# Time Optimal Feedrate Generation with Confined Tracking Error based on Linear Programming *

Jian-Xin Guo, Qiang Zhang, Xiao-Shan Gao, Hongbo Li
KLMM, Academy of Mathematics and Systems Science
Chinese Academy of Sciences, Beijing 100190, China

December 28, 2012

## Abstract

In this paper, the problem of time optimal feedrate generation under confined feedrate, axis accelerations, and axis tracking errors is considered. The main contribution is to reduce the tracking error constraint to constraints about the axis velocities and accelerations, when the tracking error satisfies a second order linear ordinary differential equation. Based on this simplification on the tracking error, the original feedrate generation problem is reduced to a new form which can be efficiently solved with linear programming algorithms. Simulation results are used to validate the methods.

**Keywords**. Tracking error; time optimal feedrate planning; CNC machining; dynamic; linear programming.

# 1 Introduction

In high speed and high precision CNC machining, it is desired to fully use the capabilities of the CNC machines and at the same time to keep the machining accuracy within a given bound. Therefore, the feedrate generation process is usually formulated as a time minimum planning problem under kinematic constraints such as velocity, acceleration, jerk, and jounce bounds and machining precision constraints such as chord error, tracking error, and contouring error bounds.

Optimal feedrate generation under kinematic constraints was widely studied and efficient algorithms were designed in the cases of confined acceleration (see for instance [1, 2, 3, 4]), confined jerk [5, 6, 7, 8, 9, 10, 11], and confined jounce [12]. After the sampling period of the CNC machine is given, the chord error constraint can be approximately converted into a constraint about the velocity [4, 7]. On the other hand, the tracking error and contouring error are quite different from the above constraints in that they depend on the dynamic parameters of the motor and the controllers used in the CNC machines and hence are more difficult to deal with.

The common method to reduce tracking and contouring 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. Lots of algorithms along this line were developed. Koren proposed the cross-coupled control strategy [13] to directly minimize the contour error. This method is widely adopted in the study of CNC error control. Visioli [14] proposed a tuning method for PID controllers to address the set-point following and the load disturbance rejection problem. Chuang et al [15] proposed a model-referenced adaptive control strategy combined with cross-coupled control of axial motion to improve contouring performance. Kulkarni et al [16] used optimal control methods to design a cross-coupled compensator aimed specifically at improving contouring accuracy in multi-axial feed drive. Since these methods are closed-loop and real-time, in order to use them, the users need to access to the control system. This demands more to the end-users.

An alternative approach is to consider the dynamic constraints in the velocity planning phase, which allows for performance improvements of existing industrial systems without modifying the controllers. Dong and Stori [17, 18] considered the dynamic information in the velocity planning phase by approximating the tracking error constraint with the linear part of the Taylor expansion, which is a linear combination of the velocity and the acceleration. Based on these new constraints, a time minimum feedrate profile is generated. Ernesto and Farouki [19] solved the problem of compensating for inertia and damping of the machine axes by a priori modifying the commanded tool path for CNC machines governed by typical feedback controllers so that the generated tool path is the desired one. The critical point approach was used in both [21] and [22] to compute feedrate with confined jerk and contour error. In [21], Lin et al used the contour error to determine the maximal feedrates at the critical points of the tool-path and use a jerk confined profile to determine the feedrate between two critical points. In [22], Tsai et al used the chord error to determine the maximal feedrates at the critical points and to reduce the contour error by reducing the maximal feedrates at the critical points when the countering error exceeds the limit.

In this paper, we consider the time minimum feedrate planning problem under confined feedrate, axis acceleration, and axis tracking error. The main contribution of the paper is to reduce the tracking error constraint to new forms which can be efficiently solved with linear programming algorithms. When the PD controller is used, the tracking error satisfies a second order linear ordinary differential equation. In this case, we prove that if a linear combination of the velocity and acceleration is bounded by $E$, then the tracking error is also bounded by $E$. As a consequence, the tracking error constraint is reduced rigorously to a constraint about the velocity and acceleration. The tracking error constraint is further relaxed so that the optimal feedrate generation problem can be discretized into a linear programming problem, which has a unique global optimal solution. Simulation results are used to show that the method can be used to solve large scale problems efficiently. We remark that when the tracking error is small enough, the contour error is also small enough. The main reason to constrain the tracking error is that the corresponding model is simple and can be solved efficiently.

Comparing to the work [17], our relaxation of the tracking error is theoretically guaranteed, while the one given in [17] is an approximation. Also, our approach reduces the optimal feedrate generation problem to a linear programming problem which can be solved with an algorithm of polynomial time computational complexity. The method proposed in [17], although more general than ours, is less efficient. Comparing to [21] and [22], our approach generates a time minimum feedrate, while the feedrate generated with methods in [21] and [22] is not time minimum.

The paper is organized as follows. In Section 2, the basic CNC controller dynamic model is described. In Section 3, simplifications for the tracking error constraints are presented. In
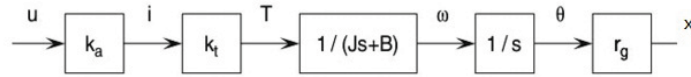
Figure 1: The Motor

Section 4, the optimal feedrate generation problem is reduced to a linear programming problem. In Section 5, simulation results via Matlab/Simulation are presented. In Section 6, concluding remarks are given.

# 2    Machine dynamics

In this section, we will introduce the structure of the motor and controller used in this paper.

## 2.1    Actuator model

In this paper, a 3-axis CNC machine is considered. The three translational axes are assumed to be decoupled and their mechanism parameters are the same for brevity. Here, we mainly discuss the dynamics of the $x$-axis, and similar principles apply to the $y$-axis and $z$-axis.

