

## 基于并行 BP 神经网络的路由查找算法

胥小波<sup>1,2</sup>, 郑康锋<sup>1,2</sup>, 李丹<sup>1,2</sup>, 杨义先<sup>1,2</sup>, 钮心忻<sup>1,2</sup>

(1. 北京邮电大学 信息安全中心, 北京 100876; 2. 北京邮电大学 灾备技术国家工程实验室, 北京 100876)

**摘 要:** 结合 Bloom-filter 算法和并行反向传播神经网络, 提出了一种新的基于并行神经网络的路由查找算法 (BFBP)。该算法满足路由查找的需求, 只需学习路由条目的网络 ID, 且易于扩展到 IPv6 地址查询。研究结果表明, 相比于已有的神经网络路由查找方法, 该算法需要学习的条目数平均减少了 520 倍, 提高了学习效率, 为神经网络应用于路由查找创造了有利条件。

**关键词:** 神经网络; 路由查找; 反向传播; Bloom-filter 算法

中图分类号: TP393.03

文献标识码: A

文章编号: 1000-436X(2012)02-0061-08

## Routing lookup algorithm based on parallel BP neural network

XU Xiao-bo<sup>1,2</sup>, ZHENG Kang-feng<sup>1,2</sup>, LI Dan<sup>1,2</sup>, YANG Yi-xian<sup>1,2</sup>, NIU Xin-xin<sup>1,2</sup>

(1. Information Security Center, Beijing University of Posts and Telecommunications, Beijing 100876, China;

2. National Engineering Laboratory for Disaster Backup and Recovery, Beijing University of Posts and Telecommunications, Beijing 100876, China)

**Abstract:** A new routing lookup algorithm based on Bloom-filter algorithm and parallel back-propagation neural networks (BFBP) was proposed. The algorithm could meet the challenges of routing lookup and just had to learn the network ID moreover, it was equally attractive for IPv6. The results show that compared to other routing lookup methods based on neural network, BFBP algorithm reduces the average number of items which neural network has to learn by 520 times, improves learning efficiency of neural networks, and creates favorable conditions for the application of neural network in the area of routing lookup.

**Key words:** neural network; routing lookup; back-propagation; Bloom-filter algorithm

### 1 引言

路由查找是指路由转发设备根据数据分组的目的 IP 地址查找路由表, 选择最优路由条目的过程。它完成从源端到目的地的选路过程, 为网络搭建“信息高速公路”, 在数据报文的转发过程中发挥重要作用。

随着 Internet 的迅猛发展, 通信量呈每 6 个月

倍增的趋势, 显著高于著名的 Moore 定律。链路的传输速率随着光通信技术的突破性发展, 已不再是主要的技术性能瓶颈, 而路由转发处理则成了制约 Internet 发展的关键因素。研究表明<sup>[1]</sup>, 存储器访问速度的增长大约是每年提高 7%, 而 Internet 出口带宽的需求却大约每 22 个月翻一番, 即每年提高约 50%。因此, 仅依赖于提高存储器访问的速度来弥补或缓解 Internet 增长的需求是不现实的。

收稿日期: 2011-01-13; 修回日期: 2011-08-17

基金项目: 国家自然科学基金资助项目 (61070208); 中央高校基本科研业务费专项基金资助项目 (BUPT2009RC0218, BUPT2010PTB0502); 国家发改委信息安全专项基金资助项目; 教育部科学技术研究重点基金资助项目

**Foundation Items:** The National Natural Science Foundation of China (61070208); The Fundamental Research Funds for the Central Universities (BUPT2009RC0218, BUPT2010PTB0502); The National Development and Reform Commission Information Security Special Fund; The Key Project of Chinese Ministry of Education

本文研究了一种基于并行神经网络的路由查找算法,利用神经网络来提取路由条目中所隐含的内在规律。并行的神经网络由多个小的神经网络组成,使得每个神经网络需要学习的条目数减少,从而加速了学习过程,提高学习效率。神经网络的结构有利于并发查询,提高查询速度,实现仅查询一次就可得出最优路由。

## 2 研究背景

为了解决 IP 地址资源紧缺,减小路由表规模,互联网引入了无类域间路由 (CIDR, classless inter-domain routing) 地址划分。随之而来的问题是:路由条目前缀长度可以是 8 至 32 之间的任意长度,路由表查询要按照最长前缀匹配的原则进行查找。但是传统的路由器执行最长前缀匹配时间很长,使得路由表查找成为路由器转发速度的瓶颈。

