مجله علمی پژوهشی «علوم و فناوری‌های پدافند نوین»
سال ششم، شماره ۱، بهار ۱۳۹۴؛ ص ۷۱-۶۵

افزایش کارایی دسته‌بند درختی مبتنی بر TCAM به کمک برش‌های پویا در فضای ژئومتریک

شکور وکیلیان[۱]، مهدی عباسی[۲*]، علی فانیان[۳]
۱- دانشجوی کارشناسی ارشد، ۳- استادیار دانشکده مهندسی برق وکامپیوتر، دانشگاه صنعتی اصفهان
۲- استادیار، گروه مهندسی کامپیوتر، دانشکده مهندسی، دانشگاه بوعلی سینا، همدان
(دریافت: ۹۳/۱۲/۲۵، پذیرش: ۹۴/۰۱/۲۹)

چکیده

دسته‌بندی بسته‌ها یکی از پردازش‌های اساسی است که اغلب توسط پردازنده‌های شبکه اجرا می‌شود. در پیاده‌سازی سخت‌افزاری الگوریتم‌های دسته‌بندی بسته‌ها، از حافظه‌های تداعی‌گر چندوضعیتی (TCAM) برای جست‌وجوی موازی و افزایش سرعت پردازش استفاده می‌شود. در معماری دسته‌بند که عموماً از دو مرحله تشکیل شده است، نخست درخت‌های تصمیم‌گیری ایجاد شده و قوانین دسته‌بند بین برگ‌های آن توزیع می‌شوند. در مرحله دوم، متناظر با برگ‌های درخت، قوانین در بلوک‌های متفاوتی از TCAM جای می‌گیرند. در این مقاله با ارائه الگوریتمی پویا، بهترین بیت‌ها برای برش فضای ژئومتریک قوانین با هدف توزیع متوازن قانون‌ها و کاهش تکرار آن‌ها در درخت تصمیم انتخاب می‌شوند. کارایی معماری پیشنهادی به کمک برش پویا، با معماری‌های موجود مقایسه شده است. مقایسه نتایج نشان می‌دهد، روش پیشنهادی قوانین را در بلوک‌های TCAM، متوازن‌تر توزیع می‌نماید. در نتیجه، میزان حافظه و توان مصرفی مورد نیاز تا حد چشمگیری کاهش می‌یابد.

کلیدواژه‌ها: دسته‌بندی بسته‌ها، درخت تصمیم‌گیری، حافظه محتوایی آدرس پذیر چند وضعیتی، توان مصرفی.

Increasing the Efficiency of TCAM-Based Packet Classifiers Using Dynamic Cut Technique in Geometric Space

Sh. Vakilian, M. Abbasi[*], A. Fanian
Isfahan University of Technology
(Received: 16/03/2015, Accepted: 18/04/2015)

Abstract

Packet classification is one of the main processes that often run on network processors. In hardware implementation of packet classification algorithms, Ternary Content Addressable Memories (TCAMs) are used to implement parallel search and process packets rapidly. In classifier architecture, first, decision tree is created and classifier rules are distributed among its leaves. In the second stage, rules are included in different blocks of TCAM corresponding to leaf of the tree structure. In this study, a new dynamic algorithm is offered to select the best bits for cutting in representation of rules in geometric space to distribute them equally and reduce their duplication in the decision tree. Efficiency of the proposed architecture which uses dynamic cuts has been compared with recent architectures. Comparing results shows that the proposed method can distribute rules in TCAM block more balanced than recent architectures. Therefore, memory and power consumption requirements are reduced considerably.

Keywords: Packet Classification, Decision Tree, TCAM, Consumption Power.

[*]Corresponding Author E-mail: Abbasi@basu.ac.ir

۱. مقدمه

طیف وسیعی از ابزارهای پردازشگر بسته‌ها شامل مسیریاب‌ها، دیواره‌های آتش، سیستم‌های تشخیص نفوذ، سیستم‌های مدیریت حساب کاربران و سیستم‌های مدیریت شبکه از دسته‌بندی بسته‌ها استفاده می‌کنند. در دسته‌بندی بسته‌ها، نوع هر بسته بر اساس اطلاعات موجود که در پارامترهای مختلف سرآیند تولید شده بسته توسط فرستنده، توسط الگوریتم‌هایی که به صورت نرم‌افزاری و یا سخت‌افزاری پیاده‌سازی می‌شوند قابل تحلیل است. در این تحلیل، اطلاعات موجود در فیلدهای مشخص سرآیندها با قوانین اولویت‌دار انطباق داده می‌شوند تا قانونی که بیشترین تطبیق را داشته باشد انتخاب شود. در نهایت به بسته مذکور یک برچسب نوع یا جریان، متناظر با قانون انتخاب شده داده می‌شود.

