

基于 EtherCAT 的 6-UPS 并联机器人控制系统设计

周亮¹,刘凯¹,吴洪涛¹,蔡宝玉²,陈柏¹

(1. 南京航空航天大学 机电学院,江苏 南京 210016;

2. 江苏泰源数控机床有限公司,江苏 泰州 225402)

摘要:针对机器人控制系统领域代码复用率低、拓展性差的问题,以 6-UPS 并联机器人为研究对象,采用开源机器人控制系统 ROS 为基本框架,借助 IGH EtherCAT 主站实现 ROS 系统与机器人之间的通信,在 Gazebo-ROS 中建立机器人的仿真模型,通过 QT-ROS 开发相应的人机界面,根据机器人运动学模型编写运动学节点、轨迹规划节点,实现机器人的仿真与实际控制并对底层驱动系统进行实时同步性测试。实验结果表明:控制系统的实时性和稳定性误差均在 μs 数量级,满足机器人控制系统的需求;通过设置中间路径点,测试仿真机器人以及真实机器人的运动情况,轨迹误差保持在一个脉冲当量以内,具有较高的控制精度。所设计的机器人控制系统可以有效地实现机器人的控制。

关键词:机器人控制系统;ROS;EtherCAT;并联机器人

中图分类号:TP242 **文献标志码:**A **文章编号:**1671-5276(2023)04-0156-04

Design of EtherCAT-based 6-UPS Parallel Robot Control System

ZHOU Liang¹, LIU Kai¹, WU Hongtao¹, CAI Baoyu², CHEN Bai¹

(1. College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; 2. Jiangsu Taiyuan CNC Machine Tool Co., Ltd., Taizhou 225402, China)

Abstract: To address the current problems of low code reuse and poor expandability in the field of robot control systems, with the 6-UPS parallel robot as the research object, and by adopting the open source robot control system ROS as the basic framework, the simulation model of the robot was established in Gazebo-ROS by realizing the communication between the ROS system and the robot based on the IGH EtherCAT master. The QT-ROS was used to develop the corresponding human-machine interface, and according to the kinematic model of the robot, kinematic nodes and trajectory planning nodes were written, thus the simulation and actual control of the robot was realized and the underlying drive system was tested for real time and synchronisation. The experimental results showed that the real-time and stability errors of the control system were within μs order of magnitude and met the requirements of the robot control system. The motions of the simulated robot and the real robot were tested by setting intermediate path points, showing that the trajectory error was kept within one pulse equivalent with high control accuracy.

Keywords: robot control system; ROS; EtherCAT; parallel robot

0 引言

近几十年以来,随着机器人领域的快速发展,机器人所携带的硬件设备也逐渐增多,若从头开始设计机器人的各个功能模块就是一个重复造轮子的过程^[1]。2010 年 Willow Garage 公司发布了开源机器人操作系统 ROS,该系统提供了硬件抽象、底层设备控制、进程间信息传递和包管理的功能,各个功能包负责不同的模块,采用松散耦合的方式进行通信,提高了机器人开发过程中的代码复用率。

在机器人控制系统领域,随着工业 4.0 的到来,对机器人有了更高的网络化、稳定性、实时性要求。跟随时代的需要,工业以太网协议的出现很好地解决了这个问题。市场上常见的工业以太网协议有 EtherCAT、ProfiNet、

Ethernet/IP、ModBusTCP 等,其中德国倍福公司的 EtherCAT 因其良好的性能以及开放性,现已得到了广大驱动器厂商的支持,在机器人控制器通信领域也得到了广泛的应用。

针对目前传统机器人控制系统适用面窄,代码复用率低、拓展性差的问题,本文以 6-UPS 并联机器人为研究对象,采用 ROS 机器人操作系统为基本框架,通过对 Linux 进行实时化改造,以 EtherCAT 总线作为通信方式,设计开发了一种开放性、高性能、低成本的机器人控制系统。

1 控制系统组成

1.1 控制系统硬件组成

控制系统硬件采用支持 EtherCAT 总线的松下 A6B 伺

基金项目:国家自然科学基金资助项目(51375230)

