文章编号: 1001- 2486(2005) 01- 0064- 05

# 基于并行流水的转发引擎设计与性能分析*

张晓哲, 彭 伟, 朱培栋
(国防科技大学 计算机学院, 湖南 长沙 410073)

摘 要: 光通信技术对核心路由器报文转发能力不断提出更高的要求。10Gbps 光传输技术已经使现有的各种软硬件路由查找方法成为核心路由器转发能力的瓶颈, 而更高性能的光传输技术则已经突破了存储器访问速度的极限, 使得基于单片存储器的路由转发方法无法应付未来日益增长的需求。在硬件存储器价格非常低的前提下, 提出一种使用多个存储器并行流水查找的硬件转发实现结构。通过使用 Internet 上真实报文数据进行的性能模拟可以看出, 随着并行度的增加, 整个转发结构可以获得近似于线性的性能加速比。

关键词: 核心路由器; 转发引擎; DRAM; 模拟器
中图分类号: TP393. 4 文献标识码: A

# Parallel Pipeline Forwarding Engine Design and Performance Evaluation

ZHANG Xiao-zhe, PENG Wei, ZHU Pei-dong
(College of Computer, National Univ. of Defense Technology, Changsha 410073, China)

**Abstract:** With the rapid development of communication technology, optical transmit technique demands higher requirement of the core router's forwarding performance. OC192 POS interfaces have made the core router's forwarding engine a new bottleneck. Further development of optical transmit technique has exceeded the maximum accessing ability of DRAM, which makes impossible the design of the new forwarding engine with one chip of DRAM. New hardware parallel forwarding engine design is offered on the basis of Gupat's DIR-24-8-BASIC forwarding architecture, taking advantage of the parallel of multiplex DRAM chips. In order to get really performance of the parallel forwarding engine design, IP packet header trace and routing table's dump of Internet core router node are used as the input of the target system's simulator. The result shows that the parallel forwarding engine can achieve linear speedup with the increase of parallel basic forwarding tables.

**Key words:** core router; forward engine; DRAM; simulator

根据摩尔定律, CPU 速度每 18 个月翻 1 倍, 而 Internet 的带宽则翻 4 倍, IP 加光网将构成未来 Internet 核心网已成为不争的事实。网络带宽迅速增加, 对作为 Internet 网络核心设备 ——路由器的性能要求越来越高。高速转发引擎是实现高端路由器的关键技术之一, 高速转发引擎要在极短的时间内完成 IP 报文的完整性检查、校验和计算; 查表决定下一跳 IP 地址和交换端口号; 包分类和过滤规则检查等, 其中查表是耗时较多的操作。对使用频率较高的 64 字节小报文, 一个 10Gbps POS 接口在满负荷情况下每秒需要处理将近 20M 个报文查表请求, 已经达到了 50ns DRAM 访问速度的极限。而 OC768 (40Gbps) 传输技术以及未来 OC3072 技术, 则对核心路由器查表处理能力提出更高的性能要求。目前现有的各种软件、硬件查表算法远远不能满足未来需求, 因此基于硬件存储器提出更高性能的路由查找算法是急需解决的问题。随着互联网技术的不断发展, IPv4 地址空间将被耗尽, 为此 IPng 工作组提出了 IPv6 协议。它具有长达 128 位的地址空间, 可以彻底解决 IPv4 地址资源不足的问题。由于 IPv4 的广泛使用, 在相当长的时间内 IPv4 仍然将与 IPv6 并存。因此研究 IPv4 高速转发引擎仍然对 Internet 的发展具有重要意义。本文基于 Ipv4 不会快速消失的前提下, 在 Gupta 硬件转发结构算法[1] 的基础上提出了

* 收稿日期: 2004- 10- 20
基金项目: 国家自然科学基金资助项目(90204005); 国家 863 高技术资助项目(2003AA121510)
作者简介: 张晓哲(1976—), 男, 博士生。

