

基于 Socket 通信的工业机器人监控系统研究

徐建明, 潘湘飞

(浙江工业大学 信息工程学院, 杭州 310023)

摘要: 针对 Epson-G6 和 Staubli-TX90 工业机器人, 研究一种基于 Socket 通信的工业机器人监控系统; 系统分为本地监控端和 Web 远程监控端; 基于 TCP 协议和 Socket 通信技术, 编写了机器人控制器服务端程序和 MFC 类库开发的客户端程序, 客户端分为通讯功能模块, 末端坐标轨迹存储模块和基于 OpenGL 类库的运动姿态展现模块; 在此基础上, 基于 J2EE 架构和 WebSocket 通信协议编写了 Web 远程监控端程序, 分为用户管理模块和实时监控模块; 系统实现了机器人末端坐标轨迹的采集并远程存储至 PostgreSQL 数据库, 同时通过逆运动学计算展现了机器人三维运动姿态, 并在 Web 页面以动态曲线方式监控机器人的末端坐标轨迹。

关键词: 工业机器人; Socket 通信; J2EE 架构; WebSocket 协议; 远程监控

Research of Industrial Robot Monitoring System Based on Socket Communication

Xu Jianming, Pan Xiangfei

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: Aiming at Epson-G6 and Staubli-TX90 industrial robots, a kind of industrial robot monitoring system based on Socket communication is studied. It includes a local monitoring terminal and a web remote monitoring terminal. Based on TCP protocol and Socket communication, the server-side in robot controllers and the client using MFC Framework are respectively programmed. The client contains the communication module, the end coordinate trajectories memory module and the motion posture presentation module based on OpenGL class library. On this basis, the web remote monitoring terminal is designed with J2EE architecture and WebSocket protocol, it is made up of a user management module and a real-time monitoring module. The system realizes the acquisition of the end coordinate trajectories of the robots, the storage of the trajectories to a PostgreSQL database and the monitoring of the trajectories by dynamic curve on web pages. At the same time, the 3D motion attitudes of two robots are also presented by the inverse kinematics calculation.

Keywords: industrial robot; Socket communication; J2EE architecture; WebSocket protocol; remote monitoring

0 引言

随着工业自动化发展, 在工业生产中, 工业机器人已经成为一种像数控机床一样必不可少的设备。中国制造 2025 规划提出中国制造业需要逐步升级到工业 4.0 时代。工业 4.0 的显著标志是“互联网+制造业”, 其中重要一环便是推动工业机器人在工业生产中的大量应用, 并通过互联网络实现机器人间的互联互通。因而研究针对工业机器人的远程监控系统显得越来越重要。

目前工业自动化领域对监控系统的研究多采用 C/S 模式。张爱民等人基于 Socket 套接字设计了一种工业机器人远程监控与诊断系统^[1], 实现对现场工业机器人运行状态信息的查看。然而采用 C/S 模式的系统, 客户端必须安装相应的软件, 使得系统的扩展性和维护性不高。饱胜利等人提出一种采用 B/S 模式基于 J2EE 架构的实时监控系统, 实现对卷包生产车间设备实时数据的远程监控^[2]。浙江大学的骆晓娟等人提出一

种采用 Ajax 和 B/S 构架的实时监测系统, 基于 .Net 平台利用 Ajax 技术在 Web 页面实现对监测数据的实时动态显示^[3]。采用 B/S 模式的系统, 软件安装, 维护和升级更加容易, 客户端只需拥有浏览器即可访问系统。如果将 B/S 模式引入基于传统 C/S 模式的机器人监控系统能够大大提高系统的维护性和扩展性。

本文以 STAUBLI-TX90 和 EPSON-G6 为监控对象, 基于“C/S+B/S”模式研究一种工业机器人远程监控系统。系统分为本地监控端和 Web 远程监控端, 本地监控端基于 TCP 协议, 采用 Socket 编程接口和多线程架构实现与工业机器人的通信。Web 远程监控端基于 J2EE 架构, 后端程序采用 SSH 框架和 WebSocket 通信协议编程, 前端 JSP 页面采用 JQuery EasyUI 框架设计。页面基于 Ajax 异步方式请求非实时数据, 基于 WebSocket 通信协议请求实时监控数据。WebSocket 是 HTML5 新标准中的 Web 通信机制, 能够在浏览器和服务器间建立长连接, 实现稳定全双工实时通信, 相对于 Ajax 轮询方式更加高效^[4]。

