文章编号: 1002-0446(2005)05-0436-05

机器人图形化编程与三维仿真环境*

邱长伍[1], 曹其新[1], 长孙郁男[2], 横山和彦[2]

(1. 上海交通大学机器人研究所, 上海 200052; 2. 日本安川电机株式会社机器人技术开发部, 北九州 日本 803-8530)

摘 要: 介绍一种采用 JAVA开发、基于 RTLinux平台的适合多机器人多任务的图形化编程与三维仿真环境. 采用图标和连线方式快速建立具备流程图特点的控制程序. 运行管理内核实现程序图结构的信息流向跟踪,对程序运行采用边解释边执行机制. 机器人运动学指令由底层 RTLab执行,结果返回上层;由 Java3D实现的三维仿真环境可动态显示机器人的运动过程. 该环境支持离线编程和在线调试. 目前支持日本安川公司研制的 Motoman UPJ和 SmartPal类人轮式移动服务机器人.

关键词: 机器人; 图形化编程; 三维仿真; JAVA; RTLinux

中图分类号: TP24 **文献标识码**: B

Graphical Programming and 3D Simulation Environment for Robot

QIU Chang-wu[1], CAO Qi-xin[1], IKUO Nagamatsu[2], KAZUHIKO Yokoyama[2]

(1. *Research Institute of Robotics, Shanghai Jiaotong University, Shanghai* 200052, *China;*
2. *Robotics Technology R&D Department, YASKAWA Electric Corporation, Kitakyushu City* 803-8530, *Japan*)

Abstract: A graphical programming and 3D simulation environment is introduced for multi-robot and multi-mission, which is developed in JAVA language and is based on RTLinux platform. The icon-based and link-line method is adopted to edit the flowchart-like control applications quickly. The running management kernel tracks the information of program chart and adopts a mechanism which can interpret and process the program simultaneously. The kinematics command is executed by RTLab in the bottom layer and the processing result is returned to the upper layer. The three-dimensional simulation environment implemented by JAVA 3D displays the virtual robot movement dynamically. This environment can support both off-line programming and on-line debugging. Now, it can be used for Yaskawa robots Motoman UPJ and SmartPal, of which the latter one is a human like wheeled service robot.

Keywords: robot; graphical programming; 3D simulation; JAVA; RTLinux

1 引言(Introduction)

图形化的编程方式目前已在众多领域得以应用,熟知的有 LabView、UML(Universal Modeling Language)、MATLAB/Simulink等. Visual C++、Visual Basic、Delphi等高级语言也都采用了图形化编程方式. 针对工业机器人控制应用程序编制规格, 1999年,欧洲协作研究计划 AMIRA(Advanced Man Machine Interfaces for Robot System Applications)提出了机器人系统应用程序用户界面风格规范,此后 Icon-based编程方式在机器人技术领域引起了广泛关注[1]. 目前 ISO(International Standard Organization) 15187一直在不断完善和补充关于工业机器人图形化编程中 Icon的相关标准和规范.

应用 Icon-based图形化编程技术可避免传统编程方式对句法、语法的严格要求,冗长的代码序列对程序结构和问题求解的结构化策略的形象表达较差等弊端[2]. 使用图形模块将一个动作步骤形象地表示出来,使其与方法步骤直接吻合,这样的模块式语言最有利于问题解决思路的流畅实现. 用户无需考虑语法、句法的严格要求,可将精力用于如何更有效、更优化地实现问题求解.

针对以上情况,作者设计了面向机器人的图形化编程与三维仿真集成环境. 机器人控制程序具有流程图特点. 用户用对应的图形模块表示他要实现

* 收稿日期: 2005-01-12

© 1994-2009 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

的逻辑序列,或动作序列,输入输出的参数通过有限的几次提示对话完成,通过鼠标点击连线,将所有的图标模块按照逻辑顺序连接起来,实现机器人控制程序的快速编制,辅以三维仿真环境显示执行过程.

