

文章编号: 2095-2163(2020)07-0150-05

中图分类号: TP393

文献标志码: A

# 一种基于 SDN 的自适应网络仿真平台

陈晓壮, 席亮, 李鸿鹄, 林中霖, 胡琮梅, 刘思岐

(哈尔滨理工大学 计算机科学与技术学院, 哈尔滨 150080)

**摘要:** 本文提出了一种基于 SDN 的自适应网络仿真平台, 主要包括图形化控制平台、控制器模块以及虚拟网络环境。该平台将控制器模块与图形化控制平台分离, 并将其分别运行在不同的虚拟机中。图形化控制平台通过 RPC 调用控制器模块, 并利用 Json 传输数据; 而对其它模块的调用则通过 GET、POST 提交请求, 到相关页面等方式来获取数据, 并将其设置或显示到图形化控制平台。控制平台通过捕获用户基于任务需求设定的各种配置信息, 生成设置报文并发送到控制器端, 以激发网络进行自适应调整。

**关键词:** 软件定义网络; 仿真平台; 任务驱动; 自适应

## An adaptive network simulation platform based on software defined network

CHEN Xiaozhuang, XI Liang, LI Honghu, LIN Zhonglin, HU Congmei, LIU Siqi

(School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

**[Abstract]** An adaptive network simulation platform based on software defined network(SDN) is proposed in this paper, which consists of graphical control platform, controller module and virtual network environment. The platform separates the controller module from the graphical control platform and runs them in different virtual machines. The graphical control platform calls the controller module through RPC, transmits data by Json, calls other modules to obtain data by means of GET and POST submission requests to relevant control interface, and display the control set and results to the graphical control platform. By capturing various configuration information set by users based on task requirements, the control platform generates setting messages and sends them to the controllers to stimulate the network for adaptive adjustment.

**[Key words]** Software defined network; Simulation platform; Task-driven, adaptive

## 0 引言

随着互联网的普及, 用户数量不断攀升, 流量需求与日俱增, 网络不堪重负。负荷过度的网络无法满足用户需求。云计算的出现以及相应业务的迅速发展, 对网络和服务器的计算能力提出了新要求。特别是物联网和人工智能的兴起, 让计算和网络服务的高效、便捷、高速率等要求成为首要需求。然而, 网络体系庞大、架构臃肿、不够灵活, 不能适应不断涌现出的新业务需求, 服务质量得不到保证; 传统网络补丁式的更新方式不能快速全面地更新网络设备。而且, 传统互联网把控制逻辑和数据转发紧耦合在网络上, 导致网络控制管理复杂化。如动态流映射、弹性 IP 地址管理和广泛的 QoS 功能, 也使得网络控制新技术的更新和发展很难直接部署于现有网络上, 其灵活性和扩展性很难适应网络的飞

速发展<sup>[1]</sup>。

为满足网络发展的要求, Stanford 大学提出了 OpenFlow 的概念, 并基于其技术实现网络的可编程能力, 使得网络变得像软件一样可以灵活编程和修改。软件定义网络 (software defined network, SDN) 的概念由此而生<sup>[2]</sup>。不同于传统网络分布式结构, SDN 结构分为应用层、控制层和转发层, 将网络控制层集中, 使用软件化方式实现将网络控制层交付给应用层, 实现了更大程度上的灵活控制<sup>[3]</sup>。SDN 网络具备三大特征: 控制与转发分离、集中控制、开发可编程接口。SDN 是下一代互联网的关键技术, 是国内外网络技术发展的焦点, 各大企业都在研究属于自己的 SDN 项目, 推动未来互联网技术创新<sup>[4]</sup>。

鉴于此, 为满足对未来 SDN 的研究, 更改传统

**基金项目:** 黑龙江省大学生创新创业训练计划项目(201810214111)。

**作者简介:** 陈晓壮(1997-), 男, 本科生, 主要研究方向: 软件定义网络; 席亮(1983-), 男, 博士, 副教授, 主要研究方向: 人工智能及应用、网络与信息安全、软件定义网络; 李鸿鹄(1996-), 男, 学士, 初级工程师, 主要研究方向: 软件定义网络; 林中霖(1998-), 男, 本科生, 主要研究方向: 软件定义网络; 胡琮梅(1996-), 女, 硕士研究生, 主要研究方向: 软件定义网络; 刘思岐(1996-), 男, 硕士研究生, 主要研究方向: 软件定义网络。

