

Snap-Bounded and Time-Optimal Feedrate Scheduling for Robotic Milling of Complex Surface Parts With Analytical Solution

Mansen Chen , Yuwen Sun , Jinting Xu , and Jun Liu , *Senior Member, IEEE*

Abstract—Feedrate scheduling is crucial for improving productivity and accuracy in robotic milling applications. However, due to the nonlinear relationship between the joint space and task space, how to plan a time-optimal feedrate profile with quick analytical solution while ensuring kinematic control up to the snap level remains quite a challenge. To solve these concerns, a snap-bounded and time-optimal feedrate scheduling model is first presented in this article. For accelerating the solving process of the nonlinear model, a synchronous linearization approach is also introduced to help relax the highly nonlinear constraints in both joint space and task space into linear ones. Thereby, the originally complex feedrate scheduling issue is converted to a finite-state convex optimization problem, and an analytical solution to the feedrate profile could be computed efficiently using a straightforward linear programming algorithm. Finally, comparative simulation and experiment are carried out to verify the effectiveness of the proposed method.

Index Terms—Convex optimization, feedrate scheduling, linear programming, nonlinear model, robotic milling.

I. INTRODUCTION

WITH the rapid development of aerospace and power technologies, more structural parts are designed with large-scale and complex surface [1], such as rocket cabins and turbine blades. Industrial robots owing to their higher flexibility, better adaptability, and lower capital cost, etc., are becoming

Received 14 August 2024; revised 12 October 2024; accepted 30 December 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 52205517, Grant 91948203, and Grant 52375484, in part by the Research Grant Council of Hong Kong under Grant 11217922, Grant 11212321, and Grant 21212720, and in part by the Innovation and Technology Commission of Hong Kong under Grant PRP/078/22FX. Paper no. TII-24-4121. (Corresponding authors: Yuwen Sun; Jun Liu.)

Mansen Chen is with the Department of Mechanical Engineering, City University of Hong Kong, Hong Kong, and also with Shenzhen Research Institute, City University of Hong Kong, Shenzhen 518057, China (e-mail: mansen.chen@cityu.edu.hk).

Yuwen Sun and Jinting Xu are with the School of Mechanical Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: ywsun@dlut.edu.cn; xujt@dlut.edu.cn).

Jun Liu was with the Department of Mechanical Engineering, City University of Hong Kong, Hong Kong, and also with Shenzhen Research Institute, City University of Hong Kong, Shenzhen 518057, China. He is now with the Department of Data and Systems Engineering, The University of Hong Kong, Hong Kong (e-mail: dr.jun.liu@hku.hk).

Digital Object Identifier 10.1109/TII.2025.3528566

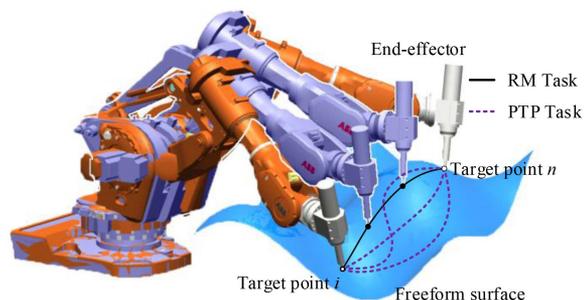


Fig. 1. Trajectory planning of IRs for different engineering tasks.

preferable alternatives to replace conventional computer numerical control (CNC) machine tools for manufacturing this type of parts. In regular point-to-point applications of robots (e.g., drilling and spot welding), the tool installed on the end-effector is only required to accurately pass through the target points, while the movement between adjacent target points is allowed to be free, as shown in Fig. 1. In contrast, robotic milling operation has stricter requirement on surface contour accuracy, in which the end-effector need trace the whole tool path continuously and accurately, not only a set of discrete target points. Nevertheless, due to the articulated serial structure, industrial robots (IRs) inevitably suffer from the inherent low stiffness and poor absolute accuracy. Improper parameters setting and load disturbance probably cause unwanted vibrations on the mechanical structure and degrade the surface quality. To handle these issues, various studies such as robot-kinematics calibration [2], tool pose optimization [3], dynamic error compensation [4], and redundancy optimization [5], [6], [7] were conducted, aimed at strengthening the capability of IRs in milling operation. Therein, feedrate scheduling, which is able to deliver a fast movement at the end-effector while maintaining a smooth joint kinematic property, plays an important role in satisfying the growing quality demands of products. Thus recently, it gains considerable attention and becomes a focus of interest.

A. Related Works

Till now, existing works can be broadly categorized into two types: 1) methods of feedrate scheduling based on integral operation; and 2) those based on optimization principles. A distinct feature of the first category of methods is to predefine a series of acceleration/deceleration (AD) profile templates for

maneuvering the movement of cutting tool before the real run, where the AD profile templates are often expressed in the form of trapezoidal [8] and S-curved [9], [10], [11], [12], [13] functions. As a matter of practice, the transient jump of acceleration in trapezoidal AD control method [8] possibly excites the resonance modes of the mechanical system, which not only gives a rise to larger tracking errors, but also leads to stronger impact on cutting tool and servos. In contrast, jerk-bounded feedrate scheduling with S-curved AD profile due to its higher level of kinematic continuity is more preferred [9]. In order to mitigate the jerk impulse appearing at the initial and final instants of motion, the fourth order feedrate scheduling method [10] and sinusoidal AD control methods [11], [12], [13] are also intensively explored. Theoretically, the smoothness of trajectory can be further enhanced using higher-order polynomial models, but the increased number of AD statuses results in a complicated selection of the switching times. In addition, the optimality of these methods is greatly influenced by the employed AD templates.

To fulfill several user-specified demands, such as minimum machining time [14], [15], [16], minimum joint torque [17], and minimum jerks [18], [19], etc., trajectory planning based on optimality principles are versatile and potent, through which the concerned feedrate scheduling problem can be transformed into a constrained nonlinear optimization problem. At present, various methods including quadratic sequence programming (SQP) [20], [21], particle swarm optimization (PSO) [22], [23], heuristic search (HS) [24], and reproducing kernel Hilbert space [25], [26] are applicable to tackle nonlinear problems. However, a critical barrier in addressing these issues is that the process tends to be costly and time consuming [27]. To handle this limitation, increasing attention has been paid to the linear programming (LP) of feedrate [28], [29], [30], [31] owing to its guaranteed global convergence and faster computational efficiency. For example, Fan et al. [29] and Zhao et al. [30] proposed classical numerical optimization methods for time-optimal feedrate generation, taking into account kinematic jerk limits. Nagy and Vajk [31] proposed a LP-based path tracking algorithm and analyzed its potential for real-time application in embedded robotic systems. In addition, machine learning or deep learning techniques [32], [33] have also found applications in solving these complex problems. For example, Chai et al. [34] developed a recurrent deep neural network-based method to plan the optimal motion trajectory with guaranteed real-time performance. Tan et al. [35] introduced a metaheuristic-based recurrent neural network model to directly handle the nonlinear manipulability optimization problem without sacrificing the feasible range.

B. Existing Problems

It is worth mentioning that, due to the nonlinear relationship between the joint space and task space, most existing LP-based feedrate optimization methods have to suffer from numerical approximation, in which the joint kinematical constraints are evaluated using the finite difference method. This inevitably results in a loss of control accuracy, particularly when considering high order constraints. On the other hand, for realizing the linearization of jerk constraints, the square of feedrate is often taken as the optimization variable. During the process,

an additional curve-fitting operation on the optimized discrete feedrate values is also necessary for the final executable profile generation. These factors make it challenging for existing methods to achieve snap control, despite the benefits of higher kinematic continuity for high quality machining.

C. Contributions and Organization of This Article

To solve these problems, this article proposes a novel smooth feedrate scheduling method for robotic milling of complex surface parts. Different from existing methods, the proposed method is not only capable of achieving a direct control of robotic kinematics up to snap level in both task space and joint space, but also can directly obtain an analytical solution to the time-optimized feedrate profile described by spline curves without any numerical approximation and loss of accuracy. To be best of the authors' knowledge, there has been no reported research or method that realizes the linearization of joint snap constraint, and solve the snap-bounded feedrate scheduling problem in linear programming manner. The rest of this article is organized as follows. Section II presents the basic mathematical formalisms. Section III establishes the snap-bounded feedrate optimization model. In Section IV, the linearization expressions of the involved nonlinear kinematic constraints are analytically derived, followed by a detailed implementation of feedrate optimization. Section V presents the results of simulation and experiment. Finally, Section VI concludes this article.

