

DOI: 10.11830/ISSN.1000-5013.201912036



连续微线段柔性加减速 自适应前瞻规划算法

叶衡, 谢明红

(华侨大学 机电及自动化学院, 福建 厦门 361021)

摘要: 针对目前数控加工代码大多是连续微线段的情况, 提出一种新型的自适应前瞻规划算法. 首先, 采用简化的 S 曲线加减速减少计算时间, 提高加工过程的平稳性, 并采用二分法求解段内可达到的最高速度; 然后, 采用识别速度敏感点的方法分割前瞻区间, 实现自适应动态规划, 进一步优化前瞻区间的终点速度, 提高加工效率; 最后, 通过模拟实验验证算法的可行性. 结果表明: 相较于传统规划算法与其他前瞻规划算法, 文中前瞻规划算法可显著提高加工效率.

关键词: 微线段; 前瞻算法; S 曲线加减速; 自适应; 数控机床; 高速加工

中图分类号: TG 659 **文献标志码:** A **文章编号:** 1000-5013(2021)01-0001-08

Self-Adaptation Look-Ahead Planning Algorithm Based on Continuous Micro Line Flexible Acceleration and Deceleration

YE Heng, XIE Minghong

(College of Mechanical Engineering and Automation, Huaqiao University, Xiamen 361021, China)

Abstract: In view of the fact that most numerical control machining codes are continuous micro lines, a new self-adaptation look-ahead planning algorithm was proposed. Firstly, the simplified S-curve acceleration and deceleration method was used to reduce the calculation time and improve the smoothness of the machining process, and the dichotomy method was used to solve the maximum speed that can be achieved in the segment. Then, the method of identifying speed sensitive points was used to segment the look-ahead interval to realize self-adaptation dynamic planning, which further optimized the terminal speed of the look-ahead interval and improved the processing efficiency. Finally, the feasibility of the algorithm was verified by simulation experiments. The results show that the proposed planning algorithm can significantly improve the machining efficiency compared with the traditional planning algorithm and other look-ahead planning algorithm.

Keywords: micro line; look-ahead algorithm; S-curve acceleration and deceleration; self-adaptation; computerised numerical control machine; high speed machining

大多数数控机床只具有直线和圆弧插补功能, 随着加工零件曲面曲线的日益复杂, 直接描述零件轮廓已不太可能. 目前, 数控机床在加工零件时, 先由计算机辅助制造(CAM)软件将工件的复杂轮廓曲线逼近成大量连续且短小的微线段, 这些线段的长度通常为 0.1~1.0 mm. 采用传统的加减速与插补算法时, 数控机床将每段微线段内独立加减速, 起始点速度设置为零, 机床在每段加工轨迹内频繁启停, 极

收稿日期: 2019-12-29

通信作者: 谢明红(1968-), 男, 研究员, 博士, 主要从事数控技术、CAD/CAM 及计算机视觉的研究. E-mail: xmh@hqu.edu.cn.

基金项目: 国家自然科学基金面上资助项目(51875218); 华侨大学研究生科研创新基金资助项目(18013080038)

大限制了机床能达到的进给速度,使机床各轴的加减速能力无法得到充分利用,同时,机床的频繁启停会造成震动与冲击,降低工件的加工质量.为了解决这些问题,国内外学者进行了大量的研究^[1-3].较为常用的前瞻算法的核心是加减速算法和拐点处的速度约束.王耀庭等^[4]提出一种基于直线加减速方式的高速嵌入式数控系统速度前瞻算法,但直线加减速方式柔性差,对机床仍有冲击.钟前进等^[5]研究 S 曲线加减速,并将其用于数控系统,提高了数控加工的速度和柔性,但该方法计算复杂,不利于实时处理.司慧晓等^[6]将完整的 S 曲线加减速进行简化.

目前,具有前瞻功能的数控机床大多采用固定段数的前瞻方式,前瞻段数达数千甚至上万,极大地占用数控系统的内存资源,降低处理效率.因此,王海涛^[7]根据加工轨迹拐角的大小区分前瞻区间,自适应地调整前瞻段数^[7],但其速度敏感点的识别方式仍可能使前瞻段数过多.许东伟^[8]提出一种以最大速度减速至零走过的段数区分前瞻区间的方法,可缩短前瞻区间,但在敏感点为零时,该方法的速度规划无法达到最优.基于此,本文提出一种连续微线段柔性加减速自适应前瞻规划算法.

1 S 曲线加减速

1.1 加减速原理

直线加减速^[9]柔性较差,而完整的 S 曲线加减速则计算复杂^[10].简化的 S 曲线加减速,如图 1 所示.图 1 中: s 为位移; v 为进给速度; a 为加速度; r 为加加速度; R 为加加速度最大值; $-A$ 为系统允许反向进给加速度的最大值; A 为系统允许进给加速度的最大值.简化的 S 曲线加减速分为加加速段 T_1 、减加速段 T_2 、匀速段 T_3 、加减速段 T_4 、减减速段 T_5 等 5 个时间段^[11].

