

基于布鲁姆过滤器算法和三态内容寻址存储器的高效范围匹配方法

戴紫彬 刘航天*

(信息工程大学密码工程学院 郑州 450004)

摘要: 该文基于布鲁姆过滤器算法和三态内容寻址存储器(Ternary Content Addressable Memory, TCAM)技术提出一种高效范围匹配方法,解决了目前 TCAM 范围匹配方案存在的存储利用率低、功耗大的问题。设计基于最长共同前缀的分段匹配算法(Segmented Match on Longest Common Prefix, SMLCP)将范围匹配拆分为前缀匹配和特征区间比对两步,TCAM 空间利用率达到 100%。根据 SMLCP 算法设计了 BF-TCAM 模型,使用布鲁姆过滤器对关键字过滤,屏蔽无关项参与比较,大幅降低功耗。使用流水线缩短关键路径长度,使查找操作在一个时钟周期内完成。研究表明,所提方法实现了零范围扩张,工作功耗较传统 TCAM 降低 50%以上。

关键词: 范围匹配;布鲁姆过滤器;三态内容寻址存储器;零范围扩张;低功耗

中图分类号: TP393.08

文献标识码: A

文章编号: 1009-5896(2016)08-1872-08

DOI: 10.11999/JEIT151264

Efficient Range Matching Method Based on Bloom Filter and Ternary Content Addressable Memory

DAI Zibin LIU Hangtian

(Cryptography Engineering College, Information Engineering University, Zhengzhou 450004, China)

Abstract: An efficient range matching method based on Bloom Filter algorithm and Ternary Content Addressable Memory (BF-TCAM) technology is proposed to resolve the problem that there generally exist low memory using ratio and high power dissipation in current TCAM range matching methods. An algorithm of Segmented Match on Longest Common Prefix (SMLCP) splits range matching into two steps — prefix matching and feature range comparison, resulting in 100% TCAM space using ratio. BF-TCAM is designed according to SMLCP algorithm, which filters searching key words by Bloom filter to avoid that unrelated items participate in comparison, so as to reduce greatly power dissipation. Critical paths are streamlined so that searching operation can be completed during one clock cycle. Research results demonstrate that BF-TCAM achieves zero range expansion, meanwhile power dissipation falls more than 50%.

Key words: Range matching; Bloom Filter (BF); Ternary Content Addressable Memory (TCAM); Zero range expansion; Low power dissipation

1 引言

范围匹配广泛应用于网络 3 到 4 层的报文分类,根据源端口和目的端口字段匹配端口范围,实现访问控制、安全过滤、带宽控制等功能^[1,2]。在存储保护方面也有较多应用,比如审查进程发起的访存操作地址是否匹配其权限内的存储空间实现安全访问控制^[3,4]。这些实时应用对查找性能要求很高,高速的范围匹配是实现实时应用的技术支撑。

目前业界普遍使用三态内容寻址存储器(Ternary Content Addressable Memory, TCAM)实现高速查找表。TCAM 突出的问题在于它不适用于范围匹配,只能实现精确匹配和前缀匹配。比如

TCAM 可以方便地表达报文分类中的协议字段和 IP 地址字段,但端口范围无法直接实现。目前已有方案是将范围匹配转换成前缀匹配和精确匹配,一个范围字段往往映射出多条匹配表项,称为范围扩张(range expansion)。范围扩张降低了 TCAM 空间使用效率,造成了较大的配置负载,提高了更新代价,同时使 TCAM 的功耗问题雪上加霜。

文献[5]提出一种基于域转换的范围匹配算法 DTRM(Domain Transformation for Range Match),充分利用 TCAM 表项中的冗余位压缩规则集,理想情况下可使范围扩张因子达到 1.21 以下,TCAM 空间利用率提高到 82%以上。文献[6]提出一种基于 TCAM 的范围匹配方法 C-TCAM (Compressed TCAM),通过二级压缩将 2 个扩展后

的表项压缩成一个, 最坏情况下范围扩张因子为 $W-1$ 或 $W-2$ (W 为关键字二进制编码位宽, 如端口号位宽为 16), 同时减少查找过程中无效表项参与比较来降低功耗。文献[7]提出了基于格雷码的范围编码方法 SRGE(Short Range Gray Encoding), 利用格雷码相邻编码之间只有一位不同的特点对规则集进行压缩, 但随着范围长度增加格雷编码效率大打折扣, 只适用于较小范围, 最坏情况下扩展因子达到了 $2W-2$ 。文献[8]提出一种用于范围查找的新型存储器结构 SBiCAM(Smart Binary Content Addressable Memory), 无需规则扩展, 直接实现关键字与表项并行比较大小, 打破了传统的 TCAM 位比较结构, 因而可实现高效的范围匹配。但 SBiCAM 是一种新型电路, 投入实际应用尚需时日, 从 TCAM 的发展历程可以判断 SBiCAM 即使投入生产也要经历一段漫长历程才能达到较理想的性能以满足应用需求。

