隧道传输系统中基于表项优化的高效转发模型

陈文龙[1]，齐宏伟[1]，徐明伟[2]，徐 恪[2]

(1. 首都师范大学信息工程学院 北京 海淀区 100048; 2. 清华大学计算机科学与技术系 北京 海淀区 100084)

【摘要】异构网络环境中，隧道报文在路由器数据层的处理涉及多次不同表项查找，报文转发时延和表项存储容量都面临着极大的挑战。该文主要研究隧道报文处理的核心功能及各项功能步骤间的关联，提出了表项聚合和表项拆分理论，并分析了二者的效用、代价及适用场景。并设计了基于表项优化的隧道设备高效转发模型，针对4over6过渡网关给出了具体实现方法。原型系统实验和分析证明了隧道网络表项优化机制和理论的正确性、高效性。表项优化研究也为其他网络设备的各类表项处理优化提供了重要参考。

关 键 词 表项聚合; 转发; 表项拆分; 隧道传输

中图分类号 TP393 **文献标志码** A doi:10.3969/j.issn.1001-0548.2015.05.016

An Effective Forwarding Model for Tunneling Transmission System Based on Table Optimization

CHEN Wen-long[1], QI Hong-wei[1], XU Ming-wei[2], and XU Ke[2]

(1. Information Engineering College, Capital Normal University Haidian Beijing 100048;

2. Department of Computer Science and Technology, Tsinghua University Haidian Beijing 100084)

Abstract The property that multi-lookup forwarding tables on data layer poses a demanding challenge to forwarding time-delay and device storage consumption in heterogeneous network system. To deal with the problem, this paper proposes a theory of aggregation and split model based on forwarding table by analyzing the kernel functional modules and relations inner the tables. The application scenarios of the theory are studied and the model is applied to 4over6 transition gateway to transition from IPv4 to IPv6. The prototype system test results verify that the optimization model using aggregation and split is more effective than that non-use. The proposed model could be a reference to other related researches.

Key words aggression; split; table optimized; tunneling

隧道传输技术已被广泛应用于互联网各种新型数据传输模型中。例如，面向下一代互联网的IPv4/IPv6过渡方案4over6[1-2]、虚拟专用网络VPN[3]都是隧道传输技术的重要应用。

IETF工作组提出的4over6方案中，源端系统发出的IPv4报文经所属边缘网络的隧道网关(Tunnel Gateway, GW_T)实施封装，封装报文经IPv6网络发送至目的端所属边缘网络的GW_T，解封装得到原始的IPv4载荷报文，并根据载荷的目的地址发送至目的端。其中，报文封装转发过程是关键步骤，GW_T涉及转发表和封装映射表的查询。VPN[4]技术是在公共的互联网中建立起端对端的专用隧道传输网络。VPN技术基于报文的隧道封装/解封装，为用户提供“专有的”传输通道。VPN设备或软件模块要多次查询转发表和封装表。

目前，隧道网络存在的共性问题是隧道网关涉及多个表项的查找，使得报文线速转发更为困难，并且存在冗余存储现象，这些问题制约了隧道技术的发展。更值得关注的是，IPv4/IPv6隧道实施中IPv6[5]的128 Bytes地址结构，会加剧此类问题。4over6隧道过渡技术已在下一代互联网CERNET2中广泛部署，使得该问题的解决更加迫切。

本文通过分析典型隧道传输过程，针对现有设计方案存在的存储冗余、报文转发速度受限的共性问题，提出表项聚合和拆分理论，设计了隧道网络的表项优化模型，包括表项存储及隧道转发步骤的优化。该模型只需对隧道传输网络中的隧道网关设备进行改进，同现有网络保持了最大兼容。

收稿日期：2014 – 12 – 22；修回日期：2015 – 07 – 01

基金项目：国家自然科学基金(61373161, 61272446, 61300171)；北京市教委科技计划(KM201410028015)

作者简介：陈文龙(1976 –)，男，博士，副教授，主要从事网络体系结构、网络协议方面的研究.

