

2-16-2023

Design and Implementation of Industrial Robot Remote Monitoring System in Cloud Manufacturing

Yongkui Liu

1.School of Mechano-Electronic Engineering, Xidian University, Xi'an 710071, China; yongkuiliu@163.com

Lin Zhang

2.School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China;

Yingfu Liu

3.Industrial Cloud Manufacturing (Sichuan) Innovation Center Co. Ltd, Chengdu 610000, China;

Jianyong Feng

4.Ningbo Shuaitelong Group Co. Ltd, Ningbo 315157, China;

See next page for additional authors

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Design and Implementation of Industrial Robot Remote Monitoring System in Cloud Manufacturing

Abstract

Abstract: Considering the lack of the existing research on cloud manufacturing monitoring system and the lack of scalability and flexibility of existing remote monitoring system, and taking deep reinforcement learning-based industrial robot intelligent grasping as an application scenario, *a micro-service architecture-based remote monitoring system for cloud manufacturing is developed, to carry out the requirement analysis and design of the monitoring system*, and the remote monitoring of industrial robot intelligent grasping processes is realized. Test results show that the system can meet the monitoring requirements of resource providers, platform operator(s) and service consumers.

Keywords

cloud manufacturing, remote monitoring, intelligent grasping, micro-service, industrial robot

Authors

Yongkui Liu, Lin Zhang, Yingfu Liu, Jianyong Feng, Bo Yu, and Wenbo Niu

Recommended Citation

Yongkui Liu, Lin Zhang, Yingfu Liu, Jianyong Feng, Bo Yu, Wenbo Niu. Design and Implementation of Industrial Robot Remote Monitoring System in Cloud Manufacturing[J]. Journal of System Simulation, 2023, 35(2): 318-329.

云制造环境下工业机器人远程监控系统设计与实现

刘永奎¹, 张霖², 刘迎福³, 冯坚勇⁴, 于博¹, 牛文博¹(1. 西安电子科技大学 机电工程学院, 陕西 西安 710071; 2. 北京航空航天大学 自动化科学与电气工程学院, 北京 100191;
3. 工业云制造(四川)创新中心有限公司, 四川 成都 610000; 4. 宁波帅特龙集团有限公司, 浙江 宁波 315157)

摘要: 考虑到当前针对云制造监控系统的研究比较缺乏, 以及已有远程监控系统可扩展性和灵活性不足等问题, 以基于深度强化学习的工业机器人智能抓取为应用场景, 设计和开发了一个基于微服务架构的面向云制造的远程监控系统, 对监控系统进行了需求分析和系统设计, 实现了对工业机器人智能抓取过程的远程监控。测试表明系统能够满足云制造中资源提供者、平台运营者和服务使用者三类用户对工业机器人的监控需求。

关键词: 云制造; 远程监控; 智能抓取; 微服务; 工业机器人

中图分类号: TP311 文献标志码: A 文章编号: 1004-731X(2023)02-0318-12

DOI: 10.16182/j.issn1004731x.joss.21-1015

引用格式: 刘永奎, 张霖, 刘迎福, 等. 云制造环境下工业机器人远程监控系统设计与实现[J]. 系统仿真学报, 2023, 35(2): 318-329.

Reference format: Liu Yongkui, Zhang Lin, Liu Yingfu, et al. Design and Implementation of Industrial Robot Remote Monitoring System in Cloud Manufacturing[J]. Journal of System Simulation, 2023, 35(2): 318-329.

Design and Implementation of Industrial Robot Remote Monitoring System in Cloud Manufacturing

Liu Yongkui¹, Zhang Lin², Liu Yingfu³, Feng Jianyong⁴, Yu Bo¹, Niu Wenbo¹(1. School of Mechano-Electronic Engineering, Xidian University, Xi'an 710071, China; 2. School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China; 3. Industrial Cloud Manufacturing (Sichuan) Innovation Center Co. Ltd, Chengdu 610000, China;
4. Ningbo Shuaitelong Group Co. Ltd, Ningbo 315157, China)

Abstract: Considering the lack of the existing research on cloud manufacturing monitoring system and the lack of scalability and flexibility of existing remote monitoring system, and taking deep reinforcement learning-based industrial robot intelligent grasping as an application scenario, a *micro-service architecture-based remote monitoring system for cloud manufacturing is developed, to carry out the requirement analysis and design of the monitoring system*, and the remote monitoring of industrial robot intelligent grasping processes is realized. Test results show that the system can meet the monitoring requirements of resource providers, platform operator(s) and service consumers.

Keywords: cloud manufacturing; remote monitoring; intelligent grasping; micro-service; industrial robot

0 引言

云制造的概念由李伯虎院士带领的团队提出, 并在航天科工率先进行大量实践^[1]。在不断发展的

制造业需求和新兴技术的驱动下, 云制造的概念、内涵、技术和应用不断升级和完善^[2]。云制造将分布在不同地域的制造资源和制造能力通过物联网、虚拟化和服务化等技术手段接入到云制造平台中,

收稿日期: 2021-09-30 修回日期: 2021-12-25

基金项目: 国家自然科学基金(61973243); 宁波科技局科技创新 2025 重大专项(2019B10081)

第一作者: 刘永奎(1981-), 男, 副教授, 博士, 研究方向为云制造、制造系统智能调度与优化、工业机器人学习及智能。

E-mail: yongkuiliu@163.com

形成虚拟的制造资源和制造能力云池,再通过智能化的管理和经营,最终实现根据用户个性化需求向用户提供各类制造资源和制造能力服务^[3]。

云制造概念自提出以来引起了国内外学者的广泛关注和研究。其中,云制造中的典型特征、关键技术和体系架构等^[4]方面的研究经过不断完善和扩展,已经相对成熟。在云制造应用方面,已有研究主要集中在云制造资源服务组合和调度方面^[5-6],并且已经有基于服务关联^[7]、接口关联^[8]和业务关联等模式的服务模型及解决方案^[9]。然而,目前还缺乏对云制造中其他关键环节和不同工业场景应用的研究^[10],如具体的云制造过程监控等^[11]。