The control system is composed of two main parts. The first part is the actuator which is generally a permanent magnet synchronous motor. The second part is the controller to be discussed in the next section. Here we consider the universal form of permanent magnet synchronous motor shown in Fig. 1

The motor works as follows. The current amplifier $k_a$ converts the actuating signal $u$ into 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 $w$, determines the axis linear position $x$ through the transmission ratio $r_g$. For brevity, set $K = k_a k_t r_g$ since these three parameters often occur in the form of this product.

## 2.2    PD controller model

The PID controller is widely used CNC systems, where the parameters are respectively the proportional coefficient $k_p$, integral coefficient $k_i$, and derivative gain $k_d$. In this paper, we will use the PD controller to simplify the presentation. The illustration is shown in Fig.2, where $e = X - x$ is the *tracking error* which is the difference of the commanded axis location $X$ and the actual axis location $x$.
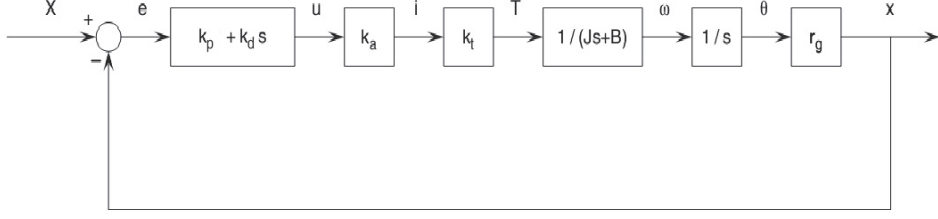
Figure 2: The PD Controller

## 2.3 Relationship between tracking error and input signal

The transfer function of the output $x$ and the input $X$ in the Laplace domain can be written as [19]

$$\frac{x}{X} = \frac{K(k_d s + k_p)}{Js^2 + (B + Kk_d)s + Kk_p}.$$

Thus, the relationship between the tracking error $e_x$ and the input signal $X$ is

$$\frac{e_x}{X} = \frac{Js^2 + Bs}{Js^2 + (B + Kk_d)s + Kk_p}.$$

In the time domain, when the initial values for $e_x$ and its derivative are zero, the above equation can be written as:

$$J\ddot{e}_x + (B + Kk_d)\dot{e}_x + Kk_p e_x = Ja_x + Bv_x.$$

or

$$\frac{J}{Kk_p}\ddot{e}_x + \frac{(B + Kk_d)}{Kk_p}\dot{e}_x + e_x = \frac{J}{Kk_p}a_x + \frac{B}{Kk_p}v_x \qquad (1)$$

where $\dot{e}_x = \frac{de_x}{dt}$, $v_x = \frac{dX}{dt}$, and $a_x = \frac{dv_x}{dt}$ are the tracking error, velocity, and acceleration of the $x$-axis respectively. We assume that $e_x(t)$ is defined in $[0, \infty)$ and $e(0) = \dot{e}(0) = 0$.

# 3 Simplifications of the tracking error constraint

In this section, we will show that the confined tracking error constraint can be replaced by a constraint about the velocity and acceleration. The new constraints will lead to optimization problems which are easier to solve. For this purpose, we need a lemma about differential equations, which will be presented in the next section.

## 3.1 A key lemma

For a function $y(t)$, the following key lemma reduces a bound of a linear combination of $y(t)$, $\dot{y}(t)$, and $\ddot{y}(t)$ to a bound for $y(t)$ itself.

**Lemma 3.1** *Let $y(t)$ be a differentiable function in the time domain $t \in [0, \infty)$ such that $y(0) = \dot{y}(0) = 0$, and $a$, $b$ positive real numbers satisfying $b^2 - 4a \geq 0$. Then $|a\ddot{y} + b\dot{y} + y| \leq E$ implies $|y| \leq E$.*

To prove the above result, we need the following lemma.

**Lemma 3.2** *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 $|r\dot{y} + y| \leq M$, then $|y| \leq M$. establishes.*

*Proof.* Consider $r\dot{y} + y \leq M$, namely, $r\dot{y} + y - M \leq 0$, and define

$$\tilde{y} = y - M.$$

Then $\dot{\tilde{y}} = \dot{y}$, so the inequation $r\dot{y} + y \leq M$ is equivalent to $r\dot{\tilde{y}} + \tilde{y} \leq 0$, or $\dot{\tilde{y}} + \frac{\tilde{y}}{r} \leq 0$. Thus we have

$$e^{\frac{t}{r}}(\dot{\tilde{y}} + \frac{\tilde{y}}{r}) \leq 0,$$

or

$$\left(e^{\frac{t}{r}}\tilde{y}\right)' \leq 0,$$

which means that $e^{\frac{t}{r}}\tilde{y}$ is a monotonously decreasing function in $[0, \infty)$. Therefore, $e^{\frac{t}{r}}\tilde{y} \leq e^{\frac{t_0}{r}}\tilde{y}(t_0)$ for $t \geq t_0$. Since $\tilde{y} = y - M$, we have

$$e^{\frac{t}{r}}(y - M) \leq e^{\frac{t_0}{r}}(y(t_0) - M),$$

or

$$y \leq \frac{e^{\frac{t_0}{r}}(y(t_0) - M)}{e^{\frac{t}{r}}} + M.$$

Setting $t_0 = 0$ and $y(0) = 0$, we have $y(t) \leq M(1 - \frac{1}{e^{\frac{t}{r}}})$. Likewise, if $r(-\dot{y}) + (-y) \leq M$, then $-y \leq M(1 - \frac{1}{e^{\frac{t}{r}}})$. As a consequence, $|y| \leq M(1 - \frac{1}{e^{\frac{t}{r}}}) \leq M$. ∎

Now, we can prove Lemma 3.1.

