

基于CORBA技术的 机器人模块化设计的研究进展

曹其新 张 镇

上海交通大学 机器人研究所

[摘 要] 随着机器人应用领域的日益普及其功能的不断完善,机器人自身结构和控制也变得越来越复杂。传统的设计和开发方法已经逐渐显示出了制约机器人的应用和发展,为此,研究者们寻求机器人模块化和组件化设计新方法,本文将从介绍以CORBA为代表的中间件在机器人领域的研究状况为切入点,综述了基于CORBA中间件的机器人模块化设计技术,并指出了通过该技术将各种软硬件的异构因素封装在统一的功能组件中,解决模块化设计的瓶颈问题。最后展望和分析了模块化、分布式的机器人组件的发展趋势。

[关键词] CORBA, 机器人, 模块化

[Abstract] With the development of robotic applications, robot functions become more and more powerful, but the robotic system becomes more and more complex. The traditional non-modularization design has restricted the robotic application and development, thus the modularization technology is introduced into the robotic domain, and encapsulates heterogeneous robotic components into the uniform standardized components. This paper introduces the current research status on CORBA in the robotic domain, presents modularization and distributed robotic architecture. The problems and developmental trends of the robotic modularization are predicted.

[Key words] CORBA, robot, modularization

0. 引言

近年来,智能机器人的应用重心已经从制造业转向非制造业,随之其研究重点也从结构环境下的固定式机械臂、机械手转向非结构未知环境下移动式的自主机器人。与传统的工业机器人相比较,新型智能机器人需要面临不确定的工作环境、完成更加复杂多变的任务。为此,许多知名企业和研究机构针对机器人某些领域的特点和要求提出了自己的解决方案,并公布了各种相关的行业或者地区标准。由于不同厂家的硬件兼容性和不同软件模块的通用问题成为了提高机器人的研发速度和降低制造成本的主要障碍。机器人的模块化设计,无疑是解决这一问题的有效方法之一。而中间件技术是实现机器人模块化的关键。当前支持模块化设计的中间件主要分为3类,

包括Microsoft推出的DCOM(Distributed Component Object Model),Sun的EJB(Enterprise JavaBeans)以及OMG(Object Management Group)的CORBA(Common Object Request Broker Architecture)。其中,DCOM技术的应用集中在Windows平台上;EJB的应用环境则局限于支持Java虚拟机的系统上;而CORBA采用远程对象调用机制,支持异构环境下分布式应用系统的开发和互操作,具有与底层硬件、操作系统、网络、通信协议和编程语言无关的特点,所以被广泛地应用于复杂机器人或多机器人的分布式系统中。CORBA支持异构平台下对象的可互操作性和移植性方面具有很大开放性与灵活性,该标准主要特点是实现了软总线结构,通过功能抽象,屏蔽和隐藏底层诸如操作系统、开发语言和具体技

术、协议等细节差异，形成一套统一的接口描述，从而解除了机器人各个部件之间紧密的耦合性；同时将各类应用模块按总线规范封装成软插件，插入总线即可实现集成运行。CORBA还针对机器人实时性、可靠性等要求，向系统提供事件服务，安全与生命周期等服务，已经成为国际机器人领域的研究热点之一。

本文主体部分内容安排如下：第1部分介绍CORBA的产生与发展情况，第2部分说明了机器人模块化设计的一般步骤，第3部分介绍了几种典型的基于CORBA机器人系统。最后，分析了机器人模块化存在问题并提出了今后的发展趋势。

1. CORBA 概述

1.1 CORBA 的产生与发展

针对早前的分布式系统跨平台、跨语言能力差的情况，OMG组织制定了工业标准CORBA规范。目的是在分布式异构环境下，基于对象的软件成员实现应用集成，并具有可重构、可移植和互操作特性。