远程监控系统大都采用机器人控制器-智能网络节点-服务器-客户端的架构来解决系统与设备的数据通信难题^[12]。然而,上述远程监控系统主要针对单一场景和任务进行设计和开发,在软件适用性、可扩展性、灵活性^[13]等方面还有待提升。此外,云制造环境对制造设备远程监控系统^[14]提出了更高的要求,包括更高的网络传输质量^[15]、更高的灵活性^[16]、更高的可扩展性^[17]、更强的稳定性,以及更好的适用性^[18]等。

针对上述问题,本文以云制造环境下工业机器人智能抓取为应用场景^[19],基于最新的微服务架构,设计和开发了一个远程监控系统,实现了对工业机器人智能抓取过程的监控。

1 工业机器人智能抓取

1.1 应用场景介绍

随着工业机器人智能化程度的增加,工业机器人对工件的智能抓取将会成为工业生产中的一种典型应用,如分拣、辅助装配等。实现对工业机器人智能抓取过程的远程监控,如获取工业机器人和工件实时位姿数据等,有助于及时发现抓取过程中存在的问题,如抓取过程中定位不准、效率低下甚至损坏工件等,并进行在线训练优化

和精准反馈控制等。

本节为云制造环境下的工业机器人远程监控提供一个典型的智能应用场景。

1.2 算法分析和应用

强化学习是从环境状态映射到动作的学习,用来解决序列决策问题。核心思想是智能体在与环境交互的过程中获得最大累计奖励以完成学习目标的最优决策。

1.2.1 马尔可夫决策过程

使用马尔可夫决策过程(Markov decision process, MDP)对强化学习问题进行建模和研究,并用四元组 (S, A, R, P) 表示马尔可夫决策过程中的变量^[20]。本文对机器人的智能抓取过程进行马尔可夫决策过程的数学建模,机器人状态空间为 S ,抓取动作空间为 A ,奖励函数为 R ,状态转移概率为 P ,以及动作执行策略为 π 。

在马尔可夫决策动态过程中,策略 $\Pi: S \rightarrow A$ 实现状态到动作的映射,是智能体选择动作的依据。强化学习的目标就是学习一种策略,使得智能体在这种策略下选择的动作获得的累积回报最大,这个策略称为最优策略 π^* 。

1.2.2 DQN 智能抓取算法的设计

云制造环境下工业机器人的抓取环境通常是高维的,深度Q网络(deep Q network, DQN)算法以深度神经网络为载体,以交互试错方式不断获取 $Q(s, a)$ 值,该 $Q(s, a)$ 值是机器人状态-动作对奖励之和的估计函数。最优策略是选择回报最大化的动作执行。DQN算法利用TD-error不断迭代 $Q(s, a)$ 实现网络参数的更新,同时逼近最优策略 π^* :

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \quad (1)$$

式中: s_t 为智能体所处的环境状态; a_t 为智能体在环境状态 s_t 下选取的动作; α 为神经网络的学习率; γ 为折扣奖励因子; r 为智能体由 s_t 状态下执

行动作 a_t 到达 s_{t+1} 状态后获得的奖惩信息。

由于 DQN 算法更新网络的依据是 TD-error, 即 TD-error 大的样本反向传播更新的梯度更大, 因此, 该机制根据每个样本的 |TD-error| 大小来排序, 更加频繁地回放这些高偏差的样本。定义采样概率:

$$P(j) = \frac{D_j^\alpha}{\sum_k D_k^\alpha} \quad (2)$$

式中: D_j 为 transition j 的优先级, 均大于 0; α 为超参数, $\alpha=0$ 时采用均匀采样, $\alpha=1$ 时采用贪婪策略采样。

$$D_j = |\delta_j| + \epsilon \text{ or } \frac{1}{\text{rank}(j)} \quad (3)$$

采取重要性采样来减小引入 $P(j)$ 产生的偏差:

$$w_j = \frac{1}{s^\beta P(j)^\beta} \quad (4)$$

式中: s 为样本总数; β 为一个超参数。

Q 网络更新变为

$$\text{Loss} = \frac{1}{k} \sum_i w_i \delta_i^2 \quad (5)$$

式中: Loss 为更新网络的反向误差; k 为抽取的样本总数; w_i 为样本重要性的权重; δ_i 为一个样本的误差。

1.2.3 DQN 智能抓取算法的实现

针对云制造环境的机器人智能抓取过程, 网络模型的核心思想是通过神经网络对每个时刻下状态-动作对 (s, a) 的值函数 Q 近似, 进而预测每个状态下的最大值函数 $Q(s)$, 将其对应的像素点映射在三维坐标中, 建立视觉信息到动作输出的最优策略 π^* 。具体实现过程如下:

(1) 网络模型搭建

对于机器人智能抓取而言, 设计的模型需要得到每个像素点精确的三维坐标。因此, 本文选用全卷积神经网络 (fully convolutional networks, FCN) 进行网络搭建。

(2) 视觉输入

预处理好的 RGB 图像和深度图像分别表示了

工作台上所有状态信息的表面特征以及高度特征, 再将其输入神经网络, 利用网络的卷积特性能够有效提取出目标物体的特征。

(3) 奖励函数与损失函数的设置

奖励设置采用稀疏的奖励。当操作成功后, 给予奖励 1; 否则, 奖励为 0。算法实现过程中, 通过每次执行抓取动作后目标物体的 z 坐标变换来判断抓取是否成功。神经网络的损失函数使用 Huber 损失函数:

$$L_i = \begin{cases} \frac{1}{2} (Q^{\theta_i}(s_i, a_i) - y_i^{\theta_i})^2, & \text{for } |Q^{\theta_i}(s_i, a_i) - y_i^{\theta_i}| < 1 \\ |Q^{\theta_i}(s_i, a_i) - y_i^{\theta_i}| - \frac{1}{2}, & \text{otherwise} \end{cases} \quad (6)$$

式中: θ_i 为第 i 次迭代时评估 Q 值网络的参数; θ_i' 为目标 Q 值网络的参数。当经过前向传播计算出最大 Q 值对应的像素点 p , 并执行动作后, 反向传播时, 只通过像素点 p 的损失函数进行反向传播, 更新对应网络的参数。其他像素点在反向传播时默认损失函数为 0。