*Proof of Lemma 3.1.* Since $a > 0$, $b > 0$, and $b^2 - 4a \geq 0$, the two roots of the quadratic equation $p^2 - bp + a = 0$ in $p$ are both positive and we use $p$ to denote one of the roots. Let $\tilde{y} = (b - p)\dot{y} + y$ and note $p^2 - bp + a = 0 \Leftrightarrow \frac{a}{b-p} = p$. We have

$$a\ddot{y} + b\dot{y} + y = (a\ddot{y} + p\dot{y}) + ((b - p)\dot{y} + y) = p\dot{\tilde{y}} + \tilde{y}.$$

By Lemma 3.2, from $|a\ddot{y} + b\dot{y} + y| = |p\dot{\tilde{y}} + \tilde{y}| < E$ and $p > 0$, we have $|\tilde{y}| < E$. Since $b - p = a/p > 0$ and $|\tilde{y}| = |(b - p)\dot{y} + y| < E$, by Lemma 3.2 again, we have $|y| \leq E$. ∎

## 3.2 Simplification of the tracking error constraint

Let the tracking error constraint be

$$|e_\tau(t)| \leq E_\tau \tag{2}$$

where $e_\tau$ and $E_\tau$ are the tracking error and tracking error bound for the axis $\tau \in \{x, y, z\}$. Furthermore, $e_\tau$ satisfies equation (1).

The tracking error constraint in (2) will lead to difficulties when using numerical methods to solve the optimization feedrate generation problem, since it generally will be reduced to complicated nonlinear constraints. As a direct consequence of Lemma 3.1, we have the following theorem which establishes a sufficient condition for the tracking error bound.

**Theorem 3.3** *Use the notations in* (1) *and* (2) *and suppose that the parameters of the motor satisfy*

$$(B + Kk_d)^2 - 4Kk_pJ \geq 0. \tag{3}$$

*If* $\left|\frac{Ja_\tau + Bv_\tau}{Kk_p}\right| \leq E_\tau$, *then the tracking error satisfies* $|e_\tau| \leq E_\tau$ *for each axis* $\tau \in \{x, y, z\}$.

**Remark 3.4** In [15], the power series solution of $e_\tau$ is computed in the Laplace domain and $e_\tau$ is taken to be the linear part which is also a linear combination of $a_\tau$ and $v_\tau$. But this linear combination is only an approximation for $e_\tau$, which does not guarantee the given tracking error bound to be valid. On the other hand, we give a theoretical guaranteed bound.

The following theorem gives a further relaxation for the tracking error bound. As will be shown in the next section, the new constraints obtained with this relaxation can be discretized into linear constraints.

**Theorem 3.5** *Assume the same condition as Theorem 3.3 and let* $\tilde{E}_\tau = \frac{E_\tau^2 Kk_p}{JA_\tau + B}$. *If* $\frac{J|a_\tau| + Bv_\tau^2}{Kk_p} \leq \tilde{E}_\tau$ *for* $\tau \in \{x, y, z\}$, *then* $|e_\tau| \leq E_\tau$.

*Proof.* Using the Cauchy inequality, we have

$$\left|\frac{Ja_\tau + Bv_\tau}{Kk_p}\right| \leq \frac{J|a_\tau| + Bv}{Kk_p} = \frac{\sqrt{J|a_\tau|} \cdot \sqrt{J|a_\tau|} + \sqrt{B}v \cdot \sqrt{B}}{Kk_p} \leq \sqrt{\frac{J|a_\tau| + Bv^2}{Kk_p}}\sqrt{\frac{J|a_\tau| + B}{Kk_p}}.$$

Since $\frac{J|a_\tau| + Bv_\tau^2}{Kk_p} \leq \tilde{E}_\tau$ and $J|a_\tau| + B \leq JA_\tau + B$, we have

$$\left|\frac{Ja_\tau + Bv_\tau}{Kk_p}\right| \leq \sqrt{\frac{\tilde{E}_\tau(JA_\tau + B)}{Kk_p}} = E_\tau.$$

By Theorem 3.3, we have $|e| \leq E_\tau$ and the theorem is proved. ∎

# 4 Optimal feedrate generation under confined tracking error based on linear programming

In this section, we will show that the optimal feedrate generation problem can be reduced to a linear programming problem which can be solved efficiently.

## 4.1 Formulation of the problem

The tool path is assumed to be a spatial parametric curve:

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

which has derivatives at least to the second order. Here, the parametric curve could be B-splines, NURBS, etc. Denote the machining velocity to be $\mathbf{v}(t) = (v_x(t), v_y(t), v_z(t))$ and the tangential velocity (machining feedrate) to be $v_f = \sqrt{v_x^2 + v_y^2 + v_z^2}$. We will use " $'$ " to denote " $\frac{\mathrm{d}}{\mathrm{du}}$ ".

Introduce the following important quantity $q$

$$\dot{u} = \frac{\mathrm{du}}{\mathrm{dt}} = \sqrt{q},$$

which will be used as the optimization variable in our feedrate planning problem. We have

$$\ddot{u} = \frac{\mathrm{d}\dot{u}}{\mathrm{dt}} = \frac{\mathrm{d}\dot{u}}{\mathrm{dt}} \cdot \frac{\dot{u}}{\dot{u}} = \frac{\mathrm{d}(\dot{u}^2)}{2\mathrm{dt}} \cdot \frac{1}{\dot{u}} = \frac{\mathrm{d}(\dot{u}^2)}{2\mathrm{dt}} \cdot \frac{\mathrm{dt}}{\mathrm{du}} = \frac{\mathrm{d}(\dot{u}^2)}{2du} = \frac{1}{2}\left(\dot{u}^2\right)' = \frac{1}{2}q'.$$

Then for the $x$-axis,

$$v_x = \dot{x} = \frac{\mathrm{dx}}{\mathrm{du}}\frac{\mathrm{du}}{\mathrm{dt}} = x'\sqrt{q} \tag{4}$$

$$a_x = \dot{v}_x = \frac{\mathrm{d}(x'\dot{u})}{\mathrm{dt}} = x''\dot{u}^2 + x'\ddot{u} = x''\dot{u}^2 + \dot{x}\ddot{u} = x''q + x'\frac{q'}{2}. \tag{5}$$