该环境是上海交大机器人研究所与日本安川电机株式会社自 2004年 8月起合作开发的二期项目 GPSR (Graphical Programming and Simulation for Robot)的开发成果. 可进行机器人离线/在线仿真、动态特性分析和模拟. 文中对 GPSR的框架和结构设计进行了概述,限于篇幅,简要叙述了 GPSR面向对象的设计方法和相关策略.

2 软件系统设计 (Design of software system)

2.1 整体框架设计

该软件系统总体由三层构成,如图 1所示. 系统运行于 RedHat Linux 9.0, 2.4.2以上内核,系统实时支持由 RTLinux 3.2完成. 其中上层和中间层由 GPSR应用软件系统与操作系统所组成. 应用软件系统主体由两大部分构成: a) GPSR集成开发环境; b) RTLab API(由日本安川电机开发研究所机器人技术开发部开发). 总体结构中的最底层由硬件资源构成,通过一块 PCI接口的 I/O卡驱动机器人伺服控制器,另一块 I/O卡驱动 16路输入与 16路输出信号,同时返回位姿检测信息、伺服控制系统状态信息和外部 I/O量.

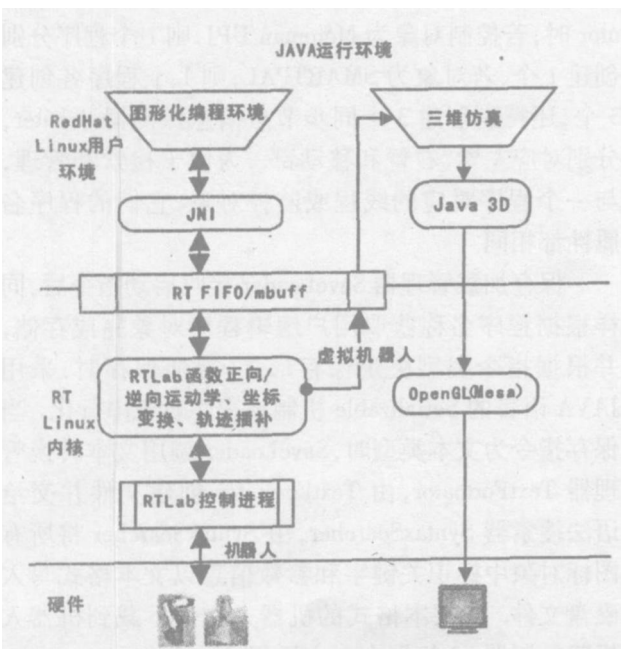

图 1　GPSR总体框架

Fig 1　General framework of GPSR

2D图形化编程环境实现了流程图式的程序图结构绘制功能,程序图结构由封装为 Icon形式的编程元素和表示不同分支属性的连线组成. RTLab API的函数接口通过 JNI(Java Native Interface)封装,使上层 JAVA实现的代码可以直接调用 C语言实现的 RTLab API代码. RTLab API具备运动学转换、运动控制、位姿检测和扩展 I/O功能[3]. 三维仿真环境通过调用由 JNI封装的位姿检测接口获取机器人实时位姿信息,同时刷新三维环境中虚拟机器人位姿参数,实现机器人动态仿真.

三维仿真环境不仅支持 Motoman UPJ型机械臂,而且支持安川公司正在开发的轮式全维移动类人服务机器人 SmartPal,可在同一窗口中或不同窗口中支持多机器人动态仿真. 在系统资源允许下,可实现 32个 Motoman UPJ或 8个 SmartPal同时运行.