第一作者简介:周亮(1997—),男,云南丽江人,硕士研究生,研究方向为机器人控制技术,nuaa_zhou@126.com。

伺服驱动器、倍福耦合模块和 IO 模块、研华 610L 工控机以及 6-UPS 并联机构,整个系统的硬件结构如图 1 所示。

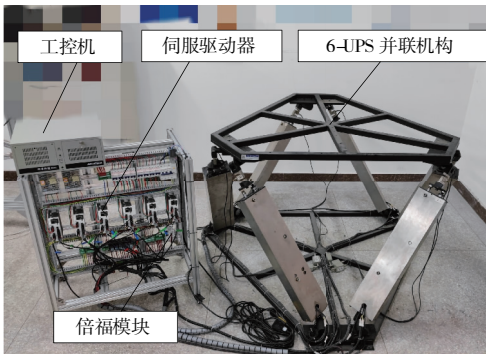


图 1 6-UPS 并联机器人

内核架构如图 3 所示。

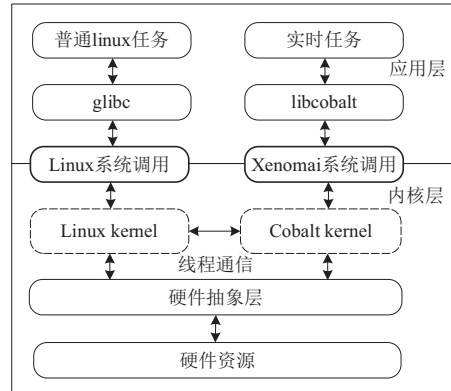


图 3 双内核系统架构

1.2 控制系统软件组成

控制系统软件框架基于 ROS 机器人控制系统,整个系统框架分为应用层、服务层、驱动层以及设备层。应用层采用 QT_ROS 开发机器人的控制系统界面;服务层主要包括机器人的运动学节点、传感器信息处理节点以及手柄控制等节点;驱动层为物理机器人和虚拟机器人的驱动节点;设备层主要包括机器人本体和仿真机器人,机器人本体通过 EtherCAT 总线进行驱动控制,仿真机器人通过自定义的 Gazebo 插件进行驱动控制。系统软件框架如图 2 所示。

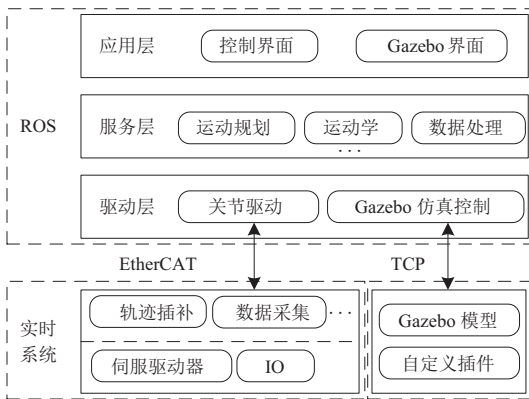


图 2 系统整体框架

将 Linux 系统改造成双内核系统基本流程如下:通过 Xenomai 下载文件中的脚本给标准的 Linux 内核打上必要的补丁;对 Linux 内核进行配置,需要对 Xenomai 使能以及关闭 CPU 频率缩放等功能;交叉编译打完补丁之后的 Linux 内核;将编译得到的文件安装并配置 GRUB 即完成了双内核系统的改造^[4]。

2.2 EtherCAT 通信实现

底层控制系统需要实现控制系统和机器人的伺服系统的通信。在 Linux 系统中,EtherCAT 主站可使用 IGH 主站和 SOME 主站,但是 IGH 主站对于 EtherCAT 协议的实现更加完整,因此本项目用 IGH 主站控制从站数据的发送和接收。EtherCAT 主站数据报读写过程如图 4 所示,在每个通信周期中,主站发出的数据依次经过每一个从站,经过每个从站的时候,从站从数据报中获取并插入数据,全部从站处理完之后,数据报原路返回并由第一个从站作为响应报文发送给主站^[5]。EtherCAT 主站在应用层支持 COE 协议,COE 协议遵循 CANopen 的应用层规范,通过对象字典便可实现伺服驱动器和 IO 模块的控制。

