

基于 ROS 的智能工业机器人系统开发平台^①

刘 凤^{1,2}, 杨东升¹, 廉梦佳^{1,2}, 张 展^{1,2}, 王丽娜^{1,2}

¹(中国科学院 沈阳计算技术研究所, 沈阳 110168)

²(中国科学院大学, 北京 100049)

摘 要: 目前, 工业机器人对于机器视觉、自主路径规划等智能化功能需求日益增长. 然而在传统工业机器人系统中添加智能化功能模块时需要修改大量的源码, 浪费了人力和成本. 本文提出的基于 ROS 的易扩展机器人系统开发平台, 能为开发者开发智能工业机器人系统提供了方便. 本平台分为服务器端和机器人端. 将机器人端作为一级节点, 与安装 ROS 的 PC 服务器端进行通信. 机器人一级节点由二级功能节点与功能模块组成. 根据此平台开发实现的 JPB06 六自由度工业机器人系统具有机器视觉、自主定位、语音控制等智能化功能, 可以满足工业机器人对于智能化和实时控制的需求.

关键词: 工业机器人; ROS; 系统开发平台; 智能化

引用格式: 刘凤, 杨东升, 廉梦佳, 张展, 王丽娜. 基于 ROS 的智能工业机器人系统开发平台. 计算机系统应用, 2017, 26(10): 77-81. <http://www.c-s-a.org.cn/1003-3254/6019.html>

Development Platform of Intelligent Industrial Robot System Based on ROS

LIU Feng^{1,2}, YANG Dong-Sheng¹, LIAN Meng-Jia^{1,2}, ZHANG Zhan^{1,2}, WANG Li-Na^{1,2}

¹(Shenyang Institute of Computer Technology, Chinese Academy of Sciences, Shenyang 110168, China)

²(University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: At present, there is an increasing demand for intelligent functions such as machine vision and autonomous path planning in industrial robots. However, adding intelligent function modules in the traditional industrial robot systems requires a lot of source code modification, wasting manpower and cost. An easy-to-extend robot system development platform based on ROS presented in this paper provides developers with the convenience of developing intelligent industrial robotic systems. The platform is divided into server-side and robot-side. The robot-side as a first-level node, communicates with the server-side installed with the ROS. The first-level node of the robot consists of the second-level functional nodes and functional modules. According to this platform, the JPB06 6-DOF industrial robot system has intelligent function modules, such as machine vision, autonomous positioning and voice control, which can meet the needs of industrial robots for intelligent and real-time control.

Key words: industrial robot; ROS; system development platform; intelligence

社会对于工业机器人智能化、自动化需求的逐渐增长, 促进了智能机器人相关基础领域的成熟, 如计算机视觉、语音识别与合成、室内移动机器人算法等, 但是这些基础领域不能很好的结合, 导致了机器人研究过程中需要统一开发测试平台. 为此, Willioy Garage

发布了开源分布式软件框架 ROS^[1](robot operating system).

ROS 起源于 2007 年斯坦福大学人工智能实验室与 Willioy Garage 公司合作的 PR(personal robot) 项目, 是运行在主操作系统上的中间件, 从本质上它是一种

① 收稿时间: 2017-01-09; 采用时间: 2017-02-23

分布式模块化的开源软件框架. ROS 为机器人研发领域提高代码复用率, 为工业机器人智能化的发展提供了方便. ROS 主要有以下几个特点. 一、所有进程以及他们所进行的数据处理, 将会通过一种点对点的网络形式表现出来. 二、支持多种语言开发. 三、集成已有的开源项目代码. 四、工具包丰富. 五、开源且免费. 但是其缺点是基于 Linux 的 ROS 实时性不能满足工业需求.