支持最长前缀匹配路由查找的算法主要包括基于软件、硬件的解决方案。基于软件的算法以基于 Trie 树结构的查找为代表,从最初的二进制 Trie<sup>[2]</sup>树结构到平衡二进制 Trie 树<sup>[3]</sup>结构再到多分枝的优先 Trie 树<sup>[4]</sup>结构。基于 Trie 树的查询算法简单,软件实现容易,路由查找时间与树的深度成正比,但是路由更新时间较长,不适合动态路由更新。

基于硬件并行查找的代表是基于 TCAM (ternary content addressable memory, 基于内容访问的三态存储器) 查询,其优点为查询一次即可得到下一跳地址。存在的问题是:由于要按照前缀长度排序,路由更新比较困难,另一方面,TCAM 功耗很大。针对这个问题,Parineet<sup>[5]</sup>提出利用前缀长度的分布规律来减少 TCAM 中路由条目位置的变化,Heeyeol<sup>[6]</sup>采用基于 HASH 算法的 TCAM 最小化算法 (HTM),将 TCAM 分块查询,降低 TCAM 的能耗。

Bloom-filter 算法在 1970 年被提出<sup>[7]</sup>,但被引入到最长前缀匹配路由查找则是近几年的一个创新<sup>[8~10]</sup>,Bloom-filter 算法可以将不同的前缀分为不同的集合,在集合中利用 hash 映射到下一跳地址。将 Bloom-filter 用于最长前缀匹配可以减小查询范围,其性能和可扩展性优于 TCAM 查找,并且易于扩展到 IPv6 路由查找。

利用神经网络来解决最短路径问题,即求路由表的算法有很多,但是将神经网络引入路由查找的研究还很少,而且还在探索阶段。Mehran 等提出了

一种基于多层反向神经网络的路由查找方法<sup>[11]</sup>,该网络拥有 12 层结构,能够达到线速查找,但是在学习性能上有待提高。在此基础上,进一步提出了 3 种用于路由查找的神经网络算法<sup>[12]</sup>,并进行了比较,证明了多层 BP 神经网络算法相比于 Hopfield 网络和单层的 BP 神经网络查找效果更好。此方法可以实现无差错的路由查找,并可以在单个芯片上集成,可实现一个时钟周期完成一次查询。但正如作者所说,在神经网络的学习速度方面限制了这个问题在真实系统中的使用,而本文为这个问题提供了一个新的解决方案。

## 3 BF-BP 系统模型

### 3.1 问题描述

最长前缀匹配比精确匹配查找更加复杂,因为它要考虑 2 个维度的问题:前缀的值和长度。由于没有办法在查找前决定某一个目标 IP 的前缀长度,因此,基于神经网络的路由查询算法需要学习路由条目所包含的 IP 地址。增大了神经网络的复杂性,不利于路由条目的更新和路由表的快速收敛。

本文解决的问题是:利用 Bloom-filter 算法将不同前缀长度的路由条目分为不同的集合,每个集合对应一个神经网络,让神经网络只学习路由表中的网络 ID,而不需要学习此条目包含的所有 IP 地址。

原方案需要学习所有路由条目的所有 IP 地址,如对于网络号为 10.0.0/8 的路由条目,传统的神经网络需要学习  $2^{(32-m)}$  个 IP 条目,即  $2^{24}$  个 IP 地址。而本文提出的算法,只学习路由条目对应的一个网络地址,大大提高了学习效率。当路由器学习到路由条目 10.0.0.0/8 时,前缀长度为 8 的神经网络只需要学习网络 ID 为 10 的一个路由条目。

### 3.2 算法模型

BFBP 算法系统框图如图 1 所示,其中,BF 代表 Bloom-filter 算法模块,BP 代表反向神经网络模块,下标  $K(K=1,2,3,\dots,32)$  表示前缀长度。通过对  $K$  的划分,将 IP 地址空间划分成了 32 个前缀空间,每个前缀空间学习相应前缀长度的网络 ID,使整个大的复杂的神经网络划分为小的简单的神经网络的组合,这样做的好处是不用将有限的资源关注于整个 IP 地址空间,而是关注每个网络前缀对应的网络 ID 号。