1 系统整体架构

整个系统对 STAUBLI-TX90 和 EPSON-G6 工业机器人的监控分为两个部分: 本地监控和 Web 远程监控。整个系统连接框图如图 1 所示。

收稿日期: 2016-12-19; 修回日期: 2017-02-27。

基金项目: 国家自然科学基金面上项目(61374103); 浙江省自然科学基金重点项目(LZ15F030003)。

作者简介: 徐建明(1970-), 男, 江西吉安人, 教授, 硕士研究生导师, 主要从事工业机器人和视觉伺服控制技术方向的研究。

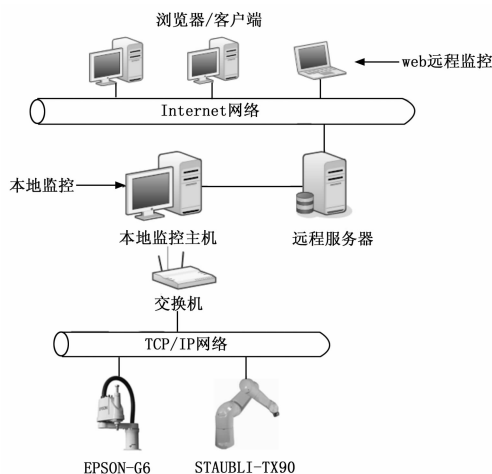


图 1 系统连接框图

本地监控部分将 STAUBLI-TX90 控制器和 EPSON-G6 控制器通过以太网端口接入交换机，本地监控主机与交换机通过网线连接，基于 TCP/IP 协议，使用 Socket 采集机器人末端坐标轨迹并存储到远程服务器的数据库中；Web 远程监控部分由远程服务器和浏览器客户端组成，机器人实时数据监控采用 WebSocket 通信机制。使用 Tomcat7 作为 Web 应用服务器，部署基于 J2EE 架构开发的远程监控程序。程序部署完成，用户通过浏览器能够实时访问机器人的监控数据。

2 本地监控实现

本地监控端采用 C/S (客户/服务器) 模式，将机器人控制器作为通信服务端，将基于 MFC^[5] 类库开发的 C++ 程序作为通信客户端，C++ 语言操作硬件更加容易，而且软件易于维护。MFC 客户端主要有 3 个核心功能：与工业机器人的通信，工业机器人末端坐标轨迹的远程存储，工业机器人三维运动姿态的实时展现。

2.1 机器人通信服务端实现

将 STAUBLI-TX90 和 EPSON-G6 的机器人控制器作为通信服务端，首先打开网络通信端口，设置控制器为服务端，侦听客户端的连接，使程序变为阻塞状态，一旦检测到客户端连接完成，程序继续执行，创建两条同步任务，在机器人运行到目标点的同时发送机器人当前实时末端坐标给客户端。

2.2 MFC 客户端实现

2.2.1 通信功能

MFC 监控客户端通信功能实现基于 Socket 接口编程，程序架构为多线程架构。MFC 中线程分为两种：界面（用户接口）线程和工作线程。前者用来独立地处理用户输入和响应用户事件，后者用于后台任务^[6]。监控客户端在界面线程创建套接字，获取机器人控制器地址和端口号，建立与机器人的连接，在工作子线程完成接收数据工作，提高系统效率。同时使用两个套接字分别与两台工业机器人建立连接，彼此间互不影响。监控客户端多线程通信的程序流程图如图 2 所示。左侧为界面线程的程序流程图，右侧为机器人工作子线程的程序流程图。