1 隧道传输分析

通常，隧道传输网络都具有基本共性：由隧道网关连接2个不同地址空间的异构网络，隧道网关对跨异构网络传输的数据进行封装/解封装及转发。需要说明，有时隧道网关功能也集成在端系统中。隧道传输过程总是涉及多次表项查找，如映射信息查找、转发表查找、下一跳MAC地址查找等。

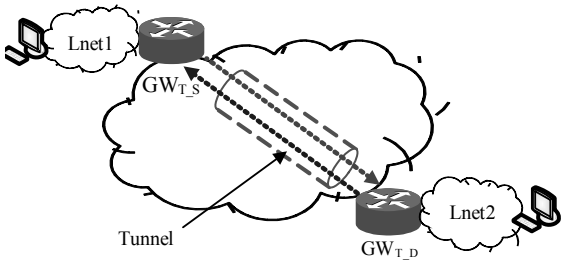

图1 隧道传输基本拓扑结构

隧道传输基本拓扑结构[6]如图1所示，隧道Tunnel是封装数据传输的通道；边缘网络(local net, Lnet)传输非封装报文；隧道网关GW_T是联接Tunnel和Lnet的网关设备。

隧道网络传输数据的流程包括：

1) 源端发送的报文经边缘网到达源端隧道网关GW_{T_S}；

2) GW_{T_S}根据收到报文的边缘网Lnet1目的标识查询隧道映射表，得到对应的隧道封装标识，实施报文封装；

3) 根据封装的目的标识，查询转发表并发送封装报文至目的隧道网关GW_{T_D}；

4) GW_{T_D}解封装取得载荷，并根据载荷报文的目的标识查询边缘网转发表，最终送达目的端。

考虑到普通转发处理功能的兼容，对未匹配到隧道转发信息的报文，将查询Lnet1的常规路由表，按常规方法转发报文。

综上分析，当前隧道传输网络的问题聚集在隧道网关中：1) 隧道网关涉及到多个表项查找，逐级查表无疑会消耗设备更多运算资源，对采用硬件转发的设备，成本必会上升，设计复杂度也难以估量；2) 各个表项之间并不孤立，而是相关联的，处理不当必会造成表项信息存储冗余，浪费存储资源；3) 表项内参数重复设置，造成冗余存储，IPv6等较长标识参数的出现会使问题更加突出。

所以，隧道传输网络的优化模型将聚焦在隧道网关，其面临两个主要问题：1) 优化表项设计，加快隧道网关查表速度，降低设计复杂度；2) 在保持对现存隧道传输网络体系兼容的前提下尽可能降低隧道网关各类表项的冗余存储。

2 表项聚合与拆分

不失一般性，将数据层表项涉及的参数分为两类：输入参数SI和输出参数SO。规定由输入和输出参数构成的集合分别为输入参数集PI和输出参数集PO。

表项查询过程可视为输入参数与输出参数间的映射：表项的输入参数SI映射得到输出参数SO，形式化表示为：

$$SI \rightarrow SO \tag{1}$$

输入参数SI是查表前已获得的一些报文信息或状态信息，如报文目的地址、报文长度等。输出参数SO是报文转发处理所需的控制信息，如报文封装标识、出接口、下一跳Next-hop、MAC地址等。由数据层表项构成的集合称作映射表。

定义 1 表项聚合。将转发处理中多个相关联的映射表项按预定规则合并，合并后的映射表能够描述原有多个映射表所包含的信息，定义该过程为表项聚合。表项聚合包含两种基本类型，Ⅰ型为：

$$\{SI_1 \rightarrow SO_1\} \Theta \{SO_1 \rightarrow SO_2\} \Rightarrow \{SI_1 \rightarrow (SO_1, SO_2)\} \tag{2}$$

Ⅱ型为：

$$\{SI \rightarrow SO_1\} \Theta \{SI \rightarrow SO_2\} \Rightarrow \{SI \rightarrow (SO_1, SO_2)\} \tag{3}$$

两种类型的表项聚合示例如图2所示。

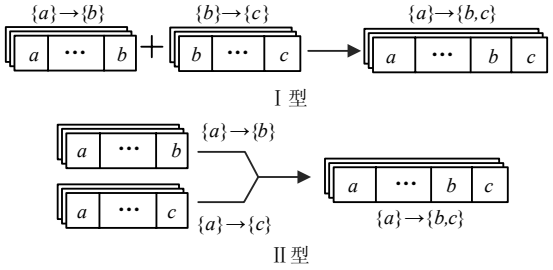

图2 两种类型的表项聚合示例

Ⅰ型表项聚合利用的是表项参数间的递推关系，实现两个表项的合并。它可应用于合并过渡网关中转发表和隧道映射表。Ⅱ型表项聚合针对具有相同输入参数的多个映射表项，可去除表项输入参数的重复冗余。