**通讯作者:** 席亮 Email: xiliang@hrbust.edu.cn

收稿日期: 2020-05-20

网络的弊端,本文提出一个基于SDN的自适应网络仿真平台,利用Mininet工具集成图形化界面完成复杂网络的自适应搭建,支持手动配置网络和实时数据显示,同时利用ODL(OpenDaylight)控制器在SDN思想基础上对宏观网络环境进行自适应调整与安全测试。该平台可满足科研人员对SDN的各项研究工作,如协议模拟、路由算法、链路自适应、SDN网络攻击与防范等。该平台基于SDN的网络分析与研究,可弥补传统网络交换设备功能单一;网络协议更新周期长;路由算法考虑不周全;无法应对新型网络攻击等问题,减少购买和维护物理设备的费用,为将来SDN的广泛实际应用添砖加瓦。并且,本项目ODL全局控制模块可单独脱离并与SDN物理设备连接,从而为下一步进行真实环境的网络管理提供技术支撑。

## 1 平台设计与实现

### 1.1 相关模型

OpenDaylight(ODL)是一套基于SDN开发的模块化、可扩展、可升级、支持多协议的控制器框架,主要包括Switch Manager、Statistics Manager、Topology Manager、Forwarding Rule Manager、ARP Manager等模块,可对其管理的网络设备进行统一管理与配置<sup>[5]</sup>。鉴于SDN设备较为昂贵,组网成本太高,因此,目前多数有关SDN的科研是基于Mininet展开的。

Mininet是一个轻量级SDN和测试平台<sup>[6]</sup>。它可以完成以一个单一系统模拟完整网络所需的内核系统和用户代码的运行,支持OpenFlow、OpenvSwitch等多种相关协议,有助于互动开发、测试和演示。然而,其图像化程序较为简陋,且运行不稳定,需要采用命令行方式,构建的网络较为抽象。同时,它不能根据环境实时检测网络运行状态和网络数据进行自适应调整,所需的实验数据只能通过间接测量获取,增加了实验的复杂性。

### 1.2 设计思想和架构

本平台基于目前主流的OpenFlow、NetConf交换协议,利用由Linux Foundation和多家网络巨头(如Cisco、Juniper和Broadcom等公司)一起创立的开源项目ODL控制器进行网络宏观自适应调整,并利用Mininet设计并实现。平台的架构方式分为3层:虚拟网络层,控制器层和用户接口层,如图1所示。各层具体工作如下:

(1)虚拟网络层:主要以Mininet作为底层网络的模拟。通过用户接口层接收用户需求,形成网络

拓扑的实时创建或修改,接受控制器层虚拟路由器、交换机的管理方案,实时检测网络运行状态、链路状态等信息反馈到用户接口层。

(2)控制器层:主要利用ODL控制器起到承上启下的作用,接受用户接口层的相关配置,在控制器层执行。同时根据用户接口层,调用北向接口返回数据结果,根据自适应算法生成相关配置,下发到仿真环境层,利用ODL控制器接口进行控制器之间的通信。

(3)用户接口层:主要以图形化界面的方式,面向用户提供服务。主要包括仿真环境的搭建和算法、网络攻击防御等相关测试;提供实验结果数据库接口,可面向专家系统进行分析。

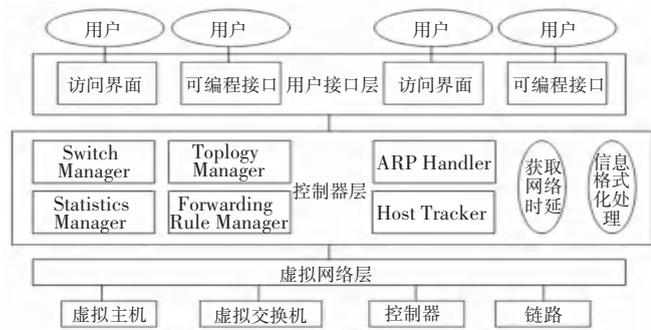


图1 平台总体框架

Fig. 1 The platform framework

### 1.3 具体实现

通过用户接口层应用程序,控制Mininet生成仿真网络拓扑结构,并与其相互联动。然后,通过调用Mininet相关接口对生成设备进行管理 and 自适应控制,及时查看虚拟主机和路由器、交换机的状态及详细信息。如,吞吐量、速率、时延等指标,配合链路检测功能生成故障检测报告。