Similar formulas can also be given for the $y$-axis and $z$-axis. So

$$v_f = \sqrt{v_x^2 + v_y^2 + v_z^2} = \sqrt{x'^2 q + y'^2 q + z'^2 q} = \sigma\sqrt{q} \tag{6}$$

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

In addition, the machining time $T$ can be written as:

$$T = \int_0^T dt = \int_0^1 \frac{du}{\dot{u}} = \int_0^1 \frac{du}{\sqrt{q}}. \tag{7}$$

The kinematic constraints considered include the tangential feedrate bound and the bounds on the axis accelerations, which mainly reflect the capabilities of the CNC machine. Furthermore, the tracking error bounds for all axes are considered, which are used to control the accuracy of the machining. Thus the feedrate generation problem can be formulated as a time-minimum problem under the kinematic and tracking error constraints, that is, to design the parametric velocity $\dot{u}$ along the tool path such that the machining time is minimized under the given constraints:

$$\min_{u(t)} T = \min_{u(t)} \int_0^1 \frac{du}{\sqrt{q}} \tag{8}$$

$$s.t. \begin{cases} |v_f| \leq V_{\max} \\ |a_\tau| \leq A_\tau \\ |e_\tau| \leq E_\tau \\ \frac{J}{Kk_p}\ddot{e}_\tau + \frac{(B+Kk_d)}{Kk_p}\dot{e}_\tau + e_\tau = \frac{J}{Kk_p}a_\tau + \frac{B}{Kk_p}v_\tau \end{cases} \tag{9}$$

where $a_\tau$ is the acceleration for $\tau \in \{x, y, z\}$, $e_\tau$ is the tracking error for $\tau \in \{x, y, z\}$, and $V_{\max}, A_\tau, E_\tau$ are the bounds for the feedrate, axis accelerations, and axis tracking errors respectively.
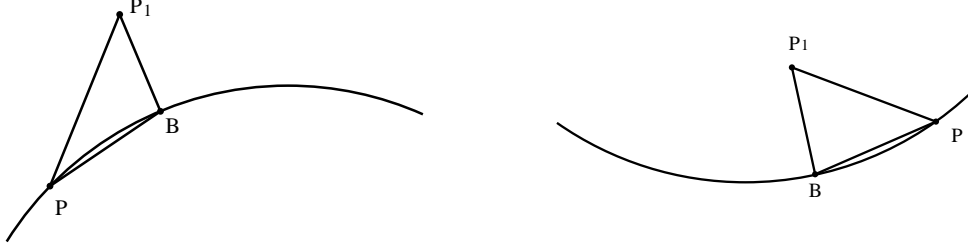
Figure 3: Contour error and tracking error

**Remark 4.1** As mentioned in Section 1, the contour error is considered in most existing work. Here, we give three reasons for controlling the tracking error. Firstly, the contour error is bounded by the tracking error in certain sense. In Fig. 3, $P$ is the intended interpolation point, $P_1$ is the actual position, and $|P_1B|$ is the contour error. It is easy to see that $|PP_1| = \sqrt{e_x^2 + e_y^2 + e_z^2}$ where $e_x$, $e_y$, $e_z$ are the tracking errors for the axes. Since $P$ and $B$ are generally very close, we can approximately assume that the tool-path near points $P$ and $B$ is a circular arc segment. In this case, it is easy to show that $|P_1B| \leq |PP_1| = \sqrt{e_x^2 + e_y^2 + e_z^2}$. Therefore, given a bound $\delta$ for the contour error, if we set $e_x = e_y = e_z = \delta/\sqrt{3}$ then the tracking error bounds implies the contour error bound. Secondly, although it is the contour error that affects the machining shape, accumulation of tracking error could also lead to problems at the connection points of two tool path segments. Thirdly, as we will show in this paper, the optimal feedrate generation problem under tracking error bounds can be solved efficiently, while the corresponding problem under contour error bounds seems more difficult to solve.

## 4.2 Simplification of the problem

In this section, we will give two new forms for problem (8), which are easier to solve. We first state two properties for problem (8).

It is shown that the time-minimum problem under confined acceleration has a bang-bang control structure [1, 2, 3]. This is also valid when the tracking error is added for the following reasons. If the tracking error reaches its bounds, then we already have a bang-bang structure. Otherwise, let us assume that the tracking error does not reach its bound in the time interval $[a, b]$. Then in the time interval $[a, b]$, the tracking error constraint is inactive and the feedrate generation problem under velocity and acceleration constraints must also be time-minimum and hence bang-bang either in the velocity or in the acceleration following [1, 2, 3]. We thus have

**Property one**. The solution to the optimal problem (8) is bang-bang-singular in the sense that for any parametric value $u \in [0, 1]$, at least one of the constraints in (9) reaches its bound.

The above property is useful in that if a numerically computed solution is bang-bang-singular, then with a high probability we can assert that it is an optimal solution.

By Theorem 3.3, the tracking error constraint $e_\tau(t) \leq E_\tau$ can be replaced by the new constraint $|\frac{Ja_\tau}{Kk_p} + \frac{Bv_\tau}{Kk_p}| \leq E_\tau$ when the predetermined controller coefficients and the machine parameters satisfying condition (3). As a consequence, instead of the initial problem (8), we can

consider the following optimization problem.

$$\min_{u(t)} T = \min_{u(t)} \int_0^1 \frac{\mathrm{d}u}{\sqrt{q}} \tag{10}$$

$$s.t. \begin{cases} |v_f| \le V_{\max} \\ |a_\tau| \le A_\tau \\ \left| \frac{Ja_\tau + Bv_\tau}{Kk_p} \right| \le E_\tau \end{cases} \tag{11}$$

From [20, p. 385], for an optimal feedrate generation problem whose constraints involve the velocity and acceleration only, the optimal solution is unique and is maximum among all feasible feedrate solutions at any time. We thus have the following property.