CORBA规范是针对OMA参考模型中的对象请求代理ORB制定的，如图1所示，OMA模型中对象请求代理ORB处于整个系统的核心位置，是对象间的通信总线，同时定义了异构环境下对象透明地发送请求和接受响应的基本机制，是构造分布式对象应用、实现异构环境下互操作的基础。

在OMA中，对象是一个封装的实体，只能通过外在的接口提供服务，而对该接口的描述是和对象的

具体实现无关。因此可以通过独立的接口定义语言（IDL）描述对象向外界提供的服务接口。服务使用者和服务提供者之间通过CORBA总线机制联系在一起。如上图所示，OMA中通过ORB交互的对象组件可以分为如下四类。

（1）对象服务(Object Services)：定义能被分布式对象应用程序使用的公共对象服务，这些服务是和特定应用领域无关的基本对象服务。如生命周期服务定义了对象创建、删除、拷贝移动的方式。

（2）公共设施(Common Facilities)：公共设施服务是从应用的最终用户角度考虑的各种服务，如设备管理、复合文档、邮件管理等。

（3）领域接口(Domain Interfaces)：面向特定应用领域的服务如机械制造、商务、医疗、电信等不同行业的公共服务。

（4）应用接口(Application Interfaces)：面向特定的具体应用。是指供应商或用户借助ORB、公共设施和公共对象服务而开发的特定产品。

1.2 CORBA 规范的特点

CORBA定义了IDL语言及映射、ORB及ORB间互操作机制。其中ORB是CORBA规范的核心部分，其体系结构如图2所示：

CORBA 规范的特点是：

（1）引入了代理（Broker）的概念。一个代理至少有三个方面的作用：完成对客户提出的抽

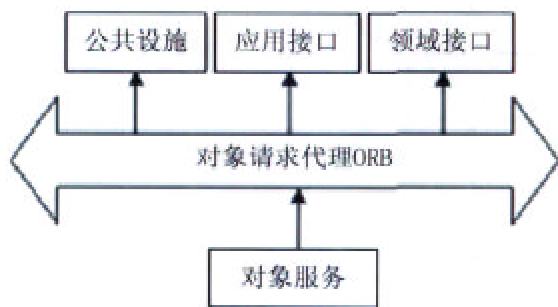


图1 OMA参考模型

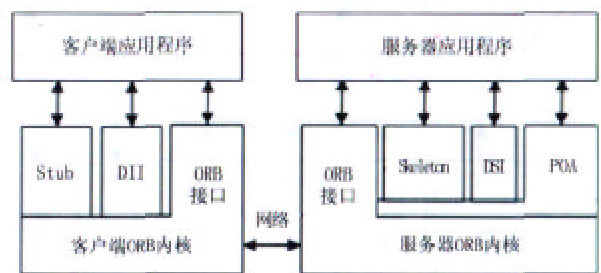


图2 CORBA体系结构

象服务请求的映射；自动寻找和定位服务器；自动设定路由,实现到路由器方的执行。这样用户在编制客户程序时就可以避免了界面过多,而只需完整的定义和说明客户需要完成的任务和目标。

(2) 实现了客户端程序与服务器端程序的完全分离,与面向过程调用机制为基础的客户/服务器模式有本质不同,客户将不再同服务器发生直接联系,而仅仅需要同代理进行交互。客户与服务器之间就可以有更加灵活的关系存在。

(3) 将面向对象的概念引入分布式计算中,提高了软件可移植性、可扩展性等性能。

(4) 提出软件总线机制。所谓软件机制总线是指CORBA规范定义了一组接口规范,任何应用程序、软件系统或工具只要具有与该接口规范相符合的接口定义,就能方便的集成到CORBA系统中,而这各组接口规范独立于任何实现语言和环境。

2. 基于CORBA的机器人模块化设计的一般步骤

2.1 硬件抽象: 在控制系统中,将硬件对象抽象映射成一般性的通用接口,屏蔽了硬件诸如系统平台、通讯协议(如RS232、1394、USB2.0等)不同的底层细节。这种接口映射使上层模块能方