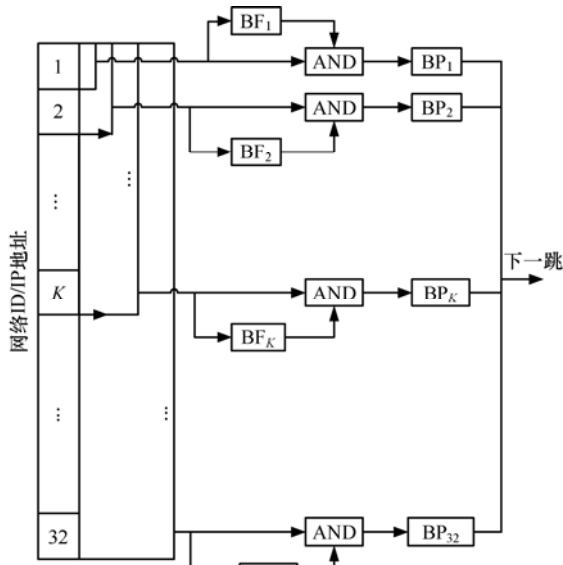


图 1 BFBP 算法模型

由于按前缀长度将 IP 地址空间进行划分并非是完全独立的。当路由器要查询一个 IP 地址时，无法确定对应的掩码长度，有可能是 32 个长度中的任意一个。因此，引入了 Bloom-filter 算法，这个算法用来记住每个前缀长度集合中所含的网络 ID 号，只有通过 BF 算法的验证，才进一步查询神经网络。

BFBP 算法模型充分利用 Bloom-filter 算法将最长前缀匹配的问题转化为多个不相关域的匹配问题，在每个域中，前缀长度相同。引入 Bloom-filter 算法有利于将多对一问题转化为一对一问题，而一对一问题可由神经网络进行学习、训练，进而提供查询。该算法采用完全并行处理架构，查找效率高，且易于扩展到 IPv6 路由查找，可应用到任意长度的 IP 地址查询。

BFBP 算法体系结构采用的 BF 模块和 BP 模块均具有伸缩性，该算法可方便扩展到 IPv6 地址查询的应用中。相应的算法模型将图 1 中最大 32 位前缀扩展到 128 位，相应的 BF、BP 模块从 32 个增加到 128 个。而由于在 IPv6 地址中，后 64bit 为接口信息 (interface ID)，可以将上述前缀长度空间缩小为 64 个。在 IPv6 大规模部署后，可针对骨干网路由表中 IPv6 地址前缀特点对 BFBP 算法的前缀范围进一步优化，从而将 BFBP 算法扩展到 IPv6 路由查询应用。

### 3.3 算法流程

路由学习阶段算法如图 2 所示，将学习到的路由条目送入相应前缀长度的 BFBP 模块，BF 模块将本前缀的网络 ID 记录在匹配标记向量中，然后 BP

算法会将此路由信息的网络 ID 学习到神经网络中。学习过程主要分为 3 个阶段：1) 从路由表中获取所有路由条目，如 1~6 行所示，其中，PrefixLen 代表路由前缀长度，Prefix 代表路由前缀，NextHop 代表路由下一跳索引；2) 为 BF 学习过程，如 7~10 行所示，由 Vector 二维数组来记录匹配标记向量组，根据前缀长度 PrefixLen 找到相应的 BF 模块，再将路由前缀 Prefix 经  $x$  个散列函数映射的单元加 1；3) 为 BP 神经网络学习过程，如 11~23 行所示，其中，11~13 行实现把路由条目输入网络，计算 BP 神经网络的输出，14~16 行实现计算输出单元的误差，17~19 行实现计算隐藏单元的误差，20~23 行根据误差反向更新 BP 神经网络的权值。

```

BFBP Router Learn Algorithm
Require
  Input: Router Items
  Output: BFBP
Get Route Items
1:FOR  $i=1:N$ 
2:   $Route \leftarrow RouterItems[i]$ ;
3:   $PrefixLen \leftarrow Route.Mask.Len$ ;
4:   $Prefix \leftarrow Route.Prefix$ 
5:   $NextHop \leftarrow Route.NextHop$ 
6:END FOR
BF Learn
7:FOR  $j=1:x$ 
8:   $h \leftarrow Hash_j(Prefix)$ 
9:   $Vector[PrefixLen][h]++$ ;
10: END FOR
BP Learn
11:FOR  $k=1:Items$ 
12:   $O_k \leftarrow BP(Route[k])$ ;
13: END FOR
14: FOR  $l=1:N$  Output
15:   $\delta_l \leftarrow O_l(1-O_l)(NextHop_l-O_l)$ 
16: END FOR
17: FOR  $m=1:Nhidden$ 
18:   $\delta_m \leftarrow O_m(1-O_m)*SumOutput(w_{lm} \delta_l)$ 
19: END FOR
20:FOR  $w_{ji}$ 
21:   $\Delta w_{ji} \leftarrow \eta \delta_j x_{ji}$ 
22:   $w_{ji} \leftarrow w_{ji} + \Delta w_{ji}$ 
23: END FOR