II. PRELIMINARIES

To perform a milling task, the reference cutter location (CL) data in the workpiece coordinate system generally consists of a position vector $\mathbf{p} = [p_x, p_y, p_z]^T$ and an orientation unit vector $\mathbf{o} = [i, j, k]^T$, where \mathbf{p} represents the coordinate of the tool center point, and \mathbf{o} represents the posture of tool axis. Without losing generality, the robotic tool path $[\mathbf{p}(u), \mathbf{o}(u)]$ in task space is defined by the normalized arc-length parameter $u \in [0, 1]$. Let $\mathbf{q}(u) = [\theta_1(u), \theta_2(u), \theta_3(u), \theta_4(u), \theta_5(u), \theta_6(u)]^T$ be the robot configuration in joint space, i.e., the angles of the actuated joints. The relation between $[\mathbf{p}(u), \mathbf{o}(u)]$, and $\mathbf{q}(u)$ can be generally expressed as

$$\mathbf{q}(u) = \text{IK}(\mathbf{p}(u), \mathbf{o}(u)) \quad (1)$$

where $\text{IK}(\cdot)$ denotes the inverse kinematic transformation of the robots. Corresponding to the robotic tool path $[\mathbf{p}(u), \mathbf{o}(u)]$, a B-spline curve is employed in this instance to define the feedrate profile as a function of the same path parameter u

$$f(u) = \sum_{i=0}^{e-1} d_i N_{i,k}(u), u \in [0, 1] \quad (2)$$

where d_i is the control point and e is the number of control points. $N_{i,k}(u)$ is the B-spline basis function defined over the knot vector $U = [u_0, \dots, u_{e+k}]$. k is the degree of the B-spline curve. According to the differential property of the feedrate curve, one can derive

$$\frac{d^\sigma f(u)}{du^\sigma} = \sum_{i=0}^{e-1} d_i N_{i,k}^\sigma(u), u \in [0, 1] (\sigma \leq k) \quad (3)$$

where σ indicates the order of derivative.

III. FEEDRATE OPTIMIZATION MODEL

In robotic milling applications, milling quality is influenced by many factors. These include, but are not limited to, the geometric constraint, process constraint and drive constraint. In order to satisfy the high-performance requirements of manufactured parts, it is preferable that feedrate scheduling is capable of fulfilling these machining-related constraints simultaneously. Meanwhile, time optimality is another crucial factor that has to be considered during the process, due to its direct impact on productivity. In order to minimize machining time, the feedrate should be maximized as much as possible. Consequently, the multiconstrained time-optimal feedrate scheduling problem can be formulated as

$$\begin{aligned} & \max \int_0^1 f(u) du \\ \text{s.t.} & \begin{cases} \varepsilon(u) < \varepsilon_{\max} \\ \begin{bmatrix} |f(u)| \\ |a(u)| \\ |j(u)| \\ |s(u)| \end{bmatrix} < \begin{bmatrix} f_{\max} \\ a_{\max} \\ j_{\max} \\ s_{\max} \end{bmatrix}; \begin{bmatrix} |\dot{q}_\tau(u)| \\ |\ddot{q}_\tau(u)| \\ |\dddot{q}_\tau(u)| \end{bmatrix} < \begin{bmatrix} V_{\tau,\max} \\ A_{\tau,\max} \\ J_{\tau,\max} \\ S_{\tau,\max} \end{bmatrix} \\ \tau = 1, \dots, 6; u \in [0, 1] \end{cases} \end{aligned} \quad (4)$$

where $\varepsilon(u)$ is the chord error which is viewed as a major source of machining inaccuracy and induced by approximating the desired path curve with a set of linear segments. $a(u)$, $j(u)$, and $s(u)$ represent the tangential acceleration, tangential jerk, and tangential snap along the tool path. $\dot{q}_\tau(u)$, $\ddot{q}_\tau(u)$, $\dddot{q}_\tau(u)$, and $\ddot{\ddot{q}}_\tau(u)$ are the velocity, acceleration, jerk, and snap of each individual joint. f_{\max} , ε_{\max} , a_{\max} , j_{\max} , s_{\max} , $V_{\tau,\max}$, $A_{\tau,\max}$, $J_{\tau,\max}$, and $S_{\tau,\max}$ are the maximum allowable bounds on feedrate, chord error, tangential acceleration, tangential jerk, tangential snap, joint velocity, joint acceleration, joint jerk, and joint snap, respectively. τ is the index of joints. Details about the constraints in (4) are described as follows.

A. Geometric Constraint

Feedrate scheduling with confined chord error is aimed at ensuring the accuracy of manufactured parts. For a robotic tool path $\mathbf{p}(u)$, the chord error is computed by

$$\varepsilon(u) = \rho(u) - \sqrt{\rho(u)^2 - (f(u)T_s/2)^2} \quad (5)$$

where T_s is the interpolator period and $\rho(u)$ is the curvature radius of $\mathbf{p}(u)$ at the parameter position u . Given a tolerance ε_{\max} , the feasible feedrate respecting the constraint of chord error can be derived as

$$f_{geo}(u) < 2\sqrt{\rho(u)^2 - (\rho(u) - \varepsilon_{\max})^2}/T_s. \quad (6)$$

B. Process Constraints in Task Space

When machining parts with sharp geometric features, frequent accelerations, and jerks can lead to fluctuations in cutting load. Therefore, it is essential to limit tangential acceleration, jerk and snap along the tool path to ensure smooth operation. This helps prevent tool wear and damage, thereby ensuring higher quality and precision in the machined parts. These kinematic properties, defined in the robotic task space, can be derived

using the differential chain rule as follows:

$$\begin{bmatrix} f(u) \\ a(u) \\ j(u) \\ s(u) \end{bmatrix} = \begin{bmatrix} f(u) \\ df(u)/dt \\ da(u)/dt \\ dj(u)/dt \end{bmatrix} = \mathbf{A}\boldsymbol{\gamma} \quad (7)$$

where \mathbf{A} is a 4×4 diagonal matrix defined by

$$\mathbf{A} = \text{diag} \begin{bmatrix} f(u) \\ f_u(u)f(u) \\ f_{uu}(u)f(u)^2 + f_u(u)^2f(u) \\ f_{uuu}(u)f(u)^3 + 4f_u(u)f_{uu}(u)f(u)^2 + f_u(u)^3f(u) \end{bmatrix} \quad (8)$$

and

$$\boldsymbol{\gamma} = \left[1, \frac{1}{\lambda}, \frac{1}{\lambda^2}, \frac{1}{\lambda^3} \right]^T. \quad (9)$$

In (8) and (9), $\text{diag}[\dots; a_i; \dots]$ denotes the diagonalization of matrix with its element a_i at the i -th diagonal position. $f_u(u)$, $f_{uu}(u)$, and $f_{uuu}(u)$ are the first, second, and third order derivatives of feedrate $f(u)$ with respect to the path parameter u , respectively. λ is the total length of the path curve $\mathbf{p}(u)$, and can be computed via $\lambda = \int_0^1 \|\mathbf{p}_u(u)\| du$.