现如今主要有四种解决方法: 第一种是将实时系统嵌入 ROS 中, 如美国西南研究院 (SwRI) 推出 ROS-Industrial^[2]. 第二种是在实时系统中植入 ROS 功能包^[3]. 第三种是开发实时软件框架, 比如北京航空航天大学推出的 RGMP-ROS^[4](RTOS and GTOS on Multi Processors-ROS) 将 ROS 节点分为实时节点与非实时节点, 并分别运行在实时系统和非实时系统中. 第四种是 2016 年发布的符合工业级标准的 ROS2.0^[5], 因为 ROS2.0 内部的 DDS(数据分发服务) 消耗过大, 并且其稳定性比 ROS1.0 低, 所以本文选取 ROS1.0.

本文结合中国科学院沈阳计算技术研究所机器人与数控机床数控系统已有的工业机器人系统开发技术, 提出新型易扩展的智能化工业机器人系统开发平台. 该平台能够提高工业机器人系统开发的灵活性.

1 ROS 开源平台

ROS 的主要目标是为机器人研究和开发提供代码复用的支持, 其最重要的贡献是提供基于异构计算集群的主机操作系统的结构化通信标准, 并且通过使用节点将代码和功能解耦, 提高了系统容错能力和可维护性, 使系统简化. 目前, 开发者和开源 ROS 开发人员为 ROS 共享社区贡献了各个领域的软件包, 如 SLAM 定位算法、坐标转换 tf 库、机器视觉开源库 OpenCV 和 OpenNI 等. 由于 ROS 代码的高可重用性和可扩展性, 基于 ROS 的机器人系统可以通过利用社区中已有的代码方便地用于二次开发和复杂应用, 实现了成熟的基础领域结合. 目前支持 ROS 的机器人已经有 40 多种, 包括 FANUC、ABB、安川、ADEPT 等大型企业的工业机器人. 以下分析传统工业机器人开发平台和单纯基于 ROS 的机器人开发平台的利弊, 并给出本文提出的系统开发平台结构.

1.1 传统工业机器人系统开发平台

传统工业机器人系统开发平台如图 1 所示. 中间

件为应用软件共享资源提供支撑, 提供其运行与开发环境. 实时操作系统实现对硬件部件的编程控制和接口操作, 为应用软件提供一个实时的执行环境和对硬件的虚拟接口, 满足工业机器人对实时性需求. 依据模块化和分层结构思想, 有利于系统整体的开放性和模块化设计. 但是使用传统工业机器人系统开发平台有以下缺点. 其一, 各大机器人公司的软件非开源, 使得开发一个机器人系统困难重重, 成本巨大. 其二, 不同的机器人厂商使用的操作系统、中间件以及编程语言各有区别, 使得各种先进的基础领域不能直接共享. 其三, 模块间联系紧密, 不利于添加或删除某一模块. 其四, 不同的控制系统, 导致了机器人系统不具备可移植性. 其五, 人机交互界面不统一使得开发者和使用者需要具备更专业水平. 综上所述, 传统工业机器人系统开发平台并不有利于开发智能化工业机器人系统.

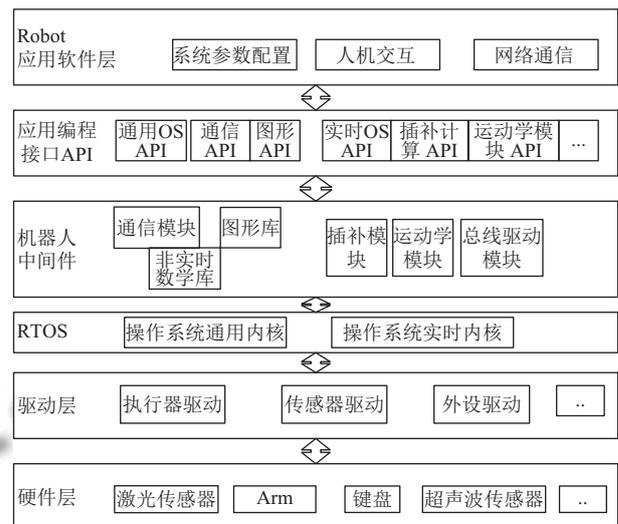


图 1 传统工业机器人系统开发平台

1.2 单纯基于 ROS 的工业机器人系统开发平台

单纯基于 ROS 的工业机器人系统开发平台如图 2 所示. 此平台提供机器人开发的统一标准, 并且由于 ROS 内部节点的概念, 使得开发者可以利用 ROS 社区共享的丰富软件包快速开发机器人系统或者修改、添加、删除功能节点. 但是, 由于 ROS 是运行在 Linux 之上的中间件, 所以通过该平台开发的机器人系统在实时性上满足不了工业机器人需求, 并且大部分功能包针对 PC 设计, 相比于嵌入式, 灵活性低. 所以本文提出新型基于 ROS 的智能工业机器人系统开发平台.

