Jerk-Limited Oscillation-Free Feedrate Scheduling Under Non-Stationary Boundary Conditions

Yunan Wang, Chuxiong Hu*, Jichuan Yu, Shize Lin, Zhao Jin, and Jizhou Yan

***Abstract*—Feedrate scheduling has significant impacts on motion efficiency, equipment vibration, and machining quality in robotic manipulation and computer numerical control machining. As the most effective methods for jerk-limited feedrate scheduling, however, optimization-based approaches face challenges such as high computational cost, artificial infeasibility, and feedrate oscillations. This paper proposes a triple linear programming (TLP) method for solving the non-convex 3rd-order problem. To avoid artificial infeasibility caused by convexification, an incremental linearization method (ILM) is developed to generate a feasible solution under non-stationary boundary conditions. Feedrate profiles are further adjusted to eliminate the oscillations caused by the discretization. In experiments on machine tools and robotic manipulators, the proposed method saves more than 10% of motion time than existing linear programming methods and reduces computational time by more than 80% than baselines based on sequential quadratic programming with better time-optimality. Furthermore, the proposed method outperforms baselines regarding feasibility and feedrate oscillations.**

***Index Terms*—Feedrate scheduling, jerk constraint, feedrate oscillation, linear programming, trajectory planning.**

I. Introduction

Feedrate scheduling is a fundamental problem in robotic manipulation [1] and computer numerical control machining [2], [3], which has significant impacts on efficiency [4], [5], equipment vibration [6], [7], and surface quality [8]. In the widely applied two-step motion planning framework, the aim of feedrate scheduling is to generate a feedrate profile to drive the equipment along a given path within a minimal time, while satisfying user-specified constraints. For example, jerk constraints are widely introduced to avoid equipment vibrations and enhance machining quality [3]. The planned profiles are subsequently tracked by low-level controllers to deal with model inaccuracies, external loads, and disturbances.

There exist some approaches to solve the feedrate scheduling problem, such as numerical integration [9], dynamic programming [6], reachability analysis [10], convex optimization [1], and curve templates like S-shaped curves [3]. When considering nonlinear dynamics and strict jerk constraints, optimization-based methods are currently the most effective ones, achieving high numerical stability. However, discretization and the non-convex nature of jerk constraints give rise to pervasive challenges [1], [11], such as high computational cost, artificial infeasibility, and feedrate oscillations.

A limited computational cost is essential for online high-speed machining and robotic manipulation facing unknown environments [12]. Since Debrouwere et al. [1] established the fundamental formulation of 3rd-order problems, extensive research has been devoted to linearization. Erkorkmaz et al. [2] used the initial solution of [1] as the final trajectory with linear programming. Ji et al. [13] assumed a spline form of feedrate profiles and transformed the problem into a linear one. However, the pseudo-jerk is introduced for convexification, leading to conservative scaling and hindering the time-optimality.

After the convexification of non-convex 3rd-order problems, the feasible regions are artificially restricted, which might lead to artificial infeasibility [11]. Considering the special structure of 3rd-order constraints, an item of parametric velocity is conservatively substituted by an upper bound which is usually obtained by a 2nd-order optimal solution [2], [13]. The above process is feasible for stationary boundary, i.e., the initial and terminal velocities as well as accelerations are zero [14]. However, the feasibility is not guaranteed under non-stationary boundary conditions, especially when the boundary velocity is large and the upper bound of the jerk is small.

Discretization of the optimization horizon leads to constraint exceedance in each grid interval [10]. Although the discretization can be refined to reduce the exceedance, the computational cost significantly increases [15]. Theoretically speaking, constraint exceedances within a certain range are acceptable since the specified constraints in machining are lower than the upper limits of mechanical performance. However, the exceedance of velocity constraints leads to feedrate oscillations, which are harmful to equipment vibration and machining quality [8]. The elimination of feedrate oscillations is critical for high-speed machining and robotic manipulation.

This paper aims to address the above challenges in jerk-limited feedrate scheduling, i.e., high computational cost, artificial infeasibility, and feedrate oscillations. A triple linear programming method is proposed to solve the non-convex 3rd-order problem. To avoid artificial infeasibility, an incremental linearization method is proposed to generate a feasible solution. Feedrate profiles are further adjusted to eliminate oscillations caused by discretization. Experiments on machine

This work was supported in part by the National Natural Science Foundation of China under Grant 624B2077, and in part by the National Key Research and Development Program of China under Grant 2023YFB4302003.

All authors are with the State Key Laboratory of Tribology in Advanced Equipment, Department of Mechanical Engineering, Tsinghua University, Beijing, 100084, China, and also with the Beijing Key Laboratory of Transformative High-end Manufacturing Equipment and Technology, Department of Mechanical Engineering, Tsinghua University, Beijing 100084, China.

* Corresponding author: Chuxiong Hu (email: cxhu@tsinghua.edu.cn).

tools and robotic manipulators demonstrate that the proposed method outperforms baselines regarding motion efficiency, computational cost, feasibility, and feedrate oscillations.

