Topological Transformation Approaches to Optimizing TCAM-Based Packet Processing Systems

[Extended Abstract]

Chad R. Meiners Alex X. Liu Eric Torng
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824, U.S.A.
{meinersc, alexliu, torng}@cse.msu.edu

Categories and Subject Descriptors

C.2.6 [**Internetworking**]: Routers; C.2.1 [**Network Architecture and Design**]: Network communications; Packet switching networks

General Terms

Algorithms, Performance, Security

Keywords

Packet Classification, TCAM, Range Expansion

1. INTRODUCTION

Ternary Content Addressable Memories (TCAMs) have become the de facto standard in industry for fast packet classification. Unfortunately, TCAMs do have limitations of small capacity, large power consumption, and high heat generation. These limitations are exacerbated by range expansion where ranges in classifier rules must be converted to ternary format for deployment in a TCAM, often resulting in many more TCAM rules. Several range reencoding schemes [1,3] have been proposed to mitigate the costs of range expansion. Unfortunately, these approaches suffer from two fundamental limitations. First, they only consider range fields and ignore all other fields; thus, they miss many optimization opportunities that can be applied to prefix fields. Second, they require either computationally or economically expensive reencoding steps that do not easily integrate into existing packet classification systems. As each packet needs to be reencoded before it can be used as a search key, previous range reencoding schemes propose to perform packet reencoding using software, which greatly increases packet processing time, or customized hardware, which is expensive from a design, cost, and implementation perspective.

In this paper, we take two novel views on range reencoding that are fundamentally different from previous range reencoding schemes. First, we view range reencoding as a topological transformation process from one colored hyperrectangle to another. Whereas previous range reencoding schemes only deal with range fields, we [2] perform reencoding on every packet field. Specifically, we propose two orthogonal, yet composable, reencoding schemes: domain compression and prefix alignment. In domain compression, we transform a given colored hyperrectangle, which represents the semantics of a given classifier, to the smallest possible "equivalent" colored hyperrectangle. In prefix alignment, on the other hand, we strive to transform a colored hyperrectangle to an equivalent "prefix-friendly" colored hyperrectangle where the ranges align well with prefix boundaries, minimizing the costs of range expansion. Second, we view range reencoding as a classification process that can be implemented with small TCAM tables. Thus, while a preprocessing step is still required, it can be easily integrated into existing packet classification systems using the same underlying TCAM technology. Furthermore, since the domain of each table is reduced significantly, our methods have the added benefit that TCAM chips can be configured to use as little as 36 bits per table entry. Given that traditional storage methods utilize 144 bits per table entry, our methods allow significant compression even if we are not able to reduce the number of rules.

2. DOMAIN COMPRESSION

The fundamental observation is that in most packet classifiers, many coordinates (*i.e.*, values) within a field domain are equivalent. We define two values v_1 and v_2 in the domain of field F_i to be equivalent if for any packet with F_i value v_1, changing the F_1 value to v_2 would not change the decision of the packet. We define an *equivalence class* E on the domain of field F_i to be the set of values in the domain of F_i that are equivalent. The idea of domain compression is to combine all the values from an equivalence class in each field domain into a single value in the transformed field domain. This type of reduction not only leads to fewer rules, but also narrower rules, which results in smaller TCAM tables. From a geometric perspective, domain compression "squeezes" a colored hyperrectangle as much as possible. For example, consider the colored rectangle in Figure 1(A) that represents the classifier in Figure 1(H). In field F_1 represented by the X-axis, all values in $[0,7] \cup [66,99]$ are equivalent; that is, for any $y \in F_2$ and any $x_1, x_2 \in [0,7] \cup [66,99]$, packets (x_1, y) and (x_2, y) have the same decision. Therefore, when reencoding F_1, we can map all values in $[0,7] \cup [66,99]$ to a single value, say 0. By identifying the equivalence classes in each field, the rectangle in Figure 1(A) is reencoded to the one in Figure 1(D), whose corresponding classifier is shown in Figure 1(I). Figures 1(B) and (C) show the two transforming tables for F_1 and F_2, respectively; note that these tables can be implemented as TCAM tables. We use "*al*" as a shorthand for "*accept with logging*", "*a*" as a shorthand for "*accept*", and "*d*" as a shorthand for "*discard*".

SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.
ACM 978-1-60558-175-0/08/08.

