

doi:10.19306/j.cnki.2095-8110.2016.06.016

## 基于 ROS 的机械臂轨迹规划研究

刘汝佳,王芳,张强,李楠  
(航天科工智能机器人有限责任公司,北京 100074)

**摘要:**针对机械臂轨迹规划中算法复杂、开发难度高的技术难点,提出基于一款开源软件平台(Robot Operating System, ROS)为机械臂搭建控制系统的方法作为解决方案。在此基础上,设计了一种基于五次多项式插值的算法来弥补该开源软件平台中关于轨迹规划功能的不足,并通过仿真实验验证了改进后的控制算法可以提高机械臂控制性能。该方法对于实现工程机械臂轨迹规划问题具有普遍的指导意义。

**关键词:**机器人;机械臂;ROS;运动控制;轨迹规划

**中图分类号:**TP242.6

**文献标志码:**A

**文章编号:**2095-8110(2016)06-0082-07

### Research on Trajectory Planning of ROS-based Robot Arm

LIU Ru-jia, WANG Fang, ZHANG Qiang, LI Nan  
(Aerospace Science & Industry Intelligence Robot Co. Ltd., Beijing 100074, China)

**Abstract:**In case of the technical difficulties of the development in manipulator trajectory planning, a solution that builds a robotic arm control system based on an open source software platform (Robot Operating System, ROS) is put forward here. On this basis, an algorithm based on quintic polynomial interpolation is presented to cover the shortage of ROS algorithm on trajectory planning functions, and the simulation results show that the new algorithms can improve the control function of manipulator arm. And the method provides a reference to engineering manipulator arm trajectory planning.

**Key words:**Robot; Manipulator arm; ROS; Motion control; Trajectory planning

## 0 引言

机械臂在工业生产中有着广泛的应用,例如使用机械臂来搬运货物、组装产品等。机械臂技术中普遍遇到的难题之一是高精度的轨迹规划<sup>[1-3]</sup>,这也是机械臂技术研究中最重要的一环。传统的研究机械臂轨迹规划问题的方法需要根据每一款具体的臂搭建符合实际情况的控制系统<sup>[4-5]</sup>,涉及末端轨迹规划、运动学逆解、干涉与约束算法以及关节细插值等过程,这一过程工作量大、耗时长,难以在高效率的现代社会普遍应用。如今机器人软件开发平台的快速发展为解决这一技术难题提供了新的思路<sup>[6-7]</sup>。

所谓的机器人软件开发平台,是指用来给多种机器人设备开发程序的软件包集。它一般具备下列特点:

- 1)统一的编程、编译环境;
- 2)可重用的组件库;
- 3)具备一定的仿真、模拟条件;
- 4)对多款机器人设备的硬件提供驱动支持(留有二次开发驱动的接口);
- 5)包含一些常用的、模块化的控制功能组件,例如导航技术、计算机视觉技术、机械臂运动控制等。

现在常用的机器人软件平台如表1所示。

表1 机器人软件开发平台  
Tab.1 The robotics software development platform

| 软件平台                      | 所属类型     |
|---------------------------|----------|
| Evolution Robotics ERSP   | 商用       |
| Microsoft Robotics Studio | 非商业用途免费  |
| OROCOS                    | 控件库,开源免费 |

收稿日期:2015-10-28;修订日期:2016-01-10。

作者简介:刘汝佳(1990-),男,硕士,主要从事机器人运动控制与软件设计方面的研究。

E-mail:sansherlock@126.com

续表

| 软件平台                  | 所属类型      |
|-----------------------|-----------|
| Skilligent            | 机器学习插件,商用 |
| OpenJAUS              | 开源        |
| ROS                   | 开源免费      |
| URBI                  | 商用        |
| Webots                | 仿真环境,商用   |
| Player, Stage, Gazebo | 开源免费      |
| iRobot AWARE          | 商用        |
| CLARAty               | 开源        |
| LabVIEW Robotics      | 商用        |

本文中选用的开发平台是由 Willow Garage 公司开发的机器人操作系统 ROS (Robot Operating System)。ROS 在构建机器人控制系统方面非常便利,具有兼容性高、开源算法完备的特点<sup>[8-10]</sup>,是目前主流的机器人开发平台之一。