由于电机在加加速段与减加速段的加加速度绝对值相等,故有

$$T_1 = T_2 = R/A. \quad (1)$$

同理,有

$$T_4 = T_5 = R/A. \quad (2)$$

故有

$$T_1 = T_2 = T_4 = T_5. \quad (3)$$

数控系统允许进给速度最大值 v_{\max} 及 R, A 是确定的,故 S 曲线加减速过程可确定.在完整的 S 曲线加减速中,加加速度 r 为

$$r = \begin{cases} R, & 0 \leq t < t_1, \\ -R, & t_1 \leq t < t_2, \\ 0, & t_2 \leq t < t_3, \\ -R, & t_3 \leq t < t_4, \\ R, & t_4 \leq t < t_5. \end{cases} \quad (4)$$

式(4)中: $t_1 \sim t_5$ 为各阶段的时间节点.

加速度函数 $a(t)$ 、进给速度函数 $v(t)$ 和位移函数 $s(t)$ 可通过积分得到,即

$$a(t) = a(t_i) + \int_{t_i}^t r(\tau_i) d\tau_i, \quad (5)$$

$$v(t) = v(t_i) + \int_{t_i}^t a(\tau_i) d\tau_i, \quad (6)$$

$$s(t) = s(t_i) + \int_{t_i}^t v(\tau_i) d\tau_i. \quad (7)$$

式(5)~(7)中: t_i 为时间节点; τ_i 表示以 t_{i-1} 为起点的时间.

刀具起始点速度均为 0(图 1),可以达到系统允许的加速度和速度,其加加速度 r 越大,达到系统允许最大加速度的时间 T_1 越小,柔性越差;反之,柔性越好.

1.2 段内加减速等效模型

在实际加工中,因各段轨迹长度限制,进给速度不一定可以达到 v_{\max} ,且电机在加工轨迹的起始点

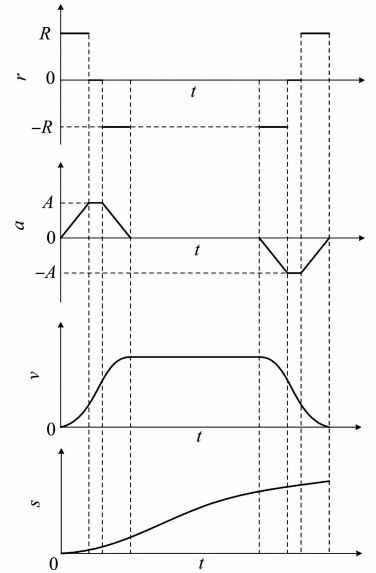


图 1 简化的 S 曲线加减速

Fig. 1 Simplified S-curve acceleration and deceleration

速度不一定为 0. 设起始速度为 v_s , 终点速度为 v_e , 可达到的最大速度为 v_m . 当 $v_s < v_e$ 时, 段内实际加减速情况, 如图 2 所示. 段内加减速曲线可能存在以下 3 种情况.

- 1) 2 段加减速(图 2(a)). 只有 T_1, T_2 或 T_4, T_5 段; $v_s > v_e$ 同理.
- 2) 完整加减速(图 2(b)). 当 $v_m = v_{\max}$ 时, 具有完整的加减速过程, $T_1 = T_2, T_4 = T_5$; $v_s > v_e$ 同理.
- 3) 4 段加减速(图 2(c)). 当 $v_m < v_{\max}$ 时, 缺少 T_3 , $T_1 = T_2, T_4 = T_5$; $v_s > v_e$ 同理.

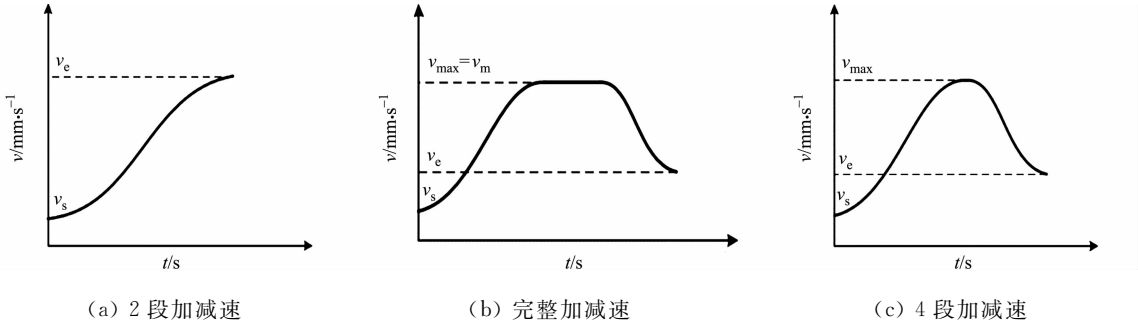


图 2 段内实际加减速情况

Fig. 2 Actual acceleration and deceleration within segment

设轨迹长度为 L ; 电机从 v_s 加速到 v_{\max} , 刀具所走过的位移为 L_{ac} ; 电机从 v_{\max} 减速到 v_e , 刀具所走过的位移为 L_{de} ; 电机从 v_s 加速或减速到 v_e , 刀具所走过的位移为 l . 由式(4)~(7)可得