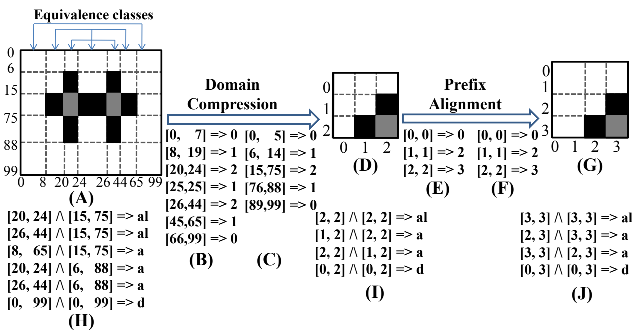

Figure 1: Example of topological transformations

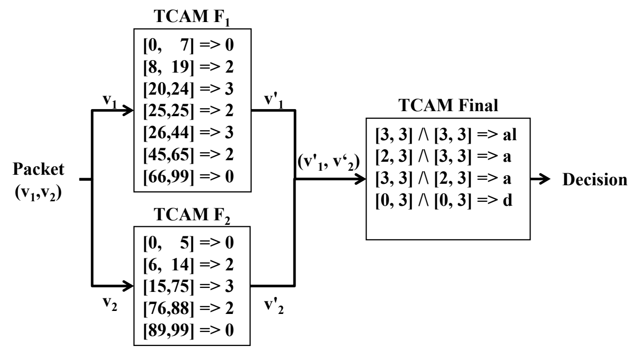

Figure 2: Pipeline architecture for reencoded lookups

A further benefit of domain compression is that it can reduce the total number of rules needed to represent the classifier. In Figure 1(H), there are no redundant rules; however, after domain compression, the rules $[26,44] \wedge [15,75] \Rightarrow al$ and $[26,44] \wedge [6,88] \Rightarrow al$ become redundant. Since the range $[26,44]$ maps to the same value as $[20,24]$, the rule $[20,24] \wedge [15,75] \Rightarrow al$ represents itself and $[26,44] \wedge [15,75] \Rightarrow al$, which makes $[26,44] \wedge [15,75] \Rightarrow al$ unnecessary. Likewise, $[20,24] \wedge [6,88] \Rightarrow al$ makes $[26,44] \wedge [6,88] \Rightarrow al$ redundant.

3. PREFIX ALIGNMENT

The basic idea of prefix alignment is to "shift", "shrink", or "stretch" ranges by transforming the domain of each field to a new "prefix-friendly" domain so that the majority of the reencoded ranges either are prefixes or can be expressed by a small number of prefixes. That is, we want to transform a colored hyperrectangle to another one where the ranges align well with prefix boundaries. This will reduce the costs of range expansion. For example, consider the packet classifier in Figure 1(I), whose corresponding rectangle is in Figure 1(D). Range expansion will yield 6 prefix rules because interval $[1,2]$ or $[01,10]$ cannot be combined into a prefix. However, by transforming the rectangle in Figure 1(D) to the one in Figure 1(G), the range expansion of the resulting classifier, as shown in Figure 1(J), will have 4 prefix rules because $[2,3]$ is expanded to $1*$. Figures 1(D) and (E) show the two transforming tables for F_1 and F_2, respectively.

We present an optimal dynamic programming solution for prefix alignment for 1-dimensional classifiers. We develop an effective hill-climbing solution for multi-dimensional classifiers using our 1-dimensional solution as a building block.

4. COMPOSITION AND ARCHITECTURE

We can compose domain compression and prefix alignment by simply composing the two transformation tables for each field, one from domain compression and one from prefix alignment. This provides us with the advantage that we can use a single TCAM lookup for each field to perform both transformation, which enables our methods to be supported by a 2-stage pipeline as shown in Figure 2. For example, the table in Figure 1(B) is composed with the table in Figure 1(E) to produce the table in TCAM F_1 in Figure 2, and the tables in Figure 1(C) and Figure 1(F) are composed to produce the table in TCAM F_2 Figure 2. We store the final classifier table, Figure 1(J), in TCAM Final in Figure 2.

5. EXPERIMENTAL RESULTS

We implemented our schemes and conducted experiments on both real-world and synthetic packet classifiers. Based on our results, our topological transformation scheme is roughly 15 times better at reducing the TCAM space needed to hold the reencoded packet classifier than the best result in prior work [1]. Even more striking, our scheme is roughly 5 times better than the best result in prior work, even when we include the TCAM space for transformers and they do not. Figure 3 shows the distribution of the expansion ratios for the set of real-life classifiers where the expansion ratio is the size of the reencoded classifier over the size of the original classifier.

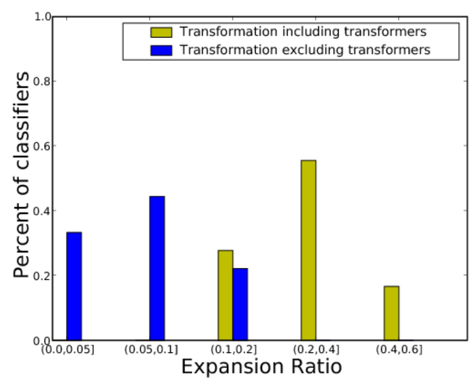

Figure 3: Distribution of expansion ratios of classifiers

6. REFERENCES

[1] A. Bremler-Barr and D. Hendler. Space-efficient TCAM-based classification using gray coding. In *Proc. 26th Infocom*, May 2007.

[2] C. R. Meiners, A. X. Liu, and E. Torng. Topological transformation approaches to optimizing tcam-based packet processing systems. Technical Report MSU-CSE-08-6, Computer Science and Engineering, Michigan State University, East Lansing, Michigan. www.cse.msu.edu/∼alexliu/publications/topo/topotech.pdf, May 2008.

[3] J. van Lunteren and T. Engbersen. Fast and scalable packet classification. *IEEE Journals on Selected Areas in Communications*, 21(4), 2003.