C. Drive Constraints in Joint Space

In addition to process constraints, excessive joint acceleration, jerk, and snap can also lead to unwanted structural vibrations and excessive tracking errors, which is adverse to milling stability, even causes damage to robot components. Hence, in order to mitigate these risks, it is also necessary to limit these joint kinematic properties based on the capabilities of the actuators. For a programmed joint path $\mathbf{q}(u)$, the joint velocity, acceleration, jerk and snap can be analytically expressed as

$$\begin{bmatrix} \dot{q}_\tau(u) \\ \ddot{q}_\tau(u) \\ \ddot{\ddot{q}}_\tau(u) \\ \ddot{\ddot{\ddot{q}}}_\tau(u) \end{bmatrix} = \begin{bmatrix} dq_\tau(u)/dt \\ d\dot{q}_\tau(u)/dt \\ d\ddot{q}_\tau(u)/dt \\ d\ddot{\ddot{q}}_\tau(u)/dt \end{bmatrix} = \mathbf{B}\boldsymbol{\eta} \quad (10)$$

with

$$\mathbf{B} = \begin{bmatrix} B_{1,1} & 0 & 0 & 0 \\ B_{2,1} & B_{2,2} & 0 & 0 \\ B_{3,1} & B_{3,3} & B_{3,3} & 0 \\ B_{4,1} & B_{4,2} & B_{4,3} & B_{4,4} \end{bmatrix} \quad (11)$$

$$\boldsymbol{\eta} = \left[\frac{q_u^\tau(u)}{\lambda}, \frac{q_{uu}^\tau(u)}{\lambda^2}, \frac{q_{uuu}^\tau(u)}{\lambda^3}, \frac{q_{uuuu}^\tau(u)}{\lambda^4} \right]^T \quad (12)$$

where $q_u^\tau(u)$, $q_{uu}^\tau(u)$, $q_{uuu}^\tau(u)$, and $q_{uuuu}^\tau(u)$ are the geometric derivatives of the τ th joint path with respect to the normalized arc length parameter u . \mathbf{B} is a 4×4 matrix with its elements expressed by

$$B_{1,1} = f(u); B_{2,1} = f_u(u)f(u)/\lambda; B_{2,2} = f(u)^2;$$

$$B_{3,1} = (f_{uu}(u)f(u)^2 + f_u(u)^2f(u))/\lambda^2;$$

$$B_{3,2} = 3f(u)^2f_u(u)/\lambda; B_{3,3} = f(u)^3;$$

$$B_{4,1} = (f_{uuu}(u)f(u)^3 + 4f_u(u)f_{uu}(u)f(u)^2 + f_u(u)^3f(u))/\lambda^3;$$

$$\begin{aligned} B_{4,2} &= (4f_{uu}(u)f(u)^3 + 7f_u(u)^2f(u)^2)/\lambda^2; \\ B_{4,3} &= 6f(u)^3f_u(u)/\lambda; B_{4,4} = f(u)^4. \end{aligned} \quad (13)$$

IV. SOLUTION OF FEEDRATE OPTIMIZATION

From (7) and (10), it could be noticed that the kinematical parameters in robotic task space and joint space are highly coupled and nonlinear. Direction solution of (4) is difficult, even impossible. Although the nonlinear optimization methods (e.g., SQP, PSO, HS) might be able to address this issue, the solution is computationally expensive, especially when snap constraints are incorporated. Based on (2) and (3), it can be deduced that for any given parameter position, the feedrate and its corresponding derivatives maintain a linear relationship with the control point variables. The nonlinear constraints in (4) can be articulated as a linear combination of the control point variables, transforming the nonlinear optimization in (4) into a linear problem. In this section, theoretical analysis will show that the minimum-time feedrate scheduling problem in (4) can be converted into a linear finite-state convex optimization problem. Thereby, the complicated feedrate scheduling problem can be effectively handled using a highly robust linear programming algorithm.

A. Proposed Theorem

To facilitate the presentations, this subsection introduces the following theorem, which will be utilized for the linearization of constraints.

Theorem 1: Let $\{\chi_i(u) | \chi_i(u) \in \mathbb{R}, i = 1, \dots, n\}$ be a set of scalar functions with respect to parameter u , and E is a positive constant. If the following inequalities are satisfied for any addition/subtraction operations:

$$|\chi_1(u) \pm \chi_2(u) \pm \dots \pm \chi_n(u)| < E \quad (14)$$

then

$$\sum_i^n |\chi_i(u)| < E. \quad (15)$$

Proof: In order to prove the above theorem, one can define

$$\begin{cases} z_1(u) = \chi_1(u) \\ z_2(u) = z_1(u) \pm \chi_2(u) \\ \dots \\ z_n(u) = z_{n-1}(u) \pm \chi_n(u). \end{cases} \quad (16)$$

Applying triangle inequality rule to $|z_n(u)|$ yields

$$|z_n(u)| \leq |z_{n-1}(u)| + |\chi_n(u)|. \quad (17)$$

Then, performing the same operation on $|z_{n-1}(u)|$ again, one can further have $|z_n(u)| \leq |z_{n-2}(u)| + |\chi_{n-1}(u)| + |\chi_n(u)|$. Repeatedly, this procedure will end up with

$$|z_n(u)| \leq \sum_i^n |\chi_i(u)|. \quad (18)$$

Equation (18) implies two important pieces of information: 1) there exists a satisfaction condition that ensures the equality hold; and 2) $\sum_i^n |\chi_i(u)|$ equals the maximal element of $\{|z_n(u)|\}$. For example, if $\chi_i(u) \geq 0 (i = 1, \dots, n)$, it is able to verify $|\chi_1(u) + \chi_2(u) + \dots + \chi_n(u)| = \sum_i^n |\chi_i(u)|$, while if $\chi_i(u) \geq 0 (i = 1, \dots, n-1)$ yet $\chi_n(u) < 0$, one can also derive $|\chi_1(u) + \chi_2(u) + \dots - \chi_n(u)| = \sum_i^n |\chi_i(u)|$. With deeper analysis, no matter $\chi_i(u)$ is positive, negative or zero, when all elements of $\{|z_n(u)|\}$ are less than E , $\sum_i^n |\chi_i(u)| < E$ is certain to be satisfied. Thus, the theorem is proved.

B. Linearization of Process Constraints

The nonlinear process constraints in (4) mainly refer to the tangential snap, jerk and acceleration. As indicated by (8), their nonlinearity arises from the high-order coupled terms $f_{uuu}(u)f(u)^3$, $f_u(u)f_{uu}(u)f(u)^2$, $f_u(u)^3f(u)$, $f_{uu}(u)f(u)^2$ and $f_u(u)^2f(u)$, etc.. In order to achieve linearization, a hybrid expression of the tangential snap $s(u)$ is derived by integrating the jerk $j(u)$ and acceleration $a(u)$ as

$$s(u) = \frac{1}{\lambda^3} f_{uuu}(u)f(u)^3 + \frac{3}{\lambda^2} f_{uu}(u)f(u)a(u) + \frac{1}{\lambda} f_u(u)j(u). \quad (19)$$

Assuming the existence of an approximate feedrate upper limit $f^*(u) (f(u) < f^*(u))$ that satisfies all the constraints, it is permissible to reformulate the inequality of the tangential snap constraint in (4) as

$$\begin{aligned} & \left| \frac{f(u)^3}{\lambda^3 f^*(u)^3} f_{uuu}(u) + \frac{3f(u)a(u)}{\lambda^2 f^*(u)^3} f_{uu}(u) \right. \\ & \left. + \frac{j(u)}{\lambda f^*(u)^3} f_u(u) \right| < \frac{s_{\max}}{f^*(u)^3}. \end{aligned} \quad (20)$$

Subsequently, using the triangular inequality, one can further have

$$\begin{aligned} & \left| \frac{f(u)^3}{\lambda^3 f^*(u)^3} f_{uuu}(u) + \frac{3f(u)a(u)}{\lambda^2 f^*(u)^3} f_{uu}(u) + \frac{j(u)}{\lambda f^*(u)^3} f_u(u) \right| \\ & \leq \left| \frac{f(u)^3}{\lambda^3 f^*(u)^3} f_{uuu}(u) \right| + \left| \frac{3f(u)a(u)}{\lambda^2 f^*(u)^3} f_{uu}(u) \right| + \left| \frac{j(u)}{\lambda f^*(u)^3} f_u(u) \right|. \end{aligned} \quad (21)$$

Considering that the synchronous constraints $j(u) < j_{\max}$, $a(u) < a_{\max}$ and $f(u) < f^*(u)$ would be fulfilled at the same time, (21) is capable of being relaxed by

$$\begin{aligned} & \left| \frac{f(u)^3}{\lambda^3 f^*(u)^3} f_{uuu}(u) \right| + \left| \frac{3f(u)a(u)}{\lambda^2 f^*(u)^3} f_{uu}(u) \right| + \left| \frac{j(u)}{\lambda f^*(u)^3} f_u(u) \right| \\ & < \left| \frac{1}{\lambda^3} f_{uuu}(u) \right| + \left| \frac{3a_{\max}}{\lambda^2 f^*(u)^2} f_{uu}(u) \right| + \left| \frac{j_{\max}}{\lambda f^*(u)^3} f_u(u) \right|. \end{aligned} \quad (22)$$

Then, according to Theorem 1, a new expression of the tangential snap constraint is derived as

$$\left| \frac{1}{\lambda^3} f_{uuu}(u) \pm \frac{3a_{\max}}{\lambda^2 f^*(u)^2} f_{uu}(u) \pm \frac{j_{\max}}{\lambda f^*(u)^3} f_u(u) \right| < \frac{s_{\max}}{f^*(u)^3}. \quad (23)$$

From (2) and (3), it could be noticed that $f_u(u)$, $f_{uu}(u)$, and $f_{uuu}(u)$ in (23) are all linearly represented by the control points $[d_1, \dots, d_{e-1}]^T$. Consequently, the linear analytical relationship between the tangential snap constraint and the feedrate (namely its control points) is thus established. Note that, aforementioned relaxed formulation does not cause a significant loss of optimality, provided that $f^*(u)$ is close enough to $f(u)$. How to determine a proper feedrate upper limit $f^*(u)$ is presented in Section IV-D.