本文为解决基于 TCAM 的范围匹配问题提供了高效解决方案, 在实现零范围扩张的同时大幅降低 TCAM 功耗。提出基于最长共同前缀的分段匹配算法(Segmented Match on Longest Common Prefix, SMLCP), 将范围匹配转化为前缀匹配和特征区间比对两个步骤, 实现了零范围扩张, 使 TCAM 空间利用率达到 100%。根据 SMLCP 算法设计了 BF-TCAM 模型, 结合 Bloom filter 的前缀分类处理优势和 TCAM 的高速查找特性, 在保持高性能的同时大幅降低了 TCAM 功耗。

2 SMLCP 算法

2.1 问题描述

首先对范围匹配问题进行形式化描述, 以准确无误地表达问题实质, 为 SMLCP 算法提供理论基础。

S 是一系列范围区间的集合, $S = \{[s_1, t_1], [s_2, t_2], \dots, [s_n, t_n]\}$, 其中 $s_i, t_i (i = 1, 2, \dots, n)$ 均为位于数轴上的整数。 $[s_i, t_i]$ 是一段连续整数的集合, $1 \leq s_i \leq t_i \leq s_j \leq t_j \leq L (1 \leq i < j \leq n)$, L 是数值的上限, S 内所有元素均为闭集且相互之间没有重叠。

范围匹配问题等价描述为: 给定一整数 x , 在 S 中定位包含 x 的元素 $[s_i, t_i]$, 即 $\exists [s_i, t_i]$ 使 $x \in [s_i, t_i] \subseteq S$, 如图 1 所示。同时 S 集支持元素的动态插入与删除。

下面给出算法涉及的几个名词解释:

(1)最长共同前缀(Longest Common Prefix, LCP): 范围是一系列连续整数, 位于数轴上的一段闭区间。最小的整数称为起点, 最大的整数称为

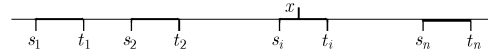


图1 范围匹配的形式化描述

终点, 构成范围的两个端点。起点与终点的二进制编码从第 1 个不同的比特位到最低位用 * 替代 (* 表示不关注), 构成三态表示, 这样的包含 * 的 W 位二进制编码称为范围的最长共同前缀 (W 为二进制编码位宽)。如 8 位范围 [22:38], 起点 22 的二进制编码为 0001_0110, 终点 38 的二进制编码为 0010_0110, 则 $LCP(22,38)=00**_****$ 。

(2)特征区间(feature range): 范围起点与终点去除最长共同前缀后组成的新区间, 它不改变范围的规模, 标识了范围内所有整数的上限与下限, 位宽为 $W-length(LCP)$ 。如范围 [22,38] 的特征区间为 [01_0110:10_0110], 位宽由 8 减为 6。

(3)偏移量(offset): 范围内的一个整数, 其二进制编码去除最长共同前缀后的数值, 标明了它在特征区间内的偏移量, 位宽与特征区间一致。如范围 [22:38] 内的整数 31, 偏移量为 $Offset(31)=01_1111$ 。

2.2 算法设计与证明

范围 $[s, t]$ 内任意一整数点 x , 其二进制编码分为 LCP 和 Offset 两段, SMLCP 算法利用这种特性, 将匹配过程分段进行: 首先查找与关键字 x 任意长度前缀相匹配的范围, 至多有 W 个范围的 LCP 与 x 相匹配, 然后再根据 x 的 Offset 精确筛选出匹配的范围。算法的关键在于第 1 步将搜索范围减少至 W 个以下, 大幅缩小了匹配范围。

定理 1 范围 $[s, t]$ 两端点 s, t 的 LCP 是范围内所有整数点的最长共同前缀且这种对应关系是唯一的。

由定理 1 可知端点的 LCP 可以唯一标识一段范围, 即 $P=LCP(s, t)$ 实现了 $[s, t]$ 到 P 的一一映射。注意 P 具有两个维度——前缀值和前缀长度, 前缀值相等且长度相同的两个范围必定完全重合。

定理 1 要从 LCP 前缀值和前缀长度两个维度予以证明, 包括:

(1)任意两段不同范围的 LCP 不同, 即若 $\forall [h, g], [s, t] \in S$ 且 $[h, g] \neq [s, t]$, 则 $LCP(h, g) \neq LCP(s, t)$;

(2)所有范围的 LCP 构成的集合称为标识集, $P=LCP(s, t)$ 是 $[s, t]$ 内所有整数点共同前缀中最长的, 即在标识集中不存在比 P 更长的 LCP 与范围内所有整数点相匹配。