**Property Two**. The optimal solution of problem (10) is unique and is maximum among all feasible solutions at any time.

Based on the above property, problem (10) is equivalent to the following problem.

$$\max_{u(t)} \int_0^1 q \, \mathrm{d}u \tag{12}$$

$$s.t. \begin{cases} |v_f| \le V_{\max} \\ |a_\tau| \le A_\tau \\ \left| \frac{Ja_\tau + Bv_\tau}{Kk_p} \right| \le E_\tau \end{cases} \tag{13}$$

Intuitively, instead of minimize the machining time, we can maximize the machining velocity. Furthermore, by Theorem 3.5, instead of the optimization problem (12), we can consider the following optimization problem:

$$\max_{u(t)} \int_0^1 q \, \mathrm{d}u \tag{14}$$

$$s.t. \begin{cases} |v_f| \le V_{\max} \\ |a_\tau| \le A_\tau \\ \frac{Ja_\tau + Bv_\tau^2}{Kk_p} \le \tilde{E}_\tau \\ \frac{-Ja_\tau + Bv_\tau^2}{Kk_p} \le \tilde{E}_\tau \end{cases} \tag{15}$$

where $\tilde{E}_\tau = \frac{E_\tau^2 Kk_p}{JA_\tau + B}$ and $\tau \in \{x, y, z\}$,

## 4.3  Discrete forms of the optimal feedrate generation problem

In order to solve problems (12) and (14), we will discretize them into mathematical programming problems. To do that the parametric interval $[0, 1]$ is divided into $N$ equal intervals with knots $u_i = \frac{i}{N}$, $i = 0, 1, \ldots, N$. The length of each interval is $\Delta = \frac{1}{N}$. When $\Delta$ is very small, the constraints can be approximately transformed into discrete inequalities. Use subscript $i$ to represent the corresponding value of a variable at $u_i = \frac{i}{N}$ (for instance, $q_i = q(\frac{i}{N})$).

When $N$ is large enough, the directives of $q$ can be approximated by finite differences:

$$\begin{cases} q_i' \approx \frac{q_{i+1} - q_{i-1}}{2\Delta} \\ q_i'' \approx \frac{q_{i+1} + q_{i-1} - 2q_i}{\Delta^2}. \end{cases}$$

Based on equations (5) - (7), the optimization problem (12) can be approximated with the following *nonlinear programming problem*:

$$\max_{q_i} \sum q_i \tag{16}$$

$$s.t. \begin{cases} \left| \sigma_i^2 q_i \right| \le V_{\max}^2 \\ \left| \tau_i^{''} q_i + \frac{\tau_i^{'}}{4\Delta}(q_{i+1} - q_{i-1}) \right| \le A_\tau \\ \left| \frac{J}{Kk_p}(\tau_i^{''} q_i + \frac{\tau_i^{'}}{4\Delta}(q_{i+1} - q_{i-1})) + \frac{B}{Kk_p}(\tau_i^{'}\sqrt{q_i}) \right| \le E_\tau \end{cases} \tag{17}$$

where $i = 0, 1, \ldots, N$ and $\tau \in \{x, y, z\}$. It is easy to see that the first two sets of constraints in (17) are linear in $q_i$, but the third set is nonlinear in $q_i$.

Based on equations (5)-(7), the optimization problem (14) can be approximated with the following *linear programming problem*:

$$\max_{q_i} \sum q_i \tag{18}$$

$$s.t. \begin{cases} \left| \sigma_i^2 q_i \right| \le V_{\max}^2 \\ \left| \tau_i^{''} q_i + \frac{\tau_i^{'}}{4\Delta}(q_{i+1} - q_{i-1}) \right| \le A_\tau \\ (\frac{J}{Kk_p}(\tau_i^{''} q_i + \frac{\tau_i^{'}}{4\Delta}(q_{i+1} - q_{i-1})) + \frac{B}{Kk_p}(\tau_i^{'2} q_i)) \le \tilde{E}_\tau \\ (-\frac{J}{Kk_p}(\tau_i^{''} q_i + \frac{\tau_i^{'}}{4\Delta}(q_{i+1} - q_{i-1})) + \frac{B}{Kk_p}(\tau_i^{'2} q_i)) \le \tilde{E}_\tau \end{cases} \tag{19}$$

where $i = 0, 1, \ldots, N$, $\tau \in \{x, y, z\}$, and $\tilde{E}_\tau = \frac{E_\tau^2 Kk_p}{JA_{\max}+B}$. For such a linear optimization problem, a local optimal solution is also the unique global optimization solution and the computational complexity to find the optimal solution is also known.

**Theorem 4.2** *For a given $N$, the worst case computational complexity to solve problem* (18) *is $O(N^{3.5})$ in terms of floating point arithmetic operations.*

Note that the number of variables and the number of constraints for the linear programming problem (18) are both $O(N)$. Based on Karmarkar's famous algorithm [23] to solve linear programming problems, the complexity to find the optimal solution for a linear programming problem is $O(N^{3.5})$ arithmetic operations for floating point numbers. Thus, we have Theorem 4.2.

**Remark 4.3** It should be noticed that the method proposed in this paper can be used to any control system where the tracking error satisfies a second order differential equation. For instance. In the control system shown in Fig.4, the PD controller is used to fulfill the position loop tracking control and the P controller is used to control the velocity. The tracking error for this system satisfied the following second order differential equation.

$$J\ddot{e} + (B + Kk_{vp}k_{pd} + Kk_{vp})\dot{e} + Kk_{vp}k_{pp}e = Ja + Bv.$$

# 5   Simulation

We implemented our algorithm and performed simulation using Matlab/Simulink. In this section, we will present the simulation results which show that the linear programming problem
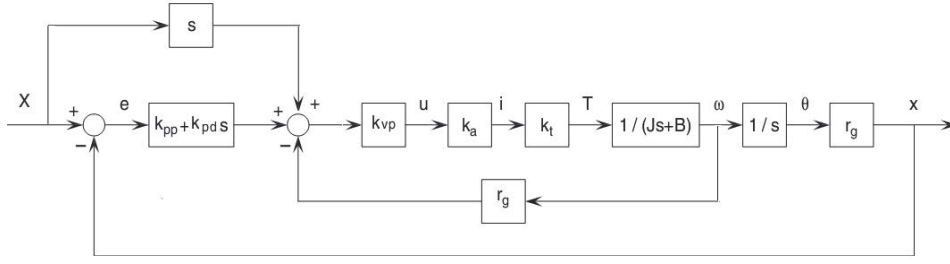
Figure 4: PD-P Controller

(18) can be efficiently solved and the feedrate provided by the algorithm is guarantied to have bounded tracking error.