برای محقق نمودن انعطاف‌پذیری لازم در جست‌وجو و انجام جست‌وجو روی بیت‌ها با مقدار بدون اهمیت[1]، طراحان از نوعی از حافظه به نام TCAM استفاده می‌نمایند [۱]. ذخیره فیلترها در TCAM، یک راه حل مبتنی بر سخت‌افزار برای دسته‌بندی بسته‌ها به حساب می‌آید که مورد علاقه محققین قرار گرفته است[۱ و ۲]. معماری TCAM، ذخیره‌سازی و جست‌وجوی منطق سه وضعیتی شامل صفر، یک و حالت بی اهمیت را در هر بیت از هر ردیف TCAM فراهم می‌نماید. فیلترها، بر اساس اولویت و به صورت نزولی در TCAM ذخیره می‌شوند. هر جست‌وجو در TCAM، یک پالس ساعت طول می‌کشد. وقتی بسته‌ای به عنوان ورودی به دسته‌بند می‌رسد، همه قانون‌ها در TCAM به صورت موازی با ورودی با توجه به بیت‌های ماسک مقایسه شده و از میان قانون‌های مطابقت داده شده، قانونی با بالاترین اولویت انتخاب می‌شود. نکته مهم در جست‌وجوی داده‌ها در TCAM آن است که وقتی تطابقی انجام می‌گیرد فقط بیت‌هایی که مقدار ۱ را در بیت‌های ماسک دارند بررسی می‌شوند. بنابراین بیت‌هایی که مقدار بیت ماسک آنها صفر باشند جست‌وجو نمی‌شوند. بنابراین، TCAM کارایی بالایی را در دسته‌بندی بسته‌ها ایجاد می‌کند.

انجام جست‌وجو به صورت موازی روی همه ورودی‌های TCAM مصرف توان بالایی را می‌طلبد [۱۰-۱]. با افزایش تعداد فیلترها توان مصرفی افزایش می‌یابد. روش‌هایی با هدف کاهش توان مصرفی مبتنی بر گروه‌بندی ورودی‌های TCAM پیشنهاد شده است [۳، ۴، ۶ و ۱۱]. با گروه‌بندی می‌توان ورودی‌های خاصی را در زمان جست‌وجو فعال یا غیر فعال نمود. بنابراین، می‌توان کارایی TCAM را با این گروه بندی بهبود داد[۴، ۶ و ۱۱]. برای کاهش توان، تولیدکنندگان TCAM روالی برای فعال کردن و جست‌وجوی قسمتی از TCAM پیاده‌سازی نموده‌اند. در این روال، ساختار داخلی تراشه‌های TCAM به بلوک‌های ثابتی تقسیم می‌شود؛ در نتیجه، تعداد متغیری از بلوک‌های TCAM در هر زمان می‌توانند فعال گردند. این روش می‌تواند کاهش توان مصرفی را هم تضمین نماید. با این وجود، ایده کاهش توان به کمک گروه‌بندی به طور کامل مورد استفاده قرار نگرفته است [۱، ۵ و ۷].

در این مقاله، برای تقلیل میزان هدر رفتن حافظه که منجر به کاهش توان مصرفی می‌شود در پیاده‌سازی الگوریتم‌های درختی در TCAM، الگوریتم جدیدی ارائه شده است. نکته کلیدی در الگوریتم ارائه شده آن است که با دیدی ژئومتریک بهترین بیت لازم برای برش فضای ژئومتریک متناظر با درخت در هر سطح را پیدا می‌کند. نتیجه این ایده، کاهش تکرار قوانین در گره‌های درخت است. از آنجا که هر یک از برگ‌های درخت، معادل با یک بلوک TCAM می‌باشد، متوازن بودن توزیع قوانین در برگ‌های درخت کمک شایانی به مصرف حافظه بهینه در هر بلوک TCAM خواهد نمود.

