

ATS 仿真系统网络化的设计与实现

徐越, 郭秀清

(同济大学 控制科学与工程系, 上海 201804)

摘要: 网络化的 ATS 仿真系统能够更好地满足在实际应用中的培训和教学需求。以上海地铁 2 号线 ATS(Automatic Train Supervision)仿真系统为例,在系统功能模块化的基础之上,结合 TCP/IP、UDP 协议和 WSAAsyncSelect I/O 模型,在现有单机版 ATS 仿真系统的基础上设计并开发了网络版,使之可以更好地满足系统的实际应用需求。

关键词: ATS; 仿真; 网络通信; WSAAsyncSelect

中图分类号: TP391.9

文献标识码: A

文章编号: 1674-7720(2013)20-0008-03

Design and implementation of the networked ATS simulation system

Xu Yue, Guo Xiuqing

(Department of Control Science and Engineering, Tongji University, Shanghai 201804, China)

Abstract: Networked ATS (Automatic Train Supervision) simulation system can better meet the teaching and training requirements in practical use. Taking the ATS simulation system of line 2 as an example, based on functional modularization in the system, with UDP, TCP protocol and WSAAsyncSelect I/O model, this paper develops a networked version of the existed standalone ATS simulation system to better meet practical use requirements.

Key words: ATS; simulation; network communication; WSAAsyncSelect

近年来,轨道交通快速进入高速期,成为带动经济增长的重要因素。列车自动监控系统(ATS)是一种智能化自动监控系统,是否对 ATS 系统正确操作,影响着列车能否安全运行,这对轨道交通运营管理人员的后勤培训提出了很高要求。这些综合性运营专业人员不仅需要熟悉 ATS 系统的工作原理,同时还需要具备实际操作能力。考虑到现场行车安全无法在目前已经投入运营的系统上进行教学培训,开发仿真培训系统就成为解决这一问题的有效途径,因此 ATS 仿真系统应运而生^[1]。

单机版 ATS 仿真系统是运行于单独一台主机上并且不与其他主机进行通信的 ATS 仿真系统,它可以提供模拟演示和故障模拟等基本功能。但是在实际的培训和教学中,单机版 ATS 仿真系统仍然存在很多局限性,例如教师无法同时对较多的学员进行示范性操作、无法模拟中央 ATS 和集中站 ATS 的拓扑环境等。这就提出了将现有单机版 ATS 仿真系统网络化的需求。网络版 ATS 仿真系统是在单机版 ATS 仿真系统的基础上,进一步构建基于客户端/服务器 C/S (Client/Server) 模型的网络化通信平台,着重研究基于 TCP/IP 协议下运行在多台主机上的 ATS 仿真系统之间的网络通信。

《微型机与应用》2013 年 第 32 卷 第 20 期

1 ATS 仿真系统网络版通信模块的总体设计

网络版 ATS 仿真系统间的通信不同于单机版 ATS 仿真系统模块之间的通信,它是基于现有的网络通信协议,采用成熟的网络编程接口实现不同主机、不同进程间的数据传输和交互。通过对 ATS 仿真系统的需求分析,将通信模块的设计分为组件状态同步、客户端的操作命令传输和权限管理三部分进行讨论。

1.1 组件状态同步的分析与设计

站场图上的组件状态同步,即将局域网内服务器上组件状态的实时数据发送到客户端,客户端根据接收到的实时数据更新组件的状态。由于可能同时会有多个客户端需要接收实时数据,如果采用 TCP(Transmission Control Protocol)连接的方式,将消耗很多服务器的资源,导致服务器处理速度变慢,同时也会加重网络的负担。并且客户端并不需要发送数据给服务器,只要保证接收到的数据的完整性即可,不需要为每一个客户端建立一个与服务器的 TCP 连接。所以,综合考虑上述因素,本文采用 UDP(User Datagram Protocol)广播的方案。UDP 广播具有资源消耗少、传输效率高等优点,能够满足这里的实时通信要求。

欢迎网上投稿 www.pcachina.com 9

软件天地 Software Technology

1.2 客户端操作命令的传输的分析与设计

中央 ATS 起着全局监管的作用,一般会把集中站的具体操作控制权交至区域控制中心。由于区域控制中心的操作需要被全局看到,包括中央 ATS 和其他区域控制中心/集中站,并能够在中央 ATS 留下操作记录。所以客户端对系统的操作并不是直接作用于本机的系统,而是将操作命令传给中央 ATS,由中央 ATS 执行命令,并将执行的结果(如组件状态的改变)广播给所有的客户端。这样便可保证区域控制中心不会越权操作,也可使各个系统之间保持一致。由于客户端与中央 ATS 之间的通信属于端到端的通信,并且需要严格保证传输数据的完整性,这就需要选择一个面向连接的、可靠的通信协议。TCP 协议是一个面向连接、可靠的通信协议,它通过自身实现的数据确认和超时重传等机制,保证了数据传输的高可靠性,并且客户端发送的操作命令的数据较小、在同一时刻也不会有很多客户端需要与中央 ATS 进行通信,并不会耗费太多系统资源,所以这里采用面向连接的 TCP 协议实现。