The planar tool-path trident in Fig.4 is from [22]. The tool-path is initialized from the origin point $(0,0)$ in order to use the inverse Laplace transformation. The following parameters are used in the simulation: $V_{\max} = 200$mm/s, $A_x = A_y = 1000$mm/s$^2$, $E_x = E_y = 0.1$mm, $K = k_a k_t r_g = 0.2$Nm$^2$ /V, $J = 0.03$kgm$^2$, $B = 0.05$kgm$^2$/s, $k_p = 1000$V/mm, and $k_d = 25$V/(mm/s). The parameter $N$ in the discretization is taken as $N = 100$. From these parameters, it can be calculated $\tilde{E}_x = \tilde{E}_y = 0.07$mm. It is easy to show that condition (3) is valid and problem (18) can be used to find an approximate solution for the original optimal feedrate generation problem.

We first compute the optimal feedrate without considering the tracking error bound with the method given in [4]. The tangent velocity and the separate axis acceleration are shown in Fig. 5 and Fig. 6 respectively. Clearly, the control is bang-bang-singular in the sense that one of the constraints reaches its bound at any time. Fig. 7 is the real time tracking error obtained by feeding the interpolation points obtained from the velocity function in Fig. 5 into Matlab/Simulink, where the interpolation period and the controlling calculated time are both 0.001s. Obviously, the peak value of the tracking error is about 0.25mm.

Next, the tracking error bound $E_x = E_y = 0.1$mm is added and the optimization problem (18) is solved using the linear toolbox in Matlab to find the feedrate. The results are illustrated in Fig. 8 - Fig. 11. From Fig. 10, it can be seen that the linear combination of $a_\tau$ and $v_\tau^2$ in (15) reaches its bounds $\tilde{E}_\tau$ at almost all places, which means that the feedrate obtained with our method is an optimal solution to problem (18) due to the discussion about the bang-bang structure in Section 4.2. From Fig. 10, we can see that the maximal tracking error is around 0.06mm. This means that although the tracking error bound is satisfied, when replacing the constraint $e_\tau \leq E_\tau$ with $\frac{Ja_\tau + Bv_\tau^2}{Kk_p} \leq \tilde{E}_\tau$, the feasible space of feedrate becomes smaller. Notice that the machining time is increased from 1.1s to 3.9s when the tracking error is reduced from 0.25mm to 0.06mm.

In Fig. 12 - Fig. 14, the sharp corners in the trident curve are illustrated. In these figures, the solid line segments marked with green stars denote the original curve, the blue dotted line denotes the output without considering tracking errors in the velocity planning, and the red dotted line denotes the output with the tracking error bound 0.1mm. We can see that the contour error is improved evidently.
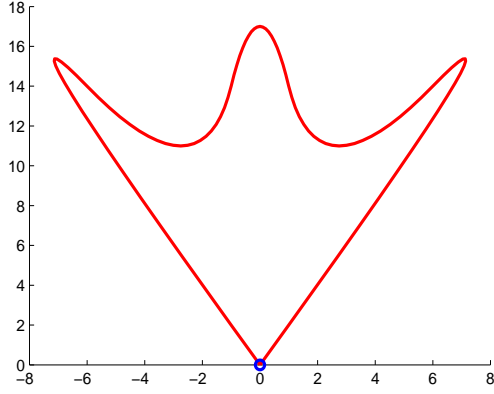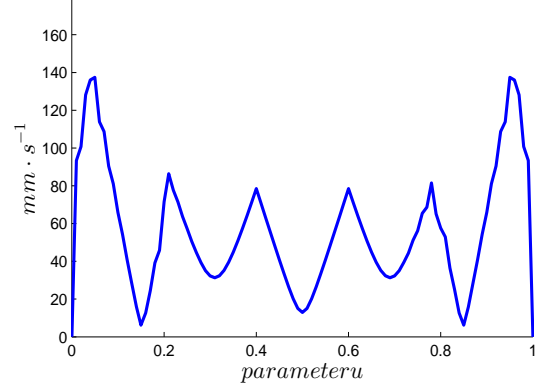
11

Figure 5: Trident Curve
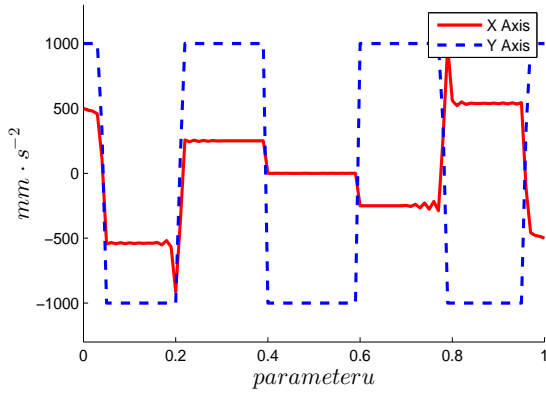


Figure 6: Tangent Velocity



Figure 7: Axis Acceleration
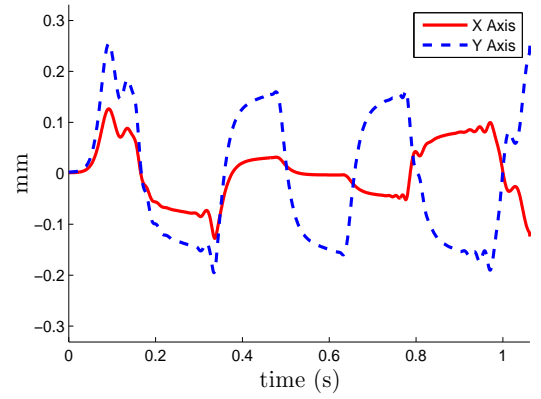


Figure 8: Tracking Error in real time

Finally, a set of data is used to compare the scalability of the problem (16) which is a nonlinear programm problem (NLP) and (18) which is a linear programm problem (LP). $N$ is set to be different values and executing times are collected and are listed in Table 1. As expected, problem (18) can be efficiently solved for large scale problems.