便的获取和管理不同的硬件资源,而不用考虑底层硬件和操作系统的差异。基于CORBA的机器人组件系统一般性结构如图3所示。

2.2 功能划分: 根据需要将硬件抽象获得的不同功能接口进行整合,按照逻辑关系划分成不同的功能组件,即面向系统内不同子任务的模块。功能组件是系统中基本单位,通过IDL语言定义接口集中的接口进行通信,向系统中其他相关组件提出要求获取信息,以及对外提供服务。功能组件分为两大类,一类称为系统模块,由CORBA中间件负责实现,提供建立CORBA分布式系统的最基本功能如命名服务、事件服务和生命周期服务等;一类称为用户开发模块,由用户自行开发的功能模块,如人机交互模块、数据处理和环境感知模块等。

2.3 各类资源的组织管理: 系统调度管理模块对系统中各个功能模块进行组织、调度和管理。

2.4 系统工作的数据流通关系: 确定这个系统各部件的工作时序。

2.5 系统集成调试与运行。

3. 几种典型的基于CORBA机器人系统

3.1 MIRO

德国Ulm大学开发的MIRO是一套基于CORBA

的移动机器人多层编程控制平台。MIRO系统结构如图4所示,分为设备层、服务层和框架层3部分,其中设备层负责将各类传感器和执行器进行接口抽象,提供各类标准设备接口,将底层与不同控制器板卡间基于不同协议的物理链接映射为服务层能够调用的标准方法。服务层负责实现设备抽象,