界面线程和工作子线程间通信基于 MFC 消息映射机制。

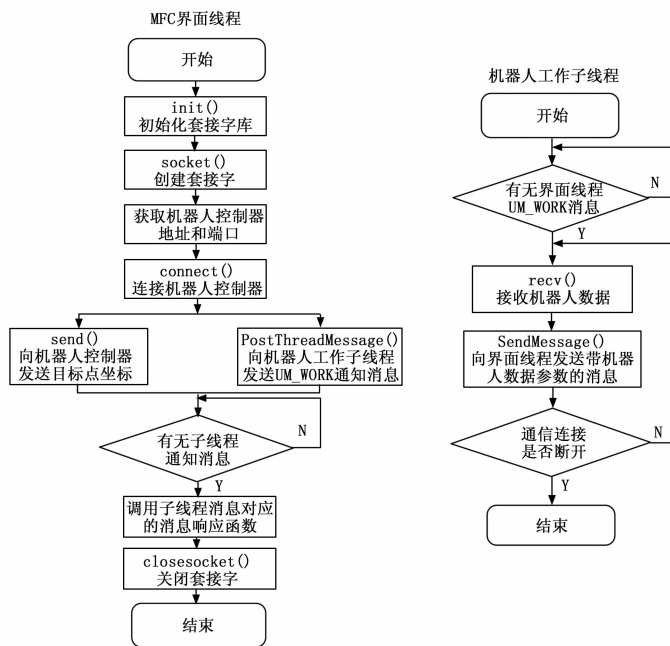


图 2 MFC 客户端多线程通信流程图

当界面线程与机器人控制器连接通信成功后调用 PostThreadMessage () 函数向工作子线程发送 UM _ WORK 通知消息，并将该消息放入消息队列，工作子线程在 While 循环条件中通过 PeekMessage () 函数不断检查消息队列中是否有界面线程的 UM _ WORK 通知消息，一旦检测到此消息，使用 Recv () 函数将接收到的机器人末端坐标数据封装到 RecvBuf () 字符数组中，使用 SendMessage () 函数将 RecvBuf 作为消息参数发送通知消息到界面线程的各功能模块。根据消息映射机制界面线程会调用对应的消息响应函数。

子线程向界面线程发送的通知消息主要有 3 个：WM _ RECVDATA 消息调用主窗口的 OnRecvData () 函数显示采集到的机器人末端坐标数据；WM _ SQLWRITE 消息调用主窗口的 OnSqlWrite () 函数，向远程数据库写入采集到的机器人末端坐标数据，实现机器人末端坐标轨迹的远程存储；WM _ OPENGLRECV 消息调用运动学子窗口的 OnOpenGLRecv () 函数，实现机器人三维运动姿态的实时展现。

MFC 客户端主窗口界面如图 3 所示。界面上方为 EPSON-G6 通信部分，下方为 STAUBLI-TX90 通信部分。在通信信息中设置机器人控制器的 IP 地址和端口号，在界面右侧设置 EPSON-G6 和 STAUBLI-TX90 机器人的目标坐标位置。

2.2.2 末端坐标轨迹存储

监控客户端将机器人在笛卡尔空间的末端坐标轨迹存储到远程服务器的 PostgreSQL 数据库中，PostgreSQL 相对于 MySQL 数据库而言，在高并发和大数据量情况下处理速度更快。相对于 ODBC，DAO 等数据库访问技术，ADO 访问数据源更加高效方便并支持远程数据库的访问，因此监控客户端采用 ADO 方式访问远程数据库^[7]。

首先在数据库中创建两张机器人数据表 epson 和 staubli，分别用来存储 EPSON-G6 和 STAUBLI-TX90 的末端坐标轨迹。调用 AfxOleInit () 函数初始化 OLE/COM 库，创建 _

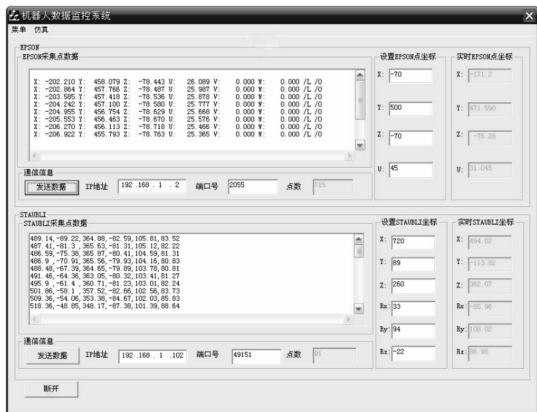


图 3 主窗口运行界面

ConnectionPtr 对象, 通过该对象的 Open () 方法建立与远程数据库的连接, 当子线程向界面线程发送 WM_SQLWRITE 消息时, 该消息会调用 OnSqlWrite () 响应函数, 在该函数中通过 _RecordsetPtr 对象的 PutCollect () 方法将机器人末端坐标轨迹写入机器人数据表。