۱-۱. مروری بر کارهای انجام شده

همان‌طور که در بخش مقدمه به آن اشاره شد، عمده مشکل استفاده از TCAM در ارتباط با میزان توان مصرفی آن بود. از این رو تحقیقات زیادی برای کاهش توان مصرفی ارائه شده است. فرانسیک و همکاران به منظور کاهش توان مصرفی، روش ویژه‌ای را برای ساخت جدول مسیریابی مبتنی بر TCAM پیشنهاد کرده‌اند[۴]. در طرح پیشنهادی آن‌ها، دو طرح برای ساخت موتورهای کم توان گسیل ساز مبتنی بر TCAM [2] ارائه شده است که عبارتند از: معماری انتخاب بیت[3] و معماری مبتنی بر درخت دودویی[4]. ایده کلیدی در هر دو معماری، تقسیم‌بندی جدول مسیریابی به بخش‌های کوچک‌تر است به نحوی که در هر جست‌وجو تنها یک بخش بررسی می‌شود. از این ایده کای و همکاران ، برای کنترل مصرف توان نیز استفاده نموده‌اند [۸].

یادی و همکاران طرحی به نام SmartPC به منظور کاهش توان مصرفی معرفی نمودند[۶]. این روش، از طریق فعال کردن بلوک‌های انتخابی TCAM با دستورالعمل‌های پیش دسته‌بندی کننده، توان مصرفی را کاهش می‌دهد. محدودیت این روش، گروه‌بندی قانون‌ها به بلوک‌های مجزا می‌باشد. برای رفع این مشکل روشی توسط ژائو روان و همکاران پیشنهاد شد[۳]. در این روش از درخت تصمیم‌گیری برای دسته‌بندی قانون‌ها استفاده می‌شود و بلوک‌های TCAM به برگ‌های درخت تصمیم‌گیری نگاشت می‌شود. این روش، ترکیبی از روش دسته‌بندی مبتنی بر نرم‌افزار (با استفاده از درخت تصمیم‌گیری) و دسته‌بندی مبتنی بر سخت‌افزار (با استفاده از TCAM) می‌باشد. نتایج تحلیلی در [۳] نشان دهنده آن است که روش مذکور در کاهش تعداد بلوک‌های مورد نیاز TCAM و همچنین کاهش توان مصرفی می‌تواند خوب عمل نماید. شکل (۱) معماری دو مرحله‌ای یک دسته‌بندی‌کننده بسته‌ها را که در [۳] پیشنهاد شده است، نشان می‌دهد. هنگام ورود بسته به دسته‌بند، قسمتی از فیلدهای سرآیند

[1] Don’t Care

[2] Low-Power TCAM Based Forwarding Engines
[3] Bit Selection Architecture
[4] Tri-Based Architecture

بسته جهت جست‌وجو در Index TCAM استفاده می‌شوند. در این شکل، آدرس IP مبدأ در کنار آدرس IP مقصد (یعنی ۶۴ بیت) جهت جست‌وجو در Index TCAM مورد استفاده قرار می‌گیرد. دلیل استفاده از این دو فیلد در جست‌وجوی اولیه آن است که تفاوت بین قانون‌های این دو فیلد بیشتر از سایر فیلدها است. به عبارت دیگر این دو فیلد تمایز بیشتری بین بسته‌های ورودی به دسته‌بند ایجاد می‌نمایند. سپس، از شناسه حاصل از تطابق در Index TCAM به عنوان ورودی برای فعال‌کردن بلوک‌های TCAM استفاده می‌شود. لازم به یادآوری است که در شکل (۱) هر بلوک TCAM در مرحله دوم دسته‌بندی، متناظر با یک برگ از درخت تصمیم است.