1.3 权限管理的分析与设计

客户端不仅仅需要作为固定的网络终端来接收数据,同时也要能够模拟集中站管理员对系统的操作。主要涉及有:用户账号和密码管理、操作权限的分配和回收、操作权限的验证和操作命令的代执行等。由于数据传输量较小,并需要保证数据传输的可靠性,故也采用面向连接的 TCP 协议实现。综合以上讨论,网络版 ATS 仿真系统间网络通信实现可以用图 1 所示的示意图来表示。

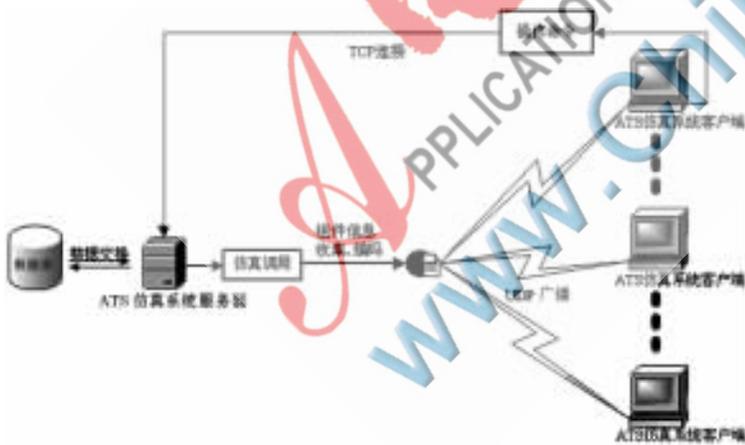


图 1 网络版 ATS 仿真系统间网络通信实现示意图

2 网络通信模块的软件实现

由于 ATS 仿真系统是一个在 Windows 平台基于 MFC 框架的桌面应用程序,同时需要处理与多个客户端之间的通信问题,所以在进行具体的软件实现之前,选择一个合适的 I/O 模型尤为重要。在现有的

通信模块中,很多都是基于阻塞模型,其优势在于简单直接,其缺点在于通常需要耗费较多的系统资源,同时如果涉及到多线程数据交互更是需要进行繁琐的同步工作。异步通知模型则可以高效解决这种情况下的并行处理问题^[2]。WSAAsyncSelect(异步选择)模型是 Winsock 提供的一个异步 I/O 模型,通过这个模型,应用程序可以接收制定套接字上的网络事件通知,这些异步通知通过窗口消息的方式传递。通过对相应消息的处理,应用程序即可实现对网络连接的控制,最突出的一方面是它可以在系统开销不大的情况下同时处理许多连接。综合考虑这些因素,本文采用 WSAAsyncSelect 模型。

2.1 协议的设计

要实现不同主机、不同进程之间的通信,首先要解决进程间的通信协议设计问题。在网络化的 ATS 仿真系统间的通信内容主要是组件状态和操作命令两种。下面就对这两种传输内容分别进行设计。

2.1.1 组件状态的协议设计

通过对 ATS 仿真系统的分析,客户端需要保持与服务器同步所需要的数据有三种:组件的编号(如编号为 123 的信号机)、组件的状态类型(如信号机的颜色)、组件的具体状态(如信号机的颜色为绿色)。所以可以将这三种数据整合成一个分组,组件的状态编码的封包格式如图 2 所示。

2.1.2 操作命令的协议设计

由于操作命令的多样化,所以将协议设计成可变的、无固定长度的分组。每一个分组包括命令类型和命令内容。命令内容会因不同的命令类型而有所不同,最后以一个‘\0’作为结束符。这样既可以方便拆包又能处理 TCP 数据流中的粘包问题^[1]。操作命令的封包格式如图 3 所示。

2.2 类设计

由于服务器端和客户端的 ATS 仿真系统都采用 WSAAsyncSelect I/O 模型,并且都有一些相同的操作,如创建套接字、创建关联窗口、关闭套接字等,所以这里封装一个 CATSGenericConnector 基类,该基类再分别派生出服务器类 CATSServer 和客户端类 CATSClient。服务器类实现了 Startservice() 和 Broadcast() 等方法,分别用于启动客户端等待客户端连接和向客户端广播组件状态信息。客户端类实现了 ConnectServer() 和 Send() 等方法,分别用于连接客户端和向服务器发送操作命令。



