

基于 OMNeT 的电力调度数据网仿真^①

杨宁¹, 李满坡¹, 宋元成², 向勇³, 张伟²

¹(东北电力调度通信中心, 沈阳 110180)

²(东北大学 信息科学与工程学院, 沈阳 110819)

³(中科院沈阳计算技术研究所, 沈阳 110168)

摘要: 结合地区电力调度数据网的组网特点, 针对虚拟专用网 BGP/MPLS VPN 技术原理, 分析了边界网关协议(BGP)和多协议标签交换(MPLS)在 OMNeT++中的实现, 进而基于 OMNeT++仿真环境对地区电力调度数据网给出了一套符合 RFC 标准的 BGP/MPLS VPN 的具体解决方案. 最后给出了 BGP/MPLS VPN 技术在实际场景中的仿真测试, 验证了该方案的可行性.

关键词: 虚拟专用网; 边界网关协议; 多协议标签交换; 电力调度数据网; OMNeT++

Implementation of Simulation using OMNeT in Power Dispatch Data Network

YANG Ning¹, LI Man-Po¹, SONG Yuan-Cheng², XIANG Yong³, ZHANG Wei²

¹(Northeast China Grid Dispatching Communication Center, Shenyang 110180, China)

²(College of information Science and Engineering, Northeastern University, Shenyang 110819, China)

³(Shenyang Institute of Computing Technology of Chinese Academy of Sciences, Shenyang 110168, China)

Abstract: Combining with the characteristics of regional network of power dispatch data network, and according to the technology principle of the virtual private network BGP/MPLS VPN, the ways of implementing Border Gateway Protocol(BGP)and Multiprotocol Label Switching(MPLS)in OMNeT++ were analyzed respectively. A new solution to implement BGP/MPLS VPN for the simulation environment of OMNeT++ was presented which is consistent with the RFC standard for regional power dispatch data network. At last, it gives the measurement of BGP/MPLS VPN simulation in the actual scenario, which proved the feasibility of this solution.

Keywords: virtual private network (VPN); border gateway protocol (BGP); multiprotocol label switching (MPLS); power dispatch data network; OMNeT++

面对日益庞大复杂的电网结构, 需要保证在电力调度数据网上的数据传输的安全性、可靠性同时满足对电网的统一调度与管理. 而 BGP/MPLS VPN 技术能够充分利用公用骨干网络强大的传输能力建立全网结构的数据专线, 保证了电力调度数据网的数据传输要求, 同时能满足不同业务的区分、隔离及 QoS 保证^[1].

BGP/MPLS VPN 是一种规模大且相当复杂的 VPN 技术, 特别是在规模较大的基于 BGP/MPLS VPN 技术的地区电力调度数据网中进行业务的分析与故障的监测等研究将相当困难, 任何网络拓扑调整(增加/

减少节点)、网络业务调整(增加新的网络业务)等都有可能改变网络的安全、可靠、性能等特性. OMNeT++ 是一款优秀的网络协议仿真软件, 近年来, 在科学和工业领域里, OMNeT++ 已逐渐成为非常流行的网络仿真平台, 在国内外已被广泛使用. 利用 OMNeT++ 对 BGP/MPLS VPN 的模型、方法以及协议进行电网调度的科学研究有重要的意义. 基于 NS-2 实现的 BGP/MPLS VPN 技术在平台选择上有相对的局限性^[2]. OMNeT++ 仿真工具中仅实现了协议的简单框架原理, 对于实现这种涉及多种协议的复杂的 VPN 技术需要

① 收稿时间:2013-07-15 收到修改稿时间:2013-08-19

在原基础上进一步的扩展与融合. 为此, 本文根据地区电力调度数据网的 BGP/MPLS VPN 技术原理, 从网络拓扑结构设计、VPN 技术实现和网络路由方案等方面, 研究了 MPLS VPN 技术在 OMNeT++ 中的仿真, 最后, 并对 BGP/MPLS VPN 场景进行仿真, 验证了仿真对实际电力调度数据组网的可行性.