范围为一个整数点时其前缀长度最长, 容易处理, 为方便证明, 首先设定范围 $[s, t]$ 至少包含两个整

数点。定理1证明如下:

首先证明(1),分为两种情况:

(a)当 $h=g=x$ 即范围 $[h,g]$ 为一整数点时,

LCP 为其自身,即 $LCP(h,g)=x$ 。

$\because P=LCP(s,t)$ 的前缀长度小于 x ,

$\therefore P \neq x$, 得证。

(b)当 $h \neq g$ 时,使用反证法,

假设 $LCP(h,g)=LCP(s,t)=P$,

令 $[h,g]$ 在数轴上位于 $[s,t]$ 的左边,即 $g < s$ 。

则 h,g,s,t 可以表示为

$h = P0^*, g = P1^*, s = P0^*, t = P1^*$ 。

$\because g < s$,

\therefore 有 $P1^* < P0^*$,

假设产生矛盾,得证。

下面证明(2), $P=LCP(s,t)$, $LCP(S)$ 为标识集。

使用反证法,假设 $\exists PQ \in LCP(S)$, 对于 $\forall x \in [s,t]$, $LCP(x)=PQ$, 其中 PQ 表示覆盖 P 的前缀,只需在 $[s,t]$ 中找出与 PQ 不匹配的整数即可证明假设错误,从而得证。

为方便证明,设定 $Q=0$ 。分两种情况:

(a)当 $[s,t]$ 包含 3 个及 3 个以上的整数点,

则有 $u=P10..0 \in [s,t]$ ($0..0$ 表示 1 个或若干个 0),

但 $LCP(u) \neq P0$ 。

假设产生矛盾,得证。

(b)当 $[s,t]$ 仅包含两个整数点,

则 $s=P0, t=P1$,

显然 $LCP(t) \neq P0$, 即得证。

综上所述,定理1得证。

LCP 适于 TCAM 存储,无需范围扩张就可以唯一表示范围,实现了 TCAM 的高效存储,使存储利用率达到 100%。对于关键字 x , 其任意长度的前缀可能与标识集中多个 LCP 相匹配,但其中至多只有一个 LCP 对应的范围包含 x 。例如 R_1 为 $[37,57]$, R_2 为 $[32,36]$, $P_1=LCP(R_1)=001^*_****$, $P_2=LCP(R_2)=0010_0^***$, 关键字 $x=0010_0101(37)$ 同时与 P_1, P_2 相匹配,但只有 R_1 命中 x 。要实现精确查找,还要对与关键字前缀相匹配的所有范围进行筛选。需要指出,并非一定是与关键字前缀相匹配的最长 LCP 包含 x , 上例中虽然 P_2 覆盖了 P_1 , 但其对应范围 R_2 并不真正包含 x 。易得知,如果存在多个范围的 LCP 与关键字前缀匹配,则这些 LCP 之间存在覆盖关系,且数量不会超过关键字的位宽。

定理2 令 $P=LCP(s,t)$ 的前缀长度为 j , 设 s', t' 表示 $s=P0s', t=P1t'$, $[0s', 1t']$ 构成了 $[s,t]$ 的特征区

间。如果关键字 x 长度为 j 的前缀与 P 相匹配,并且 $0s' \leq \text{Offset}_j(x) \leq 1t'$ 成立, $\text{Offset}_j(x)=x[W-j-1:0]$, 则 x 与 $[s,t]$ 匹配,否则不匹配。

定理2对关键字进行分段匹配:当关键字某一长度前缀与范围 LCP 相匹配且偏移量位于特征区间内,则与该范围匹配,否则不匹配。定理2易于证明,不再予以详细论证。

SMLCP 算法具体实现时,由 TCAM 和 RAM 分别存储范围 LCP 和特征区间,使用 TCAM 完成第1步关键字前缀与 LCP 的比较,缩小查找范围,然后以 TCAM 匹配表项的位置作为地址索引 RAM,查看关键字偏移量是否位于特征区间,完成第2步精确匹配。SMLCP 算法将范围匹配问题转化为前缀匹配和偏移比对两个步骤,在没有扩展规则集的情况下实现了精确查找,因而其扩张因子仅为 1.0。因为 SMLCP 算法存在离散范围的前提条件,所以在规则制定中要注意各个范围不能重叠,相互之间不可存在交集。

3 BF-TCAM 模型

3.1 模型设计

布鲁姆过滤器(Bloom filter)是一种空间效率很高的随机数据结构,利用位数组简洁地表示一个集合,能判断一个元素是否属于该集合。Bloom filter 包含一个 m 位的位数组,初始状态每一位都置为 0。为了表达含有 n 个元素的集合 $S = \{s_1, s_2, \dots, s_n\}$, Bloom filter 使用 k 个相互独立的哈希函数 Hash_i ($i = 1, 2, \dots, k$), 分别将集合中每个元素映射到 $\{1, 2, \dots, m\}$ 的范围中^[9-11]。