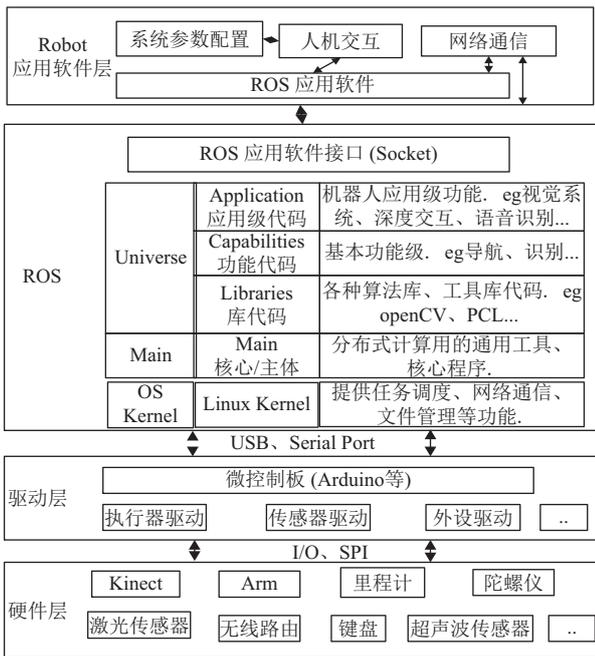


图2 单纯基于 ROS 的工业机器人系统开发平台

1.3 基于 ROS 的工业机器人系统开发平台

基于 ROS 的工业机器人系统开发平台如图 3 所示。该平台通过使用统一的软件开发框架 ROS 为开发者开发智能化工业机器人系统过程中方便地添加、删除智能化模块，并且本平台利用传统工业机器人系统的中间件与 RTOS 执行实时任务，满足工业机器人对于实时性需求。该平台不仅能满足工业机器人系统的需求，也可开发非工业机器人、多机器人系统。

2 基于 ROS 的工业机器人系统开发平台

此平台主要有以下几个特点。第一，平台主要分为 PC 服务器端和机器人端。运行 ROS 的服务器端作为开发平台的核心，提供路径规划、视觉、避障等复杂计算的功能，控制机器人端的运动轨迹，并实时展现机器人状态。相比于传统基于 ROS 的机器人系统开发平台中的将所有节点运行在一起，将高消耗节点运行在服务器端会分散 CPU 压力，提高系统稳定性。第二，在此平台中将一个要开发的机器人看成一个单独的一级节点与服务器端通信，而不是将机器人内部各个传感器、各功能模块单独看做一个节点。机器人一级节点由功能二级节点与传统工业机器人各个模块组成，主要控制机器人的运动。使用机器人一级节点比繁多的二级节点更便于多机器人系统开发。第三，相比于图 2，

此平台通过传统工业机器人中间件和 RTOS 为智能化工业机器人系统开发提供实时性。第四，相比于图 1，此平台利用 ROS 可以方便的添加智能化模块，并进行传感器数据获取与数据处理等。