```

图 2 路由学习阶段算法

路由查询阶段算法如图 3 所示，将目的 IP 输入网络，将地址中不同长度的前缀位提取出来，送入对应长度的 BF 模块进行匹配查找；当 BF 网络判断通过后，再将其送入到 BP 神经网络，输出下一跳信息。如果有 2 个以上的神经网络均有输出，那么以最长的前缀为准，满足最长前缀匹配的查询规

则。其中，PrefixLen 代表前缀长度，Network 代表在不同前缀长度下相应的网络 ID；Hash<sub>j</sub> 代表 BF 模块中第 j 个散列函数。1~2 行实现提取不同前缀长度的网络 ID，4~9 行实现通过 BF 模块对前缀进行匹配，第 10 行实现 BF 前缀匹配命中后，通过 BP 模块对下一跳索引进行查询。

```

BFBP Router Lookup Algorithm
Require
  Input: Destination IP Address
  Output: Next Hop
BP Learn
1:FOR PrefixLen=1:32
2: Network←Prefix of DesIP
3: FOR j=1:x
4:   h←Hashj(Network)
5:   IF Vector[PrefixLen][h] > 0
6:   THEN j++
7:   ELSE continue;
8:   END IF
9:   IF j is equal to x
10:  THEN NextHop←BP(Network);
11:  END IF
12: END FOR
13:END FOR
  
```

图 3 路由查询阶段算法

### 3.4 K bit-BF 模块

本文采用的具有 K bit 前缀的 BF 算法如图 4 所示，具有 4 个关键模块：K bit 前缀输入，x 个散列函数(H<sub>1</sub>,H<sub>2</sub>,...,H<sub>x</sub>)、m bit 匹配标记向量(初始为 0)、与门输出单元。图中 W 表示 IP 地址的总长度，可以为任意长度，以方便应用到 IPv6 路由查找。

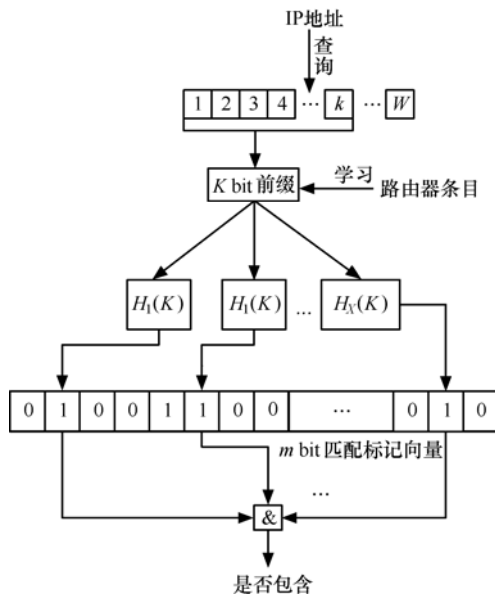


图 4 K bit 前缀 BF 算法功能结构

在学习阶段，将 K bit 前缀输入到 Bloom-filter 算法中，分别由 x 个散列函数产生值域为[1,m]的散列值，并将相应匹配标记向量中相应位置 1。为了支持路由条目的删除，将匹配标记向量改为计数向量。当增加路由时，将此条目的散列向量标记组相应位加 1；当删除路由条目时，相应计数位减 1。

在查找阶段，待查找 IP 输入到系统中，提取出网络 ID，作为 K bit 前缀输入，仅当 x 个散列单元输出值的相应值均不小于 1 时，才能说明此前缀集合中包含此网络 ID。

采用 Bloom-filter 算法能够节省 IP 地址存储空间，具有高效率的数据表示和查询结构，利用匹配数组简洁地表示前缀集合，并能够判断某个 IP 地址是否属于这个前缀集合。

### 3.5 K bit-BP 模块

K bit 前缀 BP 神经网络如图 5 所示，采用经典的 3 层 BP 神经网络结构。输入层在学习时为路由表中的条目，在查询时为从目的 IP 地址中提取的前缀，输出层为下一跳的索引号，路由器根据索引号，对数据分组进行转发。