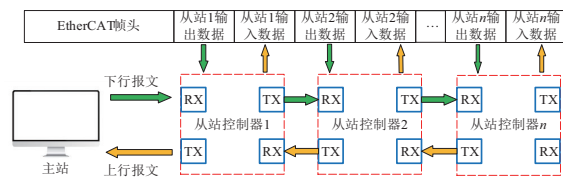


图 4 EtherCAT 数据报读写过程

2 EtherCAT 底层控制系统设计

2.1 Linux 系统实时化改造

Linux 系统相当于 Windows 而言,其开源的特性使得整个系统具有很大的灵活性,并且众多开源项目对于 Linux 都有着很好的支持。因此本项目采用 Linux 系统来运行 EtherCAT 主站,但是标准的 Linux 系统和 Windows 一样,都不具备实时性,因此需要对 Linux 系统进行实时化改造。本系统采用 Linux+Xenomai 的双内核系统^[2],常规的 Linux 内核用于处理非实时的任务,Xenomai 内核用于处理关节驱动节点中的实时任务^[3],经过改造后的系统

3 控制系统软件实现

控制系统的软件部分主要包括机器人运动学、轨迹规划算法创建、人机界面设计、EtherCAT 驱动节点设计等。通过 rqt_graph 指令可以直观地反映各个节点之间数据之间的关系,控制节点之间的关系如图 5 所示。ROS 机器人控制系统的优势在于松散耦合式的系统框架,其强大的社区与开发者提供了很多功能强大的插件。QT 作为一个跨平台的用户界面应用程序开发框架,目前已经和 ROS 很好地结合在一起。机器人人机界面采用 QT_ROS 插件进行界面的开发,可以设置连续路径运动、点到点运动。仿

真机器人界面使用 Gazebo_ROS 自带的显示界面,可以直观地看到仿真机器人的运动情况^[6]。

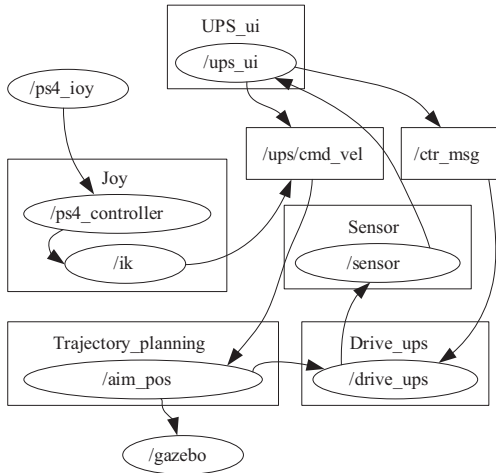


图 5 ROS 控制节点关系图

3.1 传感器节点

传感器节点用于获取机器人各支腿上接近开关的数据以及进行机器人运行状态显示。各接近开关的值通过倍福的 EL1008 模块进行采集,在每个 EtherCAT 通信周期中,通过 IGH 提供的 EC_READ_U8() 函数读取 8 个输入通道的状态。EL2008 模块用于显示机器人控制系统状态,包括 6 个伺服电机使能状态以及显示 EtherCAT 的运行状态。

3.2 轨迹规划节点

通过人机界面可以设置机器人进行点到点运动或者带中间路径点的运动。对于点到点运动,将其转化为关节空间用五次多项式进行插值,保证其速度和加速度连续平稳;对于带中间路径点的运动,将其转化为关节空间相对应的支腿长度,并采用三次样条进行插值,保证中间点过渡平滑减少机器人的抖动^[7]。

3.3 关节驱动节点实现

第 2 节的底层控制系统设计已经为关节驱动节点的设计打好了基础。为实现 EtherCAT 协议控制各伺服驱动器及 IO 模块,需要在工控机中初始化 EtherCAT 主站,在激活主站之前,注册相应的 PDO 过程数据信息,过程数据信息中包含了需要进行操作的对象字典,所采用的 PDO 配置信息如表 1 所示。随后通过 Xenomai 实时内核提供的 API 创建周期性的实时任务,在每个通信周期中实现对各从站的操作,EtherCAT 主站执行过程如图 6 所示。

3.4 Gazebo 仿真控制实现

URDF 是 ROS 中机器人描述的原生支持格式,但是该种描述格式依然缺少很多的功能,例如:无法定义闭链结构(并联机器人),缺少摩擦特性等。因此要在 ROS 里实现并联机器人仿真,就必须使用 SDF 格式的机器人描述格式,该格式的描述文件完整地描述机器人从机器人级到

