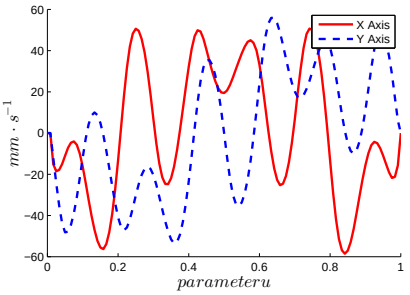


Figure 9: Velocity with tracking error constraint

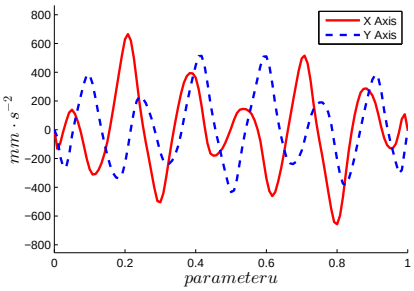


Figure 10: Acceleration with tracking error constraint

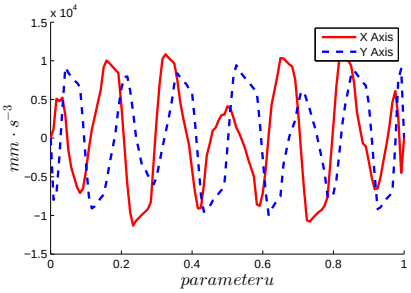


Figure 11: Pseudo jerk with tracking error constraint

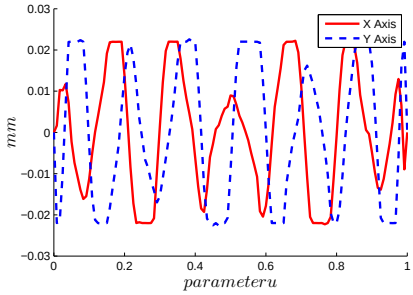


Figure 12: Pseudo tracking error with $\boldsymbol{E}_{\max} = (0.022mm, 0.022mm)$

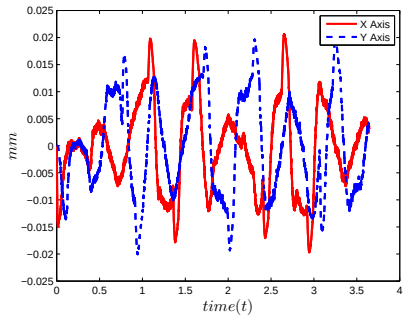


Figure 13: Real-time tracking error $\boldsymbol{E}_{\max} = (0.022mm, 0.022mm)$

**Test 2: Complex eigenvalue case.**

The proportional, integral, derivative gains of the PID controller are chosen as $k_P = 10$ V/mm, $k_I = 480$V/(mm·s), and $k_D = 0.4$ V/(mm/s) for both axes. It can be easily verified

that equation (2.11) has two conjugate complex eigenvalues. For reason of simplicity, $\delta$ is set to be 1.

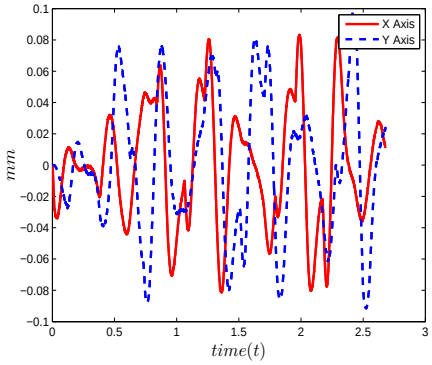


Figure 14: Real-time tracking error without tracking error constraint

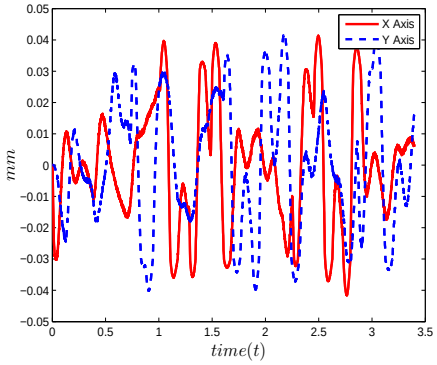


Figure 15: Real-time tracking error with $E_{\max} = 0.035$mm

Similar to Test 1, two experiments are carried out. Firstly, we run the test without considering tracking error constraints, and the result is shown in Fig. 14. In this situation, the maximum tracking error is 0.085mm. Then, nominal bounds on the tracking error $\boldsymbol{E}_{\max} = (0.035mm, 0.035mm)$ are added. The corresponding results are shown in Fig. 15, from which we can see that the maximum tracking error is about $0.04mm$. The bounds $\boldsymbol{E}_{\max}$ are violated slightly, which is due to the omission of the coefficient $\delta$ in problem (3.6).

## 5. Conclusion

How to use the full ability of the CNC machine to increase machining productivity and at the same time to achieve high machining precision is a basic research problem in CNC machining. The problem is usually formulated as a time optimal feedrate planning problem under various constraints. To generate smooth feedrate is an important way to improve machining quality, so bounds for acceleration and jerk are usually used as constraints. On the other hand, the machine dynamic may also lead to large tracking errors. In this paper, bounds of these dynamic errors are also added as constraints in the feedrate planning problem to increase the machining quality. So the minimum-time feedrate planning problem under confined velocity, acceleration, jerk, and tracking error is considered. To be practical, PID controllers are considered, which leads to a third order dynamic system.