1.2.4 网络模型训练

本文的训练仿真环境为在 Win10 系统中部署 Python3 环境并使用 Pytorch 框架, 同时安装 CUDA 和 cuDNN, 实现 GPU 加速运算; 仿真场景采用 V-REP 仿真平台进行搭建, 导入 Dobot 机器人模型、工件模型, 并配置相机以获取多角度的仿真画面和实时图像信息。基于 DQN 的机器人智能抓取算法训练框架如图 1 所示。

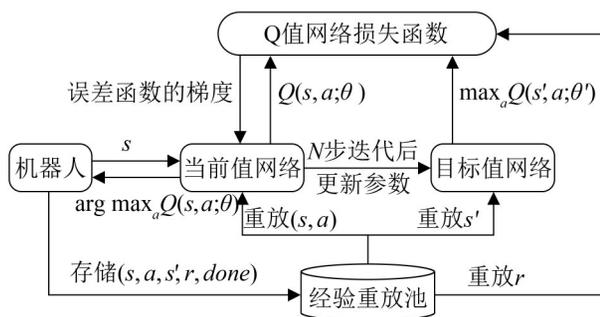


图1 算法训练框架

Fig. 1 Algorithms train frame diagrams

2 系统需求分析

2.1 系统概述

2.1.1 实现目标

在机器人智能抓取过程中, 由于机器人机载设备容量有限、可搭载模块有限、工作环境复杂导致机器人自身的处理能力难以有效应对实际抓取过程中的一些特殊情况。云制造环境的特殊性给工业机器人远程监控系统带来了新的挑战。

针对上述问题, 本文对云制造工业机器人远程监控系统的开发提出以下实现目标:

- (1) 能够满足不同用户的监控需求, 并根据角色合理分配功能权限;
- (2) 支持不同语言智能算法的灵活部署与远程调用;
- (3) 能够对机器人进行远程视频监控, 获取到机器人的实时工作画面, 并在前端界面进行展示;
- (4) 能够与物理机器人进行双向数据交互, 并且操作人员能够通过监控系统对机器人抓取过程进行远程控制。

2.1.2 角色分析

云制造中的角色主要包括平台运营者、资源提供者和服务使用者。

平台运营者: 拥有系统管理的最大权限, 能对系统中的其他用户角色进行相应监控权限的分配。

资源提供者: 拥有对机器人的最大监控权限, 能远程获取机器人的实时工况。

服务使用者: 享有系统分配的相应监控权限, 能通过系统获取到机器人智能抓取过程的实时工况。

2.2 功能性需求分析

工业机器人智能监控系统实现的主要功能性需求包括用户管理、在线训练、视频监控, 以及数据监控。

2.2.1 用户管理

角色用户管理功能需求包括用户控制和角色

控制, 其中平台运营者可以对用户和角色进行基本的设置, 包括添加、编辑、查询和删除等。此外, 平台运营者还可以对系统中的用户和系统设置的 3 类角色进行权限分配。资源提供者和服务使用者可以通过账户信息登录和退出系统。

2.2.2 在线训练

在线训练是本系统的核心功能之一, 旨在使资源提供者能够基于部署的智能算法对工业机器人进行远程训练, 以获得更好的神经网络模型并适应不同场景。

2.2.3 视频监控

对制造现场的工业机器人抓取过程进行实时监控是本系统的核心功能之一。系统用户能够通过远程监控系统获取到机器人工作的实时画面。

2.2.4 数据监控

数据监控是本系统的核心功能之一, 旨在使资源提供者(工业机器人的所有者和使用者)能够通过远程监控系统对制造过程中的机器人进行远程控制, 并获取机器人的实时工作数据。

2.3 非功能性需求分析

相对于一般制造设备监控, 云制造对远程监控提出了更高的要求。

- (1) 更高的稳定性: 保证远程系统各项服务的负载均衡以及远程监控系统上线运行的稳定性。
- (2) 更高的可靠性: 云制造环境下的智能抓取过程对数据有更高的要求, 包括实际工作数据的真实性、海量数据的有效性、数据传输过程中的可靠性等。因此, 系统对数据传输和处理有更高的要求。
- (3) 更高的实时性: 远程监控系统的许多功能模块必须收集机器人工作状态下的实时数据才有意义, 因此, 需要传输速率高、网络带宽高的通信方式和网络。
- (4) 可扩展性和适用性: 云制造环境包含大量的生产制造服务, 系统需要具有更高的可扩展性和

适用性，来适应不同的云制造场景。

(5) 可维护性：不同功能模块需要经历多次不同程度的版本迭代，因此，对监控系统的可维护性提出了更高的要求。

3 系统设计与实现

3.1 整体逻辑结构

系统采用前后端分离的方式进行开发，前端采用 Vue 作为开发框架，后端采用 Spring Boot 作为核心框架，整合 MyBatis-Plus 进行逻辑开发。系统整体开发采用模型-视图-控制器(MVC)分层设计思想，将系统分为 4 个层级，分别为数据访问对象层 (data access object, DAO)、业务层 (Service)、控制层(Controller)和表示层(View)，如图 2 所示。

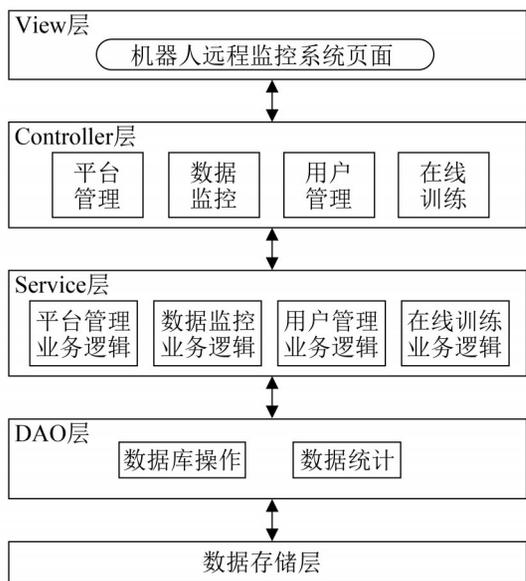


图 2 系统逻辑分层模型
Fig. 2 System logic layering mode

DAO 层负责数据持久化，与数据库连接的一些任务都封装在这层。DAO 层的设计需要先设计接口，然后定义接口的实现类，最后在模块中调用接口来进行数据业务的处理。

Service 层主要负责功能模块的业务逻辑设计。通过接口设计和实现类设计，在配置文件进行关

联，即可在操作过程中调用 Service 接口进行业务处理。Service 层的业务实现需调用已定义的 DAO 层接口。Service 层的业务逻辑封装简化了开发。