世界级的所有内容,专供 Gazebo 使用。

表 1 配置 PDO

索引	PDO 入口	名称	类型
	0x6040.0	控制字	UINT16
0x1600 RXPDO1	0x6060.0	运行模式	SINT8
	0x607A.0	目标位置	UINT16
0x1601 RXPDO2	0x6081.0	巡航速度	UDINT32
0x1A00 TXPDO1	0x60F4.0	跟随误差	DINT32
	0x6041.0	状态字	UINT16
0x1A01 TXPDO2	0x6064.0	实际位置	DINT32
	0x606c.0	实际速度	DINT32

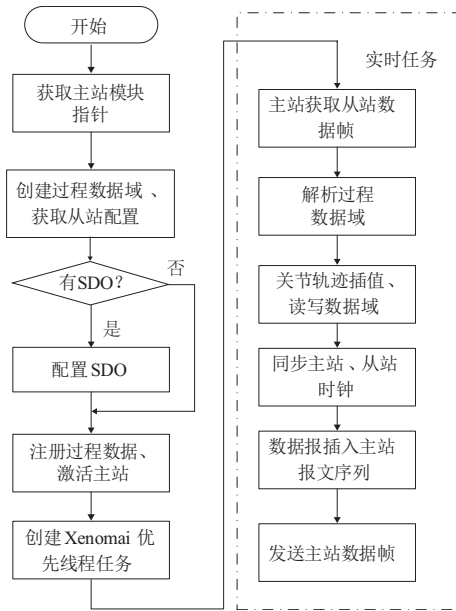


图 6 EtherCAT 主站流程

在 Gazebo_ROS 中实现并联机器人仿真控制的步骤如下:首先需要在三维建模软件中建立机器人各个部件的三维模型,并转换成 stl 格式;将三维模型导入 Blender 软件中,进行比例缩放以及原点的设置,并将处理好的文件转化为 Gazebo 能更好支持的格式 dae;计算各个零件在世界坐标系中的位姿,并在 Gazebo 中进行组装;完成关节的设置之后即得到了机器人的 Gazebo 仿真模型。要让机器人在 Gazebo 中动起来,还得有控制器。Gazebo_ROS 自带的控制器并不支持 SDF 格式,必须编写自定义的 Gazebo 控制器插件,通过自定义的插件来调用 Gazebo 预留的接口来设置关节的位置、速度、PID 参数。图 7 为该并联机器人在 Gazebo 中的仿真模型。

4 实验

在机器人控制系统中,实时性是指在规定的时间内,机器人控制系统能够完成数据的收发;同步性是指在机器人控制系统中对多轴同步运动的要求,避免由于轴之间的不同步而造成误差;可靠性是指机器人控制系统长时间运行

能够保持系统稳定,避免出现死机状况。结合所设计的 6-UPS 并联机器人,对所开发的机器人控制系统进行测试。

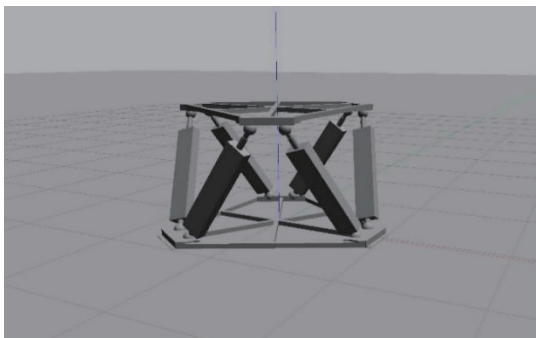


图 7 Gazebo 仿真模型

4.1 实时性测试

在 Xenomai 内核模式下,运行测试程序 latency。设定测试模式为周期性用户模式任务,测试时间为 60 min,分别测试任务周期为 0.5 ms、1.0 ms、2.0 ms 下的系统任务调度的延时性。3 个任务周期下的测试结果如表 2 所示。实验结果表明,3 种机器人常用控制周期下的延时抖动均能够维持在 31 μs 以内,满足机器人控制系统对于实时性的要求。

表 2 各周期下延时性测试结果 单位:μs

测试周期	最大延时	最小延时	平均延时
500	23.973	0.318	0.877
1 000	14.536	0.177	0.691
2 000	30.847	0.354	0.867

4.2 同步性测试

在同步性测试方面,通过对两个 EL2008 从站模块输出方波信号来观察,在两个 EL2008 模块间加入了 4 个倍福 IO 从站模块,所采用的网络拓扑结构如图 8 所示。通过上升沿或下降沿处的时间差可以用来反映同步误差,示波器的测试结果如图 9 所示,通过实验可知,从站的同步误差大约在 20 μs 左右,满足机器人控制系统对于同步性的要求。