The minimum-time feedrate planning problem in this situation is quite involved, and there exist no efficient algorithms for solving it before. The main contribution of this paper is to reduce the above minimum-time feedrate planning problem into a finite state convex optimization problem which can be solved efficiently. The work consists of two key ingredients. First, it is proved that the tracking error constraint can be reduced to a constraint about the linear combination of the acceleration and the jerk. Second, a novel method is introduced to relax the nonlinear constraints involving jerk into linear ones. Experimental

results show that the new convex optimization problem gives nice approximation to the original problem and can be solved efficiently.

It is clear that the machining time is also limited by other factors, such as the cutting force and the quality of the tool-path. It is an interesting problem to consider the time optimal feedrate planning problem under these new constraints.

## References


[1] Ahn HS, Chen YQ, Moore KL. Iterative learning control: brief survey and categorization. *Systems, Man, and Cybernetics*, 2007; 37(6):1099-1121.

[2] Annoni M, Bardine A, Campanelli S, Foglia P, Prete CA. A real-time configurable NURBS interpolator with bounded acceleration, jerk and chord error. *Computer-Aided Design*, 2012; 44:509C521.

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

[4] Boggsa PT and Tolleb JW. *Sequential quadratic programming for large-scale nonlinear optimization. Journal of Computational and Applied Mathematics*, 2000; 124(1-2):123-137.

[5] Chuang HY and Liu CH. A model-referenced adaptive control strategy for improving contour accuracy of multiaxis machine tools. *IEEE Trans. on Industry Applications*, 1992; 8(1):221-227.

[6] Dong JY and Stori JA. Optimal feed-rate scheduling for high-speed contouring. *J Manuf Sci E - T ASME*, 2007; 129(1):63-76.

[7] Dong JY, Ferreiraa PM, Stori JA. Feedrate optimization with jerk constraints for generating minimum-time trajectories. *Int J Mach Tool Manu*, 2007; 47:1941-1955.

[8] Emami MM and 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; 4(7):625-632.

[9] Ernesto CA and Farouki RT. Solution of inverse dynamics problems for contour error minimization in CNC machines. *Int J Adv Manuf Technol*, 2010; 49:589-604.

[10] Fan W, Gao XS, Yan W, Yuan CM. Interpolation of parametric CNC machine tool path under confined jounce. *Int J Adv Manuf Technol*, 2012; 62:719-739.

[11] Golnaraghi F and Kuo BC. *Automatic control systems* (9th edition). John Wiley & Sons, USA, 2010.

[12] Guo JX, Zhang Q, Gao XS, Li H. Time optimal feedrate generation with confined tracking error based on linear programming. Accepted by J Syst Sci Complex, April 2013.

[13] Jacobson N. *Basic algebra I.* W.H. Freeman and Company, San Francisco, 1974.

[14] Koren Y and Lo CC. Advanced controllers for feed drives. *Annals of the CIRP*, 1996; 41:689C698.

[15] Kulkarni PK and Srinivasan K. Optimal contouring control of multi-axial feed drive servomechanisms. *Journal of Engineering for Industry - Trans. of ASME*, 1989; 10:1115.

[16] Kulkarni PK and Srinivasan K. *High speed servo control of multi-axis machine tools. Int J Mach Tool Manu*, 2000; 40:539-559.

[17] Lee AC, Lin MT, Pan YR, Lin WY. The feedrate scheduling of NURBS interpolator for CNC machine tools. *Computer-Aided Design*m 2011; 43:612-628.

[18] Lin MT, Tsai MS, Yau HT. Development of a dynamics-based NURBS interpolator with real-time look-ahead algorithm. *Int J Mach Tool Manu*, 2007; 47:2246-2262.

[19] Riccardo S. Architectures for distributed and hierarchical model predictive control-A review. *Journal of Process Control*, 2009; 19:723-731

[20] Sencer B, Altintas Y, Croft E. Feed optimization for five-axis CNC machine tools with drive constraints. *Int J Mach Tool Manu*, 2008; 48:733-745.

[21] 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; 21:37-53.

[22] Tsai MS, Nien HW, Yau HT. Development of an integrated look-ahead dynamics-based NURBS interpolator for high precision machinery. *Computer-Aided Design*, 2008; 40:554-566.

[23] Verscheure D, Demeulenaere B, Swevers J, De Schutter J, Diehl M. Time-optimal path tracking for robots: a convex optimization approach. *IEEE Trans on Autom Control*, 2009; 54(10):2318-2327.

[24] Yau HT, Lin MT, Tsai MS. Real-time NURBS interpolation using FPGA for high speed motion control. *Computer-Aided Design*, 2006; 38(10):1123-1133.

[25] Yong T and Narayanaswami R. A parametric interpolator with confined chord errors, acceleration and deceleration for NC machining. *Computer-Aided design*, 2003; (35):1249-1259.

[26] Zhang K, Gao XS, Li H, Yuan CM. A greedy algorithm for feed-rate planning of CNC machines along curved tool paths with confined jerk for each axis. *Robotics and Computer Integrated Manufacturing*, 2012; 28:472-483.

[27] Zhang K, Yuan CM, Gao XS. Efficient algorithm for time-optimal feedrate planning and smoothing with confined chord error and acceleration. *Int J Adv Manuf Technol*, 2013; 66(9):1685-1697.