本文在研究机械臂轨迹规划问题中充分应用机器人软件开发平台的便利特性和开源成果,并在此基础上通过二次开发,改进了控制算法,对实现机械臂的平稳控制具有重要意义。

## 1 机械臂轨迹规划研究

### 1.1 机械臂控制系统

图 1 所示为基于 ROS 为机械臂搭建的运动控制系统框图。上位机的操作系统运行 ROS 软件开发平台,主要实现机器人运动学解算、位置插补以及轨迹规划等工作。驱动接口将 ROS 输出的数据通过 CAN 总线发送给驱动器,由驱动器驱动电机转动完成机械臂的运动。机械臂上的传感器将采集的位置、速度等信息实时反馈给上位机,通过闭环控制,实现对机械臂的精度控制。

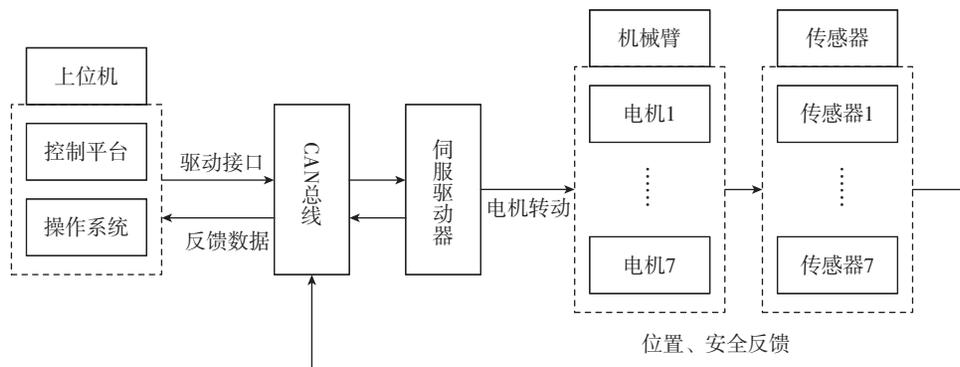


图 1 机器人运动控制系统框图

Fig. 1 The principle diagram of robot control system

### 1.2 ROS 实现机械臂轨迹规划及其存在的问题

机械臂包含两种类型的运动方式<sup>[11]</sup>。一是点对点的运动:只考虑机械臂在笛卡尔空间从起始位姿移动到目标位姿,不考虑运动过程中的位姿;二是轨迹运动:需要精确规划机械臂所走的路径。例如机械臂末端沿一条直线移动需要其姿态在移动过程中保持不变,而位置在笛卡尔空间中是一条直线。机械臂轨迹运动下的位姿显然是关于时间的函数。

无论让机械臂以何种方式运动,开发者都需要得到在关节空间中解算的一系列关节角度插值。要完成这一任务不仅要用到机器人逆运动学解,还要考虑机械臂在运动过程中的碰撞检测、轨迹规划中的运动约束(比如始终保持末端执行器的姿态不变)以及机械臂运动平稳性等问题。

MoveIt 是 ROS 一个重量级的应用模块<sup>[12]</sup>,它包含了机械臂运动控制的核心功能。

本文所研究的机械臂模型是一条 7 自由度的串联机械臂。应用 MoveIt 模块可以快速完成三方面的工作:机械臂建模、运动学解算和路径规划。MoveIt 在运动学解算以及路径规划过程中,会将各种运动约束的因素也考虑进来。例如防碰撞检测、设定关节最大运动角度和角速度等。但是 MoveIt 在仿真环境下输出的轨迹规划数据只包含 7 个关节的角度插值数据,没有角速度插值,并且输出周期为 500ms。驱动器根据此数据会驱动电机在该周期内以指定恒速到达指定位置,这样会导致位置与时间的不同步,末端精度出现较大误差。图 2 所示为机械臂跟踪 X-Y 平面内一条直线情况下关节空间内 4 号关节角度插值输出图。图 3 所示为机械臂跟踪