$$L_{ac} = (v_{\max} + v_s) \sqrt{\frac{v_{\max} - v_s}{R}}, \quad (8)$$

$$L_{de} = (v_{\max} + v_e) \sqrt{\frac{v_{\max} - v_e}{R}}. \quad (9)$$

假设 $v_s > v_e$ ($v_s < v_e$ 同理), 有

$$l = (v_s + v_e) \sqrt{\frac{v_s - v_e}{R}}. \quad (10)$$

当 $L > L_{ac} + L_{de}$ 时, 加工轨迹长度足够电机加速到 v_{\max} ; 当 $l < L < L_{ac} + L_{de}$ 时, 加工轨迹长度不够电机加速到 v_m , 此时满足

$$L = (v_m + v_s) \sqrt{\frac{v_m - v_s}{R}} + (v_m + v_e) \sqrt{\frac{v_m - v_e}{R}}. \quad (11)$$

传统方法无法精确求解 v_m , 有的文献采用牛顿迭代法, 但该法迭代次数过多, 计算过程相对耗时. 因此, 采用二分法求解, 设定误差范围为 0.1, 则有

$$f(v_m) = (v_m + v_s) \sqrt{\frac{v_m - v_s}{R}} + (v_m + v_e) \sqrt{\frac{v_m - v_e}{R}} - L. \quad (12)$$

令 $f(v_m) = 0$, 设置起始区间为 $[v_l, v_r]$, $v_l = \min(v_s, v_e)$, $v_r = v_{\max}$, 不断缩小区间至 $\Delta = |v_l - v_r| < 0.1$, 取 $v_m = \frac{v_l + v_r}{2}$. 经过仿真, 当微线段长度为 0.1~1.0 mm 时, 在设定的误差范围内, 通过二分法计算 v_m 时的迭代次数为 8~20 次, 可以满足插补的实时性计算要求.

2 自适应前瞻速度规划

2.1 拐点处速度约束

L_i 段与 L_{i+1} 段的交点为 P_i , 刀具的运动方向为 $P_{i-1} \rightarrow P_i \rightarrow P_{i+1}$. 刀具在 P_{i-1} 的速度为 v_{i-1} , 在 P_i 的速度为 v_i , 在 P_{i+1} 的速度为 v_{i+1} . 在 L_i 段, 刀具在点 P_i 的进给速度为 v_A , 方向由 P_{i-1} 指向 P_i ; 在 L_{i+1} 段, 刀具在点 P_i 的进给速度为 v_B , 方向由 P_i 指向 P_{i+1} . 在机床加工中, 设定刀具在 P_i 处的速度大小相等, 方向发生了改变, $|v_A| = |v_B|$; v_A 与 v_B 的夹角为 α , 线段 L_i 与 L_{i+1} 的夹角为 β . 拐点示意图, 如图 3 所示.

- 1) 拐角角度对速度的约束. 因为刀具在点 P_i 的速度发生变化, 且速度改变发生在 1 个插补周期

内,所以刀具在拐点 P_i 处产生加速度 a_{trans} . 为避免对机床产生冲击,应使 a_{trans} 不超过机床允许的最大加速度 a_{max} .

设插补周期为 T_s , 则 a_{trans} 满足

$$a_{\text{trans}} = \frac{|\Delta v|}{T_s} = \frac{|v_A - v_B|}{T_s} = \frac{2v_i \sin(\alpha/2)}{T_s} \leq a_{\text{max}}. \quad (13)$$

由此可得

$$v_i \leq \frac{a_{\text{max}} T_s}{2 \sin(\alpha/2)} = v_{\text{trans}}. \quad (14)$$

式(13), (14)中: Δv 为速度变化量; v_{trans} 为拐角角度的约束速度.

拐角角度的速度约束,如图 4 所示. 图 4 中: $v_{\text{trans},m}$ 拐角角度的最大约束速度.

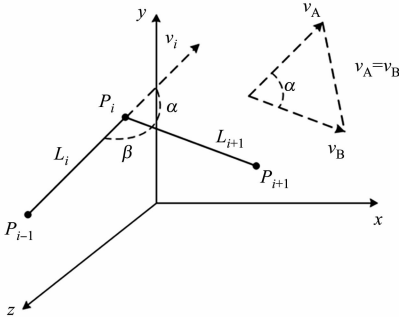


图 3 拐点示意图

Fig. 3 Diagram of corner point

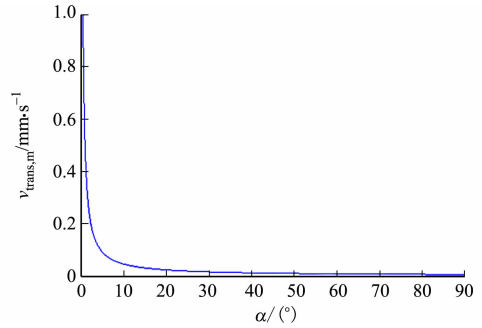


图 4 拐角角度的速度约束

Fig. 4 Speed constraint of corner angle

2) 微线段长度对速度的约束. 刀具运动的方向为 $P_{i-1} \rightarrow P_i \rightarrow P_{i+1}$, $v_{i-1} < v_i$, $v_{i+1} < v_i$, 刀具从 P_{i-1} 到 P_i 为加速过程, 受 L_i 长度的影响; 从 P_i 到 P_{i+1} 为减速过程, 受 L_{i+1} 长度的影响.