2.2 软件结构

GPSR总体划分为四层结构: 1) UI(User Interface)层; 2)组织层; 3)执行层; 4)支持层,如图 2所示. 程序编辑环境采用 MDI(Multi-Document Interface)界面风格,但三维仿真环境与程序绘制环境分别采用独立窗口实现 (当前版本的 JAVA 3D中, Canvas目前默认属性为 heavyweight,其在 JInternalFrame中显示不正常,以后的 JAVA 3D版本将解决这个问题,这样将使 GPSR具备更好的整体性[4]). UI层中包含集成环境主窗口、程序编辑窗口、仿真器窗口、树型文件浏览窗口、仿真器手动调节面板、信息窗口、程序变量及状态监视器窗口、常用工具栏、程序运行控制工具栏和编程元素工具栏. 组织层中主要包含程序运行调度管理器、编程元素管理器、保存加载管理器和虚拟机器人管理器. 执行层中主要包括程序图结构信息流向搜索器、同步节点探测器、API指令执行器、程序状态记录器、语法结构翻译器、虚拟机器人等. 支持层由 RTLab API和 RTLinux内核构成.

2.2.1 UI层

UI层各对象主要完成用户交互和信息显示. 集成环境主窗口对象 MainFrame扩展 JFrame,提供程序运行框架外观和各种应用接口 (菜单、工具栏),负责启动应用程序的各项功能,完成事件监听处理及窗口的布局管理. MainFrame中添加了 JDesktopPane控件,为程序编辑窗口提供 MDI容器. 树型文件浏览窗口、信息窗口和程序编辑窗口都扩展 JInternalFrame实现,并注册 MainFrame为窗口状态监听器,但其中任何窗口状态变化时, MainFrame都进行相应

© 1994-2009 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

的判断处理.

工具栏中提供常见的新建、保存、打开、删除等功能,在编程元素工具栏中提供 4类图标元素,包括系统、逻辑、I/O、运动控制. 系统类中提供程序开始、结束、机器人 D设置、运动学正向和逆向运算、坐标变换等功能图标;逻辑类中提供 For、endFor、If、endIf、DoWhile、endDoWhile等逻辑控制图标; I/O中提供机械手张开、机械手闭合、变量、变量运算 (数值运算和逻辑运算)、外部 I/O量输入或输出等图标;运动控制类中提供关节空间控制、笛卡尔坐标控制、增量运动控制、直线运动控制等运动控制图标. 每个图标对象分别实现相应的输入输出对话机制.

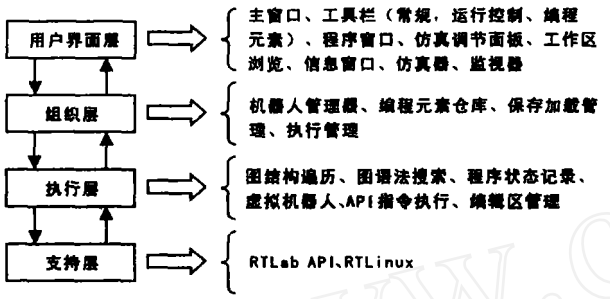

图 2 GPSR系统结构

Fig 2 The structure of GPSR system

由于 GPSR中集成了不同的机器人,为方便管理,针对不同机器人类型定制了不同的编程元素工具栏. 对于 SmartPal机器人,由于对手臂和全维移动基座采用了独立的控制内核,所以在用户程序编制中提供了同步控制标识,在这里借鉴交通信号控制特点,在 SmartPal的控制程序分支中的不同位置,添加属性不同的信号灯对象图标和并行处理的开始和结束图标.

程序编辑窗口 ProgFrame类扩展 JInternalFrame,窗口中嵌一个 JScrollPane、JPanel对象. JPanel对象作为用户编程区域,它实现鼠标事件监听,当用户在工具栏中选定一个 Icon时,当前 ProgFrame中的 ChoosetoPaint对象被赋值该 Icon的字符串标识,鼠标左键在点击编辑空白区后,以该标识检索编程元素管理器 Elementhouse获取所要绘制的 Icon对象,然后调用该 Icon的描画方法,完成图标绘制. 当在针对不同类型机器人的程序窗口之间切换时,由 MainFrame监听窗口激活事件的处理接口 internalFrameActivated (InternalFrameEvent e),控制工具栏更新为对应当前程序的机器人类型.