X-Y平面内一条直线情况下笛卡尔空间内机械臂末端运动轨迹图。

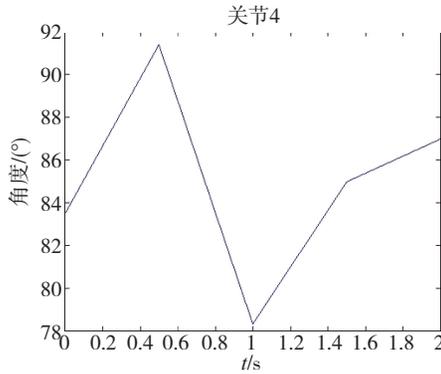


图2 关节4角度插值

Fig. 2 The angle interpolation of joint 4

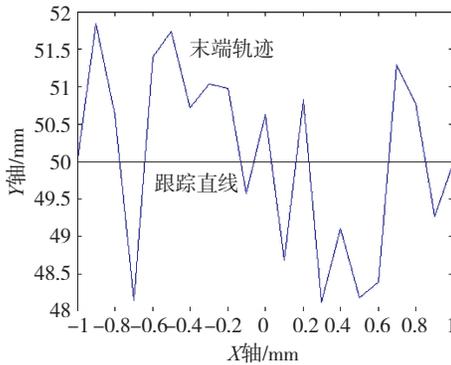


图3 跟踪直线末端轨迹

Fig. 3 The path of tracking straight

由图2和图3可以看出,由于 MoveIt 输出的关节角度插值精度低,并且没有速度插值数据(驱动器采用的是默认的初始关节速度),导致机械臂末端跟踪误差大。为了解决这一问题,我们需要在规划输出数据和驱动器之间加入一个处理环。该处理环的作用是以更小的控制周期进行数学插值,减小机械臂末端跟踪误差,同时也可以降低电机的加速度,防止在运行过程中由于频繁大幅的加减速而导致的机械臂抖动问题,如图4所示。

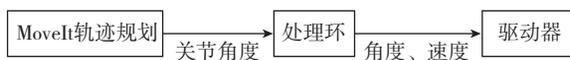


图4 轨迹规划输出示意图

Fig. 4 The output of the trajectory planning

### 1.3 改进的控制算法

#### 1.3.1 控制流程

在 MoveIt 输出的轨迹角度插值的基础上,本文

设计了五次多项式插值算法将其角度插值更加细分,并可以得到连续的角速度插值。该方法可以将仿真得到的数据直接输出给驱动器使用。完整的运动控制流程图如图5所示。

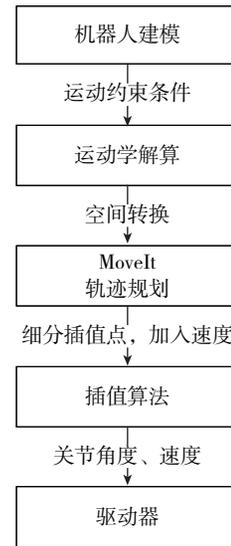


图5 运动控制流程图

Fig. 5 The flow diagram of motion control

#### 1.3.2 插值算法

设想这样的场景: MoveIt 轨迹规划库函数计算输出离散的关节角度插值,每一个插值点都可以看做是一个路径点;每个机械臂关节依次通过所有路径点,映射成不同关节关于时间的一段段连续的光滑曲线。不同关节的时间函数曲线是相互独立的,但运动周期与总的运动时间是相同的。

机械臂平稳的运动需要保证其关节角速度是连续可导的,并且角加速度是连续的。因此,可以构造五次多项式插值函数来模拟上述情景中的每段连续的时间函数。求解五次多项式方程组需要6个已知条件:分别为不同关节的每段时间函数的起始点  $t_i$  (其中,  $i \in (1, N)$ ,  $N$  是 MoveIt 输出的关节插值数目) 时刻和目标点  $t_{i+1}$  时刻的角度、速度和加速度。

任意  $t_{i+1}$  时刻的角度  $\theta_{i+1}$  就是 MoveIt 输出的路径点;  $t_{i+1}$  时刻的瞬时角速度由相邻路径点角度差除以运动周期代替(其中,  $\Delta t$  为控制周期)

$$\dot{\theta}_{i+1} = \frac{(\theta_{i+1} - \theta_i)}{\Delta t} \quad (1)$$

$t_{i+1}$  时刻的瞬时角加速度由相邻中间点角速度差除以运动周期代替

$$\ddot{\theta}_{t_{i+1}} = \frac{(\dot{\theta}_{t_{i+1}} - \dot{\theta}_{t_i})}{\Delta t} \quad (2)$$