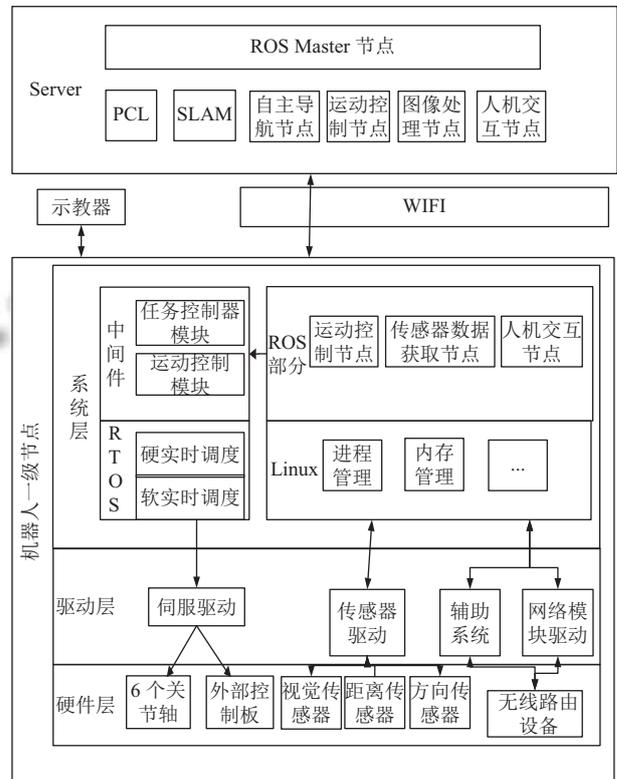


图3 基于 ROS 的工业机器人系统开发平台

2.1 硬件层

工业机器人的硬件层主要是执行器、传感器和外围设备。执行器包括辅助机器人完成运动和动作的电机和臂，对于底盘固定的工业机器人最主要的控制量就是控制机器人手臂等关节的移动，所以机器人驱动器中最根本和本质的问题就是控制电机，控制电机转的圈数，即控制机器人六个关节轴，就可以控制机器人移动的距离和方向、机械手臂的弯曲的程度或者移动的距离等。同时为了保证控制量的精确度，需要各种传感器反馈，及时对于机械臂控制做出调整。传感器，是机器人和现实世界之间的纽带，包括视觉传感器、激光传感器、里程计等。视觉传感器微软 Kinect 可以提供真实环境的三维深度数据。由超声波和激光传感器组成距离传感器以提高碰撞检测时环境数据的精确度。取向传感器，例如加速度计，里程表和陀螺仪，用于检测机器人的位姿。

2.2 驱动层

驱动层的作用是为系统层程序提供硬件的操作接口,并且实现硬件的驱动程序.驱动层主要包括总线驱动、伺服驱动、传感器驱动和网络模块驱动等.在驱动层利用控制器板如 Arduino 或 Raspberry Pi 来管理驱动器和外设. Arduino 主要应用于控制电机驱动和各类传感器(比如超声波、IMU 等),然后和 ROS 的上机位通讯,交换数据. roserial_arduino 包允许 Arduino 作为一个 ROS 的节点,可以直接发布和订阅 ROS 的消息,发布 TF 变换,以及获取 ROS 的系统时间.

2.3 系统层

在系统层,采用基于 Linux 的 ROS 核心库与基于 RTOS 的传统模块结合的方式.机器人一级节点通过 ROS 核心库与服务器通信,并且具有一定的自主运动控制机器人能力. ROS 核心库主要运行在嵌入式 Linux 中,用于接收服务器的命令,通过驱动通信向模块发送目标位姿等信息,并且通过驱动接收各种传感器的数据用于自身的局部路径规划和碰撞检测等,同时传送传感器数据给服务器用于建立现实环境的三维模型.保留基于 RTOS 的传统模块化结构是为了保障机器人的实时性和可靠性.

2.4 服务器层

服务器收集来自传感器的数据来维护机器人周围环境的三维数据,进行复杂的数据处理得出机器人目标位姿.并将目标位姿信息传递给机器人一级节点,用于运动规划,进而控制机器臂的运动.同时服务器通过三维可视化工具 RVIZ 显示三维模型和机器人实时状态. ROS master(节点管理器)运行在服务器上,控制所有机器人一级节点与其他功能节点通信.用户可以使用 ROS 开源社区提供的功能软件包来开发所需的功能,如自主导航、图像处理、碰撞检测等.服务器中的节点与机器人一级节点通信通过配置 master 节点的 ROS_MASTER_URI 来通信.基于 PC 的服务器在保证整个系统的操作能力外,具有更便利的人机交互能力和在复杂和动态环境下,比机器人主机更容易配置的优点.此开发平台也可满足多机器人系统的开发,只要将其他机器人一级节点与 master 节点通信即可.