Controller 层负责功能模块的业务流程控制。该层通过调用 Service 层的接口来实现业务流程的控制，其配置也同样是在配置文件中设置。针对具体的业务流程，设置不同的控制器，将业务流程进行抽象归纳，设计出可以重复利用的子单元流程模块。

View 层负责用户与系统进行交互。用户能对系统进行操作层面就是这层，也就是前端整体框架。

3.2 微服务架构设计

在远程监控系统总体框架的基础上，系统的微服务架构设计如图 3 所示。

本系统选用 Spring Cloud Gateway 这一微服务组件进行 API 网关的开发。API 网关作为系统的唯一入口，所有的请求都要经过 API 网关，主要负责系统的限流、鉴权、路由转发等。

在服务治理方面，系统采用 Nacos 作为微服务的注册发现中心和配置中心。在微服务系统中，只有注册到注册发现中心，启动的服务才能被发现，从而被系统操作者使用，或者被其他服务调用。配置中心可以在线以动态化和外部化的方式去进行相关服务配置和应用配置的修改，使系统的内部服务可以通过外部进行管理。在服务远程调用方面，采用 Spring Cloud OpenFeign 组件，通过接口的声明和服务发现，使不同微服务之间可通过 HTTP 的方式进行远程访问。

3.3 数据库设计

在本系统中，目标服务对象是工业机器人和系统操作人员。系统的主要服务是为工业机器人提供数据存储和计算服务，为操作人员提供监控机器人、管理系统的服务。本文通过 E-R(实体关系)图展示系统中各实体之间的联系，如图 4 所示。