以此便可以根据已知的起始点和目标点的6个条件求解五次多项式方程组系数。通过五次多项式插值进一步细分关节角度和角速度插值并将其发送给驱动器。

五次多项式及其一次、二次导函数通式如下所示。其中  $\theta_t, \dot{\theta}_t, \ddot{\theta}_t$  分别是  $t$  时刻的关节角度、角速度和角加速度;  $C_1, C_2, \dots, C_6$  是多项式待求系数。

$$\theta_t = C_1 t^5 + C_2 t^4 + C_3 t^3 + C_4 t^2 + C_5 t + C_6 \quad (3)$$

$$\dot{\theta}_t = 5C_1 t^4 + 4C_2 t^3 + C_3 t^2 + 2C_4 t + C_5 \quad (4)$$

$$\ddot{\theta}_t = 20C_1 t^3 + 12C_2 t^2 + 6C_3 t + 2C_4 \quad (5)$$

依据上述通式以及对起始点和目标点的约束条件,列出6个系数方程式,如下所示。式中,  $\theta_{t_i}, \theta_{t_{i+1}}, \dot{\theta}_{t_i}, \dot{\theta}_{t_{i+1}}, \ddot{\theta}_{t_i}, \ddot{\theta}_{t_{i+1}}$  分别表示起始点和目标点的关节角、角速度和角加速度:

$$\theta_{t_i} = C_1 t_i^5 + C_2 t_i^4 + C_3 t_i^3 + C_4 t_i^2 + C_5 t_i + C_6 \quad (6)$$

$$\theta_{t_{i+1}} = C_1 t_{i+1}^5 + C_2 t_{i+1}^4 + C_3 t_{i+1}^3 + C_4 t_{i+1}^2 + C_5 t_{i+1} + C_6 \quad (7)$$

$$\dot{\theta}_{t_i} = 5C_1 t_i^4 + 4C_2 t_i^3 + 3C_3 t_i^2 + 2C_4 t_i + C_5 \quad (8)$$

$$\dot{\theta}_{t_{i+1}} = 5C_1 t_{i+1}^4 + 4C_2 t_{i+1}^3 + 3C_3 t_{i+1}^2 + 2C_4 t_{i+1} + C_5 \quad (9)$$

$$\ddot{\theta}_{t_i} = 20C_1 t_i^3 + 12C_2 t_i^2 + 6C_3 t_i + 2C_4 \quad (10)$$

$$\ddot{\theta}_{t_{i+1}} = 20C_1 t_{i+1}^3 + 12C_2 t_{i+1}^2 + 6C_3 t_{i+1} + 2C_4 \quad (11)$$

根据克莱姆法则,解得系数通式如下所示:

$$C_1 = \frac{12(\theta_{t_{i+1}} - \theta_{t_i}) - 6(\dot{\theta}_{t_{i+1}} + \dot{\theta}_{t_i})(t_{i+1} - t_i)}{2(t_{i+1} - t_i)^5} - \frac{(\ddot{\theta}_{t_i} - \ddot{\theta}_{t_{i+1}})(t_{i+1}^2 + t_i^2) - 2(\ddot{\theta}_{t_{i+1}} - \ddot{\theta}_{t_i})t_i t_{i+1}}{2(t_{i+1} - t_i)^5} \quad (12)$$

$$C_2 = \frac{30(\theta_{t_i} - \theta_{t_{i+1}})(t_i + t_{i+1}) + (14\dot{\theta}_{t_{i+1}} + 16\dot{\theta}_{t_i})(t_{i+1}^2 - t_i^2)}{2(t_{i+1} - t_i)^5} + \frac{(3\ddot{\theta}_{t_i} - 2\ddot{\theta}_{t_{i+1}})(t_{i+1}^2 - t_i^2) + [(4\ddot{\theta}_{t_{i+1}} - \ddot{\theta}_{t_i})t_i]}{2(t_{i+1} - t_i)^5} + \frac{(\ddot{\theta}_{t_{i+1}} - 4\ddot{\theta}_{t_i})t_{i+1} + 2(\dot{\theta}_{t_{i+1}} - \dot{\theta}_{t_i})t_i t_{i+1}}{2(t_{i+1} - t_i)^5} \quad (13)$$