按前缀长度将范围分为不同集合,比如 IP 报文端口号有 16 位,将端口范围按 LCP 长度划分为 16 个子集,每个子集单独存储于 TCAM 中的一块区域。为降低 TCAM 的功耗,生产厂家在设计时将 TCAM 进行了分块,通过配置寄存器实现片选,只有选中的块参与运算。在查找时首先通过 Bloom filter 筛选出关键字可能所在的集合,然后驱动相应的 TCAM 块完成查找,这样仅被选中的 TCAM 块参与查找,功耗大幅降低。为进一步优化存储空间,可根据前缀长度的分布规律分配各个子集的存储空间。

BF-TCAM 模型由 4 类单元构成:基于 Bloom filter 的前缀预处理 BFPP(Prefix Preprocessing unit on Bloom Filter)单元, TCAM_RAM 单元,更新单元(Update Unit)和状态单元(State Unit),如图 2 所示。

BFPP 单元对关键字进行判断,筛选出关键字可能所在的前缀子集。为每个长度的前缀子集设置 Bloom filter, 分别为 $B_1, B_2, \dots, B_w, B_i$ ($i = 1, 2, \dots, w$)

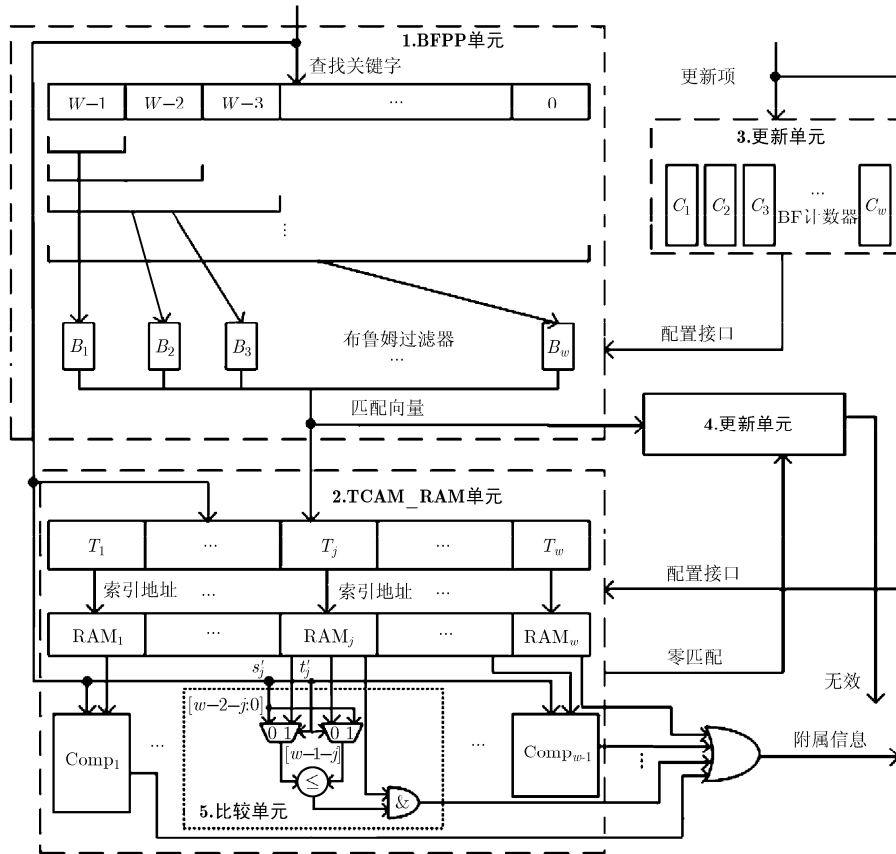


图 2 BF-TCAM 模型电路结构解析

判断关键字是否属于长度为 i 的前缀子集。所有 Bloom filter 对关键字相应长度的前缀并行运算，判断结果组成匹配向量，表明关键字命中的前缀子集。匹配向量作为 TCAM 的块区域片选信号，被选中的块区域参与关键字的前缀匹配运算。如果关键字没有命中任何 Bloom filter，则可判断关键字无效，通知状态单元。