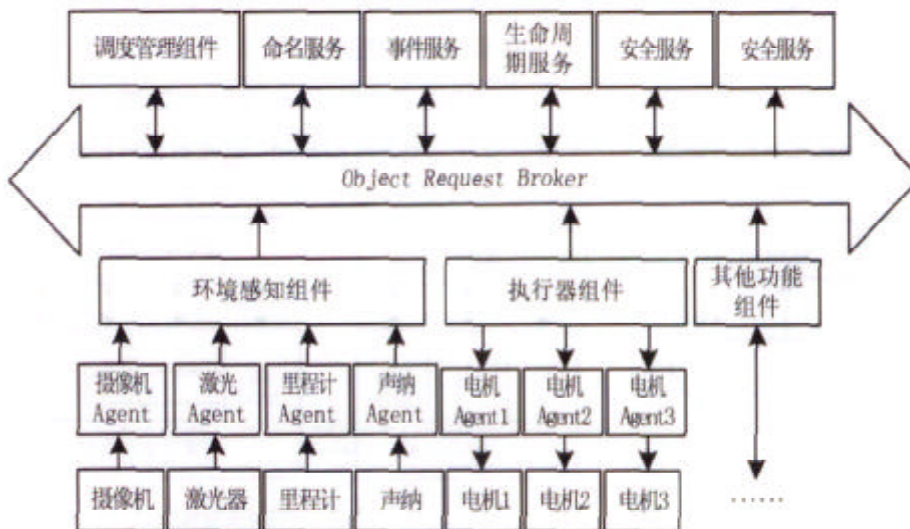


图3 机器人组件系统一般性结构图

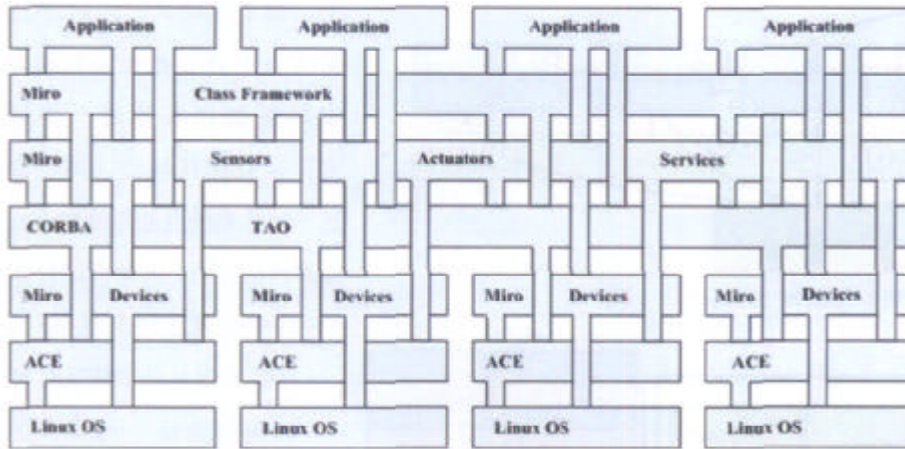


图4 Miro系统结构图



图5 B21机器人和Sparrow-99机器人

它依据实际设备对设备接口进行组合、管理,进一步封装为一些特定的传感器和执行器等设备。框架层包含不同的功能模块,提供系统级的应用服务,包括机器人环境地图获取、定位、行为生成和路径规划等。平台在ACE(Adaptive Communication Environment)环境中运行,采用CORBA的通知服务实现基于事件的通讯功能。目前该平台应用到不同机器人硬件平台(B21和Sparrow-99机器人参见图5)实现室内导航、室外救生和机器人足球等任务。

3.2 MARIE

MARIE(Mobile and Autonomous Robotics Integration Environment)是基于仲裁模式的分布式集成开发平台。MARIE是利用中间件整合现有机器人软硬件模块的又一个有代表性的系统平台,使用ACE作为其通讯类库,模块间采用异步通讯模式。它采用3层式系统结构,最低层称为核心层,负责通讯、数据处理,分布式计算及输入/输出等底层控制。中间为部件层,负责系统各模块内各

类资源的管理及使用。第三层为应用层,负责系统中模块的加载以及模块间的协同工作。目前,MARIE已经集成了地图创建、定位导航、任务规划、声音辨识与跟踪、基于触摸屏的图形化人机交互等功能,并应用到Spartacus机器

人上完成一些简单的任务。

3.3 OpenHRP

日本产业技术综合研究所AIST、东京大学和制造科学与技术中心MSTC联合开发了开放式类人机器人平台OpenHRP(Open Architecture Humanoid Robotics Platform),主要针对各种类人机器人实现运动学规划、碰撞检测、动力学计算等方面的仿真与控制。系统将类人机器人仿真及控制涉及的功能规分为5类,分别用5种独立的CORBA服务器提供相应服务,包括模型解析器、控制器、动力学计算器、碰撞检测器和在线显示器(系统结构和工作流程如图6所示)。模型解析器负责载入VRML文件中机器人与环境其他实体的3D信息,在仿真环境中构建对应的各种模型;碰撞检测器计算实体间的相对位置,检测是否发生碰撞;动力学计算器用来进行机器人前向动力学模型的计算,得到机器人不同时刻的位姿信息;控制器负责与用户进行交互,从而确定机器人需要执行的动作序列以控制机器人的运动;在线显示器用来显示机器人位姿信息。系统通过ORB将各服务器灵活连接起来,即可以支持虚拟仿真,又能对机器人进行实际控制(2种模式下的结构图如图7所示)。OpenHRP能够支持多种现有的类人机器人产品,并支持用户进行二次开发。