$$C_3 = \frac{20(\theta_{t_{i+1}} - \theta_{t_i})(t_{i+1}^2 - t_i^2) - (8\dot{\theta}_{t_i} + 12\dot{\theta}_{t_{i+1}})(t_{i+1}^3 - t_i^3)}{2(t_{i+1} - t_i)^5} + \frac{(3\ddot{\theta}_{t_{i+1}} - \ddot{\theta}_{t_i})t_i^4 - (3\ddot{\theta}_{t_i} - \ddot{\theta}_{t_{i+1}})t_{i+1}^4 - (28\dot{\theta}_{t_i} + 32\dot{\theta}_{t_{i+1}})t_i t_{i+1}^2}{2(t_{i+1} - t_i)^5} + \frac{(32\dot{\theta}_{t_i} + 28\dot{\theta}_{t_{i+1}})t_i^2 t_{i+1} + 4\ddot{\theta}_{t_{i+1}} t_i t_{i+1}^3 - 4\ddot{\theta}_{t_i} t_i^3 t_{i+1}}{2(t_{i+1} - t_i)^5} + \frac{8(\dot{\theta}_{t_i} - \dot{\theta}_{t_{i+1}})t_i^2 t_{i+1}^2 + 80(\theta_{t_{i+1}} - \theta_{t_i})t_i t_{i+1}}{2(t_{i+1} - t_i)^5} \quad (14)$$

$$C_4 = \frac{\ddot{\theta}_{t_i} t_{i+1}^5 - \ddot{\theta}_{t_{i+1}} t_i^5 + 60(\theta_{t_i} - \theta_{t_{i+1}})(t_i + t_{i+1})t_i t_{i+1}}{2(t_{i+1} - t_i)^5} + \frac{(24\dot{\theta}_{t_{i+1}} + 36\dot{\theta}_{t_i})t_i t_{i+1}^3 - (36\dot{\theta}_{t_{i+1}} + 24\dot{\theta}_{t_i})t_i^3 t_{i+1}}{2(t_{i+1} - t_i)^5} + \frac{(4\ddot{\theta}_{t_i} - 3\ddot{\theta}_{t_{i+1}})t_i t_{i+1}^4 - (4\ddot{\theta}_{t_{i+1}} - 3\ddot{\theta}_{t_i})t_i^4 t_{i+1}}{2(t_{i+1} - t_i)^5} - \frac{(12\dot{\theta}_{t_i} - 12\dot{\theta}_{t_{i+1}} + 8\ddot{\theta}_{t_i} t_{i+1} - 8\ddot{\theta}_{t_{i+1}} t_i)t_i^2 t_{i+1}^2}{2(t_{i+1} - t_i)^5} \quad (15)$$

$$C_5 = \frac{2\dot{\theta}_{t_i} t_{i+1}^5 - 2\dot{\theta}_{t_{i+1}} t_i^5 - 10\dot{\theta}_{t_i} t_i t_{i+1}^4 + 10\dot{\theta}_{t_{i+1}} t_{i+1}^4 t_i}{2(t_{i+1} - t_i)^5} + \frac{2\ddot{\theta}_{t_i} t_i t_{i+1}^5 + 2\ddot{\theta}_{t_{i+1}} t_i^5 t_{i+1} + [60(\theta_{t_{i+1}} - \theta_{t_i})]}{2(t_{i+1} - t_i)^5} - \frac{(24\dot{\theta}_{t_{i+1}} + 16\dot{\theta}_{t_i})t_{i+1} + (16\dot{\theta}_{t_{i+1}} + 24\dot{\theta}_{t_i})t_i}{2(t_{i+1} - t_i)^5} t_i^2 t_{i+1}^2 + \frac{[(3\ddot{\theta}_{t_{i+1}} + \ddot{\theta}_{t_i})t_{i+1}^2 - (\ddot{\theta}_{t_{i+1}} + 3\ddot{\theta}_{t_i})t_i^2]}{2(t_{i+1} - t_i)^5} +$$

$$\frac{4(\ddot{\theta}_{t_i} - \ddot{\theta}_{t_{i+1}})t_i t_{i+1}^2 t_i^2 t_{i+1}^2}{2(t_{i+1} - t_i)^5} \quad (16)$$

$$C_6 = \frac{2\theta_{t_i} t_{i+1}^5 - 2\theta_{t_{i+1}} t_i^5 - 10\theta_{t_i} t_i t_{i+1}^4 + 10\theta_{t_{i+1}} t_{i+1}^4 t_i}{2(t_{i+1} - t_i)^5} + \frac{2\dot{\theta}_{t_i} t_i t_{i+1}^5 + 2\dot{\theta}_{t_{i+1}} t_i^5 t_{i+1} + [20\theta_{t_{i+1}} - 20\theta_{t_i}]t_i}{2(t_{i+1} - t_i)^5} +$$

$$\frac{10\dot{\theta}_i t_{i+1}^2 - 10\dot{\theta}_{i+1} t_i^2 + 8(\dot{\theta}_{i+1} - \dot{\theta}_i)t_i t_{i+1}}{2(t_{i+1} - t_i)^5} + \frac{[\ddot{\theta}_i t_{i+1}^3 - \ddot{\theta}_{i+1} t_i^3 - (2\ddot{\theta}_i + \ddot{\theta}_{i+1})t_i t_{i+1}]}{2(t_{i+1} - t_i)^5} + \frac{(2\ddot{\theta}_{i+1} + \ddot{\theta}_i)t_i^2 t_{i+1}}{2(t_{i+1} - t_i)^5} \quad (17)$$