TCAM_RAM 单元存储了所有范围的 LCP、特征区间及附属信息，每个前缀子集对应一组 TCAM_RAM，由片选信号进行选择。其中 TCAM 中存储范围的 LCP，在 BFPP 单元选中的块区域中搜索与关键字匹配的表项，以匹配表项的地址索引 RAM。RAM 中存储了范围的特征区间及附属信息，如端口范围和控制信息。每个子集(除长为 w 的子集)设置比较器判断关键字偏移量是否位于特征区间内， w 长子集为精确匹配，不存在特征区间，无需进行比较。特征区间端点最高位为固定值(起点为 0，终点为 1)，利用此特点优化比较器减小电路资源开销：使用 1 个比较电路实现区间判断(通常设置 2 个，一个判断大于起点，另一个判断小于终点)。对于长度为 $j(1 \leq j < w)$ 的比较器，仅关键字的低 $w-j$ 比特参与比较，比较电路输入位宽为 $w-1-j$ 。若所

有参与运算的比较器判断关键字不在特征区间内，可判断关键字无效，通过零匹配信号通知状态单元。匹配特征区间的比较器输出相关附属信息。

Bloom filter 位数组中每一位可能被多个元素共享，因而某元素的删除操作可能影响到其它元素，造成 BFPP 单元无法执行删除操作。为解决该问题，将 Bloom filter 位数组的每一位扩展为 BF 计数器，每执行一次插入/删除操作相应位加/减 1。进一步分析发现，更新操作较查找操作发生频率很低，为降低开销设置更新单元维护 BF 计数器，BFPP 单元中的 Bloom filter 依然保持位数组结构。当 BF 计数器跳变为 0 或由 0 变为正数时更新 BFPP 单元的 Bloom filter。TCAM_RAM 单元同步更新。

Bloom filter 的高效是以较低的错误率换取的，存在正误差，即 Bloom filter 判断通过的关键字以较小概率是无效的。这种误差并非随机的，对于特定关键字出现误判则该关键字永远出错。具体在模型中由 BFPP 和比较器的判断结果表现：当 BFPP 匹配向量不为 0 而比较器出现零匹配时，表明关键字通过 Bloom filter 判断却没有匹配的范围表项，误差出现。然而这种误差不影响匹配结果，最终将被比较器过滤掉，但较高的错误率会使更多 TCAM

块区域参与比较,从而增加了工作功耗。状态单元记录查找状态,通过无效信号指示关键字是否有效。

使用 Bloom filter 进行预处理,可快速判断关键字可能存在的前缀子集,过滤无关子集参与运算,但同时增加了电路关键路径长度,为了不降低时钟频率引入二级流水线,将前缀预处理和 TCAM_RAM 查找分为两级进行。

3.2 算法流程

配置算法(BF-TCAM Configuration Algorithm)流程如表 1 所示。

首先获取范围集,计算每一个范围的 LCP,然后对 LCP 进行 K 次哈希计算,由 BF 计数器记录计算结果,根据计数值配置 BFPP 单元中的 Bloom filter,同时将 LCP 存入 TCAM 相应位置,特征区间和附属信息存入 RAM 相应位置。整个配置算法分为 4 个阶段:(1)获取范围集,如 1~5 行所示,共

表 1 配置算法流程

Require
Input: Range_Items
Get the Collection of Range_Items
1 FOR $i=1:N$
2 range=Range_Items[i]
3 saddr=range.StartAddr
4 eaddr=range.EndAddr
5 acs_infor=range.Accessory_Infor
Calculate LCP
6 lcp=LCP(saddr,eaddr)
7 plen=length(lcp)
BF Learn
8 FOR $j=1:K$
9 $h=Hash_j(lcp)$
10 BFC_plen[h]+
11 END FOR// j
12 FOR $q=1:m$
13 IF(BFC_plen[q]==0)
14 BF_plen[q]=0
15 ELSE
16 BF_plen[q]=1
17 ENDIF
18 END FOR// q
Configure TCAM_RAM
19 TCAM_plen[i]=lcp
20 RAM_plen[i]={saddr[$w-plen-2:0$], eaddr[$w-plen-2:0$], acs_infor}
21 END FOR// i

有 N 个范围(range), saddr 表示起始地址, eaddr 表示结束地址, acs_infor 表示附属信息;(2)计算范围 LCP,如 6~7 行所示, plen 表示 LCP 的前缀长度;(3)训练 Bloom filter,如 8~18 行所示,其中 8~11 行是更新单元中 BF 计数器的学习过程,根据前缀长度 plen 找到对应的 BF 计数器,对 LCP 进行 K 次哈希计算, Hash $_j$ 表示 BF 计数器的第 j 个哈希函数,将哈希映射的位置加 1;12~18 行是 BFPP 单元 Bloom filter 的配置过程,对应 BF 计数器不为 0 的位被置 1, m 为 Bloom filter 数组的位宽;(4)配置 TCAM 和 RAM,如 19~20 行所示,根据前缀长度(plen)找到相应 TCAM 块存入 LCP 值,范围的特征区间 saddr[$w-plen-2:0$], eaddr[$w-plen-2:0$]和附属信息(acs_infor)存入 RAM 相应位置。3,4 阶段可并行运行。