1 OMNeT++ 仿真平台

OMNeT++(Objective Modular Network Testbed in C++) 是一款开源的基于组件的、模块化的、开放的、面向对象的离散事件网络仿真工具, 具有很好的 GUI 界面, 并且可以在 GUI 中配置参数, 具备编程、调试和跟踪支持功能, 主要应用于通信网络的仿真, 也成功地应用于其他复杂 IT 系统、排队网络、硬件体系结构领域的仿真. OMNeT++ 支持用户组件库, 实现模块化的灵活重用; 具有面向对象的特性, 允许仿真内核提供扩展; 并且提供了图形化的网络编辑器和网络、数据流查看工具.

与同属于非商业软件的 NS-2 相比, OMNeT++ 具有高度的模块化, 使得它在增加一些协议时不需要重新编译整个源代码, 使用 NED 语言来定义网络的拓扑结构, 对基本的网络模块元件使用 C++ 语言来定义其行为, 且适应于 Windows 和 Linux 平台, 更好的灵活性和可适应性. 并且 NED 文件可以编译为 C++ 代码, 连接到仿真程序中; 世界各地有庞大的用户群体为 OMNeT++ 提供丰富的资源和技术支持^{[3][4]}.

INET Framework 是 OMNeT++ 仿真环境下的一个开源的通信网络仿真包, 该包包含 IPv4、IPv6、TCP、UDP 等协议的实现和一些应用模型. INET 框架中的每个简单模块之间通过发送消息进行通信, 主机、路由、交换机和其他网络设备用复合模块表示, 而这些复合模块由一些简单模块组合而成, 以此来表示协议、应用、和其他功能实体. INET 中包的组织形式按照 OSI 模型的层次结构而形成的. 本文采用 INET 框架 2.1.0 版本, 中的原有 BGPv4 和 MPLS 等标准协议的简单实现^[5]. 因此本文对 OMNeT++ 中 BGP 和 MPLS 等相关协议进行扩展以符合电力调度数据网中的 VPN 技术.

2 BGP/MPLS VPN 技术

BGP/MPLS VPN 是一种采用边界网关协议 (Border Gateway Protocol, BGP) 在边界路由器间分发

路由、使用多协议标签交换 (Multiprotocol Label Switching, MPLS) 技术在 VPN (Virtual Private Network, 虚拟专用网) 站点之间传送数据的三层 VPN 技术. 该技术使用扩展的 BGP 作为通过运营商的骨干网分发 VPN 路由信息, 使用 MPLS 在 VPN 用户站点之间转发 VPN 业务流数据, 其中涉及三种分工不同的路由器: CE (Customer Edge) 路由器、PE (Provider Edge) 路由器和 P (Provider) 路由器^[6]. BGP/MPLS VPN 的基本结构如图 1 所示.

CE 路由器是客户边缘路由器, 为用户提供到 PE 路由器的连接; PE 路由器是运营商边缘路由器, 也就是电力调度数据网中的地区变电站、风/火电站、水电厂等标签边缘路由器 (Label Edge Router, LER), 与 CE 相连, 主要负责 VPN 业务的接入, 用于存储 VRF、处理转发 VPNv4 路由, 是 MPLS 三层 VPN 的主要实现者, 它根据存放的路由信息将来自 CE 路由器或标签交换路径 (Label Switch Path, LSP) 的 VPN 数据处理后进行转发, 同时负责和其它 PE 路由器交换路由信息; P 路由器是提供商网络骨干路由器, 也就是电力调度数据网络中的标签交换路由器 (Label Switch Router, LSR), 它根据分组的外层标签通过 LDP (Label Distribution Protocol, 标签分发协议) 等协议对 VPN 数据进行透明转发. P 路由器只维护到 PE 路由器的路由信息而不维护 VPN 相关的路由信息. VPN 用户的站点是 VPN 中的一个孤立的 IP 网络, 该网络内部本身是 IP 互联的, 但是和其他站点 (或者是子网) 需要通过运营商网络实现互联互通.

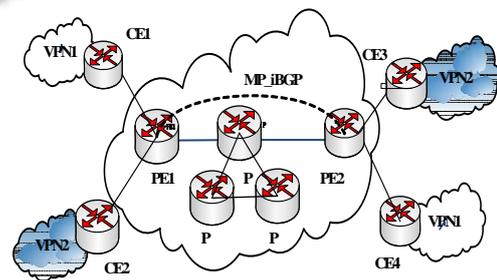


图 1 BGP/MPLS VPN 基本结构图

BGP/MPLS VPN 私网路由方案中, CE 路由器通过静态路由、OSPF、eBGP 等方式把 VPN 本地站点的私网路由信息通告给 PE, PE 路由器再利用 MBGP 与其

PE 路由器交换 VPN 路由信息. PE 之间通过 OSPF 来保证 VPN 内部节点的连通.