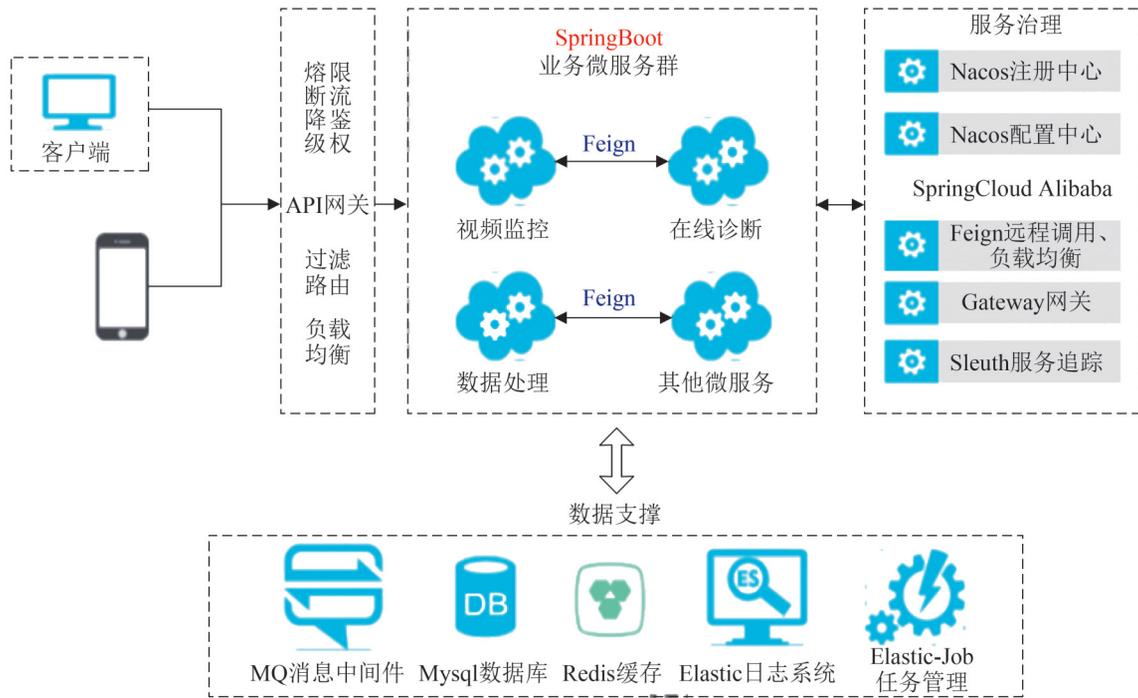


图3 系统微服务架构

Fig. 3 System microservices architecture diagram

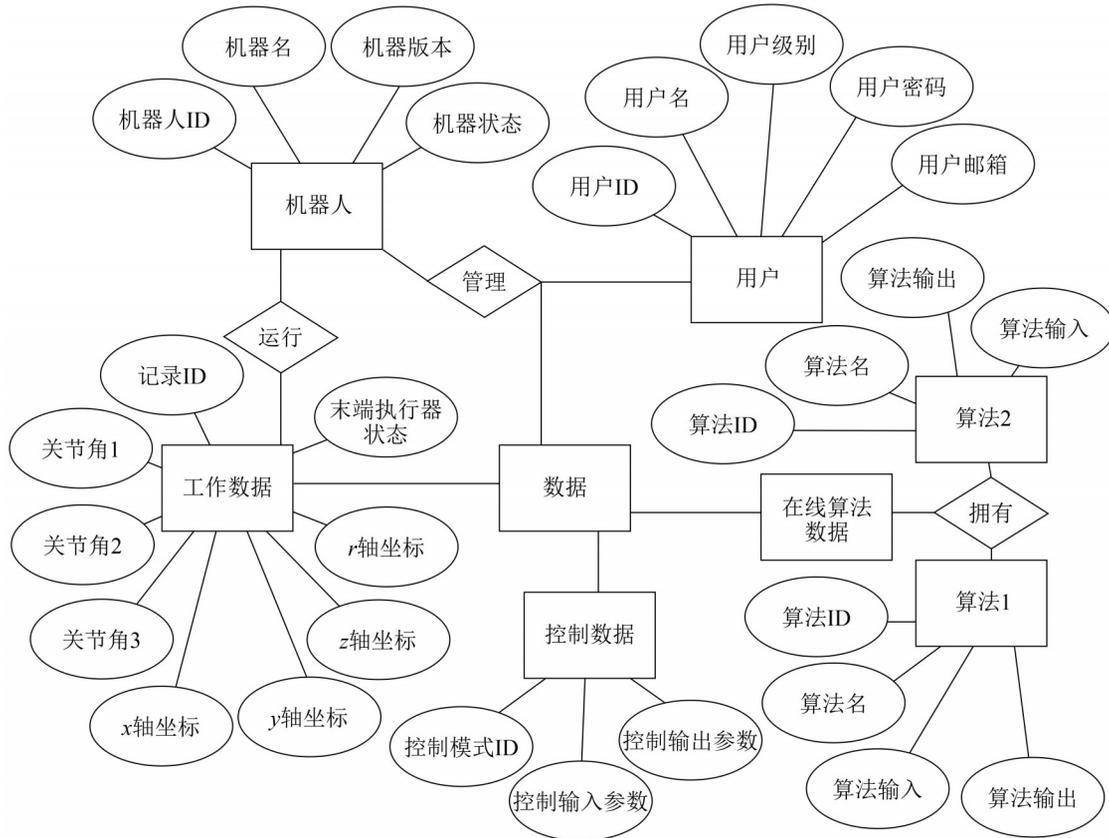


图4 系统实物关系

Fig. 4 Physical diagram of system

本节通过 E-R 图对机器人远程监控系统进行数据库概念模型设计。

(1) 机器人实体：描述工业机器人的基本信息，包括机器人 ID、机器名、机器版本号和机器人状态。

(2) 用户实体：描述系统操作人员的基本信息，包括用户 ID、用户名、用户密码、用户邮箱、用户级别等。

(3) 工作数据实体：描述机器人运行时的工作数据，包括数据记录 ID、机器人的关节角信息、机器人的笛卡尔系坐标轴数据，以及机器人当前末端执行器状态。

(4) 控制数据实体：描述机器人控制模式(运动模式)信息，包括控制模式 ID、控制时输入数据和该控制方式下的输出数据。

(5) 算法实体：描述系统中部署的智能算法的相关信息，包括算法 ID、算法名称、算法输入数据和算法输出结果。

3.4 微服务功能模块设计与实现

工业机器人远程监控系统主要的功能模块是用户管理、在线训练、视频监控和数据监控。

3.4.1 用户管理模块

系统用户管理模块包括用户控制和角色控制两部分。用户控制功能包括用户登录类和用户基本操作功能类；角色控制包括角色基本操作类和用户角色分配类。其中，Controller 类负责处理前端请求，并通过 Service 类和其实现 ServiceImpl 类处理具体的业务逻辑。

云制造工业机器人远程监控系统中的用户登录是平台运营者、资源提供者和服务使用者 3 类角色用户共有的功能模块，也是用户使用系统的第一步操作。在用户登录功能设计中，需要考虑用户登录的安全性。系统用户登录流程如图 5 所示。

云制造工业机器人远程监控系统中，角色控制是用户管理模块的核心功能，以角色列表查询为例进行角色控制功能的介绍：前端发送查询请

求经过 API 网关发送至系统用户管理微服务，通过 sysRoleController 类获取当前操作用户，如果当前用户为系统中的平台运营者，则能进行角色查询，并向 sysService 类调用 queryPage() 方法，最后经过 Dao 层进行数据库交互，获得角色信息后返回给前端界面进行角色列表展示。如果当前操作用户不是平台运营者，则不允许进行角色列表查询。

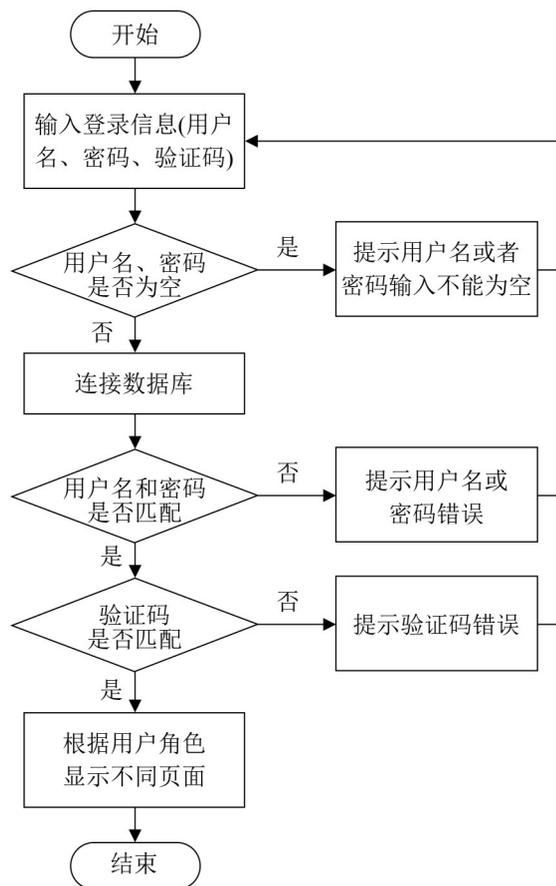


图 5 用户流程

Fig. 5 System user login flowchart

3.4.2 在线训练模块

系统后端程序采用 Java 语言编写，智能抓取算法采用 Python 语言编写，本系统需要设计 Java 程序对 Python 程序的本地调用和远程调用方法。

(1) 本地程序调用

系统调用本地部署的智能算法通过 Java 的 Runtime.getRuntime() 函数直接执行 Python 脚本文

件, 这种方式实现简单, 且由于Python程序部署在本地, 可以直接通过文件位置进行本地调用, 速度较快。

(2) 远程程序调用

系统通过gRPC(远程过程调用)的方式调用部署在远程的智能算法, 并且选用gRPC框架作为后端系统与远程智能算法的通信方式。在该方法实现过程中, 通过Proto文件定义数据结构、服务接口和数据序列化反序列化操作。在Python端, 通过grpc_tools生成序列化协议源代码; 在Java端, 通过Maven引入grpc相关的依赖, 再通过插件Protobuf对Proto文件进行编译, 生成对应的数据结构类文件以便后续对服务端程序和客户端程序的编写。