仿真器窗口与仿真器控制面板由 MainFrame经过对话后启动,其中包含预先定制的虚拟环境及其世界坐标系,并提供鼠标交互浏览功能. 信息窗口和程序变量及状态监视窗口提供程序运行过程的信息输出和显示.

仿真器手动控制面板为机器人的每个关节提供了调节控件、虚拟环境导航控制和虚拟机器人当前位姿参数提取接口. 通过选择控制对象,该控制面板的指令将发往不同的仿真器和虚拟机器人. 通过手动调节虚拟机器人位姿,然后提取其位姿参数,可选中程序编辑区域中的某个需要位姿参数的图标,然后调用 Edit菜单下的 LoadData子菜单实现自动赋值.

2.2.2 组织层

组织层主要完成程序对象和仿真机器人的组织管理,组织层接受上层控制指令启动执行层功能,执行层与程序和仿真机器人之间的交互都通过组织层间接完成. 类 RobotManager完成机器人索引管理以及机器人运动控制指令的接收和发送.

运行控制管理器 Scheduler接收要求执行的程序名,通过程序名向 MainFrame请求获得程序对象,并根据程序应用目标创建和维护与程序名对应的图结构信息流向搜索器 Pathfinder线程对象、同步节点探测器 Scouter对象、API指令执行器 Executor对象、程序监控对象 ProgRecorder,当创建 Pathfinder和 Executor时,若控制对象为 Motoman UPJ,则 1个程序分别创建 1个,若对象为 SMARTPAL,则 1个程序各创建 3个,还特别创建 3个同步节点信息探测器 Scouter,分别对应左臂、右臂和移动部. 为便于检索和管理,与一个程序对应的线程或监控对象,它们的程序名属性都相同.

保存加载管理器 SaveLoader接收启动指令后,同样根据程序名称获取用户编辑程序对象完成存储,并根据指令类型决定保存形式,普通保存时,采用 JAVA语言的 Serializable机制将程序对象串行化. 当保存指令为文本类型时, SaveLoader调用文本转换管理器 TextFormator,由 TextFormator创建文件并交给语法搜索器 SyntaxSearcher,由 SyntaxSearcher将所有图标对象中标识关键字和参数信息以文本格式写入磁盘文件. 该文本格式的机器人文件下载到机器人机载控制器,由机器人自主运行.

编程元素管理器 Elementhouse中封装了所有编程元素. 当外部要求获取以关键字标识的某编程元

© 1994-2009 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

素时,由 Elementhouse根据关键字确定创建相应的对象,并返回给请求者.

2.2.3 执行层

Pathfinder、Executor、Scouter和 ProgRecorder合作完成程序的解释执行. 除异常错误和执行结束,它们协调完成用户编辑程序的解释运行. 当 Pathfinder线程运行到程序执行终点时,Pathfinder线程停止并销毁自己和相应的 Executor和 ProgRecorder,控制权转交给 Scheduler.

在编辑控制 SmartPal机器人控制程序时,采用并行标识图标和信号灯图标标识实现同步控制,见图 3.