By following a similar procedure, the linearization of the tangential jerk constraint can also be achieved. For the sake of brevity, the tangential jerk is reformulated as

$$j(u) = \left(f_{uu}(u)f(u)^2 + \lambda f_u(u)a(u) \right) \frac{1}{\lambda^2}. \quad (24)$$

Then, utilizing the relaxation operation and *Theorem 1*, the inequality of the tangential jerk constraint in (4) is relaxed as

$$\left| \frac{1}{\lambda^2} f_{uu}(u) \pm \frac{a_{\max}}{\lambda f^*(u)^2} f_u(u) \right| < \frac{j_{\max}}{f^*(u)^2}. \quad (25)$$

For the tangential acceleration constraint in (4), its relaxed mathematical expression is directly given as (26), which could be obtained via a straightforward proportional scaling

$$\left| \frac{1}{\lambda} f_u(u) \right| < \frac{a_{\max}}{f^*(u)}. \quad (26)$$

C. Linearization of Joint Drive Constraints

Due to the increased highly coupled terms (seen in (13)), linearization of joint drive constraints is more complicated than that of process constraints. The key of solving this issue is to decouple the involved high-order terms, namely $B_{4,4}$, $B_{4,3}$, $B_{4,2}$, and $B_{4,1}$, etc.. Considering that the joint velocity is already linear to the feedrate $f(u)$, it will not be considered during the process. By integrating (7)–(9), one can derive a new expression of joint snap as follows:

$$\ddot{q}_\tau(u) = \mathbf{b}\xi + \mathbf{c}\zeta \quad (27)$$

with

$$\begin{cases} \mathbf{b} = [f_{uuu}(u)f(u)^3, 4f_{uu}(u)f(u)^3, 6f_u(u)f(u)^3, f(u)^4] \\ \xi = \left[\frac{q_u^\tau(u)}{\lambda^4}, \frac{q_{uu}^\tau(u)}{\lambda^4}, \frac{q_{uuu}^\tau(u)}{\lambda^4}, \frac{q_{uuuu}^\tau(u)}{\lambda^4} \right]^T \\ \mathbf{c} = [3f_{uu}(u)f(u)a_c(u), 7f_u(u)f(u)a_c(u), f_u(u)j_c(u)] \\ \zeta = \left[\frac{q_u^\tau(u)}{\lambda^3}, \frac{q_{uu}^\tau(u)}{\lambda^3}, \frac{q_u^\tau(u)}{\lambda^2} \right]^T. \end{cases} \quad (28)$$

Accordingly, the inequality of joint snap constraint in (4) can be reformulated as

$$|\mathbf{b}\xi + \mathbf{c}\zeta| < S_{\tau, \max}. \quad (29)$$

By scaling both sides of (29) with the feedrate upper limit $f^*(u)$, an equivalent expression of the joint snap constraint is derived as

$$|\mathbf{b}^*\xi + \mathbf{c}^*\zeta| < \frac{S_{\tau, \max}}{f^*(u)^3} \quad (30)$$

where $\mathbf{b}^* = \mathbf{b}/f^*(u)^3$ and $\mathbf{c}^* = \mathbf{c}/f^*(u)^3$. Using triangular inequality, one can have

$$|\mathbf{b}^*\xi + \mathbf{c}^*\zeta| < |\mathbf{b}^*\xi| + |\mathbf{c}^*\zeta| < |\mathbf{b}^*\xi| + |\mathbf{c}_1^*\zeta_1| + |\mathbf{c}_2^*\zeta_2| \quad (31)$$

with

$$\begin{cases} \mathbf{c}_1^* = \left[f_{uu}(u) \frac{3f(u)a_c(u)}{f^*(u)^3}, f_u(u) \frac{7f(u)a_c(u)}{f^*(u)^3} \right] \\ \zeta_1 = \left[\frac{q_u^\tau(u)}{\lambda^3}, \frac{q_{uu}^\tau(u)}{\lambda^3} \right]^T \\ \mathbf{c}_2^* = f_u(u) \frac{j_c(u)}{f^*(u)^3} \\ \zeta_2 = \frac{q_u^\tau(u)}{\lambda^2}. \end{cases} \quad (32)$$

Considering the synchronous constraints $j(u) < j_{\max}$, $a(u) < a_{\max}$ and $f(u) < f^*(u)$, it is able to further derive

$$|\mathbf{b}^*\xi| + |\mathbf{c}_1^*\zeta_1| + |\mathbf{c}_2^*\zeta_2| < |\hat{\mathbf{b}}^*\xi| + |\hat{\mathbf{c}}_1^*\zeta_1| + |\hat{\mathbf{c}}_2^*\zeta_2| \quad (33)$$

with

$$\begin{cases} \hat{\mathbf{b}}^* = [f_{uuu}(u), 4f_{uu}(u), 6f_u(u), f(u)] \\ \hat{\mathbf{c}}_1^* = \left[f_{uu}(u) \frac{3a_{c, \max}}{f^*(u)^2}, f_u(u) \frac{7a_{c, \max}}{f^*(u)^2} \right] \\ \hat{\mathbf{c}}_2^* = f_u(u) \frac{j_{c, \max}}{f^*(u)^3}. \end{cases} \quad (34)$$

Then, using *Theorem 1*, a linear expression of the joint snap constraint is accordingly derived as

$$|\hat{\mathbf{b}}^*\xi \pm \hat{\mathbf{c}}_1^*\zeta_1 \pm \hat{\mathbf{c}}_2^*\zeta_2| < \frac{S_{\tau, \max}}{f^*(u)^3}. \quad (35)$$

To linearize the joint jerk constraint, it is necessary to rewrite the expression of joint jerk in (10) as

$$\ddot{q}_\tau(u) = \varphi\varpi + \frac{1}{\lambda} \ddot{q}_\tau(u) f_u(u) \quad (36)$$

with

$$\begin{cases} \varphi = [f(u)^2 f_{uu}(u), 2f(u)^2 f_u(u), f(u)^3] \\ \varpi = \left[\frac{q_u^\tau(u)}{\lambda^3}, \frac{q_{uu}^\tau(u)}{\lambda^3}, \frac{q_{uuu}^\tau(u)}{\lambda^3} \right]^T. \end{cases} \quad (37)$$

Then, using a similar relaxation operation mentioned above, the joint jerk constraint in (4) can be reformulated as

$$\left| \hat{\varphi}\varpi \pm f_u(u) \frac{A_{\tau, \max}}{\lambda f^*(u)^2} \right| < \frac{J_{\tau, \max}}{f^*(u)^2} \quad (38)$$

with

$$\hat{\varphi} = [f_{uu}(u), 2f_u(u), f(u)]. \quad (39)$$

From (10), it could be noticed that the joint acceleration is only related to $f_u(u)f(u)$ and $f(u)^2$. Through a simple proportional scaling similar to (26), the inequality of joint acceleration constraint in (4) can be directly relaxed as

$$\left| f_u(u) \frac{q_u^\tau(u)}{\lambda^2} + f(u) \frac{q_{uu}^\tau(u)}{\lambda^2} \right| < \frac{A_{\tau, \max}}{f^*(u)}. \quad (40)$$

Based on above analysis, the feedrate optimization problem represented by (4) can be finally reduced to a compact LP issue

as in the following:

$$\begin{aligned} & \max \int_0^1 \sum_{i=0}^{e-1} N_{i,k}(u) d_i du \\ \text{s.t.} & \begin{cases} \sum_{i=0}^{e-1} N_{i,k}(u) d_i < \min(f^*(u), f_{geo}(u), f_{\max}) \\ \left| \sum_{i=0}^{e-1} \frac{\bar{q}_u^T(u) N_{i,k}(u)}{\lambda} d_i \right| < V_{\tau, \max} \\ \mathbf{H}\mathbf{D} < \Theta \\ d_i \in [0, f_{\max}] \end{cases} \end{aligned} \quad (41)$$

where \mathbf{H} is a coefficient matrix and determined by substituting the expressions of $f(u)$, $f_u(u)$, $f_{uu}(u)$, $f_{uuu}(u)$ and $f_{uuuu}(u)$ (in (2) and (3)) into (23), (25), (26), (35), (38), and (40). $\mathbf{D} = [d_1, \dots, d_{e-1}]^T$ is the control point vector in (2). Θ is a vector that represents the limits of all the nonlinear constraints in (4), namely a_{\max} , j_{\max} , s_{\max} , $A_{\tau, \max}$, $J_{\tau, \max}$, and $S_{\tau, \max}$.