将此系数通式带入到原方程式(6)~式(11)中,便可以求得7个关节在每段插值内的时间函数。

1.4 插值算法软件实现

基于上述算法控制流程图,便可以编写软件来实现机械臂的轨迹运动功能。设定机械臂运动的起始点和目标点,通过计算机解算获得在关节空间运动的参数;计算机判断机械臂是否运动到目标点;当机械臂运动到目标点,计算机停止发送运动指令。软件流程如图6所示。

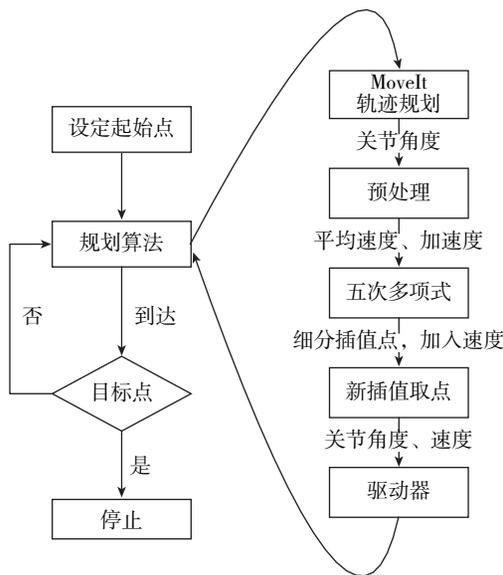
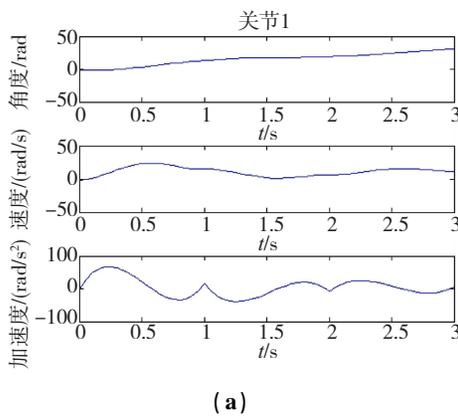


图6 软件设计流程图

Fig. 6 The flow diagram of software design



(a)