| $N$ | 100 | 150 | 200 | 300 | 500 |
|---|---|---|---|---|---|
| Time(NLP) | 20.4s | 8.10s | 10.80s | Fail | Fail |
| Time(LP) | 0.25s | 0.40s | 0.50s | 0.50s | 1.50s |

Table 1: Computation times comparison between NLP and LP for different $N$

# 6 Conclusion

An approach is proposed to reduce the tracking error of high-speed CNC systems while still maintain the machining time optimality. The PD controller is used here to illustrate the approach. Compared with the existing work, we establish a precise relationship between the tracking error bound and the bound for a linear combination of velocity and acceleration. The linear combi-
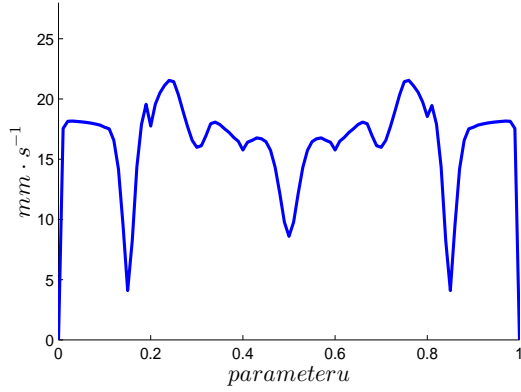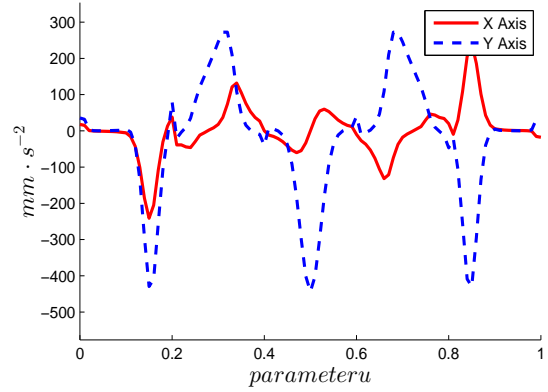
12

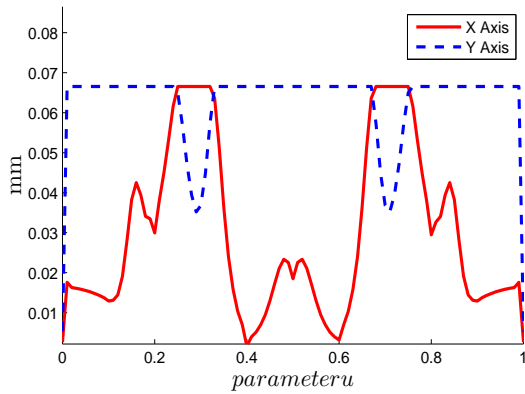Figure 9: Tangent velocity



Figure 10: Axis acceleration



Figure 11: $\tilde{e}(u) = \frac{J|a_\tau| + Bv_\tau^2}{Kk_p}$
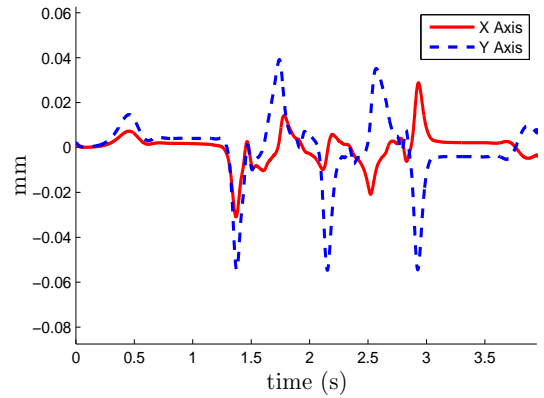


Figure 12: Tracking error with $\tilde{E}_\tau = 0.07$mm

nation is further reduced to boost the efficiency of solving the optimization algorithm. With these simplifications, the original problem is converted into a linear programming problem which can be solved efficiently for large scale problems. The main advantage of this approach is that the method can be implemented outside of the control loop, requiring no access to the control system. This is very convenient for the end-user since it allows for performance improvements of existing industrial systems without modifying their controllers.

# References

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

[2] Shiller Z. On singular time-optimal control along specified paths. *IEEE Trans. Robot. Autom.*, 10, 561-566, 1994.