D. Determination of the Feedrate Upper Limit

As mentioned before, determining the feedrate upper limit $f^*(u)$ is crucial to ensure the optimality of feedrate scheduling. Assuming that the feedrate maintains a constant value within a small sub-region of the tool path, it is easy to derive $f_u(u) = f_{uu}(u) = f_{uuu}(u) = 0$ within this specified area. In this situation, the calculation formulas of the kinematic parameters in (7) and (10) could be significantly simplified. Given a parameter position u along the tool path, an approximate feedrate limit $f^*(u)$ used for solving (41) can be computed by

$$f^*(u) = \min \left(\frac{\lambda V_{\tau, \max}}{|q_u^T(u)|}, \sqrt{\frac{\lambda^2 A_{\tau, \max}}{|q_{uu}^T(u)|}}, \sqrt[3]{\frac{\lambda^3 J_{\tau, \max}}{|q_{uuu}^T(u)|}}, \sqrt[4]{\frac{\lambda^4 S_{\tau, \max}}{|q_{uuuu}^T(u)|}} \right). \quad (42)$$

Remark 1: Based on the above insight, the proposed method has the following features.

- 1) All the relevant kinematic parameters (namely $f(u)$, $a(u)$, $j(u)$, $s(u)$, $V(u)$, $A(u)$, $J(u)$, and $S(u)$), as well as their constraints in (4) are derived analytically without any numerical approximation, which offers accurate and robust trajectory control in robotic milling.
- 2) The feasible region of decision variables is determined according to the maximum feedrate limit and considered as a basic constraint in (41). In addition, the proposed method is developed based on a standard LP model. Benefiting from above two characteristics, the proposed method could provide guaranteed feasible region of decision variables and guarantee a global convergence.
- 3) It is the first time that, the joint snap constraints of robots are linearized with respect to the primary feedrate and allowed to be handled during the time-optimal feedrate scheduling by LP algorithms.
- 4) Compared with existing methods, it is feasible for users obtain an analytical solution to feedrate profile described B-spline curve, in which the control points of feedrate profile are directly optimized during the solution process.

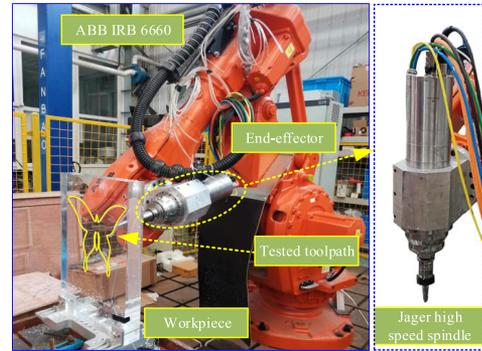


Fig. 2. Robotic milling experimental platform, consisting of an ABB IRB 6660 robot and a Jager high speed spindle.

This is favorable to avoid extra curve-fitting operation and loss of accuracy.

E. Implementation of the Proposed Method

To implement the proposed method, a scanning operation of path curve is necessary for the constraint evaluation in feedrate scheduling. Theoretically, the number of constraint evaluation points along the tool path could be identified according to the change of path curvature or determined by the capability of the solution systems, which depends on the user's requirements. Without losing generality, all the constraint evaluation points are sampled with a uniform parameter interval in this article.

V. SIMULATION AND EXPERIMENTAL RESULTS

In this section, both simulation and experiment are conducted on a typical milling example to validate the proposed method. As shown in Fig. 2, the ABB IRB 6660 robot is used to track a curved tool path under several constraints. Therein, the tool path is represented by a quartic B-spline with 493 control points and has a total length of 1462.66 mm. It can be noticed that, there are multiple corners and regions with tight curvature in this designed example. In order to adapt these geometric features of tool path, the modulation of feedrate is essential to achieve smooth joint motion and reduce the soft impact on the end-effector. That is why this curved tool path is chosen to verify the effectiveness of the proposed method. In addition, the existing AD method [13], HS method [24], LP method [30], and SQP method [21] are also implemented for comparison. More details about the results are described in the following subsections.

A. Simulation

In the simulation, to ensure a fair comparison, 300 constraint evaluation points are sampled along the tool path for all tested methods. In the LP [30], SQP [21] and proposed methods, a total of 100 control points are used to model the target feedrate curve. Due to the difference in problem modeling, the AD [13] and HS [24] methods have additional setup requirements. Specifically, in the AD method [13], a user-defined ramp coefficient $\alpha \in [0, 1]$ need be assigned to avoid producing discontinuous jerk. Without loss of generality, the value of $\alpha = 0.5$ is used in the AD method [13]. In the HS method [24],