3.4.3 视频监控模块

视频监控模块通过机器人工作台上安装的双目相机获取到机器人监控画面, 通过OBS(open broadcaster software)软件将机器人的实时工作视频流推送至Node-Media-Server流媒体远程服务器(云平台)。

在该功能模块的实现方面, 主要分为本地工作站端、流媒体服务器端和客户端。

(1) 本地工作站端

本地工作站主要负责模块中机器人实时视频流的推流工作。考虑到云制造环境对远程监控的实时性要求, 视频流的传输协议选用实时信息传输协议(real time message protocol, RTMP)。在

RTMP传输过程中, 发送端会将消息拆分成块(Chunk)进行传输, 这样将数据量大的消息拆分成块的方式可以有效避免持续发送优先级低的消息阻塞优先级高的数据, 提升数据传输的时效性。

(2) 流媒体服务流

系统选用Node-Media-Server作为流媒体服务器, Node-Media-Server支持事件回调、实时转码、服务器和流媒体信息统计等功能。

在流媒体服务器实现方面, 远程服务器中使用Docker拉取Node-Media-Server镜像并通过docker run-d-p1935:1935-p8000:8000--restart=always--name nms illuspas/node-media-server指令启动容器, 在启动容器后, 进入容器并修改默认管理员账户密码信息为本系统中平台运营者的账户信息, 配置完成之后即可正常使用。登录“服务器ip:8000/admin”网站获取当前视频流的信息, 如图6所示。

(3) 客户端

工业机器人远程监控系统选用了由BiliBili公司开源的Flv.js进行视频播放, 系统前端采用Vue框架对Flv.js进行整合, 将视频流从流媒体服务器拉取到前端界面进行机器人实时监控。

3.4.4 数据监控模块

为实现通过部署在远程服务器上的服务去控制本地工作站中的机器人, 以及对工作中的机器人实时下发指令和实时数据传输, 设计了一套有效的远程调用解决方案, 如图7所示。

App	Name	ID	IP	Audio			Video		
				codec	freq	chan	codec	size	fps
live	live	RJKJZZKM	::ffff:172.17.0.1	AAC LC	48000	2	H264 High	1280x720	60

图6 工作画面的视频流信息

Fig. 6 Video streaming information for working screen

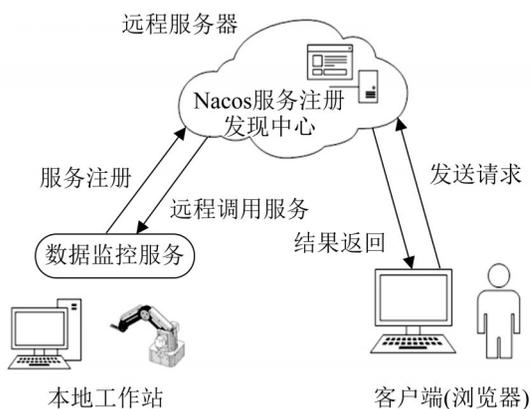


图 7 数据监控解决方案
Fig. 7 Data monitoring solutions

系统数据监控解决方案的实现：首先，通过 `DobotDll.instance.ConnectDobot` 连接机器人，该方法通过指定 `portName` 端口号对指定机器人进行连接。连接机器人后，在控制机器人之前需要进行相关参数和变量的初始化。然后，通过开发过程

中预先编写好的控制模式 API 接口对机器人具体控制方法进行选择 and 调用。最后，调用数据监听方法获取机器人实时位姿信息并保存到数据库中，再通过系统前端界面展示给操作人员。

4 系统测试

4.1 系统测试环境

本地工作站系统为 Window10 MacOS Catalina；远程服务器系统为 Cent OS Linux。软件环境：浏览器为 Google Chrome；WEB 服务器为 Tomcat；Java 运行环境为 JDK 1.8；Python 运行环境为 Python3.7；数据库为 MySQL5.7；视频流服务器为 Node-Media-Server；视频推流客户端为 OBS26.0.02(64 位)

系统微服务模块注册情况如图 8 所示。

服务名	分组名称	集群数目	实例数	健康实例数	触发保护阈值	操作
robot-training	DEFAULT_GROUP	1	1	1	false	详情 示例代码 删除
robot-gateway	DEFAULT_GROUP	1	1	1	false	详情 示例代码 删除
robot-admin	DEFAULT_GROUP	1	1	1	false	详情 示例代码 删除
robot-monitor	DEFAULT_GROUP	1	1	1	false	详情 示例代码 删除