图 2 组件状态的封包格式

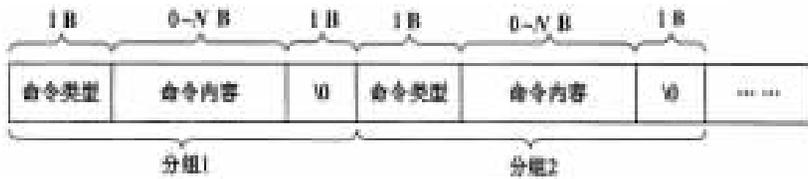


图3 操作命令的封包格式

2.3 UDP 广播的实现

UDP 广播主要是为在动态仿真过程中将服务器上运行的 ATS 仿真系统的实时信息发送到每一个客户端，客户端根据接收到的数据更新系统上的组件状态，以达到与服务器保持同步的目的。服务器在启动时将设置一个定时器，以一个固定的时间间隔作为一个传输周期，在每个传输周期内，服务器将改变的组件状态按照图 2 所示的封包格式进行统一打包，并在一个传输周期结束前将封装好的数据包广播给局域网内的每个客户端。客户端则单独开启一个线程 (ListenThread) 用于接收服务器端发送的数据，并对接收到的数据进行解码，更新相应的组件状态。

2.4 TCP 通信的实现

由通信基类 CATSGenericConnector 派生出的服务器类 CATSServer 实现了启动服务器方法 StartService()，当服务器对象调用 StartService() 方法时，服务器创建监听套接字，等待客户端的连接，并调用 WSAAsyncSelect() 与仿真系统的窗口关联。客户端调用 ConnectServer() 与服务器进行连接，并调用客户端类 CATSClient 的 Send() 和 OnRead() 方法收发数据。在客户端连接到服务器之后，服务器会将客户端的连接信息保存在一个结构 ATSCli-clientInfo 中，该结构记录了客户端的已连接套接字、IP 地址、端口号、用户名和密码等信息，用于与客户端进行收发数据、权限验证等工作。

2.5 编码和解码的实现

网络化的 ATS 仿真系统对传输数据的编码解码也要分为两个部分进行讨论。首先，对于组件状态的传输，服务器首先按照图 2 所示的封包格式对组件状态数据进行编码，客户端在接收到数据后对数据进行解码；其次，客户端会将操作命令和一些用户名、密码等信息发送到服务器端，由服务器端进行解析、执行，这样就要求客户端按照图 3 所示的封包格式，对所需要发送的数据进行编码，服务器对所收到的数据进行解码、执行、验证权限等操作。

为了满足上述要求，定义一个抽象基类，该类声明了编码和解码的接口，具体的接口实现由派生类完成。编解码类的 UML 结构如图 4 所示。

2.6 测试结果

在局域网内启动服务器上的 ATS 仿真程序，同时也在其他多台主机上的 ATS 仿真系统上以客户端用户登录，

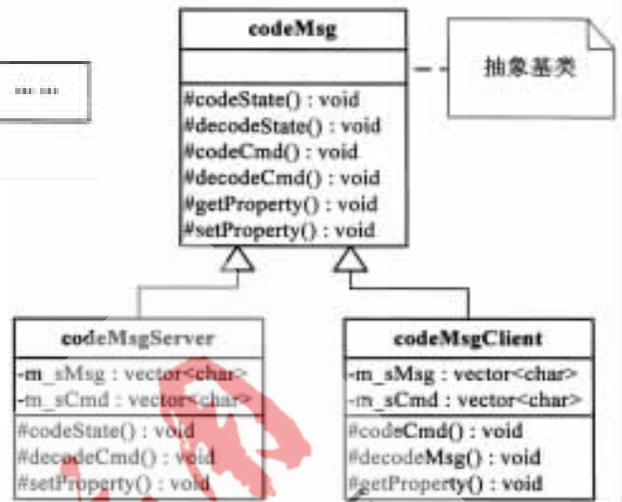


图4 编解码类的UML定义

所有客户端上的仿真程序均能立即收到服务器所发出的数据，仿真画面保持一致，没有出现不一致或者延迟的现象。客户端的操作亦能准确发送到服务器上，并且能够将操作的结果迅速地反映在多台客户端上的仿真程序上，测试达到了预期效果。

本文提出了一个将 ATS 仿真系统网络化的方案，以满足实际应用中对于 ATS 仿真系统的应用需求。该方案综合考虑了对于仿真同步和模拟中央 ATS/集中站等对通信效果的不同要求，分别应用 UDP 和 TCP 协议，并采用基于窗口的 WSAAsyncSelect I/O 模型，设计并成功地将现有的 ATS 仿真系统的单机版网络化，从而有效提高了培训效率。

参考文献

- [1] 孙志勇, 陈永生. ATS 仿真培训系统列车模拟运行的设计与实现[J]. 微型机与应用, 2012,31(14):7-9.
- [2] 邱俊源, 张跃. 异步消息驱动安全通信模块的设计与实现[J]. 计算机工程与设计, 2011,32(8):2580-2581.
- [3] 贺鹏. 城轨列车运行自动监控系统的多 Agent 模型[J]. 计算机应用与软件, 2010,27(12): 68-69.
- [4] 谢小河, 郭秀清, 郭玉臣. 基于 ATS 仿真系统的网络通信模块的研究与设计[J]. 机电一体化, 2012(8):53-54.

(收稿日期: 2013-07-17)

作者简介:

徐越,男,1988年生,硕士研究生,主要研究方向:城市轨道交通运行仿真。

郭秀清,女,1965年生,副教授,硕导,主要研究方向:过程控制与计算机控制。