II. Problem Formulation

Considering n-axis equipment like machine tools and robotic manipulators, a regular path of $\mathcal{C}^3$ continuouity is given as $\mathcal{P}: \boldsymbol{q} = \boldsymbol{q}(u)$, $u \in [0,1]$, where the configuration vector $\boldsymbol{q} \in \mathbb{R}^n$. The task is to generate a feedrate profile $u = u(t)$, $t \in [0, t_\mathrm{f}]$ to drive the equipment along $\mathcal{P}$ within minimal time t_f while satisfying some user-specified constraints. Denote $\bullet'$, $\bullet''$, $\bullet'''$, $\dot{\bullet}$, $\ddot{\bullet}$, and $\dddot{\bullet}$ as $\frac{\mathrm{d}\bullet}{\mathrm{d}u}$, $\frac{\mathrm{d}^2\bullet}{\mathrm{d}u^2}$, $\frac{\mathrm{d}^3\bullet}{\mathrm{d}u^3}$, $\frac{\mathrm{d}\bullet}{\mathrm{d}t}$, $\frac{\mathrm{d}^2\bullet}{\mathrm{d}t^2}$, and $\frac{\mathrm{d}^3\bullet}{\mathrm{d}t^3}$, respectively. Consider the following problem:

$$\min\ t_\mathrm{f}, \tag{1a}$$
$$\text{s.t. } 0 \le \dot{u} \le v(u), \tag{1b}$$
$$\boldsymbol{n}(u)\dot{u}^2 + \boldsymbol{m}(u)\ddot{u} \le \boldsymbol{g}(u), \tag{1c}$$
$$\boldsymbol{r}(u)\dot{u}^3 + \boldsymbol{s}(u)\dot{u}\ddot{u} + \boldsymbol{t}(u)\dddot{u} \le \boldsymbol{f}(u), \tag{1d}$$
$$u(0) = 0,\ \dot{u}(0) = \dot{u}_0,\ \ddot{u}(0) = \ddot{u}_0, \tag{1e}$$
$$u(t_\mathrm{f}) = 1,\ \dot{u}(t_\mathrm{f}) = \dot{u}_\mathrm{f},\ \ddot{u}(t_\mathrm{f}) = \ddot{u}_\mathrm{f}. \tag{1f}$$

The dynamic equation is implicitly included as $\frac{\mathrm{d}}{\mathrm{d}t}(u, \dot{u}, \ddot{u}) = (\dot{u}, \ddot{u}, \dddot{u})$. The constraints (1b)-(1d) are user-specified, which covers some typical constraints as follows [1]. For example,

$$\dot{\boldsymbol{q}} = \boldsymbol{q}'\dot{u},\ \ddot{\boldsymbol{q}} = \boldsymbol{q}''\dot{u}^2 + \boldsymbol{q}'\ddot{u},\ \dddot{\boldsymbol{q}} = \boldsymbol{q}'''\dot{u}^3 + 3\boldsymbol{q}''\dot{u}\ddot{u} + \boldsymbol{q}'\dddot{u}. \tag{2}$$

Evidently, the velocity constraint $\|\dot{\boldsymbol{q}}\| \le V_\mathrm{max}$, the acceleration constraint $|\ddot{\boldsymbol{q}}| \le \ddot{\boldsymbol{q}}_\mathrm{max}$, and the jerk constraint $|\dddot{\boldsymbol{q}}| \le \dddot{\boldsymbol{q}}_\mathrm{max}$ can be represented by (1b), (1c) and (1d), respectively. The boundary conditions (1e) and (1f) provide the initial and terminal positions, velocities, and accelerations. Consider the dynamic model of the equipment as

$$\begin{aligned}\boldsymbol{\tau} =& \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{q}}^\top \boldsymbol{C}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) \\ =& \left(\boldsymbol{M}\boldsymbol{q}'' + \boldsymbol{q}'^\top \boldsymbol{C}\boldsymbol{q}'\right)\dot{u}^2 + \boldsymbol{M}\boldsymbol{q}'\ddot{u} + \boldsymbol{G}.\end{aligned} \tag{3}$$

The torque constraint $|\boldsymbol{\tau}| \le \boldsymbol{\tau}_\mathrm{max}$ can be represented by (1c). Multiple constraints can be dealt with in one time, noting that constraints (1c) and(1d) are vector inequalities.

Without loss of generality, assume that $\boldsymbol{f}(u) > \boldsymbol{0}$ holds in (1d). In this case, the equipment is allowed to move arbitrarily slowly under the constraints. To eliminate the free terminal time t_f, a classical transformation [1] is applied as follows:

$$a(u) = \dot{u}^2,\ b(u) = \frac{1}{2}a'(u) = \ddot{u},\ c(u) = b'(u) = \frac{\dddot{u}}{\dot{u}}. \tag{4}$$

Then, problem (1) is transformed into the parametric domain as follows:

$$\max\ J = \int_0^1 a(u)\,\mathrm{d}u, \tag{5a}$$
$$\text{s.t. } a'(u) = 2b(u),\ b'(u) = c(u), \tag{5b}$$
$$0 \le a \le v^2(u), \tag{5c}$$
$$\boldsymbol{n}(u)a + \boldsymbol{m}(u)b \le \boldsymbol{g}(u), \tag{5d}$$
$$\boldsymbol{r}(u)a + \boldsymbol{s}(u)b + \boldsymbol{t}(u)c \le \boldsymbol{f}(u)a^{-1/2}, \tag{5e}$$
$$a(0) = \dot{u}_0^2,\ b(0) = \ddot{u}_0,\ a(1) = \dot{u}_\mathrm{f}^2,\ b(1) = \ddot{u}_\mathrm{f}. \tag{5f}$$

In (5a), the objective is to maximize the parametric velocity a over the path $\mathcal{P}$, which is capable of minimizing the time t_f in (1) in an approximate sense.

III. Methods

This section aims to solve problem (5) for a time-optimal trajectory satisfying user-specified constraints without feedrate oscillations. In Section III-A, a novel method based on triple linear programming (TLP) is proposed to solve problem (5). To avoid artificial infeasibility caused by the linearization, an incremental linearization method (ILM) is proposed to generate a feasible solution in Section III-B. Then, the proposed TLP-ILM method is compared with existing methods in Section III-C. Finally, to eliminate the feedrate oscillations caused by the discretization, a feedrate adjusting method is proposed in Section III-D.