I) v_i 受 L_i 长度的影响. 若 $L_i > L_{ac}$, 则刀具有足够的加工长度, 刀具在点 P_i 的速度不受 L_i 长度的限制, 即

$$v_i = v_{\text{max}}. \quad (15)$$

若 $L_i < L_{ac}$, 则刀具没有足够的加工长度, 刀具在点 P_i 的速度受 L_i 长度的限制, 即

$$L_i = (v_i + v_{i-1}) \sqrt{\frac{v_i - v_{i-1}}{R}}. \quad (16)$$

刀具由 v_{i-1} 加速到 v_i 的过程由 T_1, T_2 组成, 令 $T_1 = T_2 = t_m$. 由式(4)~(7)可得

$$Rt_m^3 + 2v_{i-1}t_m - L_i = 0. \quad (17)$$

通过卡丹公式求解一元三次方程, 可得

$$t_m = \sqrt[3]{-\frac{q}{2} + \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}} + \sqrt[3]{-\frac{q}{2} - \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}}. \quad (18)$$

式(18)中: $p = 2v_{i-1}/R$; $q = -L_{i-1}/R$.

由式(4)~(7)可得

$$v_i = v_{i-1} + Rt_m^2. \quad (19)$$

由此可得

$$v_i = \begin{cases} v_{\text{max}}, & L_i \geq L_{ac}, \\ v_{i-1} + Rt_m^2, & L_i < L_{ac}. \end{cases} \quad (20)$$

II) v_i 受 L_{i+1} 长度的影响. 同理可得

$$v_i = \begin{cases} v_{\text{max}}, & L_{i+1} \geq L_{ac}, \\ v_{i+1} + Rt_m^2, & L_{i+1} < L_{ac}. \end{cases} \quad (21)$$

式(21)中: $p = 2v_{i+1}/R$; $q = -L_{i+1}/R$.

2.2 自适应前瞻分段规划

前瞻的主要任务是预先获取一定数量的加工轨迹信息, 并对其进行速度规划. 当其他条件不变时,

前瞻段数越多, 则速度规划效果越优。目前, 具有前瞻功能的机床常用的方式是每次前瞻一定数量的轨迹段或规划所有的轨迹段, 但这种做法缓冲区占用的内存较大, 且固定段数的前瞻不适应加工轨迹间速度变化的要求。此外, 因为拐角角度的限制, 有些拐角处的速度极低, 甚至减速为 0, 这些拐角两端加工轨迹处的速度不会相互影响, 所以将两端处轨迹结合前瞻没有意义。为解决这些问题, 采用识别速度敏感点的方法分割前瞻区间, 实现自适应动态前瞻。

拐角速度突变示意图, 如图 5 所示。图 5 中: $v_{A,x}$, $v_{B,x}$ 分别为 v_A , v_B 在 x 轴上的分量; $v_{A,y}$, $v_{B,y}$ 分别为 v_A , v_B 在 y 轴上的分量。当拐角 $\beta < 90^\circ$ 时, 刀具在拐点 P_i 的速度在 y 轴的分量方向在一个周期内发生逆向突变, 此时, y 轴电机需要在非 0 速度下进行反转, 对电机冲击较大, 故此点速度极低。将此点设为速度敏感点, 相邻敏感点之间的加工轨迹为一个独立前瞻区间。

传统的前瞻算法将敏感点的速度设置为 0, 这种做法虽然可避免电机在非 0 速度下进行反转, 但会降低加工效率, 且造成机床频繁启停, 从而引入其他加工误差。因此, 假定刀具从敏感点运动到下一点时, 其速度可以减小为 0, 根据拐角角度和敏感点后一段加工轨迹长度的限制, 计算敏感点安全加工速度, 可得

$$v_{lim} = \min \left\{ \begin{array}{l} \sqrt[3]{L_{next}^2 R}, \\ \frac{a_{max} T_s}{2 \sin(\alpha/2)} \end{array} \right. \quad (22)$$