图 8 微服务注册列表
Fig. 8 Micro service registration list

4.2 系统功能测试

为了验证系统是否满足云制造环境下工业机器人远程监控需求，对用户管理功能、在线训练功能、监控功能分别进行了测试。

4.2.1 用户管理功能测试

系统用户管理功能测试包括用户登录、用户控制、角色控制等 3 个方面。

用户登录测试主要包括用户登录信息与数据库中存储的用户信息校验以及验证码是否生效。根据用户登录功能的测试用例对系统进行功能测

试，系统用户登录界面如图 9 所示。

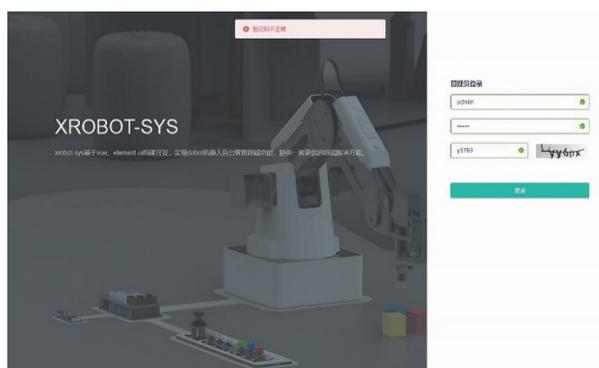


图 9 用户登录界面
Fig. 9 User login interface

用户控制功能测试主要包括对系统内所有用户进行基本的新增、删除、编辑和用户列表查询测试。根据上述测试用例对系统用户控制功能进

行测试, 测试结果均符合预期。其中, 用户列表查询界面如图 10 所示。

ID	用户名	邮箱	手机号	状态	创建时间	操作
1	admin	super@gmail.com	12345678910	正常	2016-11-11 11:11:11	修改 删除
2	userx	consumer@163.com	12345678901	正常	2021-01-15 18:33:20	修改 删除
3	consumer	123@aaa.com	12346578910	正常	2021-01-18 15:21:00	修改 删除
4	enp1	provider@163.com	18888888888	禁用	2021-03-02 13:16:17	修改 删除
5	usery	consumer@qq.com	19999999999	正常		修改 删除
6	userz	consumer@gmail.com	15555555555	禁用		修改 删除
7	enp2	provider@163.com	16666666666	正常	2021-03-02 13:50:32	修改 删除
8	admin2	super@gmail.com	19564444444	正常	2021-03-02 13:52:06	修改 删除

图 10 用户查询页面

Fig. 10 User query page

用户角色控制主要是角色权限分配, 根据上述测试用例对系统角色控制功能进行测试, 测试结果均符合预期。

4.2.2 在线训练功能测试

在线训练功能测试主要包括算法调用、参数设定、输出结果回传并展示, 以及文件对象存储等功能测试。系统在线训练功能模块的测试用例如表 1 所示。

表 1 在线训练测试用例表
Table 1 Online training test case

用例说明	测试预期结果
能否选择算法并设定参数	算法选择成功
能否正确调用在线训练算法	算法调用成功
能否接收算法输出结果	输出正确
算法输出的图片能否上传	上传成功

根据上述测试用例对系统在线训练模块进行测试, 测试结果均符合预期。

4.2.3 监控功能测试

监控功能分为视频监控和数据监控。

视频监控模块是将机器人实时工作画面传输至系统, 并通过界面进行播放。测试的结果是视频传输延迟约 3 s, 帧数可达 60 帧/s, 满足项目需求, 测试通过。

数据监控功能测试包括对机器人控制和机器人实时位姿信息获取的测试。机器人数据监控模块功能测试符合预期, 监控页面如图 11 所示。

4.3 系统非功能测试

本次测试使用 JMeter 对系统主要接口进行响应测试, 通过 50 个线程, 10 次循环向系统发送 GET/POST 请求, 以获取请求成功、失败次数, 相应时间等数据来判断系统的性能。

对系统响应时间的评价可以依据“2/5/8 原则”, 如表 2 所示。

参数名	查询	新增	批量删除	启动机器	停止机器					
<input type="checkbox"/>	动作id	关节角1	关节角2	关节角3	关节角4	笛卡尔坐 标系x	笛卡尔坐 标系y	笛卡尔坐 标系z	末端执行 器状态	操作
<input type="checkbox"/>	11	0	6.476041	13.01445 5	0.000002	231.4955 75	0	101.1277 01	<input checked="" type="checkbox"/>	修改 删除
<input type="checkbox"/>	12	0	24.90560 2	26.50623 9	0.000002	260.0000 31	0	56.11998	<input checked="" type="checkbox"/>	修改 删除
<input type="checkbox"/>	13	0	24.90560 2	26.50623 9	0.000002	260.0000 31	0	56.11998	<input checked="" type="checkbox"/>	修改 删除
<input type="checkbox"/>	14	0	24.76228 3	26.23731	0.000002	260	0	56.83998 9	<input checked="" type="checkbox"/>	修改 删除
<input type="checkbox"/>	15	0	22.64484 4	21.86789 1	0.000002	260.0000 31	0	68.59999 1	<input checked="" type="checkbox"/>	修改 删除
<input type="checkbox"/>	16	0	4.945122	21.50914	0.000002	220	0	80.39999 4	<input type="checkbox"/>	修改 删除
<input type="checkbox"/>	17	6.04042	6.740878	13.05331 2	-6.040419	230.8156 43	23.32388 7	100.9560 85	<input type="checkbox"/>	修改 删除

图 11 数据监控页面

Fig. 11 Data monitoring page

表 2 系统性能评价指标

Table 2 System performance evaluation metrics

响应时间/s	评价
≤2	响应速度快
(2~5]	响应速度较快
(5~8]	响应速度较慢
>8	响应速度过慢

非功能测试结果表明，系统中主要接口的平均响应时间均在 2 s 子内，满足“2/5/8”原则中响应速度快的指标，并且视频监控模块中实时视频延迟在 2~3 s，视频帧数为 60 帧/s。

由上述测试数据可知，本系统响应速度很快，满足要求。

5 结论

本文设计了一个面向云制造的工业机器人远程监控系统，实现了工业机器人数据采集、工作画面实时传输、智能抓取算法在线训练等功能，能较好地满足云制造环境下不同角色用户对工业机器人的远程监控需求。

以上研究还存在一些不足之处，未来计划在

以下几个方面进一步研究和改进：

(1) 在智能抓取算法方面，将问题模型以更细的粒度进行划分，通过多线程的方式将单一机器人作业扩展至多机器人协作；

(2) 在监控范围和对象方面，根据现有研究，建立多场景、多设备、多任务及多离散资源的协同监控模型，真正实现对云制造全生命周期过程进行监控；

(3) 在可扩展和适应性方面，使用分布式存储方式对海量数据进行存储和管理，并部署故障诊断、数据分析等智能模块以适应更多云制造场景。

参考文献：

- [1] 李伯虎, 张霖, 王时龙, 等. 云制造——面向服务的网络化制造新模式[J]. 计算机集成制造系统, 2010, 16(1): 1-7.
Li Bohu, Zhang Lin, Wang Shilong, et al. Cloud Manufacturing: A New Service-oriented Networked Manufacturing Model[J]. Computer Integrated Manufacturing Systems, 2010, 16(1): 1-7.
- [2] 周龙飞, 张霖, 刘永奎. 云制造调度问题研究综述[J]. 计算机集成制造系统, 2017, 23(6): 1147-1166.
Zhou Longfei, Zhang Lin, Liu Yongkui. Survey on