A. Solving Problem (5) Based on Triple Linear Programming

Evidently, problem (5) is non-convex due to the concave term $* \le \boldsymbol{f}a^{-1/2}$ in the 3rd-order constraint (5e). Note that $\boldsymbol{f}(u) > \boldsymbol{0}$ holds. In other words, the feasible region of problem (5) is non-convex due to the non-convex inequality constraint (5e). The non-convexity is harmful to the computational efficiency and the numerical stability of the optimization algorithm.

To address this issue, a novel method based on triple linear programming (TLP) is proposed to solve problem (5). First, a 2nd-order solution $\bar{a}(u)$ is solved by the following problem:

$$\max\ \bar{J} = \int_0^1 \bar{a}(u)\,\mathrm{d}u, \tag{6a}$$
$$\text{s.t. } \bar{a}' = 2\bar{b},\ 0 \le \bar{a} \le v^2,\ \boldsymbol{n}\bar{a} + \boldsymbol{m}\bar{b} \le \boldsymbol{g}, \tag{6b}$$
$$\bar{a}(0) = \dot{u}_0^2,\ \bar{a}(1) = \dot{u}_\mathrm{f}^2. \tag{6c}$$

In other words, the 3rd-order constraints (5e) are temporarily ignored. In this way, the feasible region of the 2nd-order problem (6) is convex.

Note that the 2nd-order solution $\bar{a}(u)$ is infeasible for the original 3rd-order problem (5) since the 3rd-order constraints (5e) are violated. To address the above issue, a feasible solution $(\tilde{a}, \tilde{b}, \tilde{c})$ for problem (5) is obtained by the following problem:

$$\max\ \tilde{J} = \int_0^1 \tilde{a}(u)\,\mathrm{d}u, \tag{7a}$$
$$\text{s.t. } \tilde{a}' = 2b,\ \tilde{b}' = \tilde{c},\ 0 \le \tilde{a} \le v^2,\ \boldsymbol{n}\tilde{a} + \boldsymbol{m}\tilde{b} \le \boldsymbol{g}, \tag{7b}$$
$$\boldsymbol{r}\tilde{a} + \boldsymbol{s}\tilde{b} + \boldsymbol{t}\tilde{c} \le \boldsymbol{f}l(\tilde{a}; \bar{a}), \tag{7c}$$
$$\tilde{a}(0) = \dot{u}_0^2,\ \tilde{b}(0) = \ddot{u}_0,\ \tilde{a}(1) = \dot{u}_\mathrm{f}^2,\ \tilde{b}(1) = \ddot{u}_\mathrm{f}, \tag{7d}$$

where $l(\bullet; \bar{a})$ is the linearization of $\bullet^{-1/2}$ at $\bar{a}$. Since the function $a \mapsto a^{-1/2}$ is concave, it holds that

$$l(a; \bar{a}) \le a^{-1/2},\ \forall a > 0, \bar{a} > 0. \tag{8}$$

Therefore, the feasible region of problem (7) is a subset of that in problem (5). In other words, $(\tilde{a}, \tilde{b}, \tilde{c})$ is feasible in the original problem (5) since

$$\boldsymbol{r}\tilde{a} + \boldsymbol{s}\tilde{b} + \boldsymbol{t}\tilde{c} \le \boldsymbol{f}l(\tilde{a}; \bar{a}) \le \boldsymbol{f}\tilde{a}^{-1/2}. \tag{9}$$

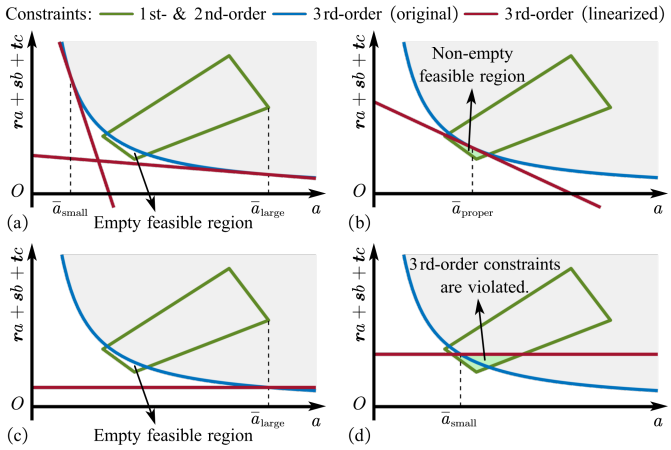

Fig. 1. Feasibility of the linearized problem. (a-b) The proposed TLP method. (c-d) Existing methods [1], [2], [13], [14].

Evidently, problem (7) is a linear programming (LP); hence, the feasible region of problem (7) is convex.

