

Algorithmic Approaches to Optimizing TCAM-Based Packet Classification

Chad R. Meiners
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824, U.S.A.
meinersc@cse.msu.edu

1. INTRODUCTION

Packet classification is the core mechanism that enables many networking services such as packet filtering and traffic accounting. Using Ternary Content Addressable Memories (TCAMs) to perform high-speed packet classification has become the *de facto* standard in industry because TCAMs can facilitate constant time classification by comparing a packet with all rules of ternary encoding in parallel. Despite their high speed, TCAMs have limitations of small capacity, large power consumption, and relatively slow access times. The well-known range expansion problem in converting range rules to ternary rules significantly exacerbates these TCAM limitations. While we can expect some gain in TCAM performance from improved hardware, the demands on TCAM performance as measured by the number of rules in packet classifiers increase far more rapidly due to the explosive growth of Internet services and threats.

My thesis addresses TCAM space reduction along two directions: equivalent transformation and architectural redesign. Space reduction is key to addressing these three issues facing TCAMs because power consumption and access time are determined by the capacity of the TCAM.

2. EQUIVALENT TRANSFORMATION

Equivalent transformation converts a large packet classifier into a smaller but semantically equivalent packet classifier. These techniques have the advantage that they can be immediately deployed to existing TCAM based packet classification hardware. Finding the minimal TCAM representation for a packet classifier is NP-hard, and the natural restriction of finding the minimal prefix packet classifier is suspected to be NP-hard for packet classifiers with more than one dimension. In this thesis, I propose three algorithmic approaches to address equivalent transformation: TCAM Razor, Bit weaving, and All-match redundancy removal.

2.1 TCAM Razor

TCAM Razor [2] is a algorithmic solution using three techniques: decision diagrams, dynamic programming, and redundancy removal. It first decomposes a multi-dimensional packet classifier into multiple one-dimensional prefix classifiers using a decision diagram. Second, each one-dimensional classifier is optimally solved. Third, TCAM Razor reconstructs a multi-dimensional classifier from the optimal one-dimensional solutions using the structure from the decision diagram. This approach is highly effective because it harnesses the power of the one-dimensional optimizers, which greatly mitigate the effects of range expansion. Finally redundancy removal removes the excess rules that were created in the third step.

TCAM Razor on average reduces packet classifiers to 24.5% of their original size for real life classifiers.

2.2 Bit weaving

Bit weaving [3] is the first TCAM-based classifier compression scheme that is not limited to producing prefix classifiers. The basic observation of bit weaving is that TCAM entries that have the same decision and have a Hamming distance of one (*i.e.*, differ by only one bit) can be merged into a single entry by replacing the bit in question with *.

Bit weaving applies two new techniques, *bit swapping* and *bit merging*, to first identify and then merge such rules together. Bit swapping first cuts a rule list into a series of partitions. Within each partition, a single permutation is applied to each rule's predicate to produce a reordered rule predicate, which forms a single prefix where all *'s are at the end of the rule predicate. This single prefix format allows the use of existing dynamic programming techniques [2] to find a minimal prefix table for each partition in polynomial time. Bit merging then searches each partition and merges together rules that differ by a single bit. Once both techniques are finished, all ternary strings are reverted back to their original bit permutation to produce the final TCAM table.

On real life classifiers Bit weaving on average reduces packet classifiers to 23.6% of their original size. When Bit weaving and TCAM Razor are combined the packet classifier are on average reduced to 21.3% of their original size.

2.3 All-match Redundancy Removal

A rule in a packet classifier is redundant if and only if it can be removed from the packet classifier without changing the packet classifier's semantic behavior. While identifying redundant rules in 1-dimensional packet classifiers is sim-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGMETRICS/Performance'09, June 15–19, 2009, Seattle, WA, USA.
Copyright 2009 ACM 978-1-60558-511-6/09/06 ...\$5.00.

ple, identifying redundant rules in multi-dimensional packet classifiers is much more difficult.

Redundancy removal is a critical component for equivalent transformation algorithms such as TCAM Razor and Bit weaving. Furthermore, redundancy detection and removal have benefits beyond minimizing TCAM entries. One exemplary use of redundancy detection is in analyzing packet classifiers for potential errors. For instance, when a rule is shadowed by rules above it, the rule becomes redundant; however, this is typically not the intent of the router or firewall administrator. Therefore, redundancy could be an indicator of errors in packet classifiers.