- Scheduling Problem in Cloud Manufacturing[J]. *Computer Integrated Manufacturing Systems*, 2017, 23(6): 1147-1166.
- [3] 周济. 智能制造——"中国制造2025"的主攻方向[J]. *中国机械工程*, 2015, 26(17): 2273-2284.
- Zhou Ji. Intelligent Manufacturing—Main Direction of "Made in China 2025"[J]. *China Mechanical Engineering*, 2015, 26(17): 2273-2284.
- [4] 李伯虎, 柴旭东, 侯宝存, 等. 云制造系统3.0——一种"智能+"时代的新智能制造系统[J]. *计算机集成制造系统*, 2019, 25(12): 2997-3012.
- Li Bohu, Chai Xudong, Hou Baocun, et al. Cloud Manufacturing System 3.0—New Intelligent Manufacturing System in Era of "Intelligence+"[J]. *Computer Integrated Manufacturing System*, 2019, 25(12): 2997-3012.
- [5] 李伯虎, 张霖, 任磊, 等. 云制造典型特征、关键技术与应用[J]. *计算机集成制造系统*, 2012, 18(7): 1345-1356.
- Li Bohu, Zhang Lin, Ren Lei, et al. Typical Characteristics, Technologies and Applications of Cloud Manufacturing[J]. *Computer Integrated Manufacturing System*, 2012, 18(7): 1345-1356.
- [6] Zhang Y, Zhang G, Liu Y, et al. Research on Services Encapsulation and Virtualization Access Model of Machine for Cloud Manufacturing[J]. *Journal of Intelligent Manufacturing*(S1572-8145), 2017, 28(5): 1109-1123.
- [7] Liu Chao, Su Ziwei, Xu Xun, et al. Service-Oriented Industrial Internet of Things Gateway for Cloud Manufacturing[J]. *Robotics and Computer-Integrated Manufacturing*(S0736-5845), 2022, 73: 3-10.
- [8] 徐建明, 俞俊铭, 董建伟, 等. 基于云平台的机器人监控系统设计[J]. *高技术通讯*, 2020, 30(9): 938-948.
- Xu Jianming, Yu Junming, Dong Jianwei, et al. Design of Robot Monitoring System Based on Cloud[J]. *High Technology Communications*, 2020, 30(9): 938-948.
- [9] 刘永奎, 王力翠, 王曦, 等. 云制造再探讨[J]. *中国机械工程*, 2018, 29(18): 2226-2237.
- Liu Yongkui, Wang Lihui, Wang Xi, et al. A Revisit to Cloud Manufacturing[J]. *China Mechanical Engineering*, 2018, 29(18): 2226-2237.
- [10] Mei D H, Song Z Y. Research and Application of WSN in Enterprise Safety Manufacture Real-Time Monitoring and Accident Pre-Warning System[J]. *Applied Mechanics & Materials*(S1660-9336), 2015, 743: 787-791.
- [11] Otsu K, Fukuda H, Kobayashi Y, et al. Remote Monitoring and Communication System with a Dolllike Robot for the Elderly[C]//IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society. Florence, Italy: IEEE, 2016: 5941-5946.
- [12] Mohan R, Suraj A K, Agarawal S, et al. Design of Robot Monitoring System for Aviation[M]//*Nanoelectronics, Circuits and Communication Systems*. Springer, Singapore, 2019: 535-547.
- [13] 朱添益, 戴逢哲, 康文, 等. 直流换流站运维监控系统智能机器人设计方法[J]. *电气技术*, 2020, 21(2): 22-25.
- Zhu Tianyi, Dai Fengzhe, Kang Wen, et al. Design of Intelligent Robot of Operation Monitoring System in the Direct Current Converter Station[J]. *Electrical Technology*, 2020, 21(2): 22-25.
- [14] Lee J, Seo Y H. Design and Development of a Monitoring System Based on Smart Device for Service Robot Applications[J]. *International Journal of Internet, Broadcasting and Communication*(S2288-4939), 2018, 10(3): 5-10.
- [15] Kekre A, Gawre S K. Solar Photovoltaic Remote Monitoring System Using IOT[C]//2017 International Conference on Recent Innovations in Signal Processing and Embedded Systems(RISE). Bhopal, India: IEEE, 2017: 619-623.
- [16] 台宪青, 吴梦悦, 王玉峰, 等. 数据中心服务器资源监控系统的设计与实现[J]. *计算机应用与软件*, 2019, 36(7): 14-20.
- Tai Xianqing, Wu Mengyue, Ma Yufeng, et al. Design and Implementation of Server Resource Monitoring System in Data Center[J]. *Computer Applications and Software*, 2019, 36(7): 14-20.
- [17] 邓昀, 李朝庆, 程小辉. 基于物联网的智能家居远程无线监控系统设计[J]. *计算机应用*, 2017, 37(1): 159-165.
- Deng Yun, Li Chaoqing, Cheng Xiaohui. Design of Remote Wireless Monitoring System for Smart Home Based on Internet of Things[J]. *Journal of Computer Applications*, 2017, 37(1): 159-165.
- [18] Perumal T, Sulaiman M N, Leong C Y. Internet of Things (IoT) Enabled Water Monitoring System[C]//2015 IEEE 4th Global Conference on Consumer Electronics (GCCE). Osaka, Japan, IEEE, 2015: 86-87.
- [19] 刘开, 徐宣国. 云制造系统中制造信息监控的框架设计[J]. *电子设计工程*, 2016, 24(7): 104-106.
- Liu Kai, Xu Xuanguo. Design on the Monitoring Frames of Manufacturing Information in the System of Cloud Manufacturing[J]. *Electronic Design Engineering*, 2016, 24(7): 104-106.
- [20] Garcia F, Rachelson E. Markov Decision Processes[M]//Sigaud, Olivier, Buffet, Olivier. *Markov Decision Processes in Artificial Intelligence*. London: Wiley, 2013: 1-38.