查找算法(BF-TCAM Lookup Algorithm)流程如表 2 所示。

首先通过 Bloom filter 过滤关键字任意长度前缀,如果没有前缀匹配,判定关键字无效。若通过过滤,然后在匹配的 TCAM 块区域中查找相应长度前缀,将匹配表项的位置作为地址索引 RAM,最后由比较器判断关键字是否位于特征区间,如果匹配成功,输出附属信息,否则判定关键字无效。整个查找算法分为两个阶段:(1)前缀预处理,如 1~19 行所示,其中 1~15 行完成 Bloom filter 运算,分别对关键字各个长度的前缀进行哈希计算和判断,将命中的前缀子集标记在匹配向量(match_vector)中,这个阶段如果没有发现匹配前缀即匹配向量为 0 可判断关键字无效,如 16~19 行所示;(2)在命中的 TCAM_RAM 中查找关键字,如 20~34 行所示,TCAM 若存在匹配表项,以其位置作为地址(ram_addr)索引 RAM,比较器判断关键字偏移量是否位于特征区间,若存在匹配的特征区间,将对应的附属信息输出,完成范围匹配,否则判定关键字无效。

BF-TCAM 支持增量更新,包括插入和删除操作。插入算法与配置算法的 2,3,4 阶段相同,完成 BF 训练和 TCAM_RAM 表项的插入。删除与插入算法相似,BF 训练阶段对 BF 计数器进行减 1 操作,如果某位归零,则相应更新 BFPP 中的 Bloom filter,同时删除 TCAM_RAM 中的表项。各个长度的前缀子集分区存放,支持随机插入与删除,无需移动无关表项,因而 BF-TCAM 模型能够高效完成更新操作。

4 性能分析

表 3 为不同算法的关键性能指标对比,其中 F 为范围匹配包含的字段个数,对于报文分类来说为

2, 包括源端口和目的端口, W 为范围字段位宽, DTRM 算法将 W 位宽的范围字段划分为 M 个较短

表 2 查找算法流程

```

Require
  Input: Search_Key
  output:acs_infor, state

Prefix Preprocess
1 FOR plen=1: W
2   prefix=Search_Key[plen-1:0]
3   FOR j=1:K
4     h=Hash_j(prefix)
5     IF BF_plen[h]==1
6       j++
7     ELSE
8       match_vector[plen]=0
9       break
10    ENDIF
11  END FOR//j
12  IF(j>K)
13    match_vector[plen]=1
14  ENDIF
15 END FOR//plen
16 IF(match_vector==0)
17   state=invalid
18   return(state)
19 ENDIF

Search in TCAM_RAM
20 FOR plen=1: W
21  IF (match_vector[plen]==1)
22   ram_addr=TCAM_plen_Search(Search_Key)
23   IF(ram_addr!=NULL)
24     sp=RAM_plen[ram_addr].s',
25     ep=RAM_plen[ram_addr].t',
26   ENDIF
27  IF(Search_Key[w-plen-1:0]<={1,ep} and
    Search_Key[w-plen-1:0]>={0,sp})
28   acs_infor=RAM_plen[ram_addr].acs_infor
29   state=valid
30  ELSE
31   state=invalid
32  ENDIF
33 END FOR//plen
34 return(acs_infor, state)

```

的比特串编码, N 为表项数目。DTRM, SRGE 和 C-TCAM 最坏情况下都会造成严重范围扩张, 以报文分类中的端口查询为例, DTRM 造成 $(2 \times 4 - 1)^2 = 49$ 倍扩张, SRGE 造成 $(2(16 - 2))^2 = 784$ 倍扩张, C-TCAM 造成 $(16 - 1)^2 = 225$ 或 $(16 - 2)^2 = 196$ 倍扩张。DTRM 和 C-TCAM 平均扩张因子优化到了较理想的水平, 但每条表项需要额外使用 $2W$ 个比特。范围扩张直接引发了高昂的更新代价, 增加或删除一个范围需要对多条表项进行操作, 同时加重了 TCAM 功耗。BF-TCAM 通过 SMLCP 算法实现了高效的范围匹配转化, 不增加任何冗余信息, 扩张因子达到 1.0, 使 TCAM 空间利用率达到 100%。SMLCP 算法将范围匹配拆分为两个步骤, 较其它算法增加了一个比较过程, 为不降低查找速度, 使用流水线缩小关键路径长度, 但不可避免地增加了电路资源消耗, 由此可见 SMLCP 的高效存储是以牺牲一定电路资源为代价的。TCAM 功耗与参与并行比较的表项数目成正相关, BF-TCAM 引入分类存储思想, 仅通过 Bloom filter 过滤的子集参与比较, 能够大大降低 TCAM 功耗, 极端恶劣情况下所有子集同时参与比较, 此时功耗达到峰值。BF-TCAM 支持增量更新, 实现了表项的随机插入与删除, 较其它算法大幅提高了更新性能。查找操作一个时钟周期即可完成, 特殊情况下流水线停顿造成一个时钟周期延时, 但目前 TCAM 时钟频率已达 500 MHz 以上, 意味着查找操作可在 4 ns 内完成, 能够满足性能需求。