3 OMNeT++中BGP和MPLS实现分析

3.1 OMNeT++中 BGPv4 的实现

本文采用的 BGPv4 模块是基于 OMNeT++的 INET 框架中的模块. 依据文献^[8]标准实现的 BGPv4 简单模块, 实现了 BGPv4 的大部分功能.

OMNeT++对 BGPv4 的实现主要包括以下部分:

1)类 TCPCommand: 在 TCP 层和 BGP 之间传递的消息设置 controlinfo. connId 用来标识应用与应用之间的本地连接, 从内部看, TCP 使用 connId 标识对来标识 socket 连接

2)BGP 有限状态机(BGPFSM): 遵循 RFC4271 标准, 包括 IdleState、ConnectState、ActiveState、OpenSentState、OpenConfirmState、Established State 六种状态.

3) 路由策略 (BGPRouting) 及所需路由表 (BGPRoutingTableEntry): 针对 BGP 路由模型, 完成相应的路由访问策略.

4) 相邻路由间的会话(BGPSession): 建立 BGP peer 间的会话、从 IBGP 和 EBGP 学习路径信息、最佳路径选择和对 IP 转发表的更改以及管理 BGP 状态机.

5) 4 种消息(BGPMessage)类型: Open、Keepalive、Update、Notification. 利用消息定义(.msg 文件)定义消息变量类型, 并在上面添加了数据文件. OMNeT++ 将消息定义转化成完全的 C++类.

OMNeT++中 BGPv4 仅仅实现了公认的三个必遵属性: ORIGIN、AS_PATH、NEXT_HOP. 为实现 BGP 对多种网络层协议的支持, BGPv4 需要对可选非过渡属性进行扩展. 本文 BGPv4 使用的报文, 与 IPv4 地址格式相关的三条信息都由 Update 报文携带, 这三条信息分别是: NLRI、路径属性中的 NEXT_HOP、路径属性中的 AGGREGATOR(该属性中包含形成聚合路由的 BGP 发言者的 IP 地址).

3.2 OMNeT++中 MPLS 实现

在 OMNET++中, MPLS 模型是一个附加模块, 用于 MPLS 的仿真, 它位于 OSI 网络模型中的第二层链路层与第三层网络层之间进行交互, 以快速的完成标签的交换, 并且从 LIB 组件和 LDP 模型中获取标签信

息. 支持 MPLS 协议的 LSR 路由器的内部仿真网络拓扑如图 2 所示.

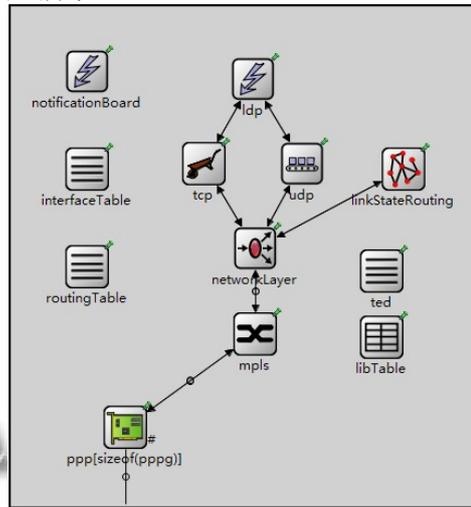


图 2 LSR 路由器的内部仿真拓扑图

MPLS 模型的实现主要包括 IClassifier 类和 networklayer/LDP 类. 其中, IClassifier 作为 MPLS 入口路由器的数据包分类器的抽象接口, 负责对数据包进行分类、压入/弹出标签、交换标签、寻找正确的标签交换路径 (Label Lwitched Paths, LSP)等操作; 标签分发协议(Label Distribution Protocol, LDP)是 MPLS 基本的信令与控制协议它规定了各种消息的格式与操作规程, 主要被应用于一个标签交换路由器 (Label Switched Router, LSR)通知另一个被标签所绑定的 LSR. 从而 LSR 利用 LDP 协议通过映射网络层路由协议信息直接通过数据链路层的交换路径来构建 LSP, 即负责标签的分发以建立 LSP.