一种基于并行思想的路由查找引擎实现结构。

## 1 相关工作

目前硬件实现的最长前缀匹配算法主要有 CAM(内容寻址存储器)、Trie 算法和哈希表索引算法。

CAM(内容寻址存储器)和传统的基于地址寻址的主存系统不同,能够在一个硬件周期内完成基于存储内容的精确匹配查找。已经提出很多基于 CAM 的最长前缀匹配算法[2, 3],但是由于 CAM 存储器的实现结构复杂,集成度比传统的 DRAM 存储器低很多,存储器容量也比 DRAM 小很多,还存在着价格昂贵和功耗高的问题,为此在商用路由器上使用大量的 CAM 来提高查找性能不是一种经济的解决方案。

David E. Taylor 等提出[4]一种基于硬件 FPGA 实现的位图压缩 Trie 算法。由于 Trie 算法查找的复杂性,一个硬件实现的路由引擎无法充分利用 SDRAM 提供的带宽资源。为此在性能测试过程中引入了并行执行的思想,在路由引擎控制器的控制下,使用 8 个路由引擎可以达到很好的性能加速比。但是并没有对并行执行的问题做进一步的探讨。Gupta 使用哈希表索引的思想[5],提出了 DIR-24-8-BASIC 转发表结构。将 32 位的 IPv4 地址按照 Internet 骨干结点的路由前缀分布规律[5]划分为 24 位长的基本索引表和路由前缀长度超过 24 位的扩展索引表。基本索引表在 DRAM 中连续存放,表项根据对应路由前缀长度的不同而存放着下一跳信息或者扩展表的索引值。基本索引表项内容通过高位的类型域来区分。对前缀长度超过 24 位的路由项,除了在基本表中占据一项外,还要在扩展表中分配连续的扩展项,长度一般为 $2^8$= 256 项。基本表的表项中填写扩展表的索引值,在扩展表被该路由覆盖的所有表项中填写下一跳的转发信息。整个转发结构的性能受限于 DRAM 存储器的访问速度,最好情况能达到一次访存完成一个目的地址的最长前缀匹配查找过程。目前常见的 DRAM 存储器速度从 100ns 到 50ns 不等,访存时间通常是输入队列速度的 10 倍左右。在报文转发过程中,大量的时钟周期浪费在等待 DRAM 访存上。为了获得更高的报文转发能力,必须在体系结构上突破 DRAM 存储器的速度限制。

## 2 并行流水查找方法

在 Gupta 算法的基础上,引入传统体系结构中非常成熟的流水线和多功能部件并行的思想,提出了能够与输入队列速度匹配的并行路由查找结构(如图 1 所示)。整个并行结构主要由下面几部分组成:

(1) 报文队列: 网络接口收到的 IP 报文头存放在报文队列中,等待引擎调度器的调度。在网络接口上出现突发流量时,报文队列还可以起到平滑流量的作用。

(2) 引擎调度器: 对报文队列中的 IP 报文头,按照轮转调度算法,使用目的地址高 24 位访问下一个空闲基本表的 DRAM 存储器,以异步的方式等待基本表返回对应的路由信息。在没有基本表空闲的情况下,停止报文队列中 IP 报文头的流出。引擎调度器内部不包含任何的报文缓冲机制,在出现扩展表忙造成报文阻塞的情况下,停止下一个访问扩展表的查找请求流出。

(3) 基本表存储模块: 每个基本表使用一块独立的 DRAM 存储器,以便通过引擎调度器的调度,充分利用各个基本表之间的并行性。基本表中路由信息的存储方式与 DIR-24-8-BASIC 转发表完全相同。