Consider a case where the 2nd-order solution $\bar{a}$ is significantly larger than the 3rd-order solution $\tilde{a}$, i.e., $\bar{a} \gg \tilde{a}$. Then, the scaling in (7c) would be over-conservative, hindering the time-optimality. In this case, a solution (a^*, b^*, c^*) is further optimized by the following LP:

$$\max \ J^* = \int_0^1 a^*(u)\,\mathrm{d}u, \tag{10a}$$

$$\text{s.t. } a^{*\prime} = 2b^*,\ b^{*\prime} = c^*, \tag{10b}$$

$$0 \le a^* \le v^2,\ \boldsymbol{n}a^* + \boldsymbol{m}b^* \le \boldsymbol{g}, \tag{10c}$$

$$\boldsymbol{r}a^* + \boldsymbol{s}b^* + \boldsymbol{t}c^* \le \boldsymbol{f}l\,(a^*; \tilde{a}), \tag{10d}$$

$$a^*(0) = \dot{u}_0^2,\ b^*(0) = \ddot{u}_0, \tag{10e}$$

$$a^*(1) = \dot{u}_\mathrm{f}^2,\ b^*(1) = \ddot{u}_\mathrm{f}. \tag{10f}$$

The difference between problems (7) and (10) is the linearization point of the 3rd-order constraint (5e). Intuitively, the feasible region of problem (10) appears to be larger than that of problem (7). In fact, since $\tilde{a}$ is feasible in problem (10), it holds that $J^* \ge \tilde{J}$. In other words, problem (10) can further optimize the trajectory based on problem (7).

Similar to [1], continuous problems (6), (7), and (10) can be directly discretized into LPs.

B. Incremental Linearization Method

For the case where the boundary conditions are nearly stationary, i.e., $(\dot{u}_0, \ddot{u}_0, \dot{u}_\mathrm{f}, \ddot{u}_\mathrm{f}) \approx \boldsymbol{0}$, the linearized problems (7) and (10) are feasible. However, for the case where the boundary conditions are non-stationary, especially when $\bar{a}$ is large, the linearized problems (7) and (10) might be infeasible, as shown in Fig. 1(a). Similarly, a method that substitutes the large $\bar{a}$ by a small one in (7c) also lacks feasibility guarantees. If problem (5) is feasible, then a proper $\bar{a}_\mathrm{proper}$ always exists, as shown in Fig. 1(b). For example, let $\bar{a}_\mathrm{proper}$ itself be a feasible solution, then the linearized problem (7) contains at least one feasible point, i.e., $\left(\bar{a}_\mathrm{proper}, \frac{1}{2}\bar{a}'_\mathrm{proper}, \frac{1}{2}\bar{a}''_\mathrm{proper}\right)$.

Algorithm 1: Feedrate Scheduling with TLP Method

Input: The parametric path $\mathcal{P}$, the boundary conditions $\boldsymbol{u} = (\dot{u}_0^2, \ddot{u}_0, \dot{u}_\mathrm{f}^2, \ddot{u}_\mathrm{f})$, the process constraints, and the number of iterations M.

Output: A trajectory $a = a^*(u)$.

Construct the 3rd-order non-convex problem (5);
Solve $\bar{a}$ by problem (6) with $\boldsymbol{u}$;
if problem (7) with $\boldsymbol{u}$ is feasible **then**
  Solve $\tilde{a}$ by problem (7) with $\boldsymbol{u}$;
  Solve a^* by problem (10) with $\boldsymbol{u}$;
else
  Solve $\bar{a}$ by problem (6) with $\boldsymbol{0}$;
  Solve $\tilde{a}$ by problem (7) with $\boldsymbol{0}$;
  Solve a_0^* by problem (10) with $\boldsymbol{0}$;
  for $p \leftarrow 1, 2, \ldots, M$ **do**
    if problem (10) with $\frac{p}{M}\boldsymbol{u}$ is feasible **then**
      Solve a_p^* by problem (11) with $\frac{p}{M}\boldsymbol{u}$;
    else
      return an infeasibility report;
    end if
  end for
  Set $a^* \leftarrow a_M^*$;
end if

However, it is challenging to find a proper $\bar{a}_\mathrm{proper}$ in practice. To address this issue, an incremental linearization method (ILM) is proposed to iteratively find a feasible initial solution. First, consider a problem with stationary boundary conditions, i.e., set $(\dot{u}_0, \ddot{u}_0, \dot{u}_\mathrm{f}, \ddot{u}_\mathrm{f}) = \boldsymbol{0}$ in problems (6), (7), and (10). Denote the solution for problem (10) as a_0^*. Then, given the number of iterations M, the p-th step is provided as follows:

$$\min \ J_p^* = \int_0^1 a_p^*(u)\,\mathrm{d}u, \tag{11a}$$

$$\text{s.t. } a_p^{*\prime} = 2b_p^*,\ b_p^{*\prime} = c_p^*, \tag{11b}$$

$$0 \le a_p^* \le v^2,\ \boldsymbol{n}a_p^* + \boldsymbol{m}b_p^* \le \boldsymbol{g}, \tag{11c}$$

$$\boldsymbol{r}a_p^* + \boldsymbol{s}b_p^* + \boldsymbol{t}c_p^* \le \boldsymbol{f}l\left(a_p^*; a_{p-1}\right), \tag{11d}$$

$$a_p^*(0) = \frac{p}{M}\dot{u}_0^2,\ b_p^*(0) = \frac{p}{M}\ddot{u}_0, \tag{11e}$$

$$a_p^*(1) = \frac{p}{M}\dot{u}_\mathrm{f}^2,\ b_p^*(1) = \frac{p}{M}\ddot{u}_\mathrm{f}. \tag{11f}$$

In other words, the 3rd-order constraints (5e) are linearized at the previous solution a_{p-1}, which is nearly feasible in problem (11) at the p-th step. At the final step M, the solution $\left(a_M^*, b_M^*\right)$ satisfies the original boundary conditions (5f).

The proposed method in Sections III-A and III-B is summarized in Algorithm 1.