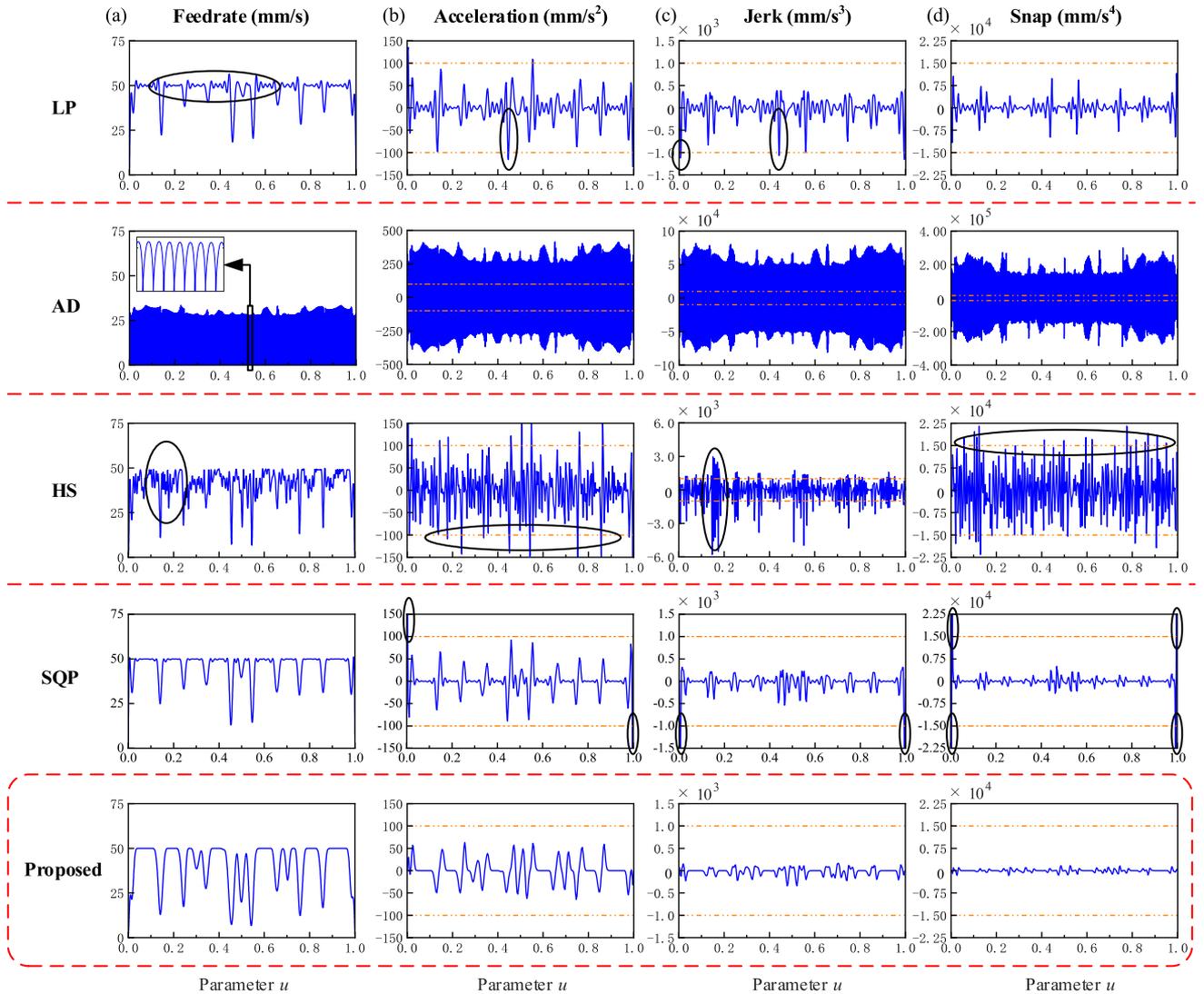


Fig. 3. Comparison of the tangential kinematic profiles of the robotic task space produced by different methods. (a) The feedrate profiles. (b) The tangential acceleration profiles. (c) The tangential jerk profiles. (d) The tangential snap profiles.

TABLE I
PARAMETERS USED IN TRAJECTORY PLANNING

Parameters	Value	Unit
f_{\max}	50	mm/s
$V_{\tau,\max}$	0.1/0.1/0.1/0.1/0.1	rad/s
$A_{\tau,\max}$	0.25/0.25/0.25/0.25/0.25	rad/s ²
$J_{\tau,\max}$	5/5/5/5/5	rad/s ³
$S_{\tau,\max}$	15/15/15/15/15	rad/s ⁴
$a_{\max}/j_{\max}/s_{\max}$	100/1000/15000	mm/s ^{2/3/4}

to guarantee the interpolation accuracy, the number of decision variables need equal the number of path sampling points plus 4, namely, $e = n + 4$. During the comparison, the snap controllability is one of the most important evaluation metrics for assessing different methods. Moreover, the improvement of productivity, computational efficiency and machining accuracy are also analyzed in the following subsections. Table I lists the associated limits imposed on robotic milling.

Fig. 3 shows the tangential kinematic profiles of the robotic task space produced by different methods. From the result, one can notice that the planned feedrate profiles using the LP method [30] and the HS method [24] have multiple fluctuation regions along the trajectory. This is mainly caused by their inherent spline fitting operation [30] and spline interpolation operation [24]. This unfavorable ripple phenomenon would become more obvious as the number of evaluation points is increased. Although the AD method [13] is capable of eliminating this phenomenon, its frequent acceleration and deceleration inevitably cause an increased machining time, and also suffer from a risk of mechanical vibrations on cutting tool. To some extent, the SQP method [21] could effectively handle this issue. Nevertheless, in order to guarantee the solution optimality, tiny but visible feedrate fluctuations often occur around its maximum. In contrast, the feedrate profile planned by the proposed method is entirely smooth and has eleven slowdown regions corresponding to the constraint-sensitive areas of the tool path.

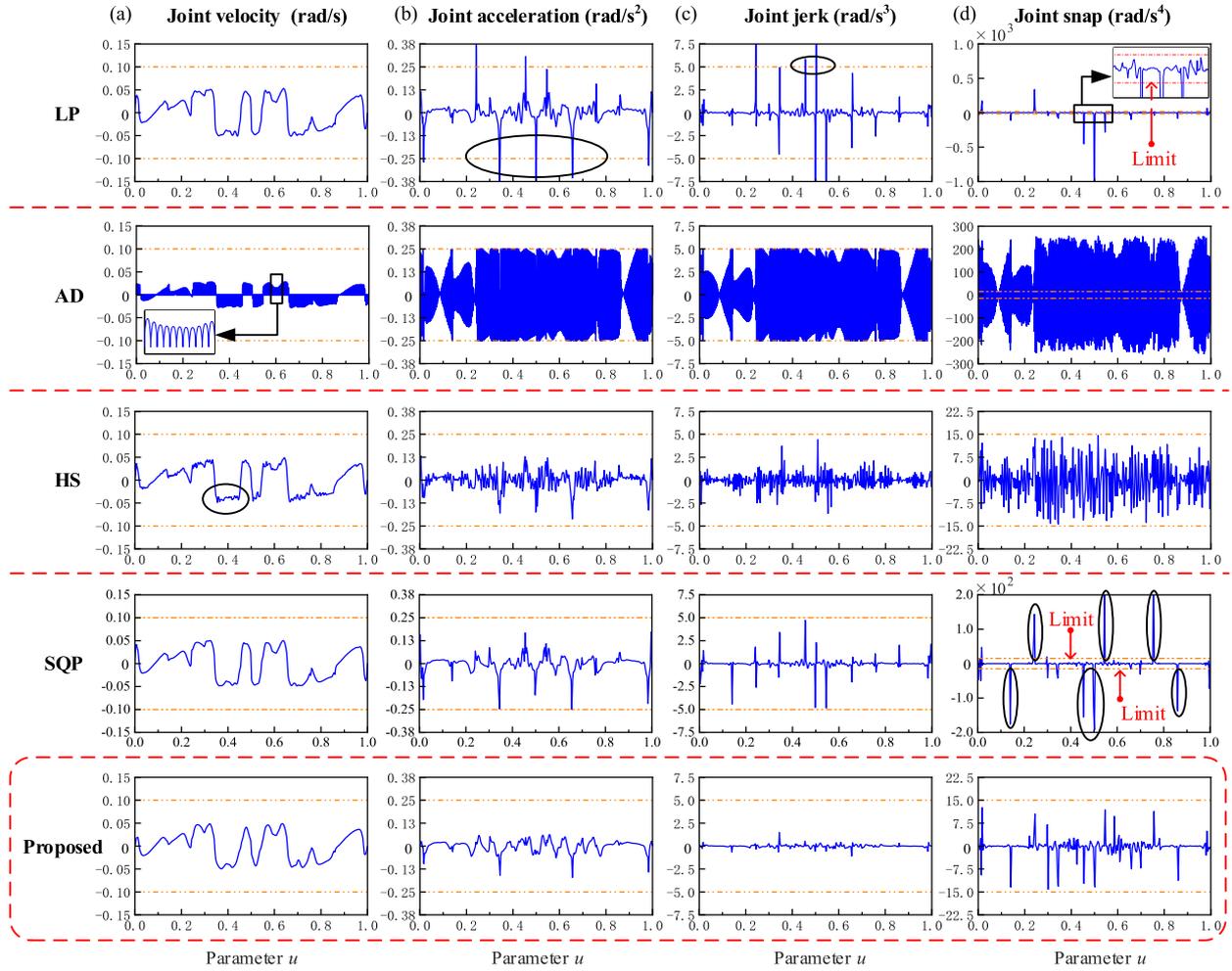


Fig. 4. Comparison of the (second joint) kinematic profiles of the robotic joint space produced by different methods. (a) The joint velocity profiles. (b) The joint acceleration profiles. (c) The joint jerk profiles. (d) The joint snap profiles.

Especially at the entrance and exit of these slowdown regions, the feedrate also maintains a soft variation. From the planned results of tangential acceleration, tangential jerk, and tangential snap, the LP [30], AD [13], HS [24], and SQP [21] methods all fail to realize accurate kinematic control in robotic task space. As a comparison, the magnitudes of these high order tangential kinematic profiles are well confined within their maximum allowable ranges using the proposed method.