3 JPB06 机器人系统

利用以上设计的机器人系统开发平台,本节实现了六自由度串联工业机器人 JPB06 系统. JPB06 机器人由机器人本体、PC、控制柜组成,具有视觉、可视

化、自主运动规划、自主定位等功能,如图 4 所示.以下通过 PC 和控制柜说明本文提出的平台的应用示例,并给出实验结果.

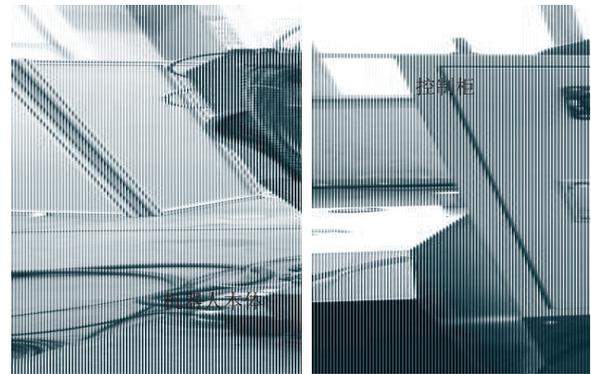


图 4 工业机器人 JPB06

3.1 PC

PC 机作为服务器端运行 Ubuntu14.04 与 ROS hydro. 主要有两个功能:通过 Zynq 中采集的数据维护机器人工作的三维环境、提供机器人全局路径规划功能,并将目标位姿信息传送到 Zynq 中对应的节点.提供人机交互服务.

3.2 控制柜

控制柜中主要有两部分组成,运行 ROS 的 Zynq-7000 EPP,和机器人控制器.

Zynq-7000 可扩展处理平台(EPP)中运行 Ubuntu-14.04,并且安装 ROS 核心包. Zynq 接收底层传感器的高速数据,传感器包括激光传感器、Kinect 传感器、里程计、陀螺仪、加速度计等. Kinect 传感器数据用于服务器端生成三维地图.激光传感器用于地图数据更精确,同时也为机器人在局部运动规划时提供更可靠的数据.里程计、陀螺仪与加速度计用于控制机器人位姿.

工业机器人功能二级节点很多,节点间的信息交互很频繁,并且信息量大,为了提升系统通信性能,做了以下两点变化.一、在机器人节点内部使用共享内存方法把传感器数据 memory-map 到内存中,只传递数据的地址和大小信息.相比同一机器中的 ROS 节点间的通信使用网络栈的 loop-back 机制,即每一个数据包都需要经过多层软件栈处理,数据延迟已经控制在 20us 以内.二、使用轻量级的序列化程序.将 ROS 的通信栈中传输数据损耗最大的序列化延时降低 50%.

为了保证 ROS 内节点的安全性,防止某一节点无

限消耗系统内存,从而导致的系统崩溃,本平台采用 Linux Container(LXC)来管理每一个 ROS 节点进程, LXC 有效地将单个操作系统管理的资源划分到孤立的群组中,以更好地在孤立的群组之间平衡有冲突的资源使用需求.并且 LXC 消耗极少量的 CPU 资源.

3.3 实验结果

本次实验通过讯飞语音控制机器人运动验证本文提出的智能工业机器人系统开发平台的可行性.语音识别的工作过程如图 5 所示. Wakeup 节点监听语音并将声音信号作为消息发送到 xfwakeup 主题, xfasr 节点订阅了 xfwakeup 主题,一边采样声音信号,一边进行识别.若识别成功,则将返回的结果发送到 xfunderstand 主题,将失败提示语发送到 xfsaywords 主题. explain 节点订阅 xfunderstand 主题,通过 control_cmd 主题发送识别结果解析后的消息.机器人控制节点接收这一消息,并根据消息内容进行不同的控制策略.