表项聚合在路由设备中应用广泛。例如，在常规路由系统中，根据目的地址和子网掩码分别映射成下一跳地址和出接口，这是两条映射。但在许多情况下，往往将这两个映射合并成一个映射，这是表项汇聚的一个典型例子。对映射表实施一定力度的聚合可以优化设备性能：1) 特定条件下，能减少映射表数量；2) 表项聚合能压缩表项数量。在现有

典型数据层处理环境中颇有意义，如基于TCAM[7]的硬件转发系统。

定义 2 表项拆分。将包含多个输入、输出参数的映射表用多个映射表等价替换，且使替换前后包含的总映射信息不变，定义该过程为表项拆分。表项拆分后将引入索引参数S_{index}，其长度决定于SO中元素个数，通常要求S_{index}必须能索引全部的SO。拆分关系描述为：

$$\{\text{SI}\}\to\{\text{SO}\}\Rightarrow\begin{cases}\text{SI}\to S_{\text{index}}\\ S_{\text{index}}\to\text{SO}\end{cases}\tag{4}$$

表项可进行多次拆分，为便于说明给出表项拆分为两项的例子。表项拆分示例如图3所示。

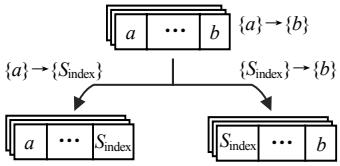

图3　表项拆分示例

表面上看，表项拆分会造成表项数量增长。然而，当参数b长度较大且取值范围较小时，参数b会大量冗余存储，通过表项拆分可以降低系统中表的总体存储开销。在基于通用CPU实施软转发时，还可减少内存查找时间，提高转发速度。

表项聚合、拆分的类型有很多种，本文只关注与隧道网络密切相关的类型。表项聚合和拆分需配合使用才能达到转发处理及表项的最大优化。

3　面向4over6隧道网关的表项优化

4over6[8-9]隧道网络是向IPv4/IPv6过渡的网络技术。以4over6为例，设计隧道网关基于表项优化的高效转发模型，该隧道网关设备也称为“过渡网关”。4over6隧道网络具有图1所示的拓扑结构，其中过渡网关是4over6网络的核心。优化的4over6过渡网关模型如图4所示。优化机制只在隧道网关的转发层实施，不涉及路由协议、路由管理等控制层功能模块[10]。

IPv4/IPv6过渡网关中，数据处理表项有两类：1) 常规IPv4/IPv6转发表；2) 隧道传输所需的IPv4/IPv6地址的映射表，称其为隧道映射表。

传统隧道网关的IPv4/IPv6转发表和隧道封装表如图4所描述。表项优化模型为：

1) 聚合IPv4转发表与隧道映射表。聚合表项中冗余参数{IPv4_Dst, mask4, next-hop, interface}，其余参数保留，产生临时性的转发表Table$_1$。

原始表项

表项	参数
IPv4转发表	< IPv4_Dst, mask4, next-hop, interface >
隧道映射表	< IPv4_Dst, mask4, IPv6_T_Dst, next-hop, interface >
IPv6转发表	< IPv6_Dst, mask6, next-hop, interface >

*IPv6_T_Dst为隧道目的地址，代表隧道对端的IP地址

优化后表项

表项	参数
一体化转发表(IPv4*)	< IPv4_Dst, mask4, IPv6_T_Dst, Index >
一体化转发表(IPv6*)	< IPv6_Dst, mask6, Index >
索引表(Index)	< Index, next-hop, interface >