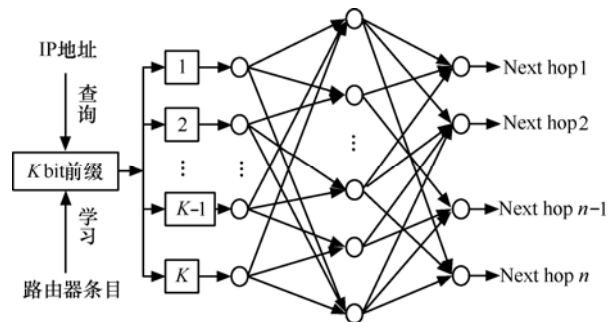


图 5 K bit 前缀 BP 神经网络功能结构

本文采用多神经网络并联结构，将一个大的神经网络分为 32 个小的神经网络，简化神经网络的复杂性，减少每个网络的学习条目，加速学习过程。输入层神经元数目取决于网络前缀长度，输出层取决于路由器输出端口数量，隐藏层与此前缀长度的路由数目成正比。

根据 Hecht-Nielsen 理论<sup>[13]</sup>，任意连续函数  $f: I^n \rightarrow R^m$  能够被具有  $2n+1$  个隐藏单元的神经网络精确表达，其中，I 属于封闭集 {0,1}，n 为输入单元个数，m 为输出单元个数。由此可得，具有 K bit 前缀的 BP 神经网络隐藏层数目上限为  $2K+1$ ，在实际系统中，可根据相应前缀长度的路由数目，设计隐藏层神经元数量。

### 3.6 误差消除学习

Bloom-filter 算法突破了传统散列函数的映射和存储元素的方式, 空间的节省和查询的高效是通过较低的错误率来换取的。文献[7]给出了正误差发生概率的近似估算值如式(1)所示。

$$fp = (1 - e^{(-nx/m)})^x \quad (1)$$

其中,  $m$  表示匹配标记向量长度,  $n$  表示路由表中某前缀长度的路由条目数,  $x$  表示散列单元个数。若取  $x=(m/n)\ln 2$ , 则正误差概率如式(2)所示。

$$fp = (0.6185)^{m/n} \quad (2)$$

由(2)式可以得出, 给定元素集的规模  $n$ , 只要对应的参数  $m$  和  $x$  足够大, 正误差可以达到足够小。根据文献[8], 内存达到 4MB, 可使冲突率趋于 0。由于路由表是缓慢增长的, 本算法采用拆分型 Bloom-filter 算法, 随着元素规模的增加而增加计数向量, 可处理动态数据。当计数向量达到处理极限时, 增加一个初始计数向量, 最终形成具有多个 Bloom-filter 计数向量组的矩阵。查找时, 向量组中任意一个向量命中, 则说明包含此网络 ID。

基于 Bloom-filter 的散列查找通过允许微小的误差存在, 在运用较少存储成本的基础上, 可以提供较高的、确定性的查找性能。但是该算法毕竟引入了查找误差, 尽管很小而且可以控制, 但这种误差并非随机的, 而是对于给定路由前缀集和待查找 IP 地址时是固定的, 即对于某些出错的 IP 地址, 永远出错。而在路由查找中, 要求零差错率。

为了实现零差错查找, 本算法一旦查出有 2 个以上的 BF 网络均通过, 而又不属于最长前缀匹配的情况, 则说明 BF 算法出现小概率误差事件。在这种情况下, 进一步学习神经网络, 使相应的神经网络纠正学习, 得出正确的输出结果。使神经网络与 BF 算法互补, 容忍 BF 算法存在的误差, 实现路由查找零差错。

### 3.7 算法复杂性分析

该算法设计采用 Bloom-filter 算法和神经网络学习模块, 初衷是利用它们的并行性, 由输入查询一次就可以直接得出结果。因此, 此算法的复杂性为  $O(1)$ , 与神经网络查询、TCAM 查询、Bloom-filter 路由查询等具有相同的时间复杂度。需要说明的是, 由于该算法采用 BF 模块和神经网络模块结合, 需要查询 2 次才能得到查询结果。也就是说, 算法时间复杂度在同一个级别上, 但是查询时间比神经

网络查询、TCAM 查询多了一倍。该算法需要 2 个时钟周期查询出匹配最长前缀的路由条目。

在空间复杂度上, 主要的存储空间是路由条目存储、Bloom-filter 匹配标记向量存储、神经网络结构存储, 后两者都为常量, 因此空间复杂度为  $O(n)$ 。在时间复杂性、空间复杂性上, 该算法与基于神经网络查询、TCAM 查询相同。但该算法主要是解决路由条目的更新问题, 可以大大提高路由条目更新的速度, 此问题在下一节重点说明。