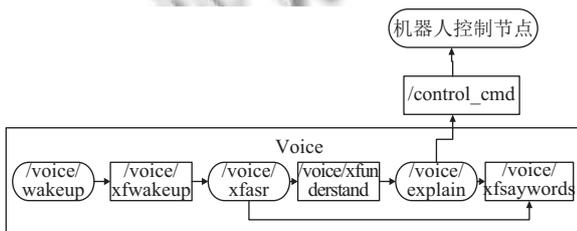


图 5 ROS 中讯飞语音包识别语音过程

测试语音为“向下”时,讯飞语音包识别出“向下”命令,并发送给机器人控制节点.机器人控制节点在接到“向下”命令时确定机器人目标位姿,然后控制机器人运动规划.(为方便展示,将此目标位姿在机器人控制节点中与“向下”命令对应.)图 6 中间两个截图为运动过程中轨迹截图.实验显示,JPB06 能够顺利根据语音指令移动到目标位姿.此实验表明了本文提出的机器人系统开发平台能够开发机器人系统.

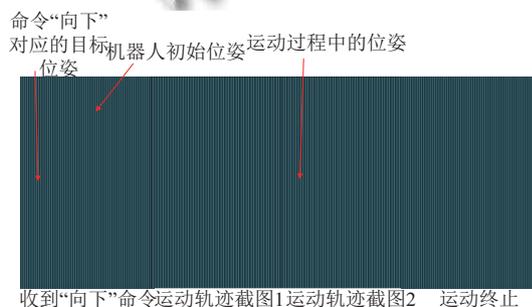


图 6 RVIZ 中显示的机器人运动过程

4 总结

本文提出一种能够提高工业机器人智能化水平的机器人系统开发平台,该平台将运行 ROS 的 PC 作为服务器端,主要负责复杂的计算、可视化和人机交互等任务.单个机器人作为机器人一级节点,由二级功能节点和任务控制等模块组成.这样系统可以在智能化和可靠性上都能满足智能工业机器人的需求.

本文主要做了语音控制运动实验验证了平台的可行性.本平台可用于开发一系列工业机器人系统,也可用于开发多机器人系统和非工业机器人系统,如足球机器人系统、服务类机器人系统、无人机系统等.使用本平台开发其他具有更多功能模块的机器人系统时,可以通过 ROS 社区提供的全球共享的开源功能包添加所需功能.

参考文献

- 1 ROS/technical overview. <http://wiki.ros.org/>. [2016].
- 2 ROS/industrial. <http://wiki.ros.org/Industrial>. [2016].
- 3 Huang Z. Development and implementation of a ROS-based software framework for real-time control of robots[MSc. thesis]. Beijing: Beihang University, 2013.
- 4 Wei HX, Huang Z, YU Q, *et al*. RGMP-ROS: A real-time ROS architecture of hybrid RTOS and GPOS on multi-core processor. Proc. of 2014 IEEE International Conference on Robotics and Automation (ICRA). Hong Kong, China. 2014. 2482-2487.
- 5 ROS/ROS2.0. <http://design.ros2.org/>. [2016].
- 6 Joseph L. Mastering ROS for Robotics Programming. Packt Publishing Ltd., 2015.
- 7 Yu Q, Wei HX, Liu M, *et al*. A novel multi-OS architecture for robot application. Proc. of 2011 IEEE International Conference on Robotics and Biomimetics (ROBIO). Karon Beach, Phuket, Thailand. 2011. 2301-2306.
- 8 Hu CX, Hu C, He DX, *et al*. A new ROS-based hybrid architecture for heterogeneous multi-robot systems. Proc. of 2015 27th Chinese Control and Decision Conference (CCDC). Qingdao, China. 2015. 4721-4726.
- 9 Boren J, Cousins S. Exponential growth of ROS[ROS Topics]. IEEE Robotics & Automation Magazine, 2011, 18(1): 19-20.
- 10 Li XM, Yang CJ, Chen Y, *et al*. Hybrid event based control architecture for tele-robotic systems controlled through internet. Journal of Zhejiang University Science, 2004, 5(3): 296-302. [doi: 10.1631/jzus.2004.0296]