2 仿真

通过仿真实验对五次多项式的插值效果进行研究验证。机械臂右臂在ROS仿真环境下按照图7所示圆弧轨迹运动。

利用 MoveIt 轨迹规划函数解算出一组7个关节的插值点,如表2所示。

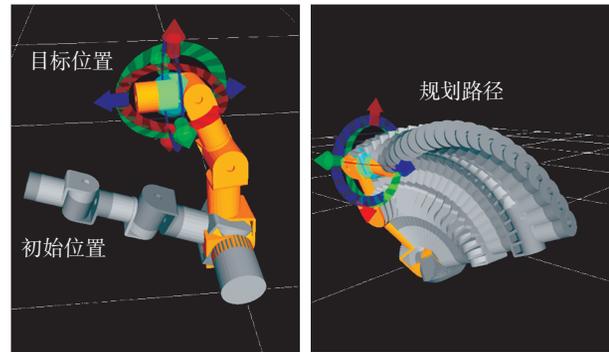


图7 机械臂运动仿真

Fig. 7 The robot arm motion simulation

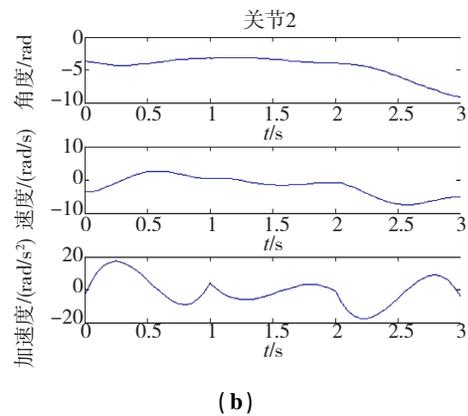
表2 关节插值点

Tab. 2 The joint interpolation

|      | $\theta_1$ | $\theta_2$ | $\theta_3$ | $\theta_4$ | $\theta_5$ | $\theta_6$ | $\theta_7$ |
|------|------------|------------|------------|------------|------------|------------|------------|
| 起始点  | -1.7       | -3.6       | -1.2       | 83.5       | -82.4      | 82.5       | 0          |
| 目标点1 | 13.0       | -3.2       | 9.8        | 74.4       | -88.2      | 82.2       | 0          |
| 目标点2 | 19.2       | -4.0       | 13.2       | 64.3       | -85.2      | 75.1       | 0          |
| 终止点  | 31         | -9.2       | 17.6       | 46.8       | -76.8      | 69.2       | 0          |

基于表2用五次多项式每50ms完成一次插值运算并使用 Matlab 绘制曲线。前6个关节角度、角速度、角加速度如图8所示(关节7在本例中没有运动,曲线省略)。

机械臂末端在此规划数值下沿圆弧剖面跟踪轨迹如图9所示。



(b)

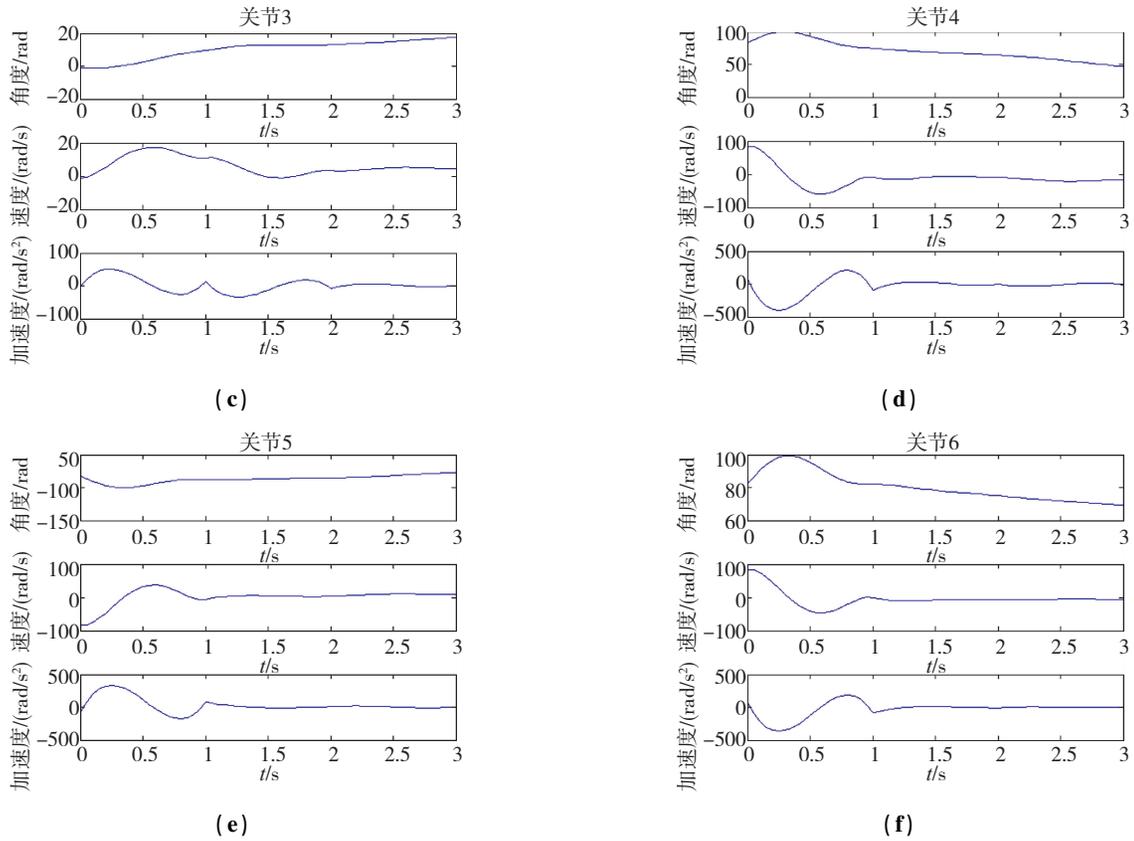


图 8 五次多项式插值关节角度、角速度、角加速度

Fig. 8 The angle, angular velocity and angular acceleration of five polynomial interpolation