以多处理器片上系统^[12](Multi-Processor System-on-Chips, MPSoC)的存储保护为背景评估 BF_TCAM 的实际功耗: MPSoC 上可并行运行多个任务, 每个任务分配有独享的内存区域, 对共享数据存储区有特定的访问权限, 引入 BF-TCAM 监控任务的访存地址可以实现对存储资源的访问控制, 进而保护系统不受非法入侵。实验中并行执行 4 个应用, 共有 551 条规则表项, 通过监测 TCAM_RAM 的片选信号即匹配向量可以计算实时参与比较的表项数目, 进而评估实时功耗。如图 3 所示, 每个应用采集 2000 次访存操作, 统计每次访存参与比较的表项数目, 图中横轴表示访存次序, 纵轴表示实时参与比较的表项数目, 可以发现应用 1 平均每次访存有 181.4 条表项参与比较, 占总规则数目的 32.9%; 应用 2 为 232.9 条, 占 42.3%; 应用 3 为 213.9 条, 占 38.8%; 应用 4 为 253.5 条, 占 46.0%。BF-TCAM 使用 Bloom filter 过滤无关项参与比较, 将实际参与比较的规则数目大幅减少, 实验表明, BF-TCAM 较传统 TCAM 实际功耗降低 50% 以上。

表3 不同算法的关键性能指标对比

指标	DTRM ^[5]	SRGE ^[7]	C-TCAM ^[6]	BF-TCAM
最坏情况下扩张因子	$(2M-1)^F$	$(2(W-2))^F$	$(W-1)^F$ 或 $(W-2)^F$	1.00
TCAM 额外比特	$2W$	0	$2W$	0
平均扩张因子	1.21^F	3.2^F	1.72^F	1.00
最坏情况下功耗	$0.3N(2M-1)^F$	$N(2(W-2))^F$	$\rho N(2(W-1))^F$ 或 $\rho N(2(W-2))^F$, ($1/2 < \rho < 1$)	N
更新代价	$O(M^F)$	$O(W^F)$	$O(W^F)$	$O(1)$
查找性能	1 时钟周期	1 时钟周期	2 时钟周期	1~2 时钟周期

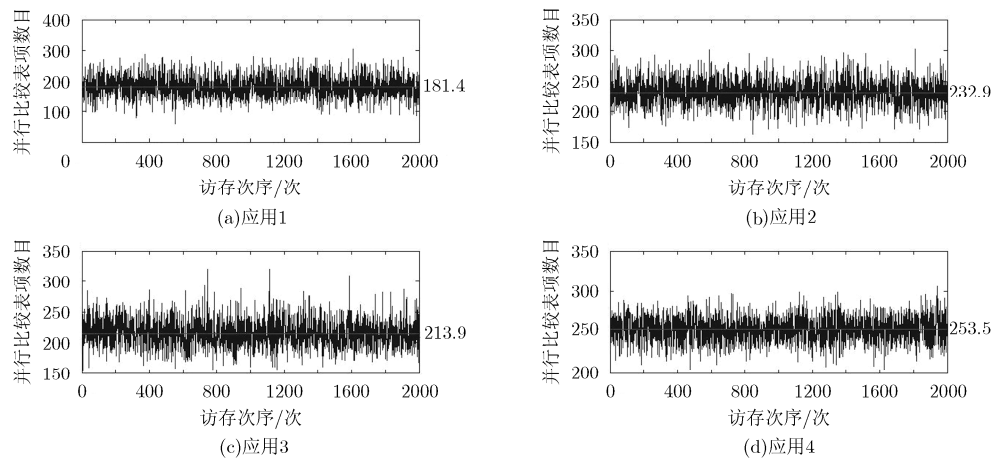


图 3 不同应用下 BF-TCAM 的实际功耗水平

5 总结

目前基于 TCAM 的范围匹配方案通过扩展规则集方式表达范围, 存储空间利用率低, 尤其对于报文分类中的端口匹配, 范围扩张因子达数百倍, 同时大幅提高了工作功耗和更新代价。针对这个问题, 本文提出了基于 BF-TCAM 的高效范围匹配方法, 利用范围区间两端点二进制编码最长共同前缀的唯一性设计了 SMLCP 算法, 将范围匹配过程分解为前缀匹配和特征区间比对两个步骤, 从而便于应用 TCAM 技术, 使 TCAM 空间利用率达到 100%。进而设计 BF-TCAM 模型, 按前缀长度对范围进行分类处理, 支持增量更新, 提高了更新性能。使用 Bloom filter 对关键字过滤, 屏蔽无关项参与比较, 从而大幅降低功耗。使用流水线技术缩短电路关键路径长度, 使查找操作在一个时钟周期内完成。BF-TCAM 模型在保持高速查找性能的同时, 使用 Bloom filter 过滤无关项, 大幅减少参与比较的规则数目, 实际功耗较传统 TCAM 降低 50% 以上。

参考文献

[1] 董永吉, 郭云飞, 黄万伟, 等. 一种新的高速报文解析结构研究[J]. 电子与信息学报, 2013, 35(5): 1083-1089.