The all-match redundancy removal algorithm[1] is the first algorithm that attempts to solve first-match packet classifier problem from an all-match packet classifier perspective. This approach improves upon prior work in two ways. First, the redundancy theorem becomes simpler. The redundancy theorem in prior work distinguishes upward and downward redundant rules, and detects them separately. In contrast, the all-match redundancy theorem presented here gives a single criterion that can detect all redundant rules. Second, the new redundancy removal algorithm is more efficient.

3. ARCHITECTURAL REDESIGN

Architectural redesign modifies the TCAM based packet classifier architecture to improve efficiency. This thesis proposes two such approaches: topological transformation, and TCAM SPLiT.

3.1 Topological Transformation

One approach in prior work for mitigating the effects of range expansion has been to reencode critical ranges. The basic idea is to reencode a given packet and use the re-encoded packet as the TCAM search key. However, previous reencoding schemes suffer from one fundamental limitation: they all ignore the decision associated with each rule and thus the classifier's decision for each packet. Disregarding classifier semantics leads all previous techniques to miss significant opportunities for space compression.

Fundamentally different from prior work, topological transformation [5] treats reencoding as a transformation from one colored hyperrectangle to another where the color is the decision associated with a given packet. Furthermore, this type of reencoding with small TCAM tables enables fast packet reencoding. Topological transformation uses two orthogonal, yet composable and complementary, reencoding approaches, domain compression and prefix alignment. In domain compression a given colored hyperrectangle that represents the semantics of a given classifier is transformed into the smallest possible "equivalent" colored hyperrectangle. This leads to each field requiring less bits as well as compressing the number of rules. Prefix alignment, on the other hand, strives for rule number compression only by transforming a colored hyperrectangle to an equivalent "prefix-friendly" colored hyperrectangle where the ranges align well with prefix boundaries, minimizing the costs of range expansion. Composing these two approaches is highly effective at minimizing the amount of TCAM space required to store a classifier.

Experiments on real life classifiers show that on average the real life classifiers can be reduced to 10.3% of their original size, and if the transforming tables are stored by alternative means, the classifiers can be reduced to 2.6% of their original size.

3.2 TCAM SPLiT

Equivalent transformation technique have a fundamental limitation that most classifiers have repetitive rules that differ by one or two fields. For example, the rules needed to block a port range for both TCP and UDP protocols cannot be combined into a single group of rules because TCP uses protocol number 6 and UDP uses protocol number 17. To overcome this fundamental limitation, TCAM SPLiT[4] abandons the conventional scheme of performing a single d -dimensional lookup on a single d -dimensional classifier in a TCAM. Instead, the classifier is split into $k \leq d$ smaller classifiers stored on a pipeline of k small and fast TCAM chips. TCAM SPLiT adapts the algorithms developed for TCAM Razor to generate compact classifier tables, and proposes table packing techniques to better utilize the TCAM bits provided for each entry.

Experiments on real life classifiers show that TCAM SPLiT on average reduces real life classifier to 8.0% of their original size. It is observed that $k = 2$ works very well for real life classifiers. As many currently deployed TCAM-based packet classification systems already use two TCAM chips, adopting TCAM SPLiT with a two-stage pipeline requires only a minor architectural adjustment.

4. REFERENCES

- [1] A. X. Liu, C. R. Meiners, and Y. Zhou. All-match based complete redundancy removal for packet classifiers in TCAMs. In *Proc. 27th Annual IEEE Conf. on Computer Communications (Infocom)*, April 2008.
- [2] C. R. Meiners, A. X. Liu, and E. Torng. TCAM Razor: A systematic approach towards minimizing packet classifiers in TCAMs. In *Proc. 15th IEEE Conf. on Network Protocols (ICNP)*, pages 266–275, October 2007.
- [3] C. R. Meiners, A. X. Liu, and E. Torng. Bit weaving: A non-prefix approach to compressing packet classifiers in TCAMs. Technical Report MSU-CSE-09-1, Department of Computer Science and Engineering, Michigan State University, January 2009. www.cse.msu.edu/~meinersc/TcamBitWeaving.ps
- [4] C. R. Meiners, A. X. Liu, and E. Torng. TCAM SPLiT: Optimizing space, power, and throughput for TCAM-based packet classification systems. Technical report, Department of Computer Science and Engineering, Michigan State University, 2009. www.cse.msu.edu/~meinersc/-TCAMSPLiT_TechReport.ps
- [5] C. R. Meiners, A. X. Liu, and E. Torng. Topological transformation approaches to optimizing tcam-based packet processing systems. In *Proc. ACM SIGMETRICS*, August 2009.