MPLS 需要调用一个 C++函数来使一个标签信息库(Lable Information Base, LIB)模块(标签信息基站)展现在路由器中. MPLS 控制协议通知调用 C++函数实现在 LIB 中的信息管理.

MPLS 模型主要完成标签交换和消息转发. 一旦接收到来自另一个 LSR 的带标签的数据包, MPLS 首先提取传入的接口和传入的标签对信息, 然后查找本地的 LIB 表. 如果输出接口和输出的标签被找到, MPLS 依据外部标签执行恰当的标签操作(PUSH、POP、SWAP)^[8].

4 OMNET++中BGP/MPLS VPN的设计实现

根据文献^[7]中作出的规范, 结合电力调度数据网

中 VPN 私网路由的技术原理, BGP/MPLS VPN 的实现主要包括在 PE 路由器上实现多个 VPN 路由转发表 (VRF) 及相应处理机制, 对现有的 BGPv4 进行多协议扩展以分发 VPN 路由信息(其中包括 VPN 使用的内部标签), 利用 MPLS 标签嵌套实现 VPN 隧道以转发数据, 实现数据网业务数据的无障碍转发. 结合 OMNeT++ 中已有的实现, 本文对 BGP 和 MPLS 模块进行相应的扩展, 以达到 BGP/MPLS VPN 技术的要求.

4.1 VRF 表

在 BGP/MPLS VPN 中, VRF(VPN Routing & Forwarding Instance, VPN 路由转发实例)表定义连接到 PE 路由器中客户结点的成员关系. 每个 PE 路由器为每个接口连接的客户结点维持 VRF 表, 有几个直接相连的客户结点就有几个 VRF 表. 只有属于同一 VPN 的各个客户结点, 才能彼此交换信息. VRF 可以看作虚拟的路由器, 包括一张独立的路由表, 一组归属于这个 VRF 的接口集合, 一组只用于本 VRF 的路由协议, 以及路由标识符 RD、路由目标属性 RT. RT(Route Target), 是每个 VRF 进行路由的标识, 主要用于控制 VPN 路由的发布和安装策略, 分为 import 和 export 两种属性, 前者表示对哪些路由进行转发, 而后者表示所发出的路由的属性. 当 PE 发布路由时, 将使用路由所属 VRF 的 RT export 规则, 直接发送给其他的 PE 设备. 对端 PE 接收路由时, 首先接收所有的路由, 并根据每个 VRF 配置的 RT 的 import 规则进行检查, 如果与路由中的 RT 属性 match, 则将该路由加入到相应的 VRF 中^[9].

本文设计实现了一个专门用来处理 VRF 的类 VRF, 并将这个类的一个实例结合到 PE 路由器的类 IClassifier 中, 对类 IClassifier 扩展一个配置 VRF 的接口, 实现对用户 VPN 的配置. 类 VRF 实现了一个以数据结构 VRFTable 为条目的 VRF 表和相应的 VRF 表操作, 同时还带有成员路由标识 RD, 输入路由目标向量 IMPORT_RT 和输入路由目标向量 EXPORT_RT. 其中, 数据结构 VRFTable 如下所示:

```
struct VRFTable
{
    int network; //网络地址
    int prefix; //地址前缀长度
    int nexthop; //下一跳地址
```

```
int label; //VPN 内部标签
int outinterface; //输出接口
};
```

VRF 表所存储和处理的路由信息都是带有 RD 的 VPN-IPV4 路由. 每个 VRF 分别配置一个唯一的 RD 值. 收到 VPN-IPV4 路由时, PE 路由器根据 RD 区分不同 VRF 路由信息, 根据比较 RT 值来确定是否执行该路由. 若该路由被执行, PE 路由器则将数据结构 VRFTable 中的网络地址、前缀长度、下一跳和内部标签安装到 VRF 表中^[10].