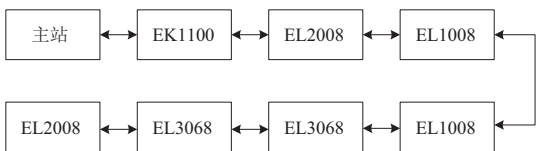


图 8 同步性测试网络拓扑结构

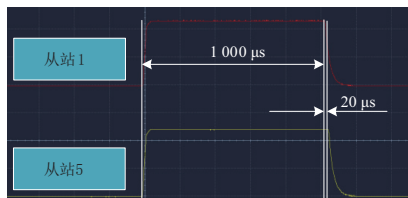


图 9 EL2008 输出同步性测试

4.3 机器人运动仿真及实验

本文搭建了 6-UPS 并联机器人的仿真模型以及控制平台,现对机器人连续路径点的轨迹规划算法进行验证。首先在人机界面输入各路径点,经过运动学解算得到的各支腿伸长量如表 3 所示。得到的各支腿关节空间序列经过三次样条插值,轨迹规划结果发送给 Gazebo 仿真机器人。由图 10 可知,机器人运动轨迹连续、过渡平滑。

表 3 机器人各路径点对应支腿伸长量 单位:mm

序列	轴 1	轴 2	轴 3	轴 4	轴 5	轴 6
1	0	0	0	0	0	0
2	200	200	200	200	200	200
3	300	100	150	300	300	100
4	200	350	400	100	400	300
5	400	400	300	350	300	200

由于篇幅的限制,本文主要对轴 1 进行了分析。轴 1 编码器的仿真结果和实际编码器数值脉冲数误差如图 11 所示。在实际运行过程中,机器人能够很好地按照仿真的曲线进行运动,误差基本在 1 个脉冲当量以内,机器人具有较高的运行精度。图 12 为 6-UPS 并联机器人在仿真及在实验中的运动姿态。

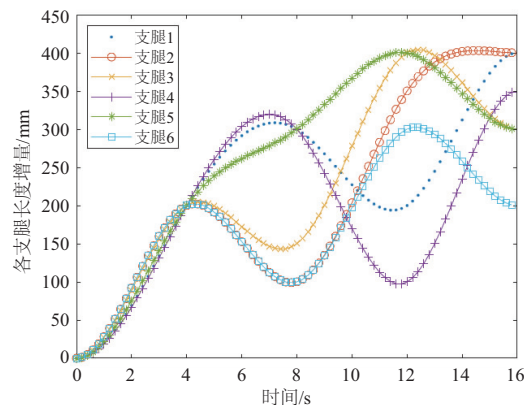


图 10 各轴长度变化三次样条曲线

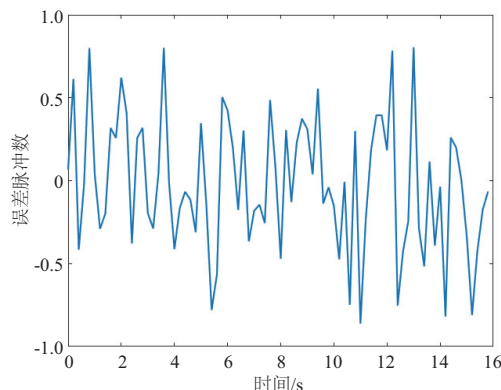


图 11 每周期应发脉冲和实发脉冲误差

(下转第 172 页)

3 结语

1) 主轴转速提升和负载转矩变大均会使主轴系统的热误差和回转误差变大,且转速对于主轴系统精度的影响更加明显。因为主轴转速的提升会导致主轴电机和轴承组的发热功率同时上升,而负载转矩的增大会使电机的电损耗增加从而导致电机生热功率上升,对轴承组的生热功率影响不大。

2) 主轴精度多目标优化算法对主轴精度的调控效果良好,且对高转速下的主轴精度提升效果更加明显,主轴的热误差和回转误差均有明显改善。同时,由于主轴精度多目标优化算法以径向回转误差为决策指标从 Pareto 解集中选取最优解,精度调控后主轴系统的回转误差对比热误差,改善效果更加稳定。

综上所述,本文提出的基于 RSM-NSGA-II 算法的主轴精度多目标优化方法可以对多工况下的主轴系统精度进行有效调控,使主轴系统的加工精度得到一定提升。

参考文献:

- [1] 白丰瑞. 电主轴回转精度退化研究及保持性评估[D]. 长春: 吉林大学, 2020.