*分别指目的地址为IPv4/IPv6的一体化转发表

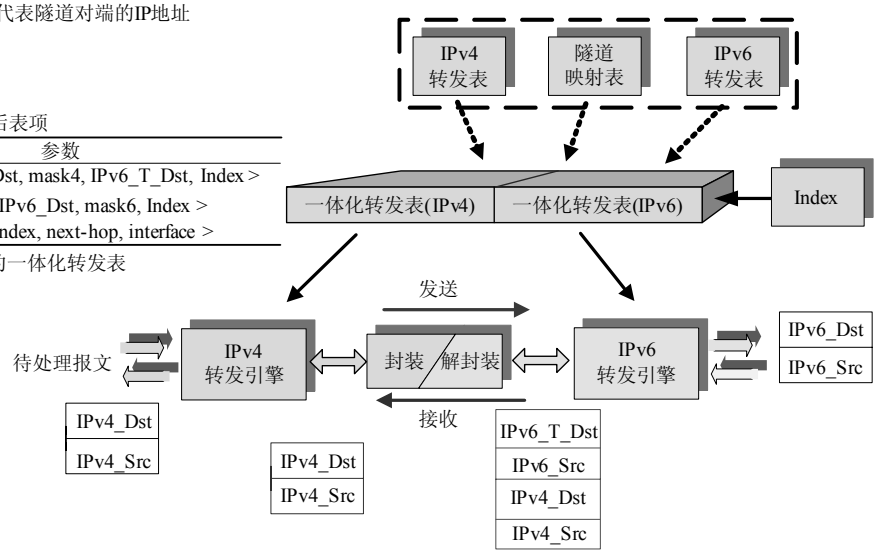

图4　优化的4over6过渡网关模型

2) 拆分Table$_1$。去除Table$_1$表项中参数{next-hop, interface}，增加参数{Index}，形成IPv4一体化转发表IPv4*。用参数{Index}、{next-hop, interface}建立Index表<Index, next-hop, interface>。

3) 拆分IPv6转发表。去除IPv6转发表中参数{next-hop, interface}，增加参数{Index}，生成IPv6一体化转发表IPv6*，类似步骤2)，建立Index表。

通过设置标志位的方式，IPv4/IPv6 索引表可以

复用。据实践经验，参数{Index}的长度设计为16 bit，并设计其最高位为IPv4/IPv6标志位，$Index_{16}$为0时，表示后面15位索引IPv4的Next-hop、Interface；$Index_{16}$为1时，表明其后索引IPv6出口信息。15 bit长度的{Index}参数能索引Next-hop、出接口Interface的数量共2^{15}个，足以满足常规使用。{Index}的长度也可由设计者根据实际情况自行设置。

在数据层面，过渡网关的原始处理流程：

GW_T先根据协议类型号判断收到报文的协议类型。

1) 收到IPv4报文

① 查询IPv4转发表。匹配到表项<IPv4_Dst, mask4, next-hop, interface>，IPv4转发引擎根据查找结果进行传统转发处理。

② 未查到IPv4转发表表项，继续查询隧道映射表。查到映射表项：<IPv4_Dst, mask4, IPv6_T_Dst, next-hop, interface>，表明目的主机可能位于隧道对端，需要封装转发。封装模块以{IPv6_T_Dst}为目的地址封装原报文，并继续查询IPv6转发表并以转发封装后报文。查询IPv4转发表与隧道映射表的先后顺序可调换，不影响转发结果。