4.2 MP-BGP 扩展和 VPN 路由分发

由于 BGP/MPLS VPN 使用的是 VPN-IPV4 地址, BGP 协议模块必须支持多协议扩展才能够用于 BGP/MPLS VPN 网络 PE 之间的路由发布. IETF 对 BGPv4 的路由属性进行了扩展, MP-BGP(Multiprotocol Extensions for BGPv4) 是 BGPv4 的扩展协议, 标准是文献^[11]. MP-BGP 是现有 BGPv4 协议基础上的增强, 使 BGP 能够为多种路由协议提供路由信息, 包括 IPv6 和组播以及 VPN 的支持. 但在 OMNeT++ 中仅实现了 3 个必遵属性(RIGIN、AS_PATH、NEXT_HOP), 没有实现 MP-BGP 的扩展, 为了实现 MP-BGP 需要对 BGP 扩展两个路由属性: MP_Reach_NLRI 和 MP_Unreach_NLR. MP-BGP 携带信息增加了一个扩展之后的 NLRI(Network Layer Reachability Information), NLRI 增加了地址族的描述、私网 label 以及 RD, 跟随之后的是 RT 的列表. 为了传输 VRF 的路由目标, 还需增加一个扩展的团体属性 RouteTarget 以对入路由和出路由进行过滤^[11].

扩展协议可达 NLRI 属性用于发布可达路由及下一跳信息, 其中 MP_REACH_NLRI 的数据结构:

```
struct BGPUUpdateMpREACHNLRI
{
    int afi; //VPN-IPV4 地址族描述
    int safi; //子地址族描述
    unsigned short lengthofnextrhopnetworkaddr; //
    路由的下一跳网络地址长度
    IPv4Address networkaddressofnextrhop; //路由
    的下一跳网络地址
    int numberofsnpas; //保留未用, 置为 0
    BGPUUpdateMpNLRI nlri; //前缀信息
    +RD+label
```

}

MP_REAVH_NLRI 数据结构中, afi 携带和网络地址相关的网络层协议标识; safiI 携带相关属性中网络层可达信息的附加信息; 网络地址长度 (lengthofnexthopnetworkaddr) 和下一跳网络地址 (networkaddressofnexthop) 描述了下一跳信息; MP-BGP 扩展后的 nlri 字段会附加 RD、标签等信息, 已完成标签的分配。

扩展协议不可达 NLRI 属性通告不可达路由, 一个含有 MP_UNREACH_NLRI 的 update 报文不需要携带 MP_UNREACH_NLRI 属性以外的任何其他路径属性。其中 MP_UNREACH_NLRI 的数据结构如下所示:

```

struct BGPUpdateMpUNREACHNLRI
{
    int afi;
    int safi;
    BGPUpdateWithdrawnRoutes
withdrawnRoutes; //撤销不可达路由信息
}

```

VPN 所使用的内部标签是由 MP_Reach_NLRI 来携带的。根据图 1 所示的 BGP/MPLS VPN 基本结构, 本文采用的 BGP/MPLS VPN 分发 VPN 路由信息分 5 步。

第 1 步: 从 VPN1 端的 CE1 路由器和入口 PE1 路由器分发。

第 2 步: 在入口 PE1 路由器输出到提供者的 BGP。PE1 路由器为直接连接的 VPN1 维持 VRF 表, 该表定义连接到 PE1 路由器上的 VPN1 用户关系。入口 PE1 路由器从它直接连接的 CE1 路由器获得路由后, 将这些路由加上特定的 RT 属性, 安装到相应的 VRF 表中, 然后将此路由转换成 VPNIPv4 路由, 输出到提供者的 BGP。

第 3 步: 穿过服务提供者骨干网络。入口 PE1 路由器能与出口 PE2 路由器通过 BGP 交换这些 VPN-IPv4 路由。

第 4 步: 在出口 PE2 路由器从提供者 BGP 输出。一旦出口 PE2 路由器从其它 PE 路由器得到 VPN-IPv4 路由, 它就为自己连接的 VPN2 安装这些与 RT 属性相连的路由。

第 5 步: 从出口 PE2 路由器向 CE2 路由器分发。被选中的 VPN-IPv4 路由被转换回 IPv4 路由, 并输入

到相应的 VRF 表中。安装到 VRF 表中的路由均可以被分配给相应的 CE2 路由器。

对于 MP-BGP 而言, 需要考虑的只是路由通告过程, 数据转发过程的具体细节(如分配内层标签、为数据包粘贴上内层标签等)由 MPLS 内部实现。

4.3 VPN-IPv4 路由模型和数据转发

为了使 PE 路由器同时具有处理 BGP 和 MPLS 的功能, 本文将两者整合在一起搭建了一个 VPN-IPv4 路由模型, 并将 PE 的节点结构扩展为如图 3 所示。VPN-IPv4 路由由复合模块由简单模块组成, 主要包括链路层的 ppp 模块、网络层的 networkLayer 模块和 ospfRouting 模块、传输层的 tcp 模块和 udp 模块、应用层的 bgp 模块和 ldp 模块以及进行数据包转发所用到的路由表及接口模块。它们一起构成一个完整 VPN-IPv4 路由模型, 各模块之间通过接口来连接。