## 4 算法性能分析

为了验证算法的学习效率, 本文将算法模型应用于真实的路由器, 表 1 给出了典型骨干网路由前缀分布情况<sup>[4]</sup>。它是根据自治系统号为 AS131072 的 APNIC R&D 路由器统计的, 总共有路由条目 323 352 条。可以发现路由前缀长度分布在 8 至 32 之间, 而且分布极不均匀, 长度在 16 至 32 的前缀约占总数的 99%, 前缀长度为 24 的路由最多, 约占总数的 52%。

表 1 APNIC R&D 路由前缀分布

前缀长度	路由个数	前缀在整个路由表中所占比例	前缀长度	路由个数	前缀在整个路由表中所占比例
8	19	0.01	21	22 771	7.04
9	10	0.00	22	29 610	9.16
10	25	0.01	23	29 282	9.06
11	67	0.02	24	169 468	52.41
12	192	0.06	25	997	0.31
13	401	0.12	26	1 245	0.39
14	692	0.21	27	679	0.21
15	1 263	0.39	28	213	0.07
16	11 058	3.42	29	56	0.02
17	5 289	1.64	30	26	0.01
18	8 989	2.78	31	0	0
19	18 339	5.67	32	8	0.00
20	22 653	7.01	其他	0	0

根据上述路由表, 可得出 BP 神经网络<sup>[12]</sup>、多层神经网络<sup>[11,12]</sup>、并行神经网络在各前缀情况下所需要学习的条目数量, 如表 2 所示。

由表 2 可得, 各算法的学习条目总数对比为: 经典神经网络学习整个路由表需要学习的条目数为 2 830 401 056, 约  $2.8 \times 10^9$  条。多层神经网络需要学习的条目数量为 88 282 090, 约  $8.8 \times 10^7$  条,

表 2 各神经网络算法所需要学习的条目数量

前缀长度	经典神经网络	经典神经网络学习 条目数量百分比	多层神经网络	多层神经网络学习 条目数量百分比	并行神经网络	并行神经网络学习 条目数量百分比
8	318 767 104	11.26	14 592	0.02	19	0.01
9	83 886 080	2.96	6 400	0.01	10	0
10	104 857 600	3.70	14 400	0.02	25	0.01
11	140 509 184	4.96	36 448	0.04	67	0.02
12	201 326 592	7.11	101 376	0.11	192	0.06
13	210 239 488	7.43	208 520	0.24	401	0.12
14	181 403 648	6.41	357 072	0.4	692	0.21
15	165 543 936	5.85	649 182	0.74	1 263	0.39
16	724 697 088	25.60	5 661 696	6.41	11 058	3.42
17	173 309 952	6.12	2 030 976	2.3	5 289	1.64
18	147 275 776	5.20	2 876 480	3.26	8 989	2.78
19	150 233 088	5.30	5 281 632	5.98	18 339	5.67
20	92 786 688	3.28	6 161 616	6.98	22 653	7.01
21	46 635 008	1.65	6 011 544	6.81	22 771	7.04
22	30 320 640	1.07	7 698 600	8.72	29 610	9.16
23	14 992 384	0.53	7 554 756	8.56	29 282	9.06
24	43 383 808	1.53	43 383 808	49.14	169 468	52.41
25	127 616	0.00	127 616	0.14	997	0.31
26	79 680	0.00	79 680	0.09	1 245	0.39
27	21 728	0.00	21 728	0.02	679	0.21
28	3 408	0.00	3 408	0.00	213	0.07
29	448	0.00	448	0.00	56	0.02
30	104	0.00	104	0.00	26	0.01
31	0	0	0	0.00	0	0
32	8	0.00	8	0.00	8	0.00
其他	0	0	0	0.00	0	0

相比于经典 BP 神经网络性能学习性能提升 32 倍。基于前缀长度的并行神经网络学习的条目数与路由条目数相同，为 323 352 条，约  $3.2 \times 10^5$ 。相对于多层神经网络，学习性能提升 520 倍，相对于经典 BP 神经网络，学习性能提升 8 753 倍。需要说明的是，本算法需要学习的 323 352 条路由，分布在 32 个小的神经网络中，最大的负载是前缀长度为 24 的路由条目数，为 169 468 条，约  $1.7 \times 10^5$  条。

按照表 2 的学习比例统计数据，可得出神经网络学习条目数的百分比图，客观展示资源分配情况，如图 6 所示。从图中可见，经典的 BP 算法没有考虑到路由地址分层结构的特点，在较短的前缀上面学习用了较多的资源。而本算法与多层神经网络算法较好地考虑了不同前缀的分布特