2.2.3 三维运动姿态展现

根据采集到的机器人末端坐标轨迹, 使用 OpenGL 类库和运动学计算展现机器人的三维运动姿态。OpenGL 是一种快速、高质量的开放三维图形程序库, VC6.0 继承了 OpenGL 的图形标准, 方便了程序设计人员实现三维图像的显示[8]。

在运动学子窗口的 OnCreate () 函数中调用 InitOpenGL () 函数设置窗口像素格式, 绘图设备上下文等, 完成 OpenGL 环境的初始化。在 RenderStockScene () 函数中绘制坐标轴, 地面等静态场景的三维模型, 在 RenderRobot () 函数中根据机器人各轴连杆变换关系绘制机器人三维模型, 将机器人各关节角的值定义为全局变量。当工作子线程向界面线程发送 WM_OPENGLRECV 消息时, 系统会调用 OnOpenGLRecv () 消息响应函数, 在该函数中调用逆运动学函数 Invkinematics () 传入机器人末端坐标, 计算出当前时刻机器人各关节角的值, 通过 SetJointAngle () 函数将计算出的值设定给关节角的全局变量。使用 InvalidateRect () 函数重绘窗口, 实现机器人三维模型的实时运动, EPSON-G6 和 STAUPLI-TX90 的实时运动姿态分别如图 4 和图 5 所示。

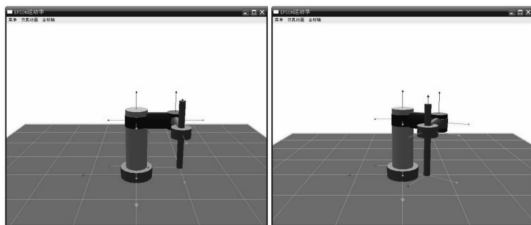


图 4 EPSON-G6 三维运动学姿态

3 WEB 远程监控实现

WEB 远程监控端采用 B/S (浏览器/服务器) 架构, 基于 MVC 模式[9] 设计, 使用 Tomcat7 作为 Web 服务器。后端程序整合 Struts2, Spring, Hibernate 框架[10] 和 WebSocket 通信机制实现, 前端 UI 界面使用 JQuery EasyUI[11] 完成。系统使用

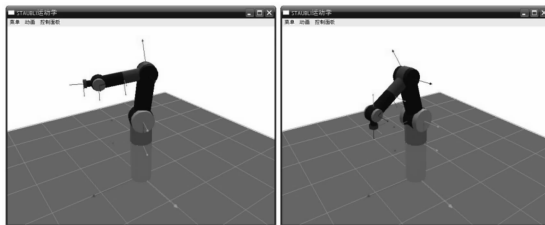


图 5 STAUPLI-TX90 三维运动学姿态

分层思想, 主要分成五层: 数据库层, 数据持久层, 业务逻辑层, 表现层和客户层。远程监控架构框图如图 6 所示。系统主要分为: 用户管理模块和实时监控模块。数据持久层使用 Hibernate 框架, 负责与数据库层的 PostgreSQL 数据库交互, 在 StaubliDao, EpsonDao 和 UserDao 类实现对机器人数据表的查询和对用户表的增删改查; 业务逻辑层使用 Spring 框架, 分别在 StaubliService, EpsonService 类处理实时监控模块的业务逻辑, 在 UserService 类处理用户管理模块的业务逻辑; 表现层使用 Struts2 框架和 WebSocket 通信机制, 处理来自客户层的 Web 请求。实时监控模块使用 WebSocket 协议与客户端建立连接, 将 EpsonSocket 和 StaubliSocket 类作为通信服务端, 用户管理模块使用 Struts2 框架拦截浏览器的 Ajax 异步请求, 将 UserAction 作为请求服务端, 在 Struts.xml 中配置 Url 地址对应的 Action 方法和默认拦截器等。