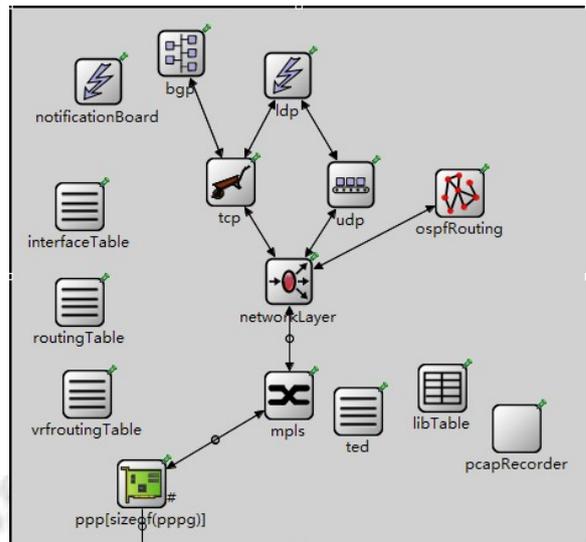


图 3 PE 路由器的内部网络拓扑图

当 CE 路由器将一个 VPN 分组转发给入口 PE 路由器后, PE 路由器查找该 VPN 对应的 VRF, 从 VRF 中得到一个 VPN 标签和下一跳出口 PE 路由器的地址, VPN 标签作为内层标签打在 VPN 分组上, 根据下一跳出口 PE 路由器的地址可以在全局路由表中查出到达该 PE 路由器应打上的域内路由的标签, 即外层标签, 于是 VPN 分组被打上了两层标签, 主干网的 P 路由器根据外层标签转发 VPN 分组, 在最后一个 P 路由器处, 外层标签弹出, VPN 分组只剩下内层标签(此过程被称作作末级弹出机制), 接着 VPN 分组被发往出口 PE 路由器。出口 PE 路由器根据内层标签查找到相应的出口

后, 将 VPN 分组上的内层标签删除, 将不含标签的 VPN 分组转发给正确的 CE 路由器, CE 路由器根据自己的路由表将分组转发到正确的目的地^[12].

根据图 1 所示的 BGP/MPLS VPN 基本结构进行的数据转发流程图如图 4 所示.

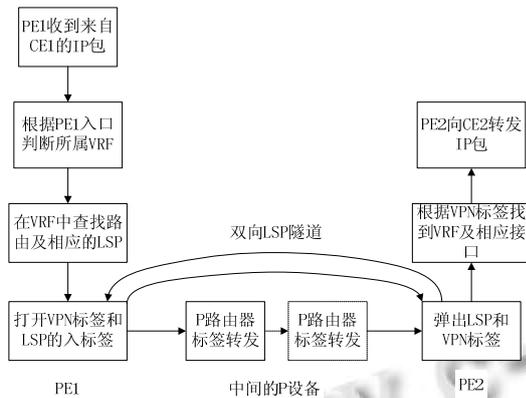


图 4 BGP/MPLS VPN 的数据转发流程图

5 仿真测试与验证

5.1 仿真测试

实际场景中, 地区电力调度数据网采用分层结构, 由三层组成, 即核心层、骨干层和接入层. 为保证网络链路的可靠性, 在网络的拓扑设计中, 遵循 N-1 的电路可靠性和 N-1 的节点可靠性设计原则, 即调度数据网络核心节点之间采用完全或不完全网状网连接, 骨干节点尽量与两个核心节点相连, 骨干节点之间可互联, 接入节点至少与一个骨干节点相连, 根据业务量及地理位置, 接入节点也可直接接入核心节点^[13].

依据实际电力调度数据网, 利用 OMNeT++网络仿真的 BGP/MPLS VPN 技术搭建网络拓扑, 结合实际参数的配置, 进行网络实际的仿真和分析, 如图 5 所示.