(1)根据平台用户需要,利用图形化界面生成Mininet可执行脚本,来生成网络拓扑结构。

(2)在拓扑结构的基础上,对网络设备进行随时的自适应管理配置,并进行实时显示。然后,通过对Mininet的API调用,将详细信息反馈到平台,以供用户了解仿真环境的实时状况。

ODL控制器设计与配置:

(1)通过ODL控制器的Routing Manager模块配置文件的方式,设计路由算法;通过调用北向接口,利用OpenFlow协议主动收集网络链路状态信息,提供QoS数据等多因素参与路由算法的生成。

(2)通过ODL控制器Switch Manager模块的调用,实时解控路由器、交换机的状态,并通过Topology Manager模块监听整个链路状态。

(3)利用交换机的流表中计数器,统计数据流

的基本信息。如,匹配该流表项的数据包数、查找次数、收发分组数、生存时间等。利用平台提供的图形化界面查看路由器执行效果。具体包括链路带宽、时延、吞吐量等。

(4)通过抓包程序抓取网络流量,进行数据内容和关键字段提取、统计与匹配。通过流量管理功能,及时与平台通信,并对流量分析,结合流量分类算法完成主流网络攻击识别与响应。

(5)通过记忆功能,支持不同算法执行效果的对比。建立完善的数据库表结构,记录完整的算法参数及结果,为调用专家策略系统提供合理的输入参数。支持图形化界面筛选所需参数或结果,利用SQL语句查询数据库中保存的数据,并形成对比。利用数据库存储事件的方式,定期自动删除多余数据,减少冗余。

#### 1.4 主要功能及基本操作

用户自定义网络拓扑并定义参数,其主要由以下两部分组成:

(1)仿真模块。根据用户需求,图形化网络拓扑结构生成 Mininet 可执行脚本,调用 Mininet 接口对生成设备进行管理。主要调用 `--topo`、`--custom`、`--switch`、`--controller`、`--mac` 等参数;在拓扑结构的基础上对网络设备进行随时的配置更改与实时显示。主要利用 Mininet 的 `dump`、`net`、`nodes`、`links` 查看网络设备状态,利用 `dpctl`、`iperf`、`py`、`link` 等模块对链路进行更改,利用 `ping` 模块对链路进行测试。通过对 Mininet 的 API 调用将详细信息反应到平台,以供用户掌握仿真环境的状况。通过图形化界面控制 ODL 控制器调用 `Switch Manager`、`Statistics Manager`、`Topology Manager`、`Forwarding Rule Manager`、`ARP Manager` 等模块来管理网络和获取网络和网络设备详细信息。

(2)验证模块。主要用来测试仿真平台健壮性和数据的真实性。通过更改 ODL 控制器路由选择算法文件的方式,用网络相关参数来验证路由算法的效率,并给出相应的路由选择意见。通过指定虚拟主机发送特定的数据来模拟网络攻击,分析对 SDN 网络造成的影响,以及建立防范措施。设计流量分类算法,对流量进行分类,提取关键字段与在识别出攻击流量后,自动在 ODL 控制器端制定相应的流表规则,禁止攻击造成更大危害。

通过编程实现自适应,并可模拟其它操作:

(1)路由选择方法。由于传统路由选择算法在设计上只考虑到了最短路径问题,没有从宏观上对

网络整体流量进行考虑,往往处于一个最短路径上的路由发生处理能力下降时,别的路由不能及时发现,造成网络数据的高延迟。本项目在设计路由选择算法时,会将网络内各路由的实时状态、吞吐量和全局网络的网络带宽、QoS 等多因素考虑在内,设计出一个可针对链路进行动态调整的算法。通过比对算法更改前后流量的传输时延、路由器的转发时延等参数,得出新路由由算法的优化效率。

(2)网络攻击模拟。针对于传统的网络攻击方式,在新型 SDN 网络下的影响,指定虚拟主机发送攻击流量,在全局范围内查看网络参数,研究网络攻击的影响。针对传统的网络攻击,测试的内容包括:ARP 欺骗、IP 伪造、DDOS 攻击等。