2) 收到IPv6报文

查询IPv6转发表，若没有匹配项，则丢弃报文；若有匹配项，分3种情况：

① 普通转发处理，根据匹配表项<IPv6_Dst, mask6, next-hop, interface>，转发IPv6报文。

② 若是普通本机接收报文，即没有4over6封装，则提交上层协议处理。

③ 若是本机接收报文且为4over6封装，表明报文经隧道封装传输而来，要先用解封装模块取出载荷，最后查询IPv4转发表转发原始报文。

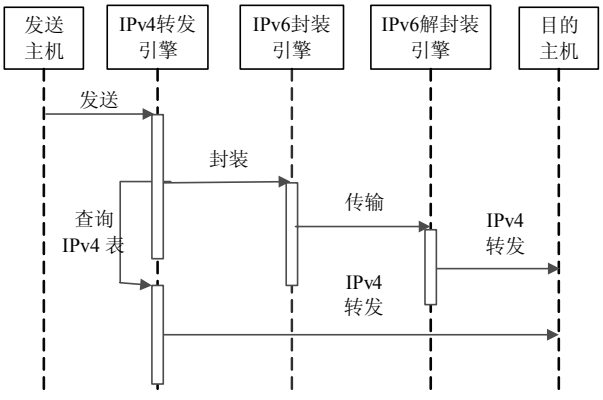

图5 原始隧道处理流程

综上所述，过渡网关最坏情况下需查询3次不同的数据表。各表项存储着冗余参数，严重制约了转发性能。原始隧道处理流程如图5所示。

实行隧道网关表项优化后能极大简化查表流程，表项有较大变化，图4中列出了优化后的表项。过渡网关的优化处理流程分为两步：

1) 收到IPv4报文

查询IPv4一体化转发表，查到<IPv4_Dst, mask4, IPv6_T_Dst, Index>，若参数{IPv6_T_Dst}为空(null)，按常规转发报文；若{IPv6_T_Dst}非空，说明要进行隧道封装，该地址为隧道目的地址。再用Index的下一跳地址{next-hop}、出接口{interface}信息发送封装报文。

2) 收到IPv6报文

若满足报文协议类型是4over6隧道报文，且目的地址为自身地址，进行报文解封装，取出载荷后转发载荷。其余不满足条件的IPv6报文查询IPv6一体化转发表，并用Index索引的{next-hop}、出接口{interface}信息转发。

本文优化技术的应用能为标识映射隧道传输网络带来如下性能变化：

1) 过渡网关隧道封装信息表、IP转发表等表项都集成为1个映射表，节省了内存开销；

2) 表项聚合使得只需1次查表，大大提高了转发速度，使跨异构网络的线速转发成为可能；

3) 一体化的转发表设计方法更利于表项维护，维护开销更小。

4 性能分析

针对内存消耗、转发时延等指标分析过渡网关优化转发模型的性能。为方便叙述，符号定义如表1所示。

表1 符号定义

符号	描述
λ_i/bit	第i个映射表中一条表项的参数长度
m_i	第i个表内表项数量
κ	聚合参数的长度
θ	拆分参数的数量
M_i	第i个映射表所耗存储

优化前隧道网关中存在3个表(IPv4转发表、隧道映射表、IPv6转发表)，3个表所占存储为：

$$\sum_{i=1}^{3} M_i = \sum_{i=1}^{3} \lambda_i m_i \tag{5}$$

优化后一体化转发表内表项的数量满足：

$$N = \max(m_i)|_{i=1,2,3} \tag{6}$$

其存储量为：

$$N(\lambda_1+\kappa)=\max(m_i)(\lambda_1+\kappa) \tag{7}$$

优化前/后转发表存储统计如表2所示，得出聚合前后存储消耗的变化Δ_{mem}：

$$\Delta_{\text{mem}}=\max(m_i)(\lambda_1+\kappa)-\sum_{i=1}^{3}m_i\lambda_i \tag{8}$$

一般认为，隧道网关的表项长短对查表速度的影响较小，可近似认为设备的表项长短和转发速度无关。在此前提下，转发时延取决于查表次数。隧道网关传统转发过程涉及多个表。具体到IPv4/IPv6过渡网关中，隧道转发一般要依次查：IPv4转发表→隧道映射表，共查表2次；而经优化后，仅需查1次一体化转发表即可获取全部路径信息，转发时延缩短。转发非封装报文都查表1次，优化前后查表次数不变，在表项长度差异对查表速度影响忽略不计的情况下，优化前后转发时延近似相等。

表2　优化前/后转发表存储统计

	聚合		拆分	
	前	后	前	后
表项的总数量	$\sum m_i$	$\max(m_i)$	$\sum m_i$	$\sum m_i+\theta$
表项的长度	λ_i	$\lambda_i+\kappa$	λ_i	$\lambda_i+S_{\text{index}}$
总内存消耗/MB	$\sum M_i$	$\max(m_i)(\lambda_1+\kappa)$	$\sum M_i$	$\left(\sum m_i+\theta\right)(\lambda_i+S_{\text{index}})$

表项拆分后表的总数量要增多，原需查询一个一体化转发表，现要多查询一个索引表。拆分会导致查表次数增加，转发时延增加，但能降低存储冗余。

转发表聚合会增加表项长度，但能加快数据转发速度，简化维护表项参数，适合基于硬件TCAM转发的隧道网关设备。转发表拆分能降低表项参数存储冗余，这是以部分转发速度为代价换取的，在对转发速度要求不高的情况下，实施转发表拆分能大幅减少内存消耗，降低成本。