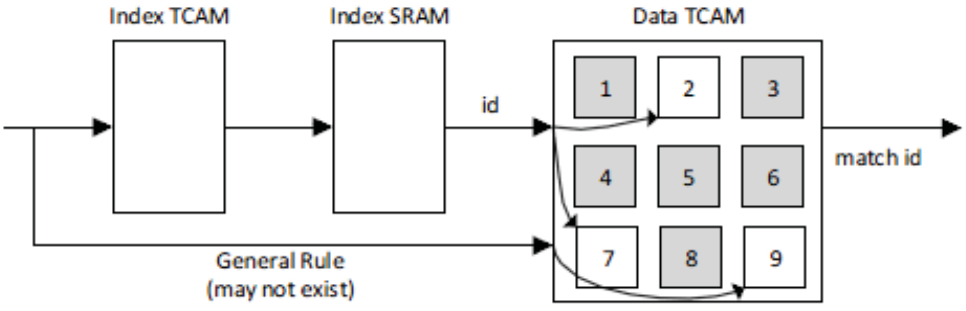
در این طرح، متناظر با بسته‌هایی که در مرحله اول دسته‌بندی نمی‌شوند بلوک‌های TCAM به صورت عمومی در نظر گرفته شده است. این بلوک‌ها، هم‌زمان با بلوک‌های خاص متناظر با برگ‌های درخت تصمیم، فعال می‌شوند. در نهایت، بهترین انتخاب از چندین بلوک فعال شده از طریق یک رمزگذار به خروجی دسته‌بند منتقل می‌گردد. مشکلات کلیدی این طرح آن است که در این روش، توزیع قانون‌ها در بلوک‌ها، به صورت نامتوازن انجام می‌شود. در نتیجه، گاهی اوقات ممکن است برخی از بلوک‌های TCAM خالی باشند. خالی ماندن بلوک‌های پیش‌بینی شده، با توجه به هزینه بالا برای هر بلوک TCAM، مناسب نبودن طرح پیشنهادی در [۳] را نشان می‌دهد. علاوه بر این، در بعضی بلوک‌ها ممکن است تعداد قانون‌های نگاشت شده، بیشتر از ظرفیت بلوک TCAM باشد. تنها راه حل پیشنهاد شده برای رفع این مشکل آن است که قانون‌های مازاد با توجه به ظرفیت بلوک، در بلوک عمومی نوشته شوند. متأسفانه این ایده منجر به بزرگ‌تر شدن بلوک عمومی و مصرف فضای زیاد در آن می‌شود.

آقای هایو سانگ و همکاران پارامتری برای برش ارائه کرده‌اند [۱۲]. این پارامتر برای برش بهینه با ایده نرم‌افزاری برای ساخت درخت مورد استفاده قرار گرفته است. مقصود از استفاده از این معیار برآورده ساختن دو هدف است. هدف اول آن است که تکرار قانون ها در برگ‌های درخت تصمیم کاهش یابد و هدف دوم این است که بتوان توزیع قانون ها را در درخت متوازن نمود. یکی از نتایج این طرح، توازن درخت ساخته شده نسبت به سایر روش های ساخت درخت مانند [۱۵-۱۳] می‌باشد. توازن درخت در روش‌های نرم‌افزاری می‌تواند به کاهش دسترسی حافظه زمان جست‌وجو منجر شود و متوسط زمان جست‌وجو را نیز کاهش دهد. لازم به ذکر است که یکی از مزیت های کار انجام شده در [۱۲] راحت بودن پیاده سازی سخت‌افزاری می‌باشد.

در هنگام جست‌وجو در درخت، هدف عبور از گره‌های درخت، پیدا کردن یک برگ می‌باشد. گره‌های برگ در روش‌های سخت‌افزاری نسبت به روش‌های نرم‌افزاری شامل قانون‌های زیادی خواهند بود. در روش‌های سخت‌افزاری ارتفاع درخت چندان مهم نمی‌باشد. زیاد بودن قانون‌ها در برگ به نوبه خود موجب کاهش تکرار قوانین نسبت به روش‌های نرم‌افزاری می‌شود. همان‌طور که تکرار قانون‌ها در روش‌های نرم‌افزاری یک مشکل به حساب می‌آید، در روش‌های سخت‌افزاری نیز این مشکل جدی بوده و کاهش این تکرارها بسیار اهمیت دارد. دلیل آن است که در معماری پردازنده‌های شبکه‌ای ظرفیت TCAM نسبت به SRAM یک منبع قیمتی محسوب می‌شود. در روش پیشنهادی هدف ایجاد توازن در توزیع قانون‌ها در برگ‌های درخت می‌باشد. هدف اصلی این ایده کلیدی استفاده بهینه از ظرفیت TCAM می‌باشد. به عنوان یک نتیجه ضمنی، رویکرد پیشنهادی با کاهش تعداد قوانین عمومی منجر به کاهش مصرف توان در تراشه TCAM نیز می‌شود.