仿真实例中 HB11 主机的运行状态图, 如图 7 所示, HB11 主机已接受来自 HA11 主机的 UDPBasicApp 数据包 rcvd:9pkts. 图 7 为 HA11 发送数据包到 HB11 的过程记录数据, 包括各路由器的数据的转发、和下一跳等信息. 在图 6 中输出的信息中, 显示 PE2 路由器中的转发数据包的信息、转发的接口信息及下一跳目的地址的信息. 表明来自 HA11 主机的数据经各路由器的转发处理后到达 HB11 主机端, 即 HA11 与 HB11 主机之间已建立 VPN-IPv4 路由.

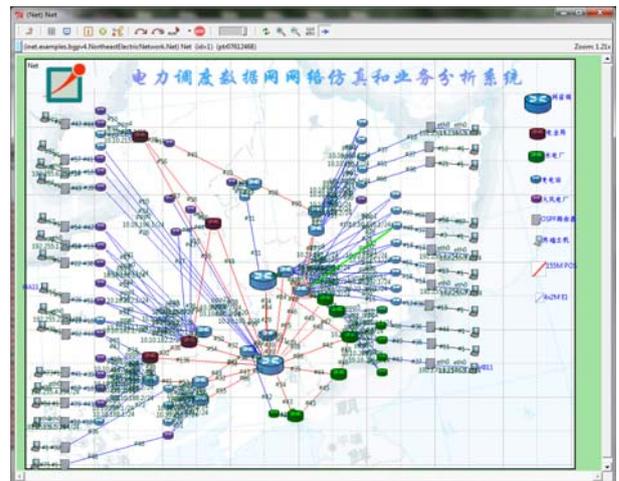


图 5 地区电力调度数据网络拓扑结构图

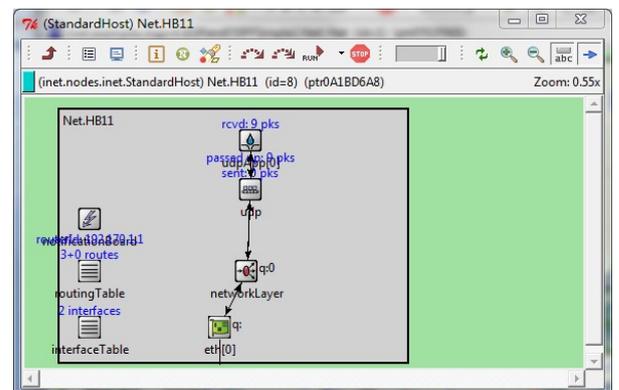


图 6 仿真中 HB11 主机的运行状态图

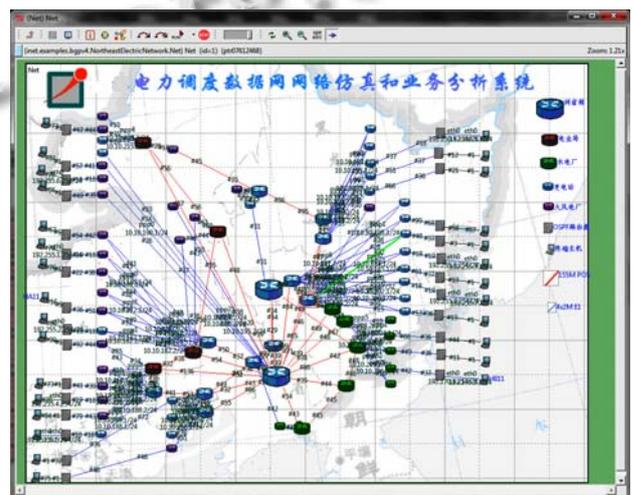


图 7 HA11 到 HB11 的路由及数据转发过程

5.2 业务仿真验证

结合电力调度数据网应用层业务特征, 在构建的

仿真网络中研究 CBR 和 VBR 业务的生成规则, 建立相应的网络业务运行模型. 在 OMNeT++ 仿真中, 包含在 `?inet.applications.*packages` 里流量生成模块 (如 `UDPBasicApp`、`TCPBasicClientApp`、`IPTraffGen`、`EtherAppCli`、`EtherAppSrv` 模块等) 依据数据包的发送间隔或者等值参数(固定值或者任意可变的参数值, 如 `exponential(1s)`)可以生成固定比特率或可变比特率的流量.