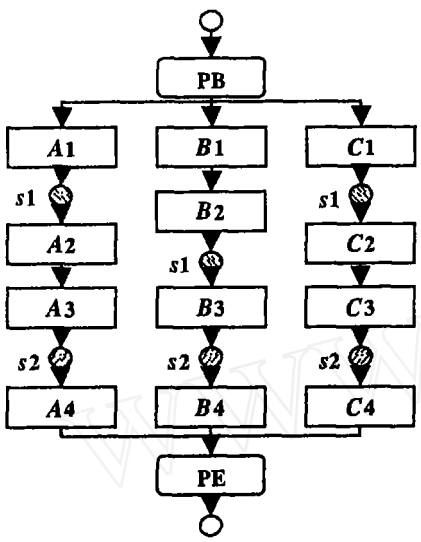

图 3　并行分支同步处理机制

Fig 3　Mechanism of synchronization in parallel branch

当 Scheduler接受执行用于控制 SmartPal类型机器人的程序任务时,它同时分别创建 3个 PathFinder、Scouter和 Excecuter 程序运行到达并行分支开始节点 PB时,Scouter开始工作,3个 Scouter分别探测 *A*、*B*、*C*分支中同步节点信号灯,并将探测到的信号灯 *s*1、*s*2及其所在分支信息注册到 ProgRecorder中,当 Scouter搜索到达并行分支结束的图标 PE时,销毁自己,通知 ProgRecorder,自己所在分支搜索完毕,当 ProgRecorder检测到所有分支都搜索完毕的信息时,ProgRecorder通知 3个分支的 PathFinder同时开始执行. 任何一条分支执行到 *s*1时,它将信号灯 *s*1属性设置为点亮,同时通知 ProgRecorder,由 ProgRecorder检测其他分支是否已到达 *s*1,若任何一条分支都未到达该 *s*1,则通知正在 *s*1停留的 PathFinder等待,否则 ProgRecorder通知所有分支上正在等待的 PathFinder继续执行.

虚拟机器人 VRobot类中封装了虚拟机器人构造、行为动作处理、碰撞检测及处理等功能. 用户程序在 GPSR中运行前,可选择启动仿真环境,同时创建虚拟机器人,提供简化和正常两种精度模型 (针对计算机计算和三维处理性能选择,正常为 1∶1比例 CAD几何面元建模,简化模型为连杆模型). 程序指令执行过程中,当执行运动指令时,启动以下代码段:

```
while(motionStatus = RTLab. RtlGetJoint(theta) >0)
{ robotManager. robotMove (progName, robotType, theta); }
```

运动指令执行完毕时,motionStatus小于零,RtlGetJoint[3]执行过程中将底层 RTLab中维护的机器人实时位姿数据写入 theta,通过 robotManager. robotMove()接口驱动虚拟机器人实现运动过程仿真. 其他 I/O信号获取原理类似.

2.2.4 支持层

支持层采用 JNI技术封装了底层 C语言实现的机器人控制核心 RTLab API. RTLab API由日本安川电机开发研究所机器人技术开发部基于 RTLinux (Real Time Linux)内核开发,是一组用于机器人学研究的软件包,由于以 RTLinux技术实现,所以具备优良的实时性能,位置控制周期为 0.5ms[5]. RTLab API提供六自由度串联机械臂正向/逆向运动学求解、运动控制 (关节空间和笛卡尔空间)和伺服系统初始化与关闭等功能,可支持 ADLINK公司的 PCI7230 I/O卡,具备 16bit的 16路输入与 16路输出. 通过 I/O扩展可实现丰富的信号检测和交互,如力传感、限位开关等信号检测和扩展设备 (如多指机械手)的输入输出控制[5]. RTLab利用 RTLinux FIFO (First-In-First-Out)管道实现对一个 Motoman机械臂的实时控制. 目前其中 16个管道用于真实机器人实时控制,其余 16个管道进行虚拟机器人实时仿真.

3　编程与仿真 (Programming and simulation)

机器人图形化程序模块的设计和组织形式的目标为遵从 ISO15187,并在此基础上进行扩展. 当前开发的 GPSR在图标外观的表达上与上述标准还有差距,但底层执行逻辑和图标意义的设计与上述标准是一致的,后期开发中将着重使图标及连线的表达更加标准化.

对流程图风格的应用程序,由于采用了图标与

© 1994-2009 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

连线的鼠标操作编辑方式,自然屏蔽了传统编程方式易产生的字符输入错误,所以 GPSR环境仅提供图标连接正确性、程序结构正确性和相关参数完备性测试功能,即可保证程序在“语法”上的正确性,功能正确性由设计者结合程序执行过程或结果自主判断.