(3)攻击防御模拟。针对上述的攻击方式,利用流量分类算法提取关键字,建立攻击指纹数据库,对流表匹配项的次数做限制,到达一定匹配数后进行攻击识别。若识别到疑似攻击分析攻击类型,则在仿真平台上给出警告以及解决方法,具体解决方法如下:

①对满足攻击条件的流量,通过 ODL 控制器进行流表下发,对攻击流量进行过滤;

②利用 Host Tracker 组件,对局域网内的攻击主机进行追踪,对其直接相连的路由器或交换机进行 MAC 地址过滤或 IP 地址过滤处理。

## 2 系统演示与测试

### 2.1 平台主要界面

根据以上功能描述,本文研发了基于 SDN 的自适应网络仿真平台。本平台主要包括以下几个界面:拓扑建立及显示界面、Host 配置界面、Switch 配置界面和 Controller 配置界面等,来完成网络拓扑搭建、基于任务驱动的网络自适应调整和网络攻击与防范等网络仿真任务。主要操作界面如图 2 所示。

### 2.2 网络自适应调整实验

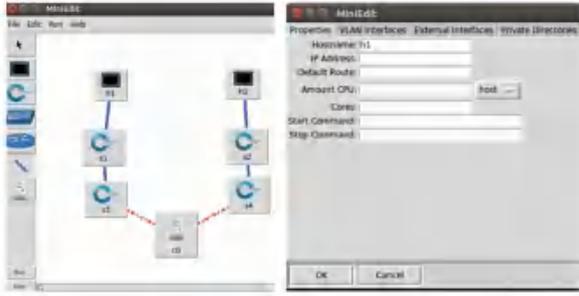
主要通过下发流表实现,如图 3 所示。其中图 3(a)为初始拓扑形式,主机 h1 到 h2 默认路线为:  $h1 \rightarrow s1 \rightarrow s3 \rightarrow h2$ 。当 s1 与 s3 连接链路出现问题时,网络进行了自适应调整,此时路线为:  $h1 \rightarrow s1 \rightarrow s2 \rightarrow s4 \rightarrow s3$ ,图 3(b)为调整后的拓扑形式及 Flow 的显示信息。

### 2.3 攻击仿真实验

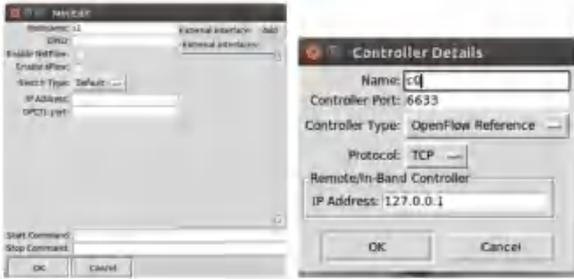
以 ARP 攻击和防范为例<sup>[7]</sup>,演示本平台的攻击与防范的仿真过程和结果。图 4 为网络拓扑图。

(1)攻击过程:设置 h1 (IP: 192.168.134.189) 与 h2 (IP: 192.168.134.190) 通信, h3 (IP: 192.168.

134.191) 进行 ARP 攻击,如图 5 所示。此时 h2 到 h1 的流量,被 h3 监听,如图 6 所示。然后, h1 的 arp 列表中, h2 和 h3 的 MAC 地址已经相同,说明 ARP 攻击已经成功,如图 7 所示。



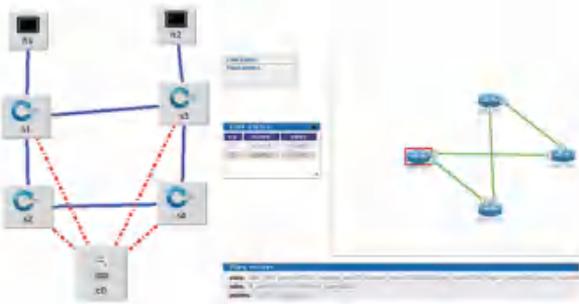
(a) 拓扑建立 (b) Host 配置  
(a) Foundation of topology (b) Host config