式(22)中: v_{lim} 为敏感点的安全过渡速度; L_{next} 为敏感点后一段加工轨迹长度。

2.3 前瞻速度规划

完成对加工轨迹的独立前瞻区间分段后, 需要对区间内的轨迹段进行速度规划, 结合前推法和回溯法^[12-18]提出前瞻规划策略。设现有一个独立前瞻区间, 拐点分别为 $P_{i-1} \sim P_{i+5}$; 轨迹段分别为 $L_i \sim L_{i+5}$; 设 α_n 为点 P_n 处拐角, $n=i-1, i, \dots, i+5$; 由 S_{if} 判断点 P_n 是否敏感点, 若为 Y, 表示是敏感点, 若为 N, 表示非敏感点; v_{trans_n} 为点 P_n 处拐角限制的最大速度(式(14)); v_{pre_n} 为点 P_n 前一段加工轨迹长度限制的最大速度(式(20)); v_{next_n} 为点 P_n 后一段加工轨迹长度限制的最大速度(式(21)); v_n 为点 P_n 的最优最大速度。前瞻前将每点的速度初始化, 即 $v_n=0$, S_{if} 为 N。前瞻规划策略有以下 6 个步骤。

步骤 1 从插补轨迹第 1 段开始, 依次计算每个拐角 α_n 。当 $\alpha_n < 90^\circ$ 时, 将 S_{if} 设置为 Y; 否则, 设置为 N。将 2 个相邻 S_{if} 标志为 Y 的点之间的轨迹段设置为 1 个独立区间。从第 1 个独立前瞻区间开始前瞻。

步骤 2 根据式(22)计算终点 P_n 处的最优最大速度 $v_n = v_{lim}$, $n=i+5$ 。

步骤 3 令 $n=n-1$, 若 $n=i-1$, 开始步骤 5; 否则, 向前读取下一点 P_n , 分别计算 v_{trans_n} , v_{next_n} 。如果 $v_{trans_n} > v_{next_n}$, 则 $v_n = v_{next_n}$, 重复该步骤; 如果 $v_{trans_n} < v_{next_n}$, 则 $v_n = v_{trans_n}$ 。令 $n=i$, 开始步骤 4。

步骤 4 若 $v_i > v_{i+1}$, 结束该步骤; 否则, 计算 v_{pre_i+1} , 判断是否 $v_{pre_i+1} > v_{i+1}$, 若是, 则结束该步骤; 否则, $v_{i+1} = v_{pre_i+1}$ 且 $i=i+1$, 重复该步骤, 直至 $v_{pre_i+1} > v_{i+1}$, 开始步骤 3。

步骤 5 令 $n=n+1$, 若 $n=i+5$, 结束该步骤; 否则, 向后读取下一点 P_n , 计算 v_{pre_n} 。若 $v_{pre_n} > v_{next_n}$, 结束该步骤; 若 $v_{pre_n} < v_{next_n}$, 则 $v_n = v_{pre_n}$, 重复该步骤。

步骤 6 该前瞻区间结束, 判断是否前瞻完成, 若完成, 则结束前瞻; 否则, 读取下一前瞻区间, 转入步骤 2。

适用于连续微线段加工的新型前瞻规划算法(文中前瞻规划算法)采用先回溯再前推的方法, 避免前瞻缓冲区过大的问题, 且进一步优化敏感点的速度, 使其不为 0, 可提高加工效率。

3 实验验证

3.1 实验平台的搭建

实验平台包括硬件平台和软件平台。硬件平台由空气开关、电源、工控机(上位机)、CX5020 型下位机(德国倍福自动化有限公司)、伺服电机、伺服驱动器等通过导线和通信线(双绞线)进行连接。实验硬

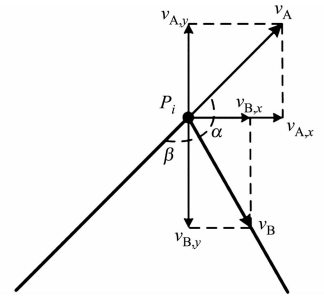


图 5 拐角速度突变示意图
Fig. 5 Diagram of abrupt change of corner speed

件平台,如图 6 所示. 软件平台的开发环境采用基于 Windows 系统的 Visual Studio,编程语言采用面向 Windows 的 Visial C++. 采用 EtherCAT 总线技术实现与伺服驱动器的实时通信,由数控系统计算插补点,发送插补周期给伺服驱动器,进而驱动电机,通过编码器采集速度.

3.2 连续微线段的取样验证

微线段的加工轨迹,如图 7 所示. 设机床最大允许加速度为 $8\,000\text{ mm}\cdot\text{s}^{-2}$,最大允许加加速度为 $80\,000\text{ mm}\cdot\text{s}^{-3}$,插补周期 $T_s=1\text{ ms}$. 在不同机床允许的最大速度(v_{\max} 分别为 $600,800\text{ mm}\cdot\text{s}^{-1}$)下,分别采用传统规划算法、其他前瞻规划算法和文中前瞻规划算法进行对比实验. 其他前瞻规划算法与文中前瞻规划算法衔接点(拐角点)的速度,如表 1 所示.