本系统的业务仿真采用可变参数的方式产生可变的 UDP 数据包流量来模拟 CBR 和 VBR 业务. 本系统业务仿真中, 通过设置 `HA11.udpApp[0].sendInterval = 5s` 和 `HA11.udpApp[0].sendInterval = exponential(1s)` 分别生成固定流量和可变流量来分别模拟 CBR 和 VBR 业务的生成. 通过图 8 可以看出 VBR 业务的线路平均带宽占用率. 通过观察线路带宽吞吐量及负载性能分析业务特征. 其中, CBR 业务的线路平均带宽占用率如图 9 所示.

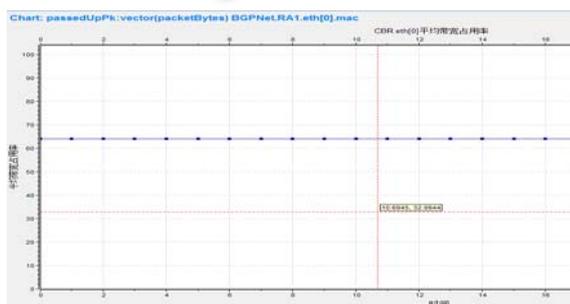


图 8 CBR 业务的线路平均带宽占用率

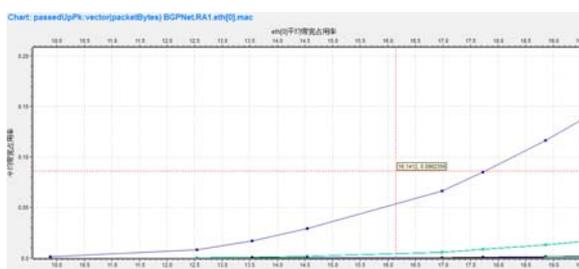


图 9 VBR 业务的线路平均带宽占用率

6 结 语

本文结合地区电力调度数据网的组网特点, 基于 OMNeT++ 仿真环境中现有的 BGPv4 和 MPLS 的基础上对其进行进一步扩展, 给出了符合 RFC 标准的

BGP/MPLS VPN 的具体实现方案, 并通过实际场景下的组网进行具体测试, 验证了仿真的可行性, 并对数据网数据业务的仿真, 为以后在基于 OMNeT++ 平台下对电力调度数据网中的性能分析, 业务分析作进一步研究具有重要意义, 例如进行实现电力调度数据网的业务区分、业务隔离、流量工程的技术研究.

参考文献

- 1 彭清卿, 向力, 卢长燕等. 国家电力调度数据网组网研究. 电力系统自动化, 2004, 28(8): 10-11.
- 2 韩波, 沈富可, 刘莉. BGP/MPLS VPN 在 NS-2 中的实现. 计算机应用, 2006, 26(4): 980-981.
- 3 OMNeT++ Home Page. <http://www.omnetpp.org/>. 2012-10.
- 4 夏峰. OMNeT++ 网络仿真. 北京: 清华大学出版社, 2012: 4-5.
- 5 Vojtech J, Andras V, Helene L, et al. INET Framework for OMNeT++. <http://omnetpp.org/>. [2013-1-25].
- 6 Rosen E, Rekhter Y. RFC2547: BGP/MPLS VPN. 1999.
- 7 Rekhter Y, Li T, Hares S. RFC4271: Border Gateway Protocol 4 (BGP-4). Jan. 2006.
- 8 Rosen E, Tappan D, Fedorkow G, et al. RFC3032: MPLS Label Stack Encoding. 2001.
- 9 董玲, 黄杨, 徐塞虹. BGP/MPLS VPN 实现细节探讨. 计算机工程与应用, 2005, 41(29): 118-119.
- 10 梁海英, 侯锐, 兰丽辉等. 基于 VRF 和 RT 实现 BGP/MPLS VPNs 中的 VPN 拓扑发现. 计算机科学, 2007, 34(2): 43-44.
- 11 Bates T, Chandra R, Katz D. RFC4760: Multiprotocol Extensions for BGP-4. January 2007.
- 12 李海华. BGP MPLS VPN 数据转发过程分析. 计算机技术与发展, 2011, 21(6): 5-6.
- 13 陈然, 刘问. 电力调度数据网 MPLS VPN 跨域互联技术. 云南电力技术, 2012, 40(2): 22-23.