(c) Switch 属性 (d) Controller 属性  
(c) Switch property (d) Controller property

图 2 主要操作界面

Fig. 2 Main operation interfaces



(a) 初始网络拓扑 (b) 自适应调整后拓扑及 Flow 显示

(a) Initial topology of network (b) Adaptively adjusted topology and flow

图 3 网络自适应调整

Fig. 3 Network adaptive adjustment

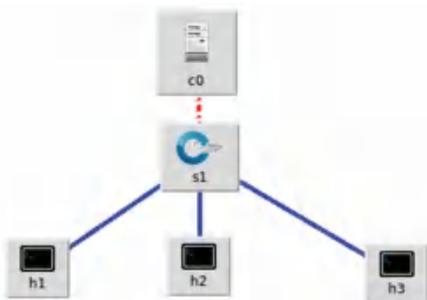


图 4 实验拓扑  
Fig. 4 Topology

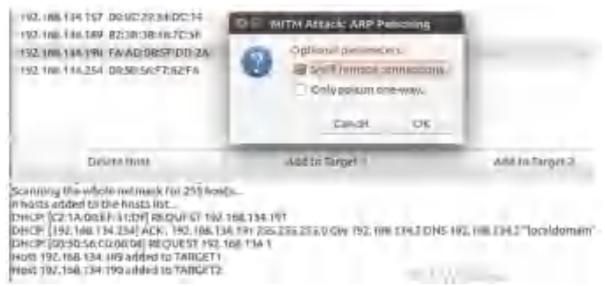


图 5 h3 ARP 攻击  
Fig. 5 ARP attack from h3

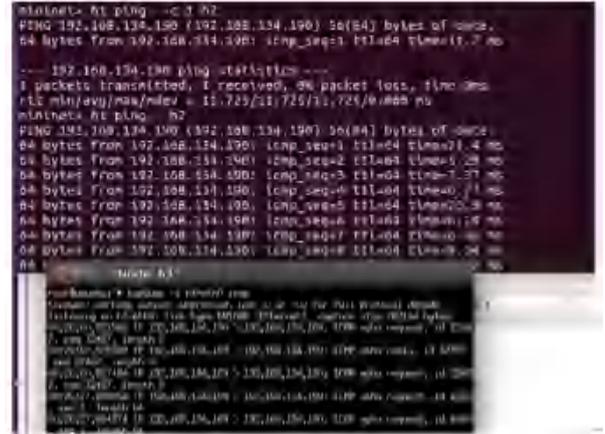


图 6 h3 监听  
Fig. 6 Listening of h3



图 7 h3 的 ARP 攻击结果  
Fig. 7 Results of ARP attack from h3

(2) 防范方法: 通过接口查看流表以了解当前网络的实际状态,如图 8 所示; 根据实际状态, 设定流表, 并通过接口下发流表, 绑定主机 MAC 地址与转发端口, 以此防范 ARP 攻击, 如图 9 所示。

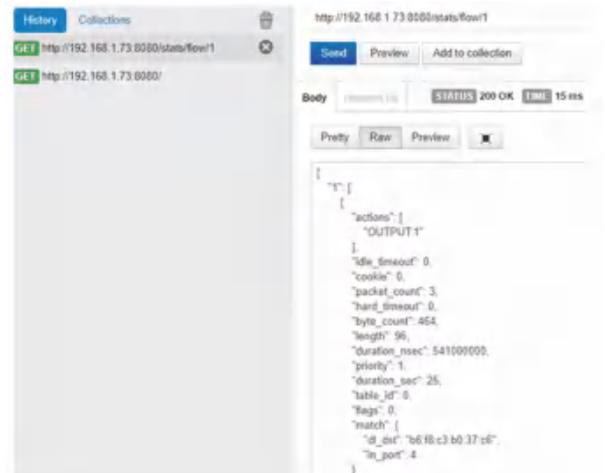


图 8 流表  
Fig. 8 Flow table

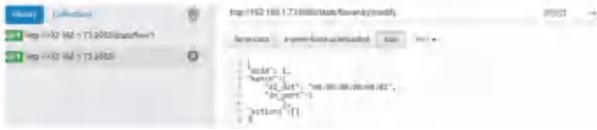


图9 防范设置

Fig. 9 Safety precautions configuration

### 3 结束语

本文设计并实现了一套具有自适应拓扑、网络攻击与防范等功能的 SDN 自适应网络仿真平台。可以自定义网络拓扑及网络参数,动态显示网络参数的改变;可以用来进行模拟真实的 SDN 网络实验,例如网络攻击防范、路由算法的设计等。并且,根据 SDN 网络的可编程性,设计自动调节网络拓扑的算法。该平台可为 SDN 各项科研实践提供仿真平台,极大降低物理设备成本,提供灵活多变的实验验证方法和对比分析手段,为下一步进行真实环境的网络管理提供技术支撑。