三维仿真环境采用 JAVA 3D技术实现,在 Linux平台上 JAVA 3D基于 OpenGL实现三维图形驱动显示. 机器人正常模型由 VRML2.0文件存储,通过 JAVA语言实现的 Xj3D软件包加载 VRML描述的机器人构件模型,在 JAVA 3D虚拟环境中进行虚拟装配,构建虚拟机器人. 三维仿真环境提供完整的机器人运动过程仿真和碰撞检测,真实表达实际机器人运行效果.

4 结论(Conclusion)

本文提出的 GPSR——机器人图形化编程与仿真环境,遵从了工业机器人应用程序开发的国际标准,符合当前机器人应用开发的主导方向,在国内还未出现同类集图形化编程与三维仿真于一体的集成化环境.

致谢

日本安川电机公司为本研究提供了整套伺服控制机器人系统,并安排本项目成果参加 2004年 3月在北九州举办的国际机器人博览会.

参考文献 (References)

[1] Kazi A, Seyfarth M. The MORPHA style guide for icon-based programming[J]. VDI Berichte, 2002, 1679: 41 - 46.

[2] Cox P T, Smedley T J. Building environment for visual programming of robots by demonstration [J]. Journal of Visual Languages and Computing, 2000, 11(5): 549 - 571.

[3] R&D Center YASKAWA Corporation. Instructions for RTLab API (Ver 1.1.2) [M]. Japan: YASKAWA Robotics Technology R&D Dept, 2004.

[4] Sun Microsystems INC. Getting Started with the Java 3D™ API [M]. USA: Sun Microsystems INC, 1999.

[5] 横山和彦,梅津真弓. 研究用ロボットパッケージRTLab[J]. 技報安川電, 2003, 261: 269 - 274.

作者简介:

邱长伍 (1978-),男,博士生. 研究领域:机器人学,机电一体化技术,系统仿真.

曹其新 (1960-),男,博士,教授. 研究领域:机器视觉与识别,智能控制,多智能体协调.

(上接第 425页)

参考文献 (References)

[1] Blidberg R. The development of autonomous underwater vehicles (AUV): a brief summary [EB/OL]. http://www.ausi.org, 2001 - 03 - 26.

[2] Chappell S, Komerska R. An environment for high-level multiple AUV simulation and communication [EB/OL]. http://www.ausi.org, 2000 - 08 - 01.

[3] Choi S, Yuh J. A virtual collaborative world simulator for underwater robots using multi-dimensional synthetic environment[A]. Proceedings of the 2001 IEEE International Conference on Robotics and Automation[C]. Seoul, Korea: IEEE, 2001. 926 - 931.

[4] Fujii T, Ura T, Kuroda Y, *et al*. Development of a versatile test-bed "Twin-Burger" toward realization of intelligent behaviors of autonomous underwater vehicles[EB/OL]. http://underwater.iis.u-tokyo.ac.jp/robot/tb/tb-chp1-e.html, 2005 - 01 - 24.

[5] Robinson T. ODIN - an underwater warfare simulation environment [A]. Proceedings of 2001 Winter Simulation Conference[C]. Arlington, Va. USA: IEEE, 2001. 672 - 679.

[6] 梁金海. 基于多智能体的军用智能水下机器人体系结构研究[D]. 哈尔滨:哈尔滨工程大学, 2003.

[7] 李皓, 康凤举, 田志超. 基于 HLA的水下航行器分布交互仿真[J]. 计算机仿真, 2003, 20(3): 5 - 8.

[8] 张斌. 多机器人系统仿真平台[D]. 北京: 中国科学院自动化所, 2000.

作者简介:

徐红丽 (1978-),女,博士生. 研究领域:水下机器人系统仿真,多水下机器人协调控制.

许真珍 (1982-),女,博士生. 研究领域:多水下机器人智能.

封锡盛 (1941-),男,博士生导师,中国工程院院士. 研究领域:水下机器人总体技术,多水下机器人控制、导航等.

© 1994-2009 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net