- [2] BLASER P, PAVLICEK F, MORI K, et al. Adaptive learning control for thermal error compensation of 5-axis machine tools[J]. Journal of Manufacturing Systems, 2017, 44: 302-309.
- [3] MAREŠ M, HOREJŠO, HAVLÍK L. Thermal error compensation of a 5-axis machine tool using indigenous temperature sensors and CNC integrated Python code validated with a machined test piece[J]. Precision Engineering, 2020, 66: 21-30.
- [4] WANG J, JIANG T, SHEN J, et al. Thermal error compensation of spindle system of computer numerically controlled machine tools through experiments and modeling[J]. Instrumentation Measure Métrologie, 2020, 19(4): 1374-1412.
- [5] CREIGHTON E, HONEGGER A, TULSIAN A, et al. Analysis of thermal errors in a high-speed micro-milling spindle[J]. International Journal of Machine Tools and Manufacture, 2010, 50(4): 386-393.
- [6] 薛海涛, 胡鹏浩, 苗恩铭, 等. 数控车床主轴热变形对零件形位加工精度影响分析[J]. 工具技术, 2011, 45(3): 35-38.
- [7] 袁哲俊, 刘华明. 刀具设计手册[M]. 北京: 机械工业出版社, 1999.
- [8] 刘强, 李忠群. 数控铣削加工过程仿真与优化: 建模、算法与工程应用[M]. 北京: 航空工业出版社, 2011.

收稿日期: 2022-02-14

(上接第 159 页)

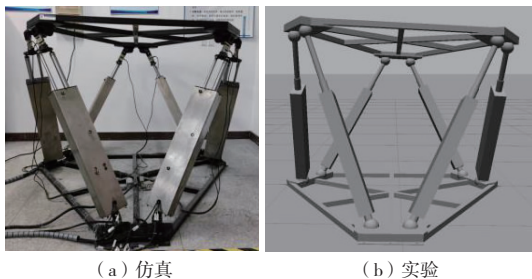


图 12 机器人运行状态

5 结语

针对传统的机器人控制系统存在的同步性差、实时性差的问题,通过开源的 Xenomai 和 EtherCAT 构建了开放的机器人控制系统,舍弃了传统的采用运动控制器+PC 的方式,采用 EtherCAT 实现了机器人的伺服控制。利用 ROS 系统资源丰富的优势,开发了机器人的上位机系统,利用 ROS 松散耦合的通信方式,开发了相应的功能包,有利于代码的复用和移植。使得整个机器人控制系统具有开源、易拓展的优势,对于工业软件国产化也具有重要的意义。各项测试实验结果表明,所设计的机器人控制系统可以有效地实现机器人的控制。

参考文献:

- [1] 孟明辉, 周传德, 陈礼彬, 等. 工业机器人的研发及应用综述[J]. 上海交通大学学报, 2016, 50(增刊1): 98-101.
- [2] CEREIA M, BERTOLOTTI I C, SCANZIO S. Performance of a real-time EtherCAT master under linux[J]. IEEE Transactions on Industrial Informatics, 2011, 7(4): 679-687.
- [3] LIM S, LEE S Y, KIM J H, et al. Development and analysis of Korea open source motion system based on real-time Ethernet[J]. The Transactions of the Korean Institute of Electrical Engineers, 2017, 66(1): 186-193.
- [4] 韩守谦, 裴海龙, 王清阳. 基于 Xenomai 的实时嵌入式 Linux 操作系统的构建[J]. 计算机工程与设计, 2011, 32(1): 96-98, 102.
- [5] 刘豪志, 刘凯, 王欢, 等. 基于嵌入式 Linux 平台 EtherCAT 主站设计[J]. 机械与电子, 2020, 38(3): 44-47, 52.
- [6] ZHANG G J, LI Z Q, NI F L, et al. A real-time robot control framework using ROS control for 7-DoF light-weight robot[C]// 2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM). Hong Kong, China: IEEE, 2019: 1574-1579.
- [7] 孙凌云, 罗福源, 刘鹏. 基于 V-REP 和 MATLAB 的机器人建模及轨迹规划仿真实验[J]. 机械制造与自动化, 2022, 51(1): 108-112.

收稿日期: 2021-11-23