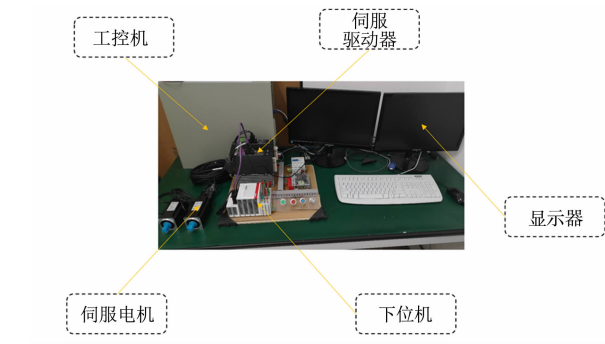


图 6 实验硬件平台
Fig. 6 Experimental hardware platform

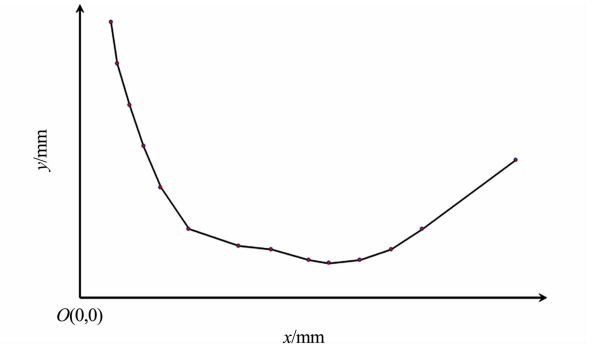


图 7 微线段的加工轨迹
Fig. 7 Machining trace of micro line

表 1 前瞻规划算法衔接点的速度

Tab. 1 Junction points speed of look-ahead planning algorithm

衔接点坐标/mm	v_{trans} (其他前瞻规划算法)/ $\text{mm}\cdot\text{s}^{-1}$		v_{trans} (文中前瞻规划算法)/ $\text{mm}\cdot\text{s}^{-1}$	
	$v_{\max}=600\text{ mm}\cdot\text{s}^{-1}$	$v_{\max}=800\text{ mm}\cdot\text{s}^{-1}$	$v_{\max}=600\text{ mm}\cdot\text{s}^{-1}$	$v_{\max}=800\text{ mm}\cdot\text{s}^{-1}$
(2.0,8.0)	0	0	0	0
(2.4,6.8)	0	0	7.533 91	7.533 91
(3.2,5.6)	18.020 80	26.649 60	29.417 70	29.417 70
(4.1,4.4)	0	0	20.323 90	20.323 90
(5.2,3.2)	0	0	8.324 15	8.324 15
(7.0,2.0)	0	0	4.655 16	4.655 16
(10.2,1.5)	16.656 70	33.313 30	18.628 60	33.313 30
(12.3,1.3)	22.395 10	46.634 80	26.057 60	49.159 50
(14.7,1.1)	21.712 80	28.396 10	25.677 40	31.193 30
(16.0,1.0)	0	0	15.792 10	15.792 10
(18.0,1.1)	0	0	20.224 20	20.224 20
(20.0,1.4)	0	0	14.040 40	14.040 40
(22.0,2.0)	33.011 40	44.344 40	66.022 70	66.022 70
(28.0,4.0)	0	0	0	0

当 $v_{\max}=800\text{ mm}\cdot\text{s}^{-1}$ 时,传统规划算法的插补输出情况,如图 8 所示. 图 8 中: v, v_x, v_y 分别为传统规划算法的插补输出速度(进给速度)及其在 x, y 轴上的分量; a, a_x, a_y 分别为插补输出加速度(进给加速度)及其在 x, y 轴上的分量.

由图 8 可知:采用传统规划算法进行加减速插补时,刀具在各衔接点的速度降为 0,各微线段内独立加减速,因为微线段长度限制,刀具在各段内可达到的最大进给速度很小,加速度波动较大,机床加工效率低下,电机在拐角处频繁启停造成机床震动,影响工件加工质量.

当 $v_{\max}=800\text{ mm}\cdot\text{s}^{-1}$ 时,其他前瞻规划算法的插补输出情况,如图 9 所示. 由图 9 可知:其他前瞻规划算法通过设置拐点,将加工轨迹分为独立区间,并在独立区间内独立加减速,相较于传统规划算法,可实现独立区间内的连续速度规划,提高加工效率,减少电机启停的次数,提高机床的加工精度,但仍不能保证在衔接点处的速度均大于 0. 电机对机床仍有冲击,不利于机床进一步提高速度和精度.