To further illustrate the benefits of the proposed method, Fig. 4 shows an example of the second joint kinematic results of the robot produced by different methods. Notably, several isolated instances of acceleration and jerk constraint violation occur with the LP method [30], primarily due to its reliance on difference approximation operations. Apart from this, all the mentioned methods are capable of limiting the joint velocity, acceleration, and jerk. Nevertheless, only the HS method [24] and the proposed method successfully control the joint snaps within the allowable range. From the perspective of motion smoothness, it can be noticed that, the joint velocity produced by the proposed method is obviously smoother than that of HS method [24]. For detailed comparison, the statistical results of the maximum magnitude of the snaps of all involved joints and their average

TABLE II
MAXIMUM MAGNITUDE OF JOINT SNAPS PRODUCED BY DIFFERENT METHODS

Methods	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6	Unit
LP [30]	502.677	1354.434	1258.535	484.724	313.648	1302.075	rad/s ⁴
AD [13]	257.145	257.976	299.583	238.329	209.836	308.770	rad/s ⁴
HS [24]	14.993	14.591	14.562	13.774	10.174	14.907	rad/s ⁴
SQP [21]	181.957	280.504	308.879	161.440	114.292	317.134	rad/s ⁴
Proposed	14.426	14.141	14.421	12.493	9.003	14.816	rad/s ⁴

TABLE III
AVERAGE ABSOLUTE DEVIATIONS (AADs) OF JOINT SNAPS PRODUCED BY DIFFERENT METHODS

Methods	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
LP [30]	9.559	12.249	12.199	7.907	5.691	12.704
AD [13]	52.094	77.794	84.888	43.693	33.722	86.843
HS [24]	3.949	4.921	5.223	3.212	2.446	5.337
SQP [21]	4.243	7.830	7.592	4.353	2.517	7.893
Proposed	1.124	1.244	1.180	0.998	0.629	1.216

absolute deviations (AADs) are shown in Table II and Table III, respectively. According the results, it can be seen that, all joint snap constraints are fulfilled using the proposed method.

TABLE IV
CONSTRAINT CONTROL CAPABILITY COMPARISON BETWEEN THE PROPOSED METHOD AND EXISTING LP, AD, HS, SQP METHODS

Methods	Free from numerical approximation?	Snap-bounded in joint space?	Snap-bounded in task space?
LP [30]	✗	✗	✓
AD [13]	✓	✗	✗
HS [24]	✓	✓	✗
SQP [21]	✓	✗	✗
Proposed	✓	✓	✓

TABLE V
PERFORMANCE COMPARISON BETWEEN THE PROPOSED METHOD AND EXISTING LP, AD, HS, SQP METHODS

Methods	Machining time (s)	Computational time (s)	Maximum error (mm)
LP [30]	32.38	0.073	0.4364
AD [13]	97.76	0.014	1.3093
HS [24]	50.36	1565.566	0.2018
SQP [21]	33.42	1.213	0.3856
Proposed	45.08	0.112	0.1183

Even compared with the other methods, the index of AAD generated by the proposed method is also the minimum, which indicates a reliable snap control ability of the proposed method. In Table IV, it gives a summary of the above comparative results.

Table V shows the optimal machining time obtained by different methods for the reference task. The results indicate that the LP method [30] offers the shortest task time, but it comes at the expense of compromising constraints control accuracy (as shown in Fig. 4). When the feedrate profiles of the AD [13] and HS methods [24] are used in practice, their corresponding task time is 97.76 s and 50.36 s, respectively. In contrast, the proposed method consumes 45.08 s to accomplish the same task, which contributes to almost 53.89% and 10.5% increase of productivity, respectively. Since the SQP method [21] does not consider the snap constraints in their model, the task time is slightly shorter than that of the proposed method (33.42 s versus 45.08 s).

In addition, the computational complexity of the proposed model (41) is also analyzed. According to the simulation setup, the number of decision variables is e , and a total of n sampling points is set along the tool path. For each sample point, it needs to construct 101 inequalities to characterize the machining-related constraints. Except for this, $2e+2$ inequalities are also required to characterize the boundary conditions. Considering that (41) is equivalent to a standard LP problem, one can infer that the proposed model has a complexity of $O(e^{3.5})$. The computational overheads for different methods are summarized in Table V, which is measured in MATLAB environment running on a computer with 1.6 GHz Core i5 processor. Since the model computational complexity does not include the preprocessing events, in order to make a fair comparison, the computational overhead here refers to the real time spent on solving the model. Note that, among all the methods mentioned, only the AD method [13] is dedicated to real-time applications. Thereby, the computational efficiency of the AD method [13] is the highest, as summarized in Table V. Despite this, the proposed method also demonstrates

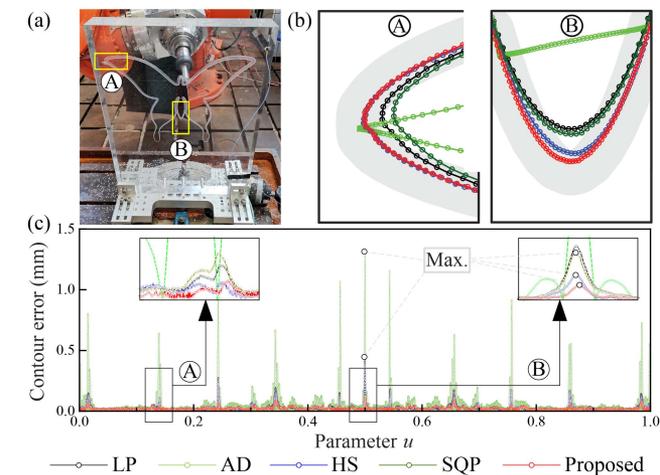


Fig. 5. Experimental results. (a) The machined part. (b) The comparison between the measured trajectories. (c) The resulting contour errors.

a satisfactory computational performance, which only requires about 0.112 s to determine the final optimum solution. When compared with the HS [24] and the SQP method [21], the computational efficiency is significantly improved.

These above simulation results confirm that the proposed method not only has the ability to generate a snap-bounded feedrate profile, ensuring robust control of robotic kinematics in both task space and joint space, but also can maintain a satisfactory performance in terms of machining and computational efficiency.

B. Experimental Results

To validate the feasibility of the proposed method, comparative experiments are further conducted in this section. The optimized feedrate profiles shown in Fig. 4 are directly used to control the experimental platform, and the platform allows the measurement of the position of each joint from the encoder at a user-defined sampling interval. During the process, the sampling interval is set as 4 ms. Fig. 5(a) shows the final machined part, and Fig. 5(b) shows the zoom shots of the measured trajectories corresponding to the marked A/B zones in Fig. 5(a). In contrast to the other methods, the proposed method yields the best performance, where its corresponding trajectory is consistent with the desired trajectory, and no overshoot or excessive deviation occurs around the tight curvature zones (e.g., region B). For quantitative comparison, Fig. 5(c) plots the curves of the resulting contour errors, and the results are provided in Table V. From the results, it can be observed that the maximum contour errors for the LP [30], AD [13], HS [24], SQP [21] and the proposed methods are 0.4364 mm, 1.3093 mm, 0.2018 mm, 0.3856 mm, and 0.1183 mm, respectively. Through computation, the reduction of the error magnitude using the proposed method is up to 41%, compared with the best of other methods. The results confirm the effectiveness of the proposed method for ensuring the trajectory following accuracy of the robots.

VI. CONCLUSION

In this article, the problem of time-optimal feedrate scheduling for robotic milling was investigated. A primary contribution of this work was the development of a novel method for transforming the complicated nonlinear feedrate scheduling problem into a linear convex optimization problem with improved computational efficiency. Compared with existing methods, the proposed method can not only realize kinematic control up to the snap level, but also maintain global optimality in productivity. In addition, the proposed method was free of numerical approximation, and able to offer an analytical solution to the feedrate profile described by Bspline without any loss of accuracy. Both comparative simulations and experiments were conducted to validate the proposed method, and the results confirmed its strong capability in limiting the highly-coupled nonlinear constraints. For future work, the proposed method will be extended to address more challenging issues in dual-robot mirror milling of thin-walled parts, such as kinematic control of collaborative robots and contour error compensation of thin-walled parts. Also, the functional redundancy of the robots will be fully exploited to enhance the mirror milling performance.