(上接第 149 页)

#### 3.4.2 几种波分复用器/解复用器的比较

(1)介质薄膜滤波型波分复用器。优点:光纤参数对器件的设计没有影响;波长间隔数值没有固定要求,信道数灵活;插入损耗、偏振相关损耗不大;温度特性好。

缺点:在 100 吉赫以下时,因为成就频率间隔太困难,信道数被限制;组装设备时间长,信道的数量决定了设备损耗以及造价的高低。

(2)光栅型波分复用器/解复用器。光栅型波分复用器与解复用器的优点:

- ①光栅结构可以精准把握中心反射波长;
- ②反射带宽可以被做的非常小,适合 DWDM 系统;
- ③反射率高达约 100%;
- ④灵活便捷的与常见输送光纤接洽;
- ⑤对偏振不敏感。

光栅型波分复用器与解复用器的缺点是回波反射很高,在使用时必须额外使用光隔离器。

(3)阵列波导光栅型光学波分复用器。AWG 型光波分复用器具有低串扰、可靠性强、小体积、重复性好、低消耗、波长间隔小、信道数多、通带平坦等优点<sup>[8]</sup>,适合用在速度高和容量很大的 WDM 系统

### 参考文献

- [1] JAHANTIGH M N, RAHMANI A M, NAVIMIROUR N J, REZAEI A. Integration of Internet of Things and cloud computing: a systematic survey[J]. IET Communications, 2020, 14(2): 165-176.
- [2] DAS T, SRIDHARAN V, GURUSAMY M. A survey on controller placement in SDN[J]. IEEE Communications Surveys and Tutorials, 2020, 22(1): 472-503.
- [3] ALSAEEDI M, MOHAMAD M M, AL-ROUBAIEY A A. Toward adaptive and scalable OpenFlow-SDN flow control: a survey[J]. IEEE Access, 2019, 7: 107346-107379.
- [4] OKTIAN YE, LEE S, LEE H, LAM J. Distributed SDN controller system: A survey on design choice[J]. Computer Networks, 2017, 121: 100-111.
- [5] KIM T, MYUNG J, YOO S E. Load balancing of distributed datastore in OpenDaylight controller cluster[J]. IEEE Transactions on Network and Service Management, 2019, 16(1): 72-83.
- [6] LANTZ B, O'CONNOR B. A Mininet-based virtual testbed for distributed SDN development[J]. ACM SIGCOMM Computer Communication Review, 2015, 45(4): 365-366.
- [7] SHAH Z, COSGROVE S. Mitigating ARP cache poisoning attack in software-defined networking (SDN): a survey[J]. Electronics, 2019, 8(10): 1-26.

中,已成为目前研究、开发与应用的焦点。

### 4 结束语

本次研究主要根据波分复用系统的基本原理与组成,了解且预设了光纤在未来光通信发展中的战略地位,了解到了光纤接入网在现在电信网中的重要作用,以及光纤通信主要使用的一些技术及其原理。探讨系统的关键器件的工作原理与技术参数,给出了波分复用系统对设备的要求以及波分复用系统的设计。

### 参考文献

- [1] Gerd Keiser, 凯泽, 李玉权, 等. 光纤通信(第三版)[M]. 光纤通信(第三版). 电子工业出版社, 2002.
- [2] Robert Elsenpeter, Toby J. Velte. 光网络实用指南[M]. 北京:清华大学出版社, 2003.
- [3] 张劲松, 陶智勇, 韵湘. 光波分复用技术[M]. 北京:北京邮电大学出版社, 2002.
- [4] 王辉. 光纤通信. 第2版[M]. 电子工业出版社, 2009.
- [5] 纪越峰. 光通信. 光波分复用系统[M]. 北京:北京邮电大学出版社, 1999.
- [6] A. W. Snyder and J. D. Love. Optical Wavelength Theory[M]. Chapman & Hall, New York, 1983.
- [7] 张卫钢. 通信原理与通信技术[M]. 西安:西安电子科技大学出版社, 2003.
- [8] 张言荣. 智能建筑综合布线技术[M]. 北京:中国建筑工业出版社, 2002.