[3] Timar S.D., Farouki R.T. Time-optimal traversal of curved paths by Cartesian CNC machines under both constant and speed-dependent axis acceleration bounds. *Robotics and Computer-Integrated Manufacturing*, 23(5), 563-579, 2007.
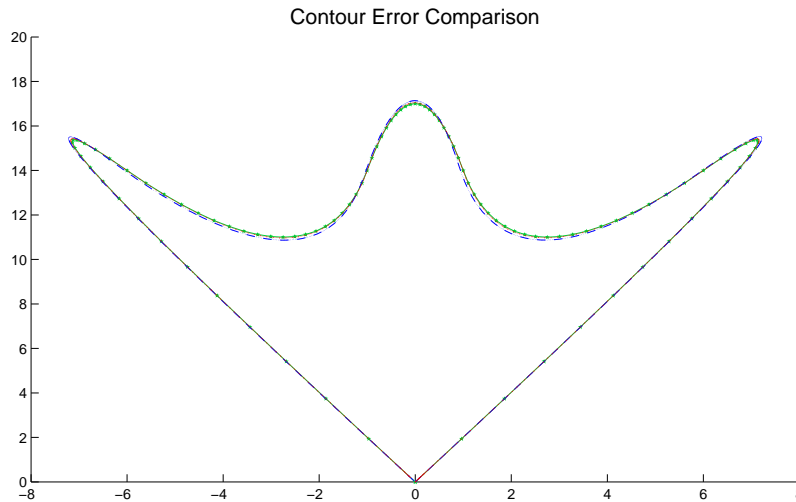
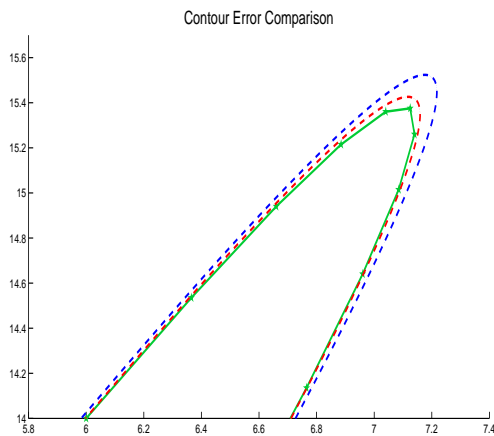Figure 13: Contour Error Comparison
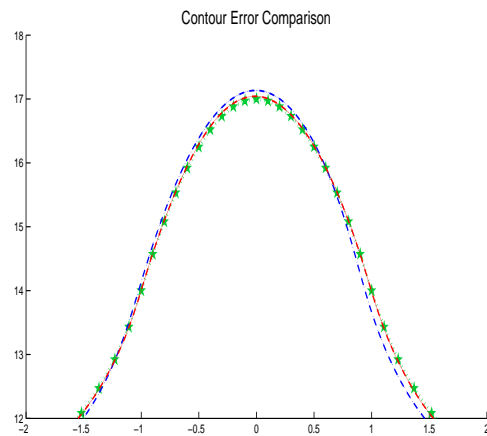


Figure 14: Right Corner



Figure 15: Middle Ridge

[4] Zhang K., Yuan C.M., Gao X.S. Efficient algorithm for time-optimal feedrate planning and smoothing with confined chord error and acceleration. *Int. J. Adv. Manuf. Technol.*. DOI 10.1007/s00170-012-4450-3.

[5] Erkorkmaz K., Altintas Y. High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation. *Int. J. of Mach. Tools and Manu.* 41(2001)1323-1345

[6] Sencer B., Altintas Y., Croft E. Feed optimization for five-axis CNC machine tools with drive constraints. *Int. J. of Mach. Tools and Manu.* 48 (2008) 733-745.

[7] Lai J.Y., Lin K.Y., Tseng S.J., Ueng W.D. On the development of a parametric interpolator with confined chord error, feedrate, acceleration and jerk. *Int. J. Adv. Manuf. Technol.* 37(2008)104-121

[8] Emami M.M., Arezoo B. A look-ahead command generator with control over trajectory and chord error for NURBS curve with unknown arc length. *Computer-Aided Design* 4(7)(2010)625-632.

[9] Lee A.C., Lin M.T., Pan Y.R., Lin W.Y. The feedrate scheduling of NURBS interpolator for CNC machine tools. *Computer-Aided Design* 43(2011)612-628.

[10] Gasparetto A., Lanzutti A., Vidoni R., Zanotto V. Experimental validation and comparative analysis of optimal time-jerk algorithms for trajectory planning *Robotics and Computer-Integrated Manufacturing* 28(2012)164-181.

[11] Zhang K., Gao X.S., Li H., Yuan C.M. 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*, 28 (2012) 472-483.

[12] Fan W., Gao X.S., Yan W., Yuan C.M. Interpolation of parametric CNC machine tool path under confined jounce. *Int. J. Adv. Manuf. Technol.* 62(2012) 719-739.

[13] Koren Y. Cross-coupled biaxial computer control for manufacturing systems. *ASME Transactions, Journal of Dynamic Systems, Measurement and Control.* 102(4)(1980) 265-272.

[14] Visioli A. Optimal tuning of PID controllers for integral and unstable processes. *MIEEE Proceedings Control Theory and Applications*, 2006; 148(2):180-184.

[15] Chuang H.Y. and Liu C.H. A model-referenced adaptive control strategy for improving contour accuracy of multiaxis machine tools. *IEEE Trans. on Industry Applications*, 1992; 28(1):221-227.

[16] Kulkarni P.K., Srinivasan K. Optimal contouring control of multi-axial feed drive servomechanisms. *Journal of Engineering for Industry-transactions of the ASME*, 1989; 10:1115.

[17] Dong J., Stori J.A. Optimal feed-rate scheduling for high-speed contouring. *Journal of Manufacturing Science and Engineering*, 2007; 129(1):63-76.

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

[19] Ernesto C.A., Farouki R.T. Solution of inverse dynamics problems for contour error minimization in CNC machines. *Int. J. Adv. Manuf. Technol.*, 2010; 49:589-604.

[20] Dong J, Stori J.A. A Generalized Time-Optimal Bidirectional Scan Algorithm for Constrained Feed-Rate Optimization. *Journal of Dynamic Systems, Measurement, and Control.*, 2006; 128(2):379-390.

[21] Lin M.T., Tsai M.S., Yau H.T., Development of a dynamics-based NURBS interpolator with real-time look-ahead algorithm. *Int. J. of Mach. Tools and Manu.* 2007;47:2246-2262.

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

[23] Karmarkar N. A new polynomial time algorithm for linear programming. *Combinatorica* 4(4)(1984)373-395.