C. Comparison with Existing Methods

In some existing works on optimization-based 3rd-order trajectory planning [1], [2], [13], [14], the initial solution $\hat{a}$

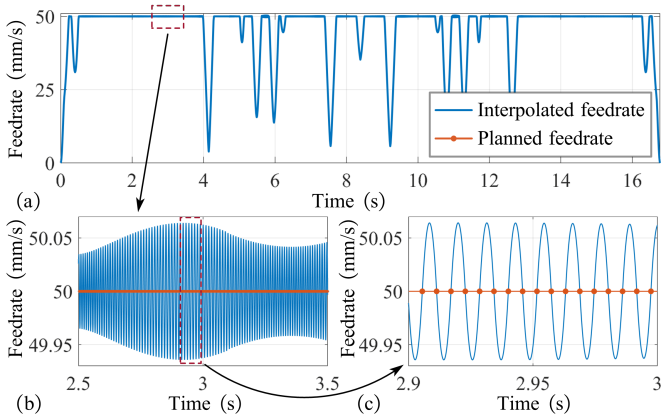

Fig. 2. The phenomenon of feedrate oscillations in optimization-based trajectory planning. The maximal feedrate is 50 mm/s.

is obtained as follows:

$$\max_{(\hat{a},\hat{b},\hat{c})} \quad \hat{J} = \int_0^1 \hat{a}\,(u)\,\mathrm{d}u, \tag{12a}$$

$$\text{s.t.} \quad \hat{a}' = 2b,\ \hat{b}' = \hat{c},\ 0 \le \hat{a} \le v^2,\ \boldsymbol{n}\hat{a} + \boldsymbol{m}\hat{b} \le \boldsymbol{g}, \tag{12b}$$

$$\boldsymbol{r}\hat{a} + \boldsymbol{s}\hat{b} + \boldsymbol{t}\hat{c} \le \boldsymbol{f}\bar{a}^{-1/2}, \tag{12c}$$

$$\hat{a}\,(0) = \dot{u}_0^2,\ \hat{b}\,(0) = \ddot{u}_0,\ \hat{a}\,(1) = \dot{u}_\mathrm{f}^2,\ \hat{b}\,(1) = \ddot{u}_\mathrm{f}. \tag{12d}$$

In these methods, it is noted that $\forall a = a\,(u)$ is feasible in problem (5), $\forall u \in [0,1]$, $a\,(u) \le \bar{a}\,(u)$ holds point-by-point where $\bar{a}$ is the 2nd-order optimal solution in problem (6). The above observation can be proved by considering a feasible solution $\max\{\bar{a}, a\}$ in problem (6). Therefore, it is acknowledged that

$$\bar{a}^{-1/2} \le a^{-1/2},\ \forall a \text{ feasible in problem (5)}, \tag{13}$$

resulting in the conservative scaling of the 3rd-order constraint (12c). Note that the feasible region of problem (12) is convex and is a subset of that in problem (5). However, the scaling (13) is more conservative than (8) since $l\,(a;\bar{a}) > \bar{a}^{-1/2}$.

Denote the feasible region of problems (5), (6), (7), and (12) as $\mathcal{D}$, $\overline{\mathcal{D}}$, $\widetilde{\mathcal{D}}$, and $\widehat{\mathcal{D}}$, respectively. Then, it holds that

$$\widehat{\mathcal{D}} \subset \widetilde{\mathcal{D}} \subset \mathcal{D} \subset \overline{\mathcal{D}}. \tag{14}$$

Therefore, $\hat{J} \le \tilde{J} \le J^*$ holds. In other words, the proposed TLP method is capable of obtaining a feasible solution with better time-optimality than existing methods based on conservative scaling (13).

Considering the case under non-stationary boundary conditions, the feasibility is further compared in Fig. 1. If a large $\bar{a}$ is applied in (12c), as shown in Fig. 1(c), then the feasible region might be empty. The above case is common when $\dot{u}_0, \dot{u}_\mathrm{f} \gg 0$. If a small $\bar{a}$ is applied in (12c), as shown in Fig. 1(d), then the feasible region can be non-empty. However, the original 3rd-order constraints (5e) might be violated even for a feasible solution for problem (12). In contrast to the proposed ILM, the above existing method lacks the ability to generate a feasible solution under non-stationary boundary conditions.

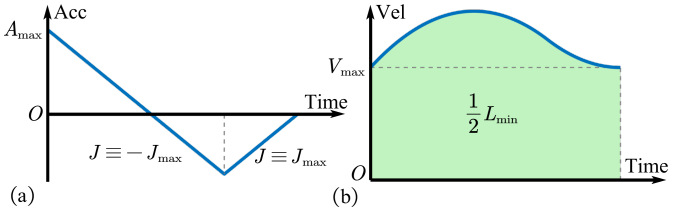

Fig. 3. The computation of L_min in (15b). (a) The acceleration profile. (b) The velocity profile.

D. Elimination of Feedrate Oscillations

In optimization-based trajectory planning methods, the discretization of optimization problems can lead to constraint violations [10]. For example, a feedrate profile is shown in Fig. 2(a), where the feedrate constraint is set as 50 mm/s. Although the constraint is generally satisfied, approximately 1% oscillations occur near the upper bound, as shown in Fig. 2(b). In fact, the planned feedrate profile satisfies the constraint at grid points, but the feedrate oscillates in discretization intervals, as shown in Fig. 2(c). Note that it is the interpolated feedrate that has a direct impact on the machining quality.