综上分析，优化措施方法的选择取决于设计目的。聚合与拆分互不排斥，二者可在不同时间段使用，当隧道网关存在多个转发表时，可选择性的实施聚合与拆分。实际应用中，聚合和拆分往往是分不开的。

隧道网关是否实施优化取决于所处网络的流量分布情况。当部分表项的命中率很高时，表明报文路径集中度很高，应当进行优化；反之，不宜进行聚合，因为存储增加的代价远大于转发速度的提升带来的优势。转发表的拆分也类似。

5　实验分析

基于隧道传输网络的表项优化工作是实现一体化标识网络寻径的一部分。一体化标识网络目的是为异构网络提供统一高效的数据传输服务，本文所做工作是对该体系中的过渡网关进行优化。原型系统构建在面向过渡的IPv4/IPv6路由器上。在此原型系统上，测试优化隧道网关的性能。

实验基于Bit-Engine Netwire4600系列设备，按照前面描述的4over6隧道优化模型设计了原型系统，在此平台上可以完成转发涉及的转发表、隧道映射表的聚合、拆分实验。性能测试实验示意图如图6所示，将IXIA网络测试仪的两个1 000 Mb/s接口连接至过渡网关的两个1 000 Mb/s接口，测试IPv4报文转发(缘网转发)4over6隧道转发性能。

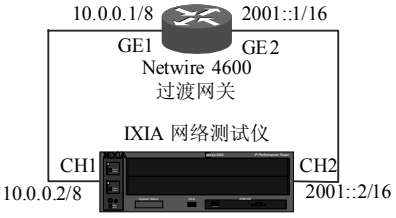

图6　性能测试实验示意图

实验测试了不同长度报文的转发能力。考虑到4over6需为IPv4报文加40 Bytes的IPv6头部，取报文长度范围64～1 478 Byte。

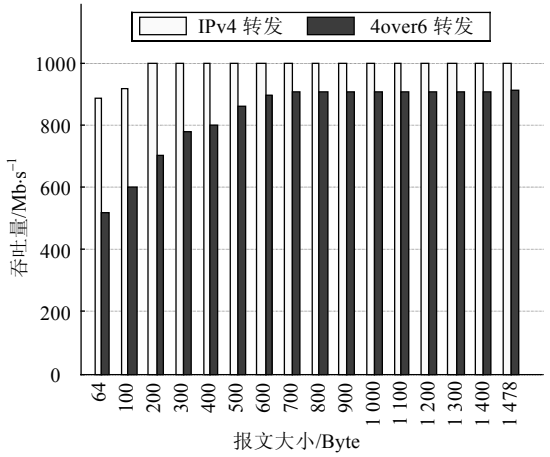

图7　隧道网关转发带宽测试结果

隧道网关转发带宽的测试结果如图7所示，在模拟边缘网转发的测试中，64 Byte报文的转发带宽在900 Mb/s左右，报文长度在200 Byte后近乎实现了千兆网络的线速转发；在4over6隧道封装转发测试中，64 Byte报文转发带宽为520 Mb/s，当报文长度达到700 Byte后转发带宽可以稳定在900 Mb/s，报文长度增加后封装处理效率会明显提升。4over6隧道转发时，过渡网关会在报文前封装40 Byte的IPv6头部，但与之相连的IXIA测试仪的报文注入接口统计的是IPv4流量，因此能够认为4over6隧道转发也达到了线

速转发。

隧道网关转发时延的测试结果如图8所示，在边缘网模拟中，报文长度在512 Byte以下时的转发时延基本保持在40 μs左右，从512 Byte开始报文转发时延呈线性增加；在4over6隧道封装转发测试中，报文小于512 Byte时转发时延在40 μs左右，由于随着报文长度增转发时间会显著增加，当报文长度到达512 Bytes后转发时延开始以线性规律增加，这是由于报文增长后报文的收发处理时间也相应增加。对不同大小的报文分别进行IPv4转发和4over6隧道转发的时延近乎相等，可见进行隧道封装的IPv6头部对整体转发的时延影响较小。