REFERENCES

- [1] Q. Bi, X. Wang, Q. Wu, L. Zhu, and H. Ding, "FV-SVM-based wall-thickness error decomposition for adaptive machining of large skin parts," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2426–2434, Apr. 2019.
- [2] Z. Jiang, W. Zhou, H. Li, Y. Mo, W. Ni, and Q. Huang, "A new kind of accurate calibration method for robotic kinematic parameters based on the extended Kalman and particle filter algorithm," *IEEE Trans. Ind. Electron.*, vol. 65, no. 4, pp. 3337–3345, Apr. 2018.
- [3] L. Lu et al., "Joint-smooth toolpath planning by optimized differential vector for robot surface machining considering the tool orientation constraints," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 4, pp. 2301–2311, Aug. 2022.
- [4] Y. Zeng, W. Tian, and W. Liao, "Positional error similarity analysis for error compensation of industrial robots," *Robot. Comput.-Integr. Manuf.*, vol. 42, pp. 113–120, 2016.
- [5] Y. Zhang, S. Li, J. Gui, and X. Luo, "Velocity-level control with compliance to acceleration-level constraints: A novel scheme for manipulator redundancy resolution," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 921–930, Mar. 2018.
- [6] J. Peng, Y. Ding, G. Zhang, and H. Ding, "Smoothness-oriented path optimization for robotic milling processes," *Sci. China Technol. Sci.*, vol. 63, no. 9, pp. 1751–1763, 2020.
- [7] Y. Zhang, S. Li, J. Zou, and A. H. Khan, "A passivity-based approach for kinematic control of manipulators with constraints," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3029–3038, May 2020.
- [8] H.-J. Heo, Y. Son, and J.-M. Kim, "A trapezoidal velocity profile generator for position control using a feedback strategy," *Energies*, vol. 12, no. 7, pp. 1–14, 2019.
- [9] R. H. Castain and R. P. Paul, "An on-line dynamic trajectory generator," *Int. J. Robot. Res.*, vol. 3, no. 1, pp. 68–72, 1984.
- [10] P. Lambrechts, M. Boerlage, and M. Steinbuch, "Trajectory planning and feedforward design for electromechanical motion systems," *Control Eng. Pract.*, vol. 13, no. 2, pp. 145–157, 2005.
- [11] A. Valente, S. Baraldo, and E. Carpanzano, "Smooth trajectory generation for industrial robots performing high precision assembly processes," *CIRP Ann.*, vol. 66, no. 1, pp. 17–20, 2017.
- [12] S. Perumaal and N. Jawahar, "Synchronized trigonometric s-curve trajectory for jerk-bounded time-optimal pick and place operation," *Int. J. Robot. Automat.*, vol. 27, no. 4, pp. 385–395, 2012.
- [13] Y. Fang, J. Qi, J. Hu, W. Wang, and Y. Peng, "An approach for jerk-continuous trajectory generation of robotic manipulators with kinematical constraints," *Mech. Mach. Theory*, vol. 153, 2020, Art. no. 103957.
- [14] J. Zhou, H. Cao, P. Jiang, C. Li, H. Yi, and M. Liu, "Energy-saving trajectory planning for robotic high-speed milling of sculptured surfaces," *IEEE Trans. Automat. Sci. Eng.*, vol. 19, no. 3, pp. 2278–2294, Jul. 2022.
- [15] R. Chai et al., "A two phases multiobjective trajectory optimization scheme for multi-UGVs in the sight of the first aid scenario," *IEEE Trans. Cybern.*, vol. 54, no. 9, pp. 5078–5091, Sep. 2024.
- [16] R. Chai, A. Tsourdos, A. Savvaris, S. Chai, Y. Xia, and C. P. Chen, "Design and implementation of deep neural network-based control for automatic parking maneuver process," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 4, pp. 1400–1413, Apr. 2022.
- [17] L. Petrović, I. Marković, and I. Petrović, "Mixtures of Gaussian processes for robot motion planning using stochastic trajectory optimization," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 52, no. 12, pp. 7378–7390, Dec. 2022.
- [18] C. Dai, S. Lefebvre, K.-M. Yu, J. M. Geraedts, and C. C. Wang, "Planning jerk-optimized trajectory with discrete time constraints for redundant robots," *IEEE Trans. Automat. Sci. Eng.*, vol. 17, no. 4, pp. 1711–1724, Oct. 2020.
- [19] J. Huang, P. Hu, K. Wu, and M. Zeng, "Optimal time-jerk trajectory planning for industrial robots," *Mech. Mach. Theory*, vol. 121, pp. 530–544, 2018.
- [20] H. Liu, X. Lai, and W. Wu, "Time-optimal and jerk-continuous trajectory planning for robot manipulators with kinematic constraints," *Robot. Comput.-Integr. Manuf.*, vol. 29, no. 2, pp. 309–317, 2013.
- [21] B. Sencer, Y. Altintas, and E. Croft, "Feed optimization for five-axis CNC machine tools with drive constraints," *Int. J. Mach. Tools Manufacture*, vol. 48, no. 7-8, pp. 733–745, 2008.
- [22] J. Xiao, S. Liu, H. Liu, M. Wang, G. Li, and Y. Wang, "A jerk-limited heuristic feedrate scheduling method based on particle swarm optimization for a 5-DoF hybrid robot," *Robot. Comput.-Integr. Manuf.*, vol. 78, 2022, Art. no. 102396.
- [23] R. Chai, A. Tsourdos, S. Chai, Y. Xia, A. Savvaris, and C. P. Chen, "Multiphase overtaking maneuver planning for autonomous ground vehicles via a desensitized trajectory optimization approach," *IEEE Trans. Ind. Informat.*, vol. 19, no. 1, pp. 74–87, Jan. 2023.
- [24] A. Bharathi and J. Dong, "Feedrate optimization for smooth minimum-time trajectory generation with higher order constraints," *Int. J. Adv. Manuf. Technol.*, vol. 82, pp. 1029–1040, 2016.
- [25] S. Momani, O. Abu Arqub, and B. Maayah, "Piecewise optimal fractional reproducing kernel solution and convergence analysis for the Atangana-Baleanu-Caputo model of the liénard's equation," *Fractals*, vol. 28, no. 08, 2020, Art. no. 2040007.
- [26] O. Abu Arqub and N. Shawagfeh, "Solving optimal control problems of Fredholm constraint optimality via the reproducing kernel Hilbert space method with error estimates and convergence analysis," *Math. Methods Appl. Sci.*, vol. 44, no. 10, pp. 7915–7932, 2021.
- [27] Y. Sun, M. Chen, J. Jia, Y.-S. Lee, and D. Guo, "Jerk-limited feedrate scheduling and optimization for five-axis machining using new piecewise linear programming approach," *Sci. China Technol. Sci.*, vol. 62, no. 7, pp. 1067–1081, 2019.
- [28] Y. Sun, Z. Shi, and J. Xu, "Synchronous feedrate scheduling for the dual-robot machining of complex surface parts with varying wall thickness," *Int. J. Adv. Manuf. Technol.*, vol. 119, no. 3, pp. 2653–2667, 2022.
- [29] W. Fan, X.-S. Gao, C.-H. Lee, K. Zhang, and Q. Zhang, "Time-optimal interpolation for five-axis CNC machining along parametric tool path based on linear programming," *Int. J. Adv. Manuf. Technol.*, vol. 69, pp. 1373–1388, 2013.
- [30] X. Zhao, H. Zhao, J. Yang, and H. Ding, "An adaptive feedrate scheduling method with multi-constraints for five-axis machine tools," in *Proc. Intell. Robot. Applications: 8th Int. Conf.*, 2015, pp. 553–564.
- [31] Á. Nagy and I. Vajk, "LP-based velocity profile generation for robotic manipulators," *Int. J. Control*, vol. 91, no. 3, pp. 582–592, 2018.
- [32] R. Chai, D. Liu, T. Liu, A. Tsourdos, Y. Xia, and S. Chai, "Deep learning-based trajectory planning and control for autonomous ground vehicle parking maneuver," *IEEE Trans. Automat. Sci. Eng.*, vol. 20, no. 3, pp. 1633–1647, Jul. 2022.
- [33] R. Chai, Y. Guo, Z. Zuo, K. Chen, H.-S. Shin, and A. Tsourdos, "Cooperative motion planning and control for aerial-ground autonomous systems: Methods and applications," *Prog. Aerosp. Sci.*, vol. 146, 2024, Art. no. 101005.
- [34] R. Chai, H. Niu, J. Carrasco, F. Arvin, H. Yin, and B. Lennox, "Design and experimental validation of deep reinforcement learning-based fast trajectory planning and control for mobile robot in unknown environment," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 5778–5792, Apr. 2024.
- [35] J. Tan, M. Shang, and L. Jin, "Metaheuristic-based RNN for manipulability optimization of redundant manipulators," *IEEE Trans. Ind. Informat.*, vol. 20, no. 4, pp. 6489–6498, Apr. 2024.