To address the above issue, an oscillation elimination method is proposed. Denote the discretization grid as $\{u_i\}_{i=0}^N$ and the planned trajectory as $a_i^* = a^*\,(u_i)$. Consider a tolerance ε, the maximal feedrate V_max, the maximal acceleration A_max, the maximal jerk J_max. $[u_{i_1}, u_{i_2}]$ is determined as a continuous segment maintaining the maximal feedrate if

$$\|\boldsymbol{q}'\,(u_i)\|\,a_i^* \ge V_\mathrm{max} - \varepsilon,\ \forall i_1 \le i \le i_2, \tag{15a}$$

$$\sum_{i=i_1+1}^{i_2} \|\boldsymbol{q}\,(u_i) - \boldsymbol{q}\,(u_{i-1})\| \ge L_\mathrm{min}. \tag{15b}$$

L_min is the minimal length for adjusting the feedrate.

For a determined continuous segment $[u_{i_1}, u_{i_2}]$, the feedrate profile is substituted by one with a constant feedrate V_max in $[u_{i_1}, u_{i_2}]$. At the initial point u_{i_1}, denote the feedrate as $F_{i_1} = \|\boldsymbol{q}'\,(u_{i_1})\|\sqrt{a_{i_1}^*}$ and the tangential acceleration $A_{i_1} = \|\boldsymbol{q}'\,(u_{i_1})\|'\,a_{i_1}^* + \|\boldsymbol{q}'\,(u_{i_1})\|\,b_{i_1}^*$. The kinematic state is driven from (F_{i_1}, A_{i_1}) to $(F_\mathrm{max}, 0)$ as fast as possible within a 2-phase jerk profile, i.e., $J \equiv -J_\mathrm{max}$ for t_1 time and then $J \equiv J_\mathrm{max}$ for t_2. Then, (t_1, t_2) can be solved as follows:

$$0 = A_{i_1} + J_\mathrm{max}\,(t_1 - t_2)\,, \tag{16a}$$

$$F_\mathrm{max} = F_{i_1} + A_{i_1}t_1 + J_\mathrm{max}\left(\frac{1}{2}\,(t_1+t_2)^2 - t_2^2\right). \tag{16b}$$

A similar process is applied to the terminal point u_{i_2}. In this way, feedrate oscillations are eliminated since the interpolated points are much denser than the optimization grid points.

As shown in Fig. 3, let $A_{i1} = A_\mathrm{max}$ and $F_{i1} = V_\mathrm{max}$, and L_min is computed as follows:

$$L_\mathrm{min} = \frac{\left(9\sqrt{2}+4\right)A_\mathrm{max}^2}{6J_\mathrm{max}^2} + \frac{\left(2\sqrt{2}+2\right)V_\mathrm{max}}{J_\mathrm{max}}. \tag{17}$$

TABLE I
QUANTITATIVE RESULTS IN STANDARD PARAMETERS.

Experiments	Metrics	LP [2]	SQP [1]	TLP (Ours)
Butterfly (2-axis)	T_f (s)	14.208	12.156	11.936
	T_c (s)	0.262	2.987	0.397
Mold (3-axis)	T_f (s)	1.645×10^4	1.330×10^4	1.105×10^4
	T_c (s)	8.187×10^2	1.850×10^4	1.575×10^3
Blade (5-axis)	T_f (s)	1.887×10^3	1.662×10^3	1.649×10^3
	T_c (s)	4.535×10^2	5.315×10^3	6.302×10^2
Franka (7-axis)	T_f (s)	8.114	7.737	7.707
	T_c (s)	0.835	7.464	1.560

IV. EXPERIMENTS

A. Setup

The proposed **TLP** method is compared with the following baselines. (a) **LP** [2]: A method based on linear programming, as described in Section III-C. (b) **SQP** [1]: A method which applies the result of LP as initial solutions and solves problem (5) by sequential quadratic programming. Due to the limitation of computational resources, only one iteration is applied in the SQP method. The methods are compared regarding motion time T_f, computational time T_c, and trajectory quality. In this paper, the motion time is computed by $t_f = \int_0^1 \frac{du}{\sqrt{a(u)}}$ where the planned $a(u)$ is piecewise quadratic.

B. Parametric Study

A 2-axis butterfly curve is used as the parametric path $\mathcal{P}$. Let maximal feedrate V_{max}=100 mm/s, axial acceleration A_{max}=800 mm/s^2, and axial jerk J_{max}=3000 mm/s^3. The feedrate distributions of the three methods are shown in Fig. 4(a). It can be observed that the proposed TLP method achieves a feedrate profile similar to SQP. The motion time T_f and the computational time T_c are provided in Tab. I. TLP saves 16.0% of motion time than LP, while reducing computational time by 86.7% compared to SQP within similar motion time.

To evaluate feedrate oscillations, the classification of feedrate in the proposed TLP method is shown in Fig. 4(b), and the profiles of the three methods are shown in Fig. 4(c)-(d). Compared to the two baselines, the proposed TLP exhibits little feedrate oscillations, incurring only single peaks at the two ends of the constant feedrate segments.

Fix V_{max}, J_{max} and vary A_{max}. The motion time T_f with different constraints is shown in Fig. 4(e). Theoretically speaking, T_f should decrease with increasing A_{max}. However, for LP, the scaling (12c) is too conservative, leading to a longer motion time. The above anomalous effect is slight in the proposed TLP, as the linearization (11d) is much tighter.

Consider non-stationary boundary conditions, where $\dot{u}_0 > 0$ and $\ddot{u}_0 = \dot{u}_f = \ddot{u}_f = 0$. It can be observed from Fig. 4(f) that the proposed TLP is capable of finding a feasible solution for $1 \le \dot{u}_0^2 \le 20$, which is attributed to the developed ILM. In contrast, both LP and SQP are infeasible when $\dot{u}_0^2 \ge 15$.