点，学习路由条目所耗资源与路由表中路由条目的分布情况成正比。

基于表 2 的数据统计，可得出各前缀长度下，3 种神经网络算法所需要学习的路由条目数量图，如图 7 所示，曲线与学习算法的对应关系与图 6 相同。另外，由于需要学习的路由条目数量级差别比较大，本图纵坐标取对数后均等划分；由于对 0 取对数无意义，对取值为 0 的数据进行了处理，使之取值为 1。由图 7 可见，经典 BP 算法的学习量最大，多层 BP 神经网络次之，并行神经网络最小。在各前缀长度下，本算法所需要学习的路由数量远小于其他 2 种神经网络路由学习算法。在前缀长度为 8 到 24 之间，路由数目所占的比重较大，需要学习的路由条目相差在  $10^2$  数量

级以上。可以看出，基于并行 BP 神经网络的路由查找算法极大地减少了需要学习的路由条目数量，提高了路由更新的效率。

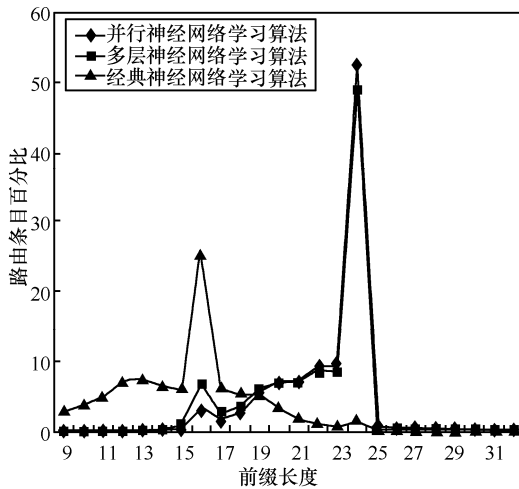


图 6 神经网络学习路由条目百分比

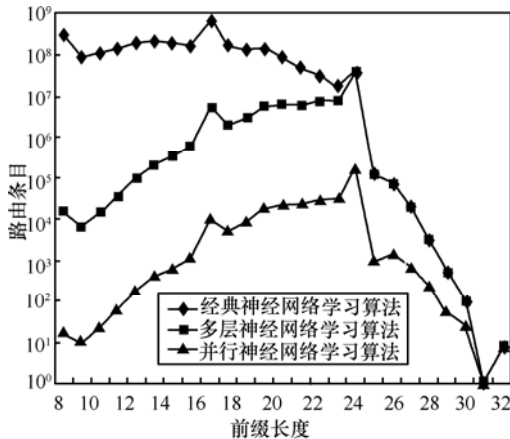


图 7 神经网络学习路由条目数量

为了测试 BFBP 算法对不同网络环境的适应性，本文在不同的网络环境下，采用真实路由器产生的路由表进行统计实验。图 8 为不同前缀长度下，所需要学习的平均路由条目数，曲线与学习算法的对应关系与图 6 相同，纵坐标取对数后均等划分。从图中可以看出，在不同的前缀长度，BFBP 算法相对于已有的 2 种神经网络路由学习算法有不同程度的性能提升。但随着前缀长度的减小，提升幅度降低，特别是在前缀长度大于 24 以后，提升幅度小于 128 倍。这是因为在前缀长度较大时，路由条目所包含的 IP 地址数量较少。综上所述，BFBP 算法适用于具有短前缀路由条目的路由器中，即至少包含前缀长度小于 24 的路由条目。因此该算法在支持路由汇聚的路由器和骨干网路由器中具有较好的适用性。

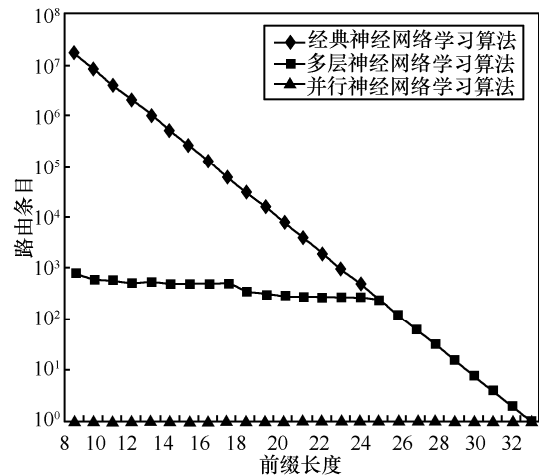


图 8 前缀相关学习路由条目平均数量

### 5 结束语

针对神经网络路由查找时，需要学习路由条目中所有 IP 地址，学习量大，路由更新速度慢等问题，本文提出了一种新的 BFBP 路由查找算法。该算法利用 BF 过滤器将 IP 空间划分为独立的前缀空间，每一个空间的前缀长度相同，每个神经网络只需学习路由条目中的网络 ID，而不需要学习所有的 IP 地址，将有限的资源集中于学习网络 ID。研究结果表明，相比于已有的神经网络路由查找方法，该算法需要学习的条目数平均减少了 520 倍，提高了神经网络的学习效率，有利于路由更新速度，为神经网络在路由查找方面的应用创造了有利条件。