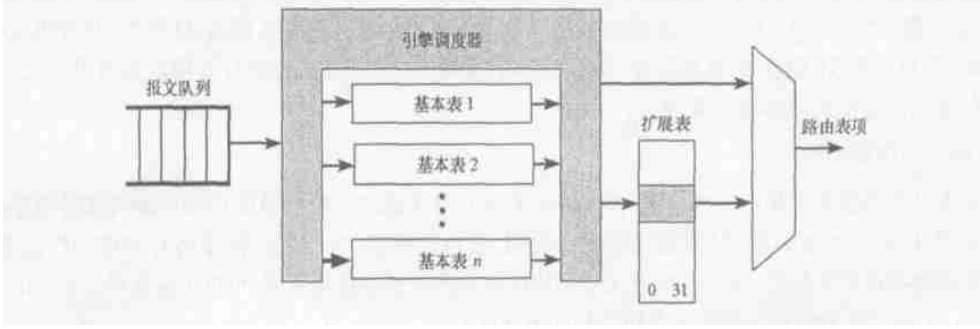


图 1 并行流水转发引擎

Fig. 1 Parallel pipeline forwarding engine

(4) 扩展表存储模块: 整个转发结构只使用了一个独立的扩展表, 扩展表的路由存储方式、扩展块的分配管理方式也与 DIR-24-8-BASIC 转发表完全相同。

并行路由查找结构在接收到 IP 报文时, 将 IP 报文头存放在报文队列中等待调度。引擎调度器从报文队列中取出报文头, 以轮转的方式选择一个空闲的基本表, 发出路由查表请求。每个基本表在完成路由查找操作后, 根据查表结果决定是否需要继续访问扩展表。对一次查找就可以完成最长前缀匹配查找过程的报文, 将报文加入到内部交换网络的输入队列中等待转发。对需要访问扩展表的 IP 报文, 重新将其交给引擎调度器。由引擎调度器对扩展表发出查表请求, 由扩展表完成最终的路由查找操作。在处理路由更新时, 由于每个基本表存储的路由信息完全相同, 引擎调度器对所有基本表同时发出更新操作, 由每个基本表独立完成路由更新。因此对多个基本表的更新操作时间与对单个基本表的情况完全相同。整个并行路由查找结构的路由更新复杂度与 Gupta 算法相同。

表 1 统计了在 MAE-West[7] 8 天的 Trace 数据中随机选取的 10 万个连续 IP 报文, 基于 Gupta 转发结构的模拟查表结果。模拟中使用的路由信息来源于 IPMA[6] 对 MAE-West 每隔 3 小时的路由统计结果。从表中可以看出, 扩展表的使用频率非常低, 在并行路由查找结构中使用冗余的扩展表几乎不能获得任何性能的提高。但是单扩展表结构存在着一定的风险, 如果一段时间内出现大量需要访问扩展表的 IP 报文, 扩展表访问速度问题会造成转发性能急剧下降。最坏情况下, $n$ 个基本表并行转发结构性能会退化为单个 Gupta 转发结构的性能。

表 1 访问扩展表的报文比例

Tab. 1 The access ratio to extent table

| 数据来源时间 | 报文数 | 扩展表访问次数 | 占总报文数的百分比 |
|---|---|---|---|
| 2000 年 11 月 15 日 | 10 万 | 3195 | 3.195% |
| 2000 年 12 月 15 日 | 10 万 | 908 | 0.908% |
| 2001 年 1 月 15 日 | 10 万 | 428 | 0.428% |
| 2001 年 3 月 15 日 | 10 万 | 721 | 0.721% |
| 2001 年 4 月 15 日 | 10 万 | 487 | 0.487% |
| 2001 年 5 月 15 日 | 10 万 | 602 | 0.602% |
| 2001 年 7 月 15 日 | 10 万 | 269 | 0.269% |
| 2001 年 8 月 15 日 | 10 万 | 510 | 0.51% |