C. Case Study

Consider the 3-axis machine tool shown in Fig. 5(a1), where a single-point milling process of a Mercedes-Benz mold

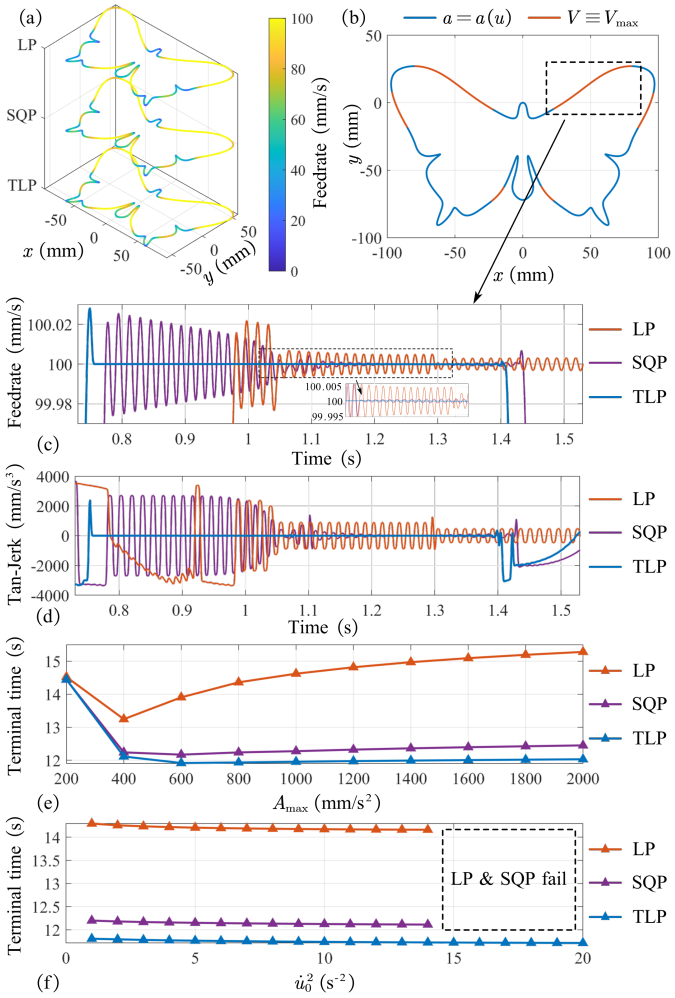

Fig. 4. Results of parametric study. (a) Feedrate distributions. (b) Classification of feedrate in TLP. (c-d) Feedrate and tangential jerk profiles. (e-f) Motion time with different constraints and boundary conditions.

is considered. Set the maximal feedrate as 3,000 mm/min, the axial acceleration as 500 mm/s^2, and the axial jerk as 10,000 mm/s^3. The feedrate distribution of the proposed TLP method is shown in Fig. 5(b). The planned acceleration and jerk profiles are shown in Fig. 5(c), where the probability density functions (PDF) show that the constraints are satisfied. The planned trajectory satisfies the constraints, which indicates that the proposed TLP is applicable to 3-axis machining. It can be observed from Tab. I that TLP saves 32.8% and 16.9% of motion time compared to LP and SQP, respectively, within limited computational time.

A blade is considered in a 5-axis machine tool shown in Fig. 5(a2). Set the maximal feedrate as 1,000 mm/min, the axial orientation acceleration as 5 rad/s^2, and the axial orientation jerk as 120 rad/s^3. The feedrate distribution and the planned trajectory of TLP is shown in Fig. 5(c) and (e), respectively. The proposed TLP can generate a feasible trajectory within kinematic constraints, which can be observed from PDFs. By Tab. I, TLP saves 12.6% of motion time compared to LP, while reducing computational time by 88.1% compared to SQP.