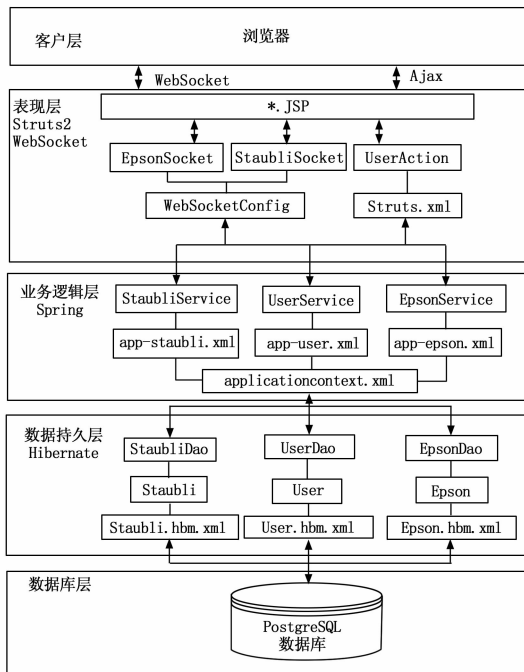


图 6 WEB 远程监控端架构框图

3.1 用户管理实现

用户管理功能包括用户登录校验和用户增删改查。服务端程序使用 Struts2 的拦截器机制完成用户登录校验。在 Struts.xml 中配置自定义 LogonCheckInterceptor 拦截器类, 将 Struts2 默认拦截器栈设置为自定义拦截器栈, 拦截除 myuserLogin () 登录方法外的所有方法。

在自定义拦截器类的 doIntercept () 方法中通过判断 ses-

sion 中的 user 属性是否为空来完成用户登录校验。分别调用 ajaxaddUser (), ajaxdelUser (), ajaxeditUser () 和 easyUI-ListUser () 方法完成对用户表的增删改查, 所有方法都是通过 Ajax 异步方式请求服务端。

3.2 实时监控实现

为了以动态曲线的方式实时展现机器人末端坐标轨迹, 客户端传统做法是用 Ajax 轮询的方式不断向服务器发送异步请求, 但高频率的轮询不仅会加重服务器的负载, 而且会将大量时间耗费在频繁的请求和响应时延上, 使得效率相对低下。系统采用 HTML5 新标准中的 WebSocket 通信机制实现实时通信。服务端使用 Tomcat7 提供的已实现 WebSocket API 接口标准的实现类, 基于注解方式开发。

服务端首先侦听是否有客户端的连接请求, 如果有则建立与客户端的连接通道, 然后继续侦听是否有客户端发送的消息, 如果有则在 WebSocket 的 onMessage 方法中不断推送封装成 JSON^[12] 格式的机器人最新数据到客户端。调用业务逻辑层的 getLastLine () 方法获取机器人数据表最新的一条记录, 调用 JSONArray 的 fromObject () 方法将该条记录转换为 Json 格式, 调用 session 的 sendText () 方法发送 Json 格式的机器人数据到客户端。

浏览器客户端在页面上点击建立连接按钮, 通过新建 WebSocket 对象建立与服务端的连接, 在其构造参数中传入需要连接的服务端 url 地址, 通过 send () 方法向服务端发送通知消息, 在 onMessage () 方法中解析服务端返回的机器人数据。通过 JavaScript 的 eval () 方法将从服务端返回的 Json 字符串转换为 Json 对象。通过该 Json 对象解析出机器人在笛卡尔空间各轴末端坐标并存储到数组中, 将其装载到 Highcharts 图表库动态曲线的数据列上, 每隔 0.5 秒重载数据列。Web 页面上 EPSON-G6 和 STAUBLI-TX90 在笛卡尔空

间的末端坐标轨迹实时曲线分别如图 7 和图 8 所示, 横坐标为当前时间, 纵坐标为机器人实时末端坐标。点击断开连接按钮, 调用 Websocket () 的 close () 方法关闭服务端与客户端的连接通道。