BFBP 算法可用于解决多维分类问题，利用 BF 算法从某一维度上将问题分为不同的域，在每个域中利用神经网络学习其他维度的信息，达到降低维度，降低学习复杂性的目的。并行 BP 神经网络是普通 BP 网络在空间上的拓展，该模型对多个维度的分类问题具有良好的适应性。由于采用完全并行处理的架构，算法可扩展性强，可应用于任意长度的分类问题。本文应用该模型在路由查找问题上提出了一种新的路由查找解决方案，有关该模型在其他方面的应用是值得进一步研究的问题。下一步计划将此算法模型推广到更多的多维分类算法中。

### 参考文献：

[1] HENNESSY J L, PATTERSON D A. Computer Architecture: a Quantitative Approach[M]. Beijing: the China Machine Press, 2007.

- 288-289.
- [2] SHAFFER C A. A Practical Introduction to Data Structures and Algorithm Analysis[M]. New Jersey: Prentice Hall, 2010.123-135.
- [3] LIM H, KIM H, YIM C. IP address lookup for Internet routers using balanced binary search with prefix vector[J]. IEEE Trans Comm, 2009,57(3): 618-621.
- [4] LIM H, YIM C. Priority tries for IP address lookup[J]. IEEE Trans Compu, 2010, 59(6): 784-794.
- [5] PARINEETH M R. Fast updating algorithm for TCAMs using prefix distribution prediction[A]. International Conference on Electronics and Information Engineering (ICEIE)[C]. Kyoto, Japan, 2010. 1400-1404.
- [6] HEEYEOL Y. A memory and time-efficient on-chip TCAM minimizer for IP lookup[A]. Design, Automation & Test in Europe Conference & Exhibition (DATE)[C]. Dresden, Germany, 2010. 926-931.
- [7] BLOOM B H. Space/time trade-offs in hash coding with allowable errors[J]. Communications of the ACM, 1970, 13(7): 422-426.
- [8] DHARMAPURIKAR S, KRISHNAMURTHY P, TAYLOR D E. Longest prefix matching using bloom filters[A]. Proceedings of ACM SIGCOMM'03[C]. Karlsruhe, Germany, 2003. 201-212.
- [9] SONG H, DHARMAPURIKAR S, TURNER J, *et al.* Fast hash table lookup using extended Bloom filter[A]. An Aid to Network Processing. Proceedings of ACM SIGCOMM'05[C]. Philadelphia, USA, 2005. 181-192.
- [10] DHARMAPURIKAR S, KRISHNAMURTHY P, DAVID E. Longest prefix matching using Bloom filters[J]. IEEE-ACM Trans on Networking, 2006, 14(2): 397-409.
- [11] MAHRAMIAN M M, YAZDANI N, TAHERI H, *et al.* A multilayer neural network for IP lookup and packet classification[A]. The 9th Asia-Pacific Conference on Communications[C]. Malaysia, 2003. 924-928.
- [12] MAHRAMIAN M M, YAZDANI N, TAHERI H, *et al.* Three new neural network based algorithms for IP lookup and packet classification[J]. Iranian Journal of Science and Technology, 2005, 29(B1): 11-22.

- [13] FAUSETT L. Fundamentals of Neural Networks Architectures, Algorithms and Applications[M]. Prentice Hall, 1993.
- [14] BGP routing table analysis report [EB/OL]. <http://bgp.potaroo.net/>.

#### 作者简介:



**胥小波** (1985-), 男, 四川盐亭人, 北京邮电大学博士生, 主要研究方向为信息安全、人工智能算法。



**郑康锋** (1975-), 男, 山东烟台人, 北京邮电大学讲师, 主要研究方向为网络与信息安全。

**李丹** (1986-), 女, 甘肃陇西人, 北京邮电大学博士生, 主要研究方向为信息安全、人工智能算法。



**杨义先** (1961-), 男, 四川盐亭人, 北京邮电大学教授、博士生导师, 主要研究方向为信息安全、密码学。

**钮心忻** (1961-), 女, 浙江湖州人, 北京邮电大学教授、博士生导师, 主要研究方向为数字水印、信息隐藏和隐写分析。