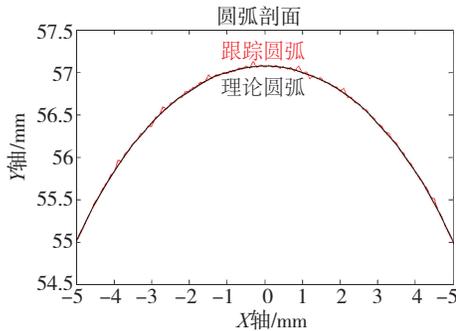


图 9 跟踪圆弧末端轨迹

Fig. 9 The path of tracking circle

从图 8 中可以看出,多项式插补后的角度、角速度和角加速度曲线都是连续、可导的,这样会减小运动过程中由于机械臂电机起停等因素导致的冲击,延长电机寿命,增强机械臂在运动过程中的平稳性。

在添加了新的插值算法之后,从图 9 可以看出,机械臂的跟踪圆弧与理论圆弧的距离误差最大不超过 0.1mm,并且运动过程中不会产出累积误差而导致机械臂无法到达指定目标点。对运动仿真模拟的结果验证了五次多项式插值算法,保证了机械

臂轨迹运动的高精度。

### 3 结论

本文基于 ROS 平台进行机械臂轨迹规划的二次开发,通过对轨迹跟踪算法的完善,将插补速度加入,保证了机械臂轨迹规划的跟踪精度,提高了机械臂运动的平稳性,并且通过仿真实验验证了该算法对于机械臂轨迹跟踪优化的有效性,后续将在机械臂平台进行相应的实际实验验证。同时,本文所做的研究表明基于 ROS 软件开发平台为机械臂快速搭建控制系统的方法提高了开发效率,减小了开发难度,为机械臂开发人员提供了一套可供参考的方案。

#### 参考文献

[1] Wen-Fu X U, Wang X Q, Xue Q, et al. Study on trajectory planning of dual-arm space robot keeping the base stabilized [J]. Acta Automatica Sinica, 2013, 39(1):69-80.

[2] Tang X, Shao Z. Trajectory generation and tracking control of a multi-level hybrid support manipulator in FAST [J]. Mechatronics, 2013, 23(8):1113-1122.

[3] 郑利红, 荣国宏, 李元宗. 基于改进的 Khatib 方法的冗余机器人手避障轨迹规划[J]. 太原理工大学学报, 2015, 32(1):

- 29-32.
- [4] 刘华山. 旋转关节串联机器人多轴运动控制技术研究[D]. 杭州:浙江大学, 2010.
- [5] 秦锋, 徐向荣. 基于样条函数法的机器人运动轨迹规划[J]. 机械科学与技术, 1996, 15(4):561-565.
- [6] Metta G, Sandini G, Vernon D, et al. The iCub humanoid robot: an open platform for research in embodied cognition[C]// Proceedings of the 8<sup>th</sup> Workshop on Performance Metrics for Intelligent Systems. ACM, 2008:50-56.
- [7] Ha I, Tamura Y, Asama H. Development of open platform humanoid robot DARwIn-OP [J]. Advanced Robotics, 2013, 27(3):617-628.
- [8] 曹正万, 平雪良, 陈盛龙, 等. 基于ROS的机器人模型构建方法研究[J]. 组合机床与自动化加工技术, 2015(8):51-54.
- [9] Hart S, Dinh P, Hambuchen K. The affordance template ROS package for robot task programming[C]// 2015 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2015: 6227-6234.
- [10] Tsarouchi P, Makris S, Michalos G, et al. ROS based coordination of human robot cooperative assembly tasks-an industrial case study[J]. Procedia CIRP, 2015, 37:254-259.
- [11] 马江. 六自由度机械臂控制系统设计与运动学仿真[D]. 北京:北京工业大学, 2009.
- [12] Coleman D, Sucas I, Chitta S, et al. Reducing the barrier to entry of complex robotic software: a moveit! case study[J]. Eprint Arxiv, 2014.
- [13] Okada Kei. ROS(Robot Operating System) [J]. Journal of the Robotics Society of Japan, 2012, 30(9):830-835.
- [14] 蔡自兴, 徐光祐. 人工智能机器应用 [M]. 北京:清华大学出版社, 2010.
- [15] 张建伟, 张立伟, 胡颖, 等. 开源机器人操作系统—ROS [M]. 北京:科学出版社, 2012.