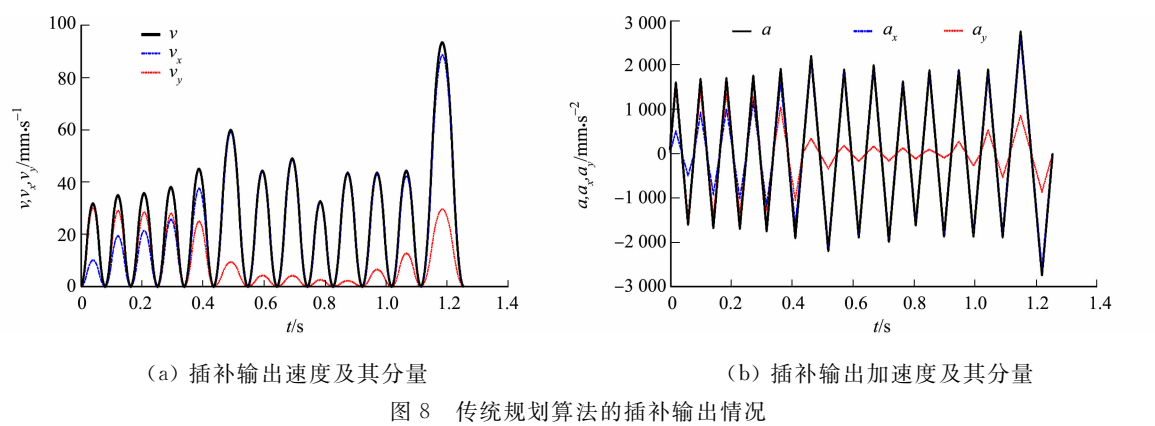


图 8 传统规划算法的插补输出情况

Fig. 8 Interpolation output situation of traditional planning algorithm

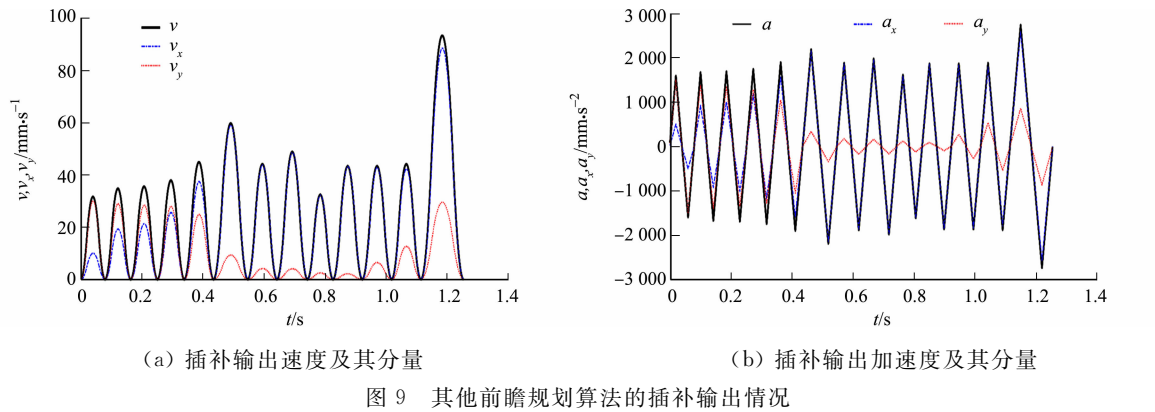


图 9 其他前瞻规划算法的插补输出情况

Fig. 9 Interpolation output situation of other look-ahead planning algorithm

当 $v_{\max}=800\text{ mm}\cdot\text{s}^{-1}$ 时,文中前瞻规划算法的插补输出情况,如图 10 所示.由图 10 可知:采用文中前瞻规划算法对加工轨迹进行规划时,除起点和终点外,衔接点速度均大于 0,且段内可达到的进给速度明显增大,加工效率较高,加速度波动小,机床不容易产生冲击.

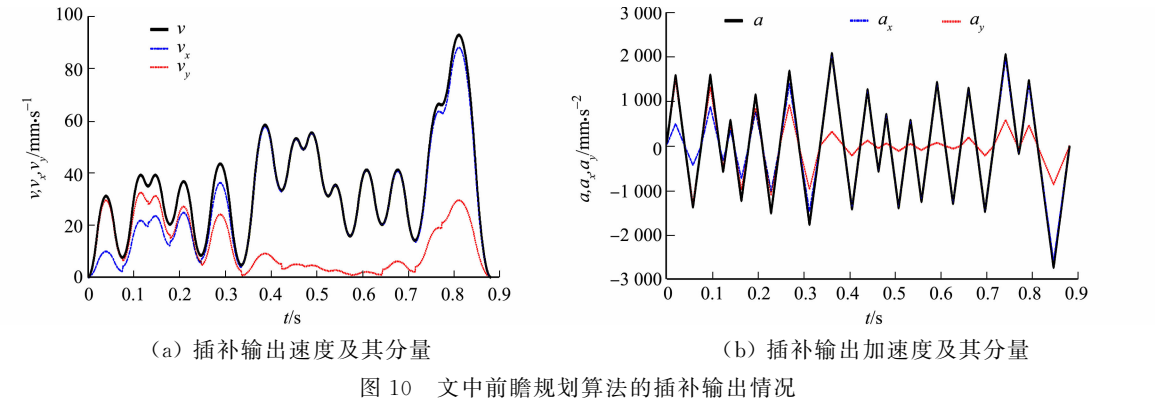


图 10 文中前瞻规划算法的插补输出情况

Fig. 10 Interpolation output situation of proposed look-ahead planning algorithm

3 种规划算法的输出结果,如表 2 所示.由表 2 可知:相较于传统规划算法和其他前瞻规划算法,文中前瞻规划算法可以缩短加工时间.当 $v_{\max}=600\text{ mm}\cdot\text{s}^{-1}$ 时,文中前瞻规划算法的加工效率比传统规划算法提高了 28.6%,比其他前瞻规划算法提高了 8.7%;当 $v_{\max}=800\text{ mm}\cdot\text{s}^{-1}$ 时,文中前瞻规划算法的加工效率比传统规划算法提高了 29.8%,比其他前瞻规划算法提高了 13.7%.