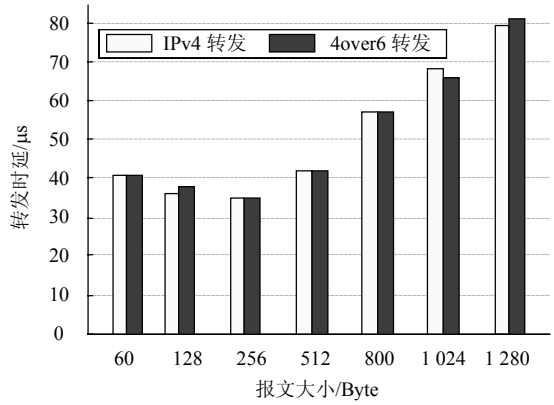

图8 隧道网关转发时延的测试结果

综合分析以上实验数据，证明本文提出的基于表项优化的隧道传输系统模型能优化存储并线速转发，且实现方法较简单，同现有隧道传输网络友好兼容，是一种有效的优化方法。

6 结 论

本文在归纳分析当前隧道传输网络所存的共性问题后，提出了表项聚合和表项拆分理论，设计了基于表项优化的高效隧道转发模型。并结合4over6隧道网络，设计了该优化机制的实现方法。该优化机制在小幅度修改现有路由设备转发表的基础上，实现了性能的最大优化，保证了同现有网络设备和应用的兼容性，实现方法也较为简单。本文工作的优化方法也可用于其他网络转发设备的数据转发处理。设计者可以结合实际需求规划各种表项参数，实现选择性优化。最后，原型系统实验验证了所提理论的有效性并评估了模型的性能。

本文研究得到清华大学网络研究所的大力支持，在此表示感谢。

参 考 文 献

[1] CUI Yong, WU Peng, XU Ming-wei, et al. 4over6: Network layer virtualization for IPv4/IPv6 coexistence[J]. IEEE Network, 2012, 26(5): 44-48.

[2] WU Jian-ping, CUI Yong, LI Xing, et al. 4over6 transit solution using IP encapsulation and MP-BGP extensions[EB/OL]. [2014-12-07]. http://www.faqs.org/rfcs/rfc5747.html

[3] M BATENI, A GERBER, M HAJIAGHAYI et al. Multi-VPN optimization for scalable routing via relaying[C]//28th IEEE International Conference on Computer Communications. Brazil: IEEE, 2009.

[4] CONTA A, DEERING S. Generic packet tunneling in IPv6 specification[EB/OL]. [2014-12-07]. http://www.faqs.org/rfcs/rfc2473.html

[5]DEERING S, HINDEN R. Internet protocol, version 6 (IPv6) specification[EB/OL]. [2014-12-07]. http://www.faqs.org/rfcs/rfc2460.html.

[6] 吴建平, 李星, 崔勇等. 4over6: 基于非显示隧道的IPv4跨越IPv6互联机制[J]. 电子学报, 2006, 34(3): 455-458.
WU Jian-ping, LI Xing, CUI Yong, et al. 4over6: IPv4 network interconnection over IPv6 backbone without explicit tunneling[J]. Chinese Journal of Electronics, 2006, 34(3): 454-458.

[7] 任旭明. IPv4/IPv6路由器低功耗TCAM查表算法研究[D]. 北京: 北京邮电大学, 2013.
REN Xu-ming. Power-efficient tcam for IPv4/IPv6 routing tables[D]. Beijing: Beijing University of Posts and Telecommunications, 2013.

[8] CUI Yong, WU Jian-ping, WU Peng, et al. Public IPv4-over-IPv6 access network[EB/OL]. [2014-11-11]. http://www.faqs.org/rfcs/rfc7040.html.

[9] CUI Yong, WU Peng, XU Ming-wei, et al. 4over6: Network layer virtualization for IPv4-IPv6 coexistence[J]. IEEE Network Magazine, 2012, 26(5): 44-48.

[10] 陈文龙, 徐明伟. 面向地址空间分离网络的地址映射模型: AMIA[J]. 计算机学报, 2012, 35(1): 2-9.
CHEN Wen-long, XU Ming-wei. AMIA: Address mapping model facing the network with separated address space[J]. Chinese Journal of Computers, 2012, 35(1): 2-9.

编 辑 叶 芳