## 3 性能评测

将 IPMA[6] 对 MAE-West 骨干路由结点每隔 3 小时的路由统计结果, 作为并行路由查找结构中各个基本表、扩展表中的路由转发信息。将 NLANR 被动测量和分析项目(PMA)[7] 记录的穿过 MAE-West 骨干路由结点的 IP 报文头作为模拟程序的 IP 目的地址的来源。PMA 使用 DAG3.5[8] 报文捕获卡记录 MAE-West 路由节点上 OC12c POS 链路的 TCP/IP 报文头。每个报文记录长度为 44 字节, 包含报文的到达时间、IP 报文头(不包含 IP 选项部分) 和部分 TCP 报文头, 记录报文到达时间精度为百万分之一秒。详细的捕获记录格式请参见文献[9]。

### 3.1 转发结构模拟器

转发结构模拟器使用 C++ 语言在 Windows 平台上编译通过。模拟器使用脚本配置文件描述路由信息来源、IP 目的地址来源、时钟模块的时钟周期, 模拟结束条件, 转发结构中各种硬件的性能参数。为了模拟各种配置下的转发性能, 配置文件采用简单的转发结构描述语言, 可以生成各种并行、串行、基于 Cache 的硬件转发结构。模拟器逻辑结构如图 2 所示。

### 3.2 模拟结果分析

文献[4] 的测试结果表明, 每秒 1000 次的路由更新对转发过程几乎不产生任何的影响。在骨干网

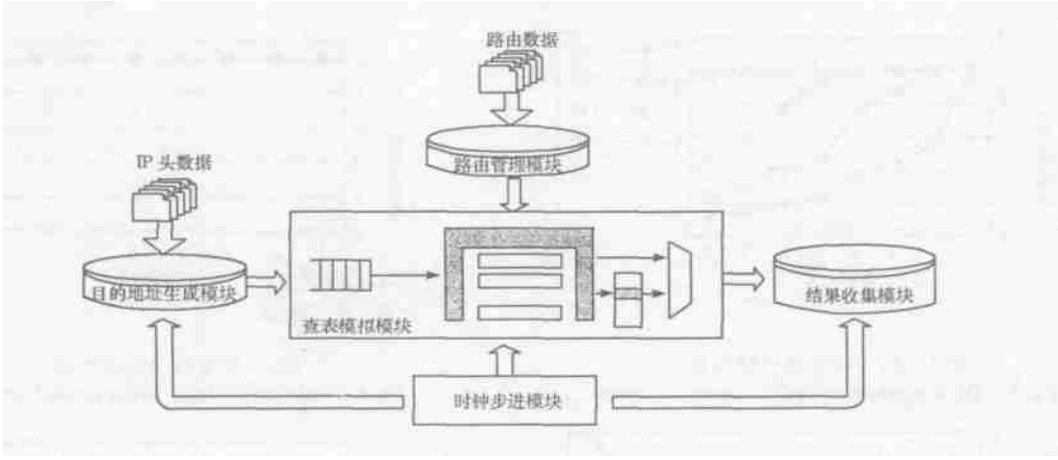


图 2 转发结构模拟器

Fig. 2 Parallel pipeline forwarding engine simulator

络上, 路由节点大量采用 BGP(边界网关路由协议) 进行互联, 每秒钟 BGP 的平均路由更新数量不会超过几百条[10], 所以在模拟过程中, 没有测试路由更新对转发性能的影响。

从 MAE-West OC12c POS 链路上采集的数据流量比较小, 首先从 2000 年 11 月到 2001 年 8 月每月 15 日的 IP 报文 Trace 数据中提取出报文目的地址和到达时间, 对每条数据流按照相同的比例缩短报文的到达间隔。在比较不同负载情况下转发性能时, 分别对这 8 天的数据流进行聚合, 形成 2 条流到 8 条流的聚合结果。下面的性能测试都是基于聚合流的。

模拟的硬件参数如表 2 所示。

表 2 模拟参数

Tab. 2 Simulation parameters

| 时钟周期 | 基本表 DRAM 速度 | 扩展表 DRAM 速度 | 输入队列长度 |
|---|---|---|---|
| 100MHz | 100ns | 100ns | 512 项 |