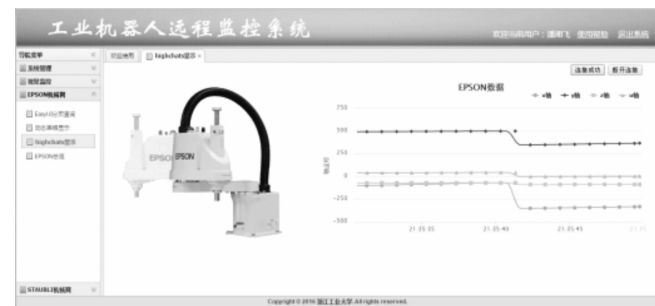


图 7 EPSON-G6 实时动态曲线

4 试验结果与分析

通过试验验证本系统的可行性, 试验结果表明本地监控 MFC 客户端能够实现与工业机器人控制器的通信, 通信效率达到每 50 ms 收发一个数据包, 丢包率不超过 5%。浏览器客户端通过 WebSocket 协议实现了与服务端建立连接通道。依据网络情况好坏, 首次建立连接耗时在 30~40 ms 之间, 连接

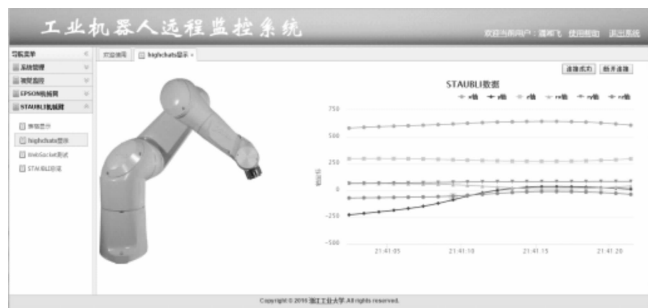


图 8 Staubli-TX90 实时动态曲线

建立完成后, 推送每条数据的网络时延极低, 基本上能保证 Web 页面上监控数据的实时性。试验结果表明借助于 Socket 多线程的通信架构, 程序模块化的设计及 WebSocket 高效的通信效率使得系统具有较高的稳定性和实时性。

5 结论

本文研究了一种基于 Socket 套接字通信的工业机器人远程监控系统。以 STAUBLI-TX90 和 EPSON-G6 为监控对象, 结合 Socket 通信技术, ADO 数据库访问技术, OpenGL 类库, J2EE 架构和 WebSocket 协议, 实现对工业机器人的本地和远程实时监控。市场上大多数工业机器人控制器都有兼容 Socket 通信的以太网接口, 因此本系统的开发在实现对不同类型工业机器人的远程并发监控方面, 具有较大的参考价值。

参考文献:

- [1] 张爱民, 孔得鹏, 王倩. 工业机器人的远程监控与诊断系统设计 [J]. 机械, 2010, 37 (10): 45-47.
- [2] 饱胜利, 钟 勇. 基于 J2EE 的实时监控系统的设计与实现 [J]. 计算机工程, 2008, 34 (5): 240-242.
- [3] 骆晓娟, 许 力. 基于 AJAX 和 B/S 构架的实时监测系统 [J]. 工业控制计算机, 2013, 26 (4): 64-65.
- [4] 薛 陇, 彬刘钊远. 基于 WebSocket 的网络实时通信 [J]. 计算机与数字工程, 2014, 42 (3): 478-481.
- [5] 徐 璇, 姜明新, 黄 静, 等. 基于 MFC 的工程软件界面设计 [J]. 电子设计工程, 2011, 19 (21): 11-13.
- [6] 王 伟. 基于 MFC 的多线程与串口通信 [J]. 电子世界, 2013 (4): 106.
- [7] 索 岩, 崔红志. 基于 MFC 的数据库动态访问技术 [J]. 科技信息, 2009 (31): 46-47.
- [8] 喻 敏, 李成刚, 李富中, 等. 基于 OpenGL 和 MFC 的机器人运动控制及标定 [J]. 机械设计与制造工程, 2015, 44 (1): 46-49.
- [9] 冉春玉, 白炳杉. MVC 模式及 Struts 框架应用研究 [J]. 武汉理工大学学报, 2004, 26 (6): 66-69.
- [10] 赵 洋, 张 丽, 王恩东, 等. 基于 Struts, Hibernate 和 Spring 的 J2EE 架构研究 [J]. 现代电子技术, 2009, 32 (2): 107-110.
- [11] 陈国耀, 王清心. 基于 jQuery 的高校网络心理咨询系统的设计与实现 [J]. 武汉工程大学学报, 2011, 33 (10): 103-106.
- [12] 高 静, 段会川. JSON 数据传输效率研究 [J]. 计算机工程与设计, 2011, 32 (7): 2267-2270.