DONG Yongji, GUO Yunfei, HUANG Wanwei, *et al.* A new high-speed packet parsing architecture[J]. *Journal of Electronics & Information Technology*, 2013, 35(5): 1083-1089.

[2] 李智涛, 徐雅静, 刘利宏, 等. 一种新的 IPv6 网络带宽测量方法[J]. 电子与信息学报, 2008, 30(9): 2283-2286.

LI Zhitao, XU Yajing, LIU Lihong, *et al.* An approach to available bandwidth measurement in IPv6 networks[J]. *Journal of Electronics & Information Technology*, 2008, 30(9): 2283-2286.

[3] Grammatikakis M D, Papadimitriou K, Petrakis P, *et al.* Security effectiveness and a hardware firewall for MPSoCs[C]. *IEEE High Performance Computing and Communications*, Paris, 2014: 1032-1039.

[4] Grammatikakis M, Papadimitriou K, Petrakis P, *et al.* Security in MPSoCs: a NoC firewall and an evaluation framework[J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2015, 34(8): 1344-1357.

[5] 田乐. 面向存储和功耗优化的 TCAM 报文分类算法研究[D]. [硕士学位论文], 解放军信息工程大学, 2013.

TIAN Le. Research on storage and power efficiency packet classification algorithm based on TCAM[D]. [Master dissertation], PLA Information Engineering University, 2013.

[6] 朱国胜, 余少华. 基于 TCAM 的范围匹配方法 — C-TCAM

- [J]. 通信学报, 2012, 33(1): 31-37.
- ZHU Guosheng and YU Shaohua. Range matching method based on TCAM: C-TCAM[J]. *Journal on Communications*, 2012, 33(1): 31-37.
- [7] BREMLERR-BARR A and HENDLER D. Space-efficient TCAM-based classification using gray coding[J]. *IEEE Transactions on Computers*, 2012, 61(1): 18-30.
- [8] RAY S S and BHATTACHARYA A. A fast range matching architecture with unit storage expansion ratio and high memory utilization using SBiCAM for packet classification[C]. IEEE India Conference, Pune, 2014: 1-6.
- [9] 侯颖, 郭云飞, 黄海, 等. 基于同源组合布鲁姆过滤器的早期流量抽样算法[J]. 通信学报, 2014, 35(10): 117-126.
- HOU Ying, GUO Yunfei, HUANG Hai, *et al.* Early traffic sampling algorithm based on SSCBF[J]. *Journal on Communications*, 2014, 35(10): 117-126.
- [10] 侯颖, 黄海, 兰巨龙, 等. 基于自适应超时计数布鲁姆过滤器的流量测量算法[J]. 电子与信息学报, 2015, 37(4): 887-894. doi: 10.11999/JEIT140820.
- HOU Ying, HUANG Hai, LAN Julong, *et al.* An adaptive timeout counter bloom filter algorithm for traffic measurement[J]. *Journal of Electronics & Information Technology*, 2015, 37(4): 887-894. doi: 10.11999/JEIT140820.
- [11] 张士庚, 刘光亮, 刘璇, 等. 大规模 RFID 系统中一种能量有效的丢失标签快速检测算法[J]. 计算机学报, 2014, 37(2): 434-444.
- ZHANG Shigeng, LIU Guangliang, LIU Xuan, *et al.* An energy-efficient and fast missing tag detection algorithm in large scale RFID systems[J]. *Chines Journal of Computers*, 2014, 37(2): 434-444.
- [12] 王一拙, 左琦, 计卫星, 等. 访存与用户行为敏感的 MPSoC 应用映射[J]. 电子学报, 2015, 43(4): 631-638.
- WANG Yizhuo, ZUO Qi, JI Weixing, *et al.* Memory-aware and user-aware mapping of applications to MPSoCS[J]. *Acta Electronica Sinica*, 2015, 43(4): 631-638.
- 戴紫彬: 男, 1966 年生, 博士, 教授, 研究方向为安全专用芯片设计.
- 刘航天: 男, 1991 年生, 硕士生, 研究方向为安全 SoC 设计与验证.