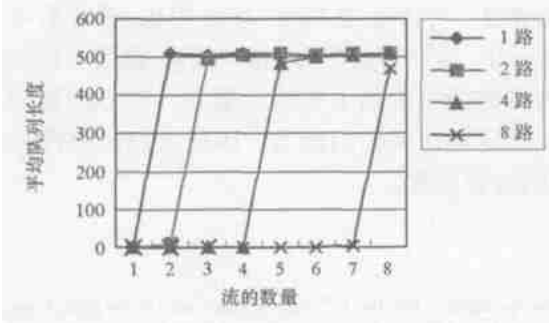


图 3 平均队列长度比较

Fig. 3 Average queue length

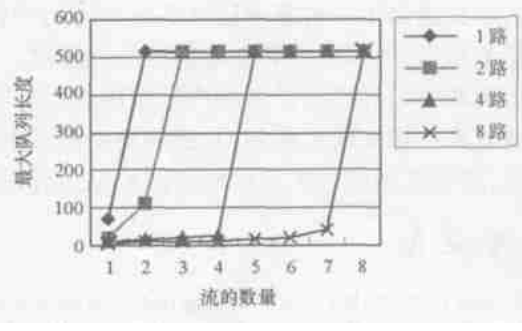


图 4 最大队列长度比较

Fig. 4 Maximum queue length

图 3~ 图 5 给出不同配置情况下基本表的性能比较, 其中“1 路”表示 Gupta 提出的硬件转发结构。图中横坐标轴表示 1 个到 8 个流的输入情况。可以看出, 随着流量增大, 在超过硬件转发能力时, 平均队列长度、最大队列长度急剧增长, 迅速使队列达到饱和, 而报文转发数量, 由于队列饱和后丢弃报文的影响, 随着报文流量的增大而减小。比较图中性能曲线, 可以看出增加基本表的数量能够获得近似于线性的加速比, 其中 8 路并行在重负载情况下, 其转发报文能力和队列长度上都优于其它配置情况。

单扩展表结构在遇到大量连续访问扩展表的 IP 报文时, 整体转发性能会下降。在理想情况下, 引擎调度器调度多个 Gupta 转发结构并行流水执行, 扩展表数量与基本表数量匹配时, 不会出现性能下降问题。8 路并行的基本表查找速度非常快, 更容易反映出由扩展表造成的性能下降, 为此图 6~ 图 8 比较 8 路并行与理想的 8 个 Gupta 转发结构并行之间的性能差异。从图 6 转发性能比较中可以看出, 在 8