表 2 3 种规划算法的输出结果

Tab. 2 Output results of three algorithms

$v_{\max}/\text{mm}\cdot\text{s}^{-1}$	规划算法	插补周期数/个	衔接点处是否存在冲击
600	传统规划算法	1 380	是
	其他前瞻规划算法	1 080	是
	文中前瞻规划算法	986	否
800	传统规划算法	1 256	是
	其他前瞻规划算法	1 022	是
	文中前瞻规划算法	882	否

4 结束语

提出一种适用于连续微线段加工的新型前瞻规划算法,采用简化的 S 曲线加减速,减少计算时间,通过识别速度敏感点自适应调整前瞻区间,解决了传统前瞻算法前瞻段数过多的问题;进一步优化前瞻区间的终点速度,使其大于 0. 相较于其他算法,文中前瞻规划算法可进一步提高衔接点速度、加工平稳性及加工效率.

参考文献:

- [1] 李浩,吴文江,韩文业,等. 基于自适应前瞻和预测校正的实时柔性加减速控制算法[J]. 中国机械工程,2019,30(6):690-699. DOI:10.3969/j.issn.1004-132X.2019.06.010.
- [2] 陈琳,黄旭丰,刘梦,等. 综合多约束条件优化连续轨迹前瞻算法[J]. 机械工程学报,2019,55(13):151-159. DOI:10.3901/JME.2019.13.151.
- [3] 陈日东,李迪,王世勇,等. 连续轨迹动态前瞻插补算法研究[J]. 电子设计工程,2017,25(7):152-155,160. DOI:10.14022/j.cnki.dzsjgc.2017.07.037.
- [4] 王耀庭,张秋菊,成津赛. 高速嵌入式数控系统速度前瞻控制算法的研究[J]. 机械科学与技术,2014,33(12):1850-1853. DOI:10.13433/j.cnki.1003-8728.2014.1218.
- [5] 钟前进,王科,丁信忠. 一种新型 S 加减速算法研究[J]. 电气传动,2019,49(6):8-12,37. DOI:10.19457/j.1001-2095.dqcd18616.
- [6] 司慧晓,吴文江,李浩. 基于圆弧过渡的五段 S 型加减速算法[J]. 组合机床与自动化加工技术,2017(3):6-9,13. DOI:10.13462/j.cnki.mmtamt.2017.03.002.
- [7] 王海涛. 数控系统速度前瞻控制算法及其实现[D]. 南京:南京航空航天大学,2011.
- [8] 许东伟. 多轴数控系统的运动控制算法研究与开发[D]. 广州:广东工业大学,2015.
- [9] LEE C S. Generation of velocity profiles with speed limit of each axis for high-speed machining using look-ahead buffer[J]. International Journal of Precision Engineering and Manufacturing,2010,11(2):201-208. DOI:10.1007/s12541-010-0023-2.
- [10] ERKORKMAZ K,ALTINTAS Y. High speed CNC system design (Part I): Jerk limited trajectory generation and quintic spline interpolation[J]. International Journal of Machine Tools and Manufacture,2001,41(9):1323-1345.
- [11] 李志杰,蔡力钢,刘志峰. 加加速度连续的 S 型加减速规划算法[J]. 计算机集成制造系统,2019,25(5):1192-1201.
- [12] 李淑梅,谢明红. 高速高精度数控系统速度平稳控制策略[J]. 华侨大学学报(自然科学版),2016,37(4):411-415. DOI:10.11830/ISSN.1000-5013.201604004.
- [13] 李淑梅. 高速高精度五轴数控系统前瞻算法研究[D]. 厦门:华侨大学,2016.
- [14] 王旭浩,张华. 多轨迹段平滑过渡的前瞻插补算法[J]. 计算机系统应用,2020,29(4):118-125. DOI:10.15888/j.cnki.csa.007350.
- [15] 朱长峰. 连续微线段高速加工数控系统路径与速度前瞻规划算法研究[D]. 杭州:浙江大学,2018.
- [16] HAO Li,WU Wenjiang,RASTEGAR J,et al. A real-time and look-ahead interpolation algorithm with axial jerk-smooth transition scheme for computer numerical control machining of micro-line segments[J]. Journal of Engineering Manufacture,2019,233(9):2007-2019. DOI:10.1177/0954405418809768.
- [17] 金永乔. 微小线段高速加工的轨迹优化建模及前瞻插补技术研究[D]. 上海:上海交通大学,2015.
- [18] 杨珧. 高速加工数控系统 NURBS 曲线前瞻直接插补关键算法研究与实现[D]. 杭州:浙江大学,2019.

(责任编辑:钱筠 英文审校:崔长彩)