A 7-axis Franka robot arm is considered in Fig. 5(a3) with

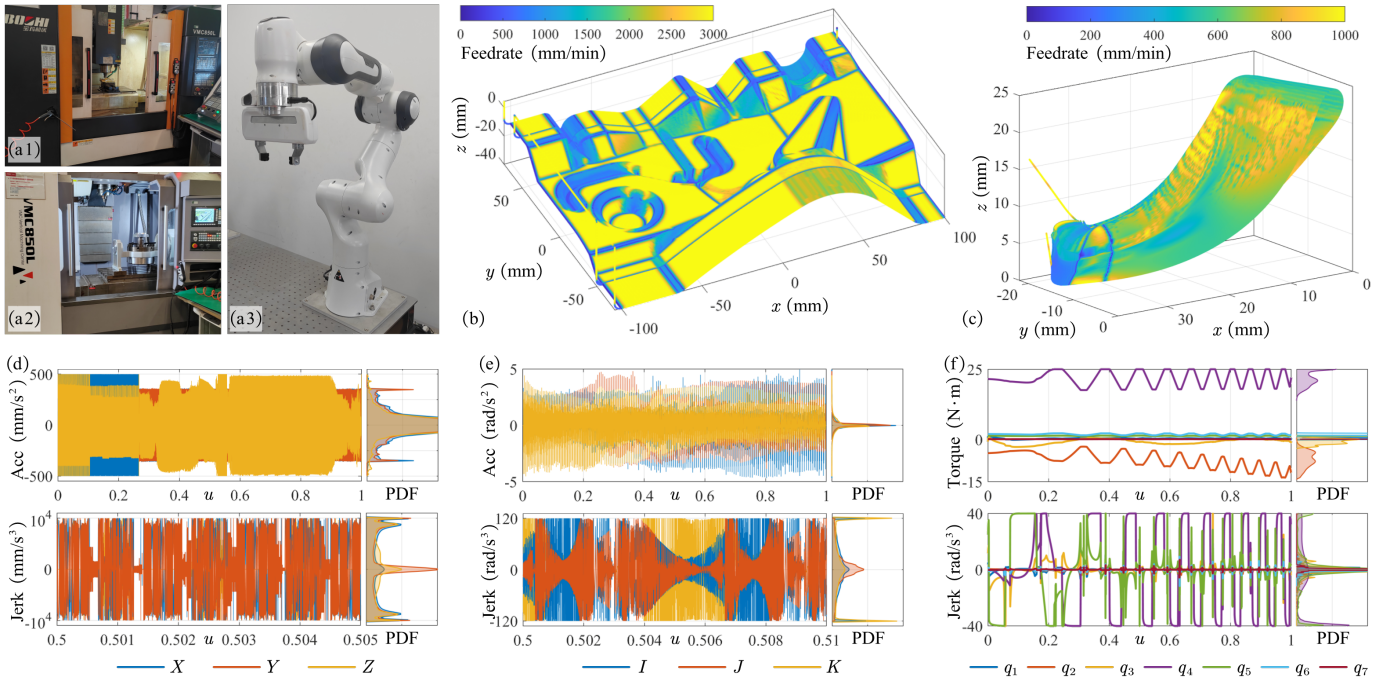

Fig. 5. Equipment and results of case study. (a) Equipment. (b)-(c) Feedrate distributions. (d)-(f) Planned trajectories and probability density functions (PDF).

a dynamic model [16]. A parametric path is followed with maximal axial torque 25 N·m and maximal axial jerk 40 rad/s^3. As shown in Fig. 5(f), kinematic and dynamic constraints are satisfied in the planned trajectory. Similar to the above cases, the proposed TLP method outperforms LP and SQP in terms of motion time and computational time, respectively.

V. Conclusion

This paper proposes a jerk-limited oscillation-free feedrate scheduling method based on triple linear programming (TLP) which is capable of finding a feasible trajectory under non-stationary boundary conditions. Compared to existing linear programming methods, the proposed TLP saves more than 10% of motion time while reducing computational time by more than 80% with better time-optimality than the baseline based on sequential quadratic programming. TLP also significantly outperforms baselines regarding feasibility and feedrate oscillations. The proposed method has the potential to be applied to various machining processes with different equipment like robotic manipulators and 5-axis machine tools.

References

[1] F. Debrouwere, W. Van Loock, et al., "Time-optimal path following for robots with convex–concave constraints using sequential convex programming," *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1485–1495, 2013.

[2] K. Erkorkmaz, Q.-G. C. Chen, et al., "Linear programming and windowing based feedrate optimization for spline toolpaths," *CIRP Annals*, vol. 66, no. 1, pp. 393–396, 2017.

[3] H. Zhao, L. Zhu, et al., "A real-time look-ahead interpolation methodology with curvature-continuous b-spline transition scheme for cnc machining of short line segments," *International Journal of Machine Tools and Manufacture*, vol. 65, pp. 88–98, 2013.

[4] S. Lin, C. Hu, et al., "Real-time local greedy search for multiaxis globally time-optimal trajectory," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 54, no. 2, pp. 960–971, 2024.

[5] Y. Wang, C. Hu, et al., "Time-optimal control for high-order chain-of-integrators systems with full state constraints and arbitrary terminal states," *IEEE Transactions on Automatic Control*, vol. 70, no. 3, pp. 1499–1514, 2025.

[6] D. Kaserer, H. Gattringer, et al., "Nearly optimal path following with jerk and torque rate limits using dynamic programming," *IEEE Transactions on Robotics*, vol. 35, no. 2, pp. 521–528, 2018.

[7] Y. Wang, C. Hu, et al., "Chattering phenomena in time-optimal control for high-order chain-of-integrator systems with full state constraints," *IEEE Transactions on Automatic Control*, 2025.

[8] Y. Wang, C. Hu, et al., "On the consistency of path smoothing and trajectory planning in cnc machining: A surface-centric evaluation," *Robotics and Computer-Integrated Manufacturing*, vol. 92, p. 102873, 2025.

[9] Q.-C. Pham, "A general, fast, and robust implementation of the time-optimal path parameterization algorithm," *IEEE Transactions on Robotics*, vol. 30, no. 6, pp. 1533–1540, 2014.

[10] H. Pham and Q.-C. Pham, "A new approach to time-optimal path parameterization based on reachability analysis," *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 645–659, 2018.

[11] D. Malyuta, T. P. Reynolds, et al., "Convex optimization for trajectory generation: A tutorial on generating dynamically feasible trajectories reliably and efficiently," *IEEE Control Systems Magazine*, vol. 42, no. 5, pp. 40–113, 2022.

[12] S. He, W. Zhao, et al., "A hierarchical long short term safety framework for efficient robot manipulation under uncertainty," *Robotics and Computer-Integrated Manufacturing*, vol. 82, p. 102522, 2023.

[13] C. Ji, Z. Zhang, et al., "A convex optimization method to time-optimal trajectory planning with jerk constraint for industrial robotic manipulators," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 4, pp. 7629–7646, 2023.

[14] W. Fan, X.-S. Gao, et al., "Time-optimal interpolation for five-axis cnc machining along parametric tool path based on linear programming," *The International Journal of Advanced Manufacturing Technology*, vol. 69, pp. 1373–1388, 2013.

[15] S. He, C. Hu, et al., "Time optimal control of triple integrator with input saturation and full state constraints," *Automatica*, vol. 122, p. 109240, 2020.

[16] C. Gaz, M. Cognetti, et al., "Dynamic identification of the franka emika panda robot with retrieval of feasible parameters using penalty-based optimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4147–4154, 2019.