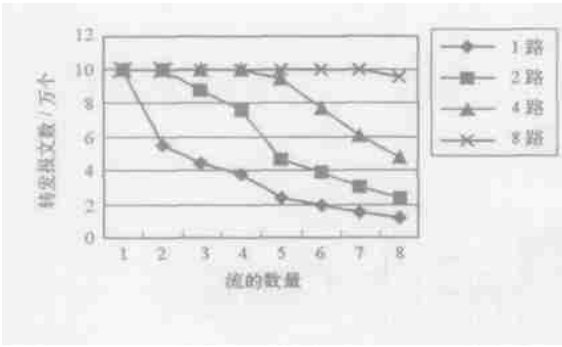


图 5 10 万个报文中转发数

Fig. 5 The forwarding ratio of ten thousand packets

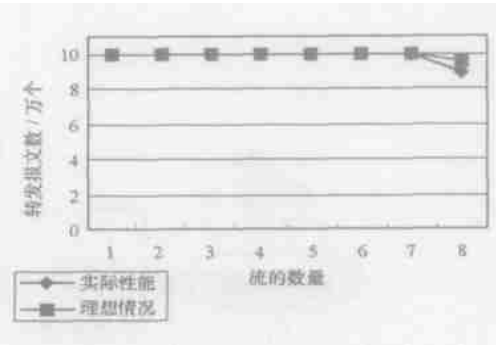


图 6 转发报文性能比较

Fig. 6 Eight paralle basic tables vs ideal condition

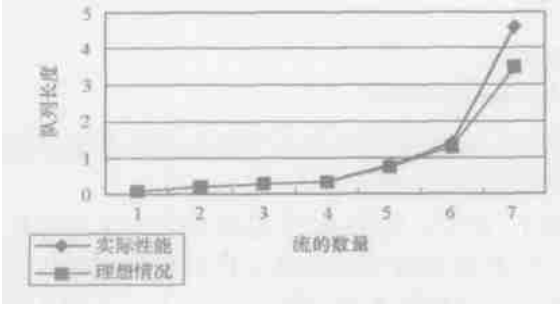


图 7 平均队列长度比较

Fig. 7 Average queue length

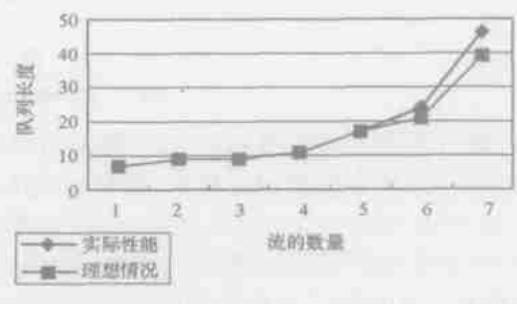


图 8 最大队列长度比较

Fig. 8 Maximum queue length

条流的情况下, 前者只有不到 6% 的性能下降。图 7 和图 8 比较了平均队列长度和最大队列长度的差异, 由于在 8 条流情况下, 队列长度迅速上升到接近 512 项, 为了更清楚地反映差别, 本文只比较了 1 至 7 条流的情况。从图中可以看出, 两者性能曲线非常接近, 输入队列长度没有明显的增长。

## 4 结论和进一步的工作

本文提出了一种基于硬件实现技术的并行路由查找结构。与传统的 Gupta 算法相比, 随着基本表数量的增加, 可以获得非常好的性能加速比。在重负载情况下, 报文转发能力和队列情况都优于传统方法。随着网络规模的扩展和网络应用的不断增加, 对路由器的能力提出了更高的要求。核心路由器不仅要有高速报文转发的能力, 还需要支持灵活高速的报文过滤和访问控制能力。因此如何扩展硬件转发表, 研究基于硬件的高性能访问控制, 是值得继续研究的重要问题。

## 参 考 文 献:

[1] Gupta P, Lin S, McKeown N. Routing Lookups in Hardware at Memory Access Speeds[A]. In: The 17th Annual Joint Conf. of the IEEE Computer and Communications Societies, San Francisco, USA, 1998: 1240- 1247.

[2] Gupta D S P. Fast Incremental Updates on Ternary-CAMs for Routing Lookups and Packet Classification[J]. In: IEEE Micro., 2001, 21(1).

[3] SiberCore Technologies Inc. SiberCAM Ultra-2M SCT2000[Z]. Product Brief, 2000.

[4] Taylor D E, Lockwood J W. Scalable IP Lookup for Programmable Routers[A]. In: The 21st Annual Joint Conf of. the IEEE Computer and Communications Societies, New York, 2002: 562- 571.

[5] Merit Networks[Z]. Inc. http://www.merit.edu.

[6] Internet Routing Table Statistics[R]. http://www.merit.edu/ipma/routing_table/.

[7] Passive Measurement and Analysis[Z]. http://pma.nlanr.net/Traces/.

[8] DAG3 architecture[R]. http://dag.cs.waikato.ac.nz/dag/dag3-arch.html.

[9] TSH Packet Format[R]. http://pma.nlanr.net/Traces/tsh.format.html.

[10] Maennel O, Feldmann A. Realistic BGP Traffic for Test Labs[A]. In: The 21st Annual Joint Conf. of the IEEE Computer and Communications Societies, New York, 2002: 31- 44.