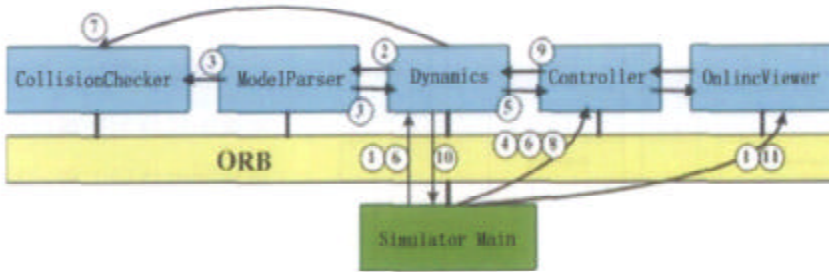


图6 OpenHRP系统结构

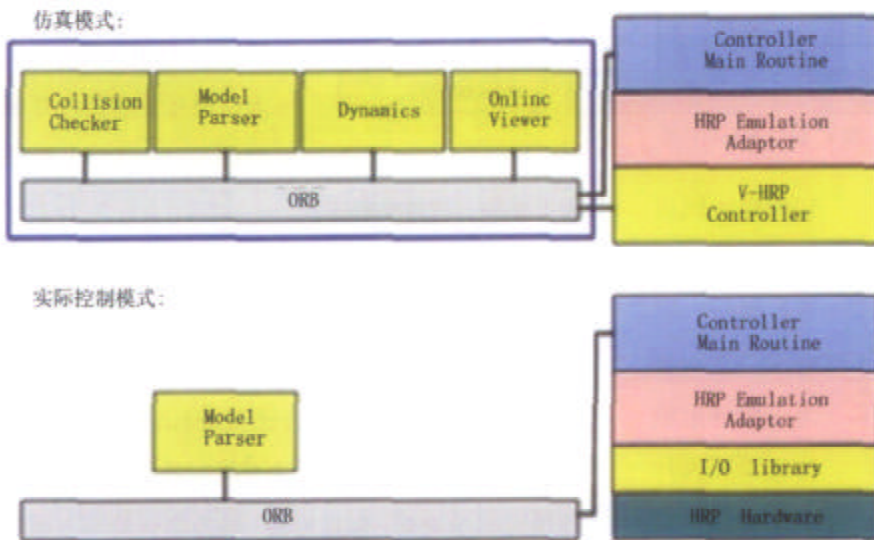


图7 仿真与实际控制模式下的系统结构图

上、以及模块执行的时序逻辑关系上需要相互协调和调度的机制仍有待进一步研究；从互操作性上，高层缺乏一种从系统性能角度对于互操作性的分析，例如再评价系统中无缝连接的性能时，

往往需要从功能模块性能的角度验证。而现在的机器人系统正需要一种模块性能的验证机制。动态结构自组织方面，当系统处于不同的工作状态或外界环境，如何适时灵活的改变自身的结构，即有选择性的激活必要的功能模块，关闭无用的模块，从而提高系统的工作效率，今后仍然需要

4. 存在的问题

对于新的非结构环境下执行复杂任务的要求，需要机器人系统实现跨系统、跨平台、跨语言、跨设备的进行资源整合与共享，导致系统在规模、复杂度、功能上的极大增长，迫使控制系统要向异构协同工作、各层次上集成、可反复重用的道路上前进。为适应这种需求，CORBA规范被引入到机器人系统集成中，支持分布式计算、模块化和构件化集成，使软件类似于硬件一样，可用不同的标准构件拼装而成。目前机器人模块化设计的研究多集中在特定应用环境下某个特定机器人系统，主要体现在底层硬件抽象、系统内功能划分、接口定义，资源管理等方面，而对系统各模块的可扩展性、互操作性及动态结构自组织方面研究较少。如系统的可扩展性，当新的功能模块加入后，系统需要在资源使用上、不同功能模块的调用关系

进行深入研究。

5. 结束语

CORBA技术与机器人技术的有机结合，是机器人集成与模块化深入研究和发展的一种必然趋势。当前，机器人领域基于CORBA模块化技术在硬件抽象、功能划分、资源管理等方面取得了一定的进展，并成功地应用到一些实际机器人系统中。在今后的研究中，需要从理论上进一步完善机器人模块化设计技术，与新的理论结合，研究新的智能控制结构，扩展新的应用领域。可以预计，基于CORBA中间件技术在机器人模块化和集成控制中具有更加广阔的应用前景。

参考文献（共18篇略）