۲. روش پیشنهادی

در این مقاله راهکاری با توسعه ایده مطرح شده در[۱۲] برای یافتن بهترین برش قوانین در مدل فضای ژئومتریک ارائه شده است. راهکار پیشنهادی با استفاده از الگوریتم‌های درختی، توزیع قانون‌ها در بلوک‌های TCAM را متوازن می‌نماید. در ادامه به معرفی مدل فضای ژئومتریک و نحوه استفاده از آن برای نمایش قوانین دسته‌بند در الگوریتم‌های درختی پرداخته شده است. در این راستا، پس از توضیح مفهوم برش در این فضا معیار برش بهینه توضیح داده شده است.



شکل ۱. معماری سخت‌افزاری دسته‌بند بسته‌های IP [۳]

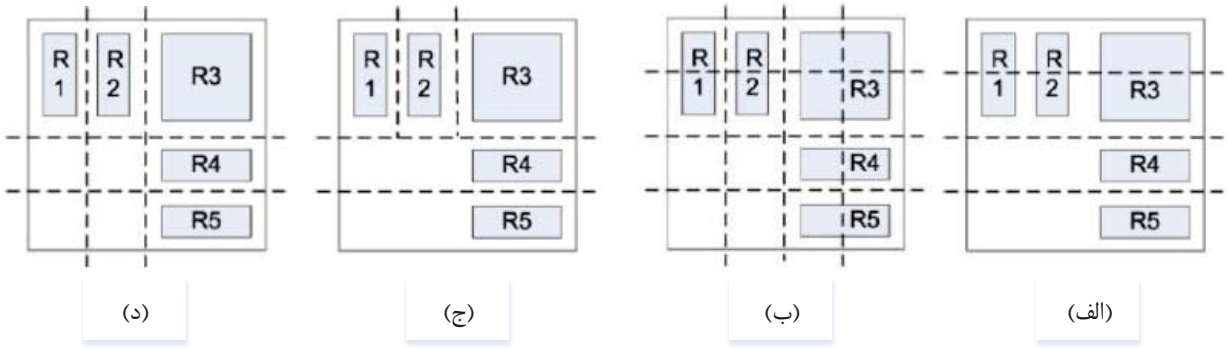
۲-۱. معیار پیشنهادی برش در مدل فضای ژئومتریک قوانین

مدل فضای ژئومتریک برای نمایش قوانین دسته‌بندی IP، از هندسه محاسباتی برگرفته شده است. برای آشنایی با این مدل، ابتدا مثالی که در شکل (۲) نمایش داده شده است، مورد بررسی قرار می‌گیرد. فرض کنید، پنج قانون دسته‌بندی IP، روی دو فیلد از سرآیند تعریف شده است. این قوانین با نام‌های R1 تا R5 در شکل مشخص شده‌اند. قوانین به صورت پیشوندی روی دو فیلد تعریف شده‌اند که این دو فیلد متناظر با آدرس IP مبدأ و آدرس IP مقصد می‌باشند. می‌توان یک فضای هندسی دوبعدی را در نظر گرفت و قوانین را به آن نگاشت نمود. در این فضای دوبعدی، محور افقی، گستره آدرس پیشوندی مبدأ و محور عمودی، گستره آدرس پیشوندی مقصد را بر اساس بیت‌های مشخص در قانون نشان می‌دهد. با این تعریف نگاشت، هر قانون به صورت یک مستطیل در فضای دوبعدی نمایش داده می‌شود. نکته مهم در نگاشت قوانین به این فضا آن است که ترتیب نگاشت قوانین به این فضا بر اساس اولویتشان در دسته‌بند است. بدین معنی که، ابتدا قوانین کم اولویت‌تر و سپس قوانین با اولویت بالاتر به این فضای هندسی نگاشت می‌شوند. در نتیجه، ممکن است بخش‌هایی از قوانین کم اولویت‌تر توسط قوانینی با اولویت بالاتر پوشش داده شوند. با در نظر گرفتن توضیحات فوق در شکل (۲) نگاشت پنج قانون R1 تا R5 به فضای هندسی نمایش داده شده است. واضح است که در این فضا، هر برش در موقعیت یک بیت خاص روی یک محور، باعث می‌شود قوانین به دو گروه تقسیم شوند. این تقسیم‌بندی دقیقا متناظر با جای دادن برخی قوانین درزیردرخت سمت چپ و برخی دیگر در زیر درخت سمت راست درخت دودویی می‌باشد.

همان‌طور که در بخش دوم به آن اشاره شد، در[۱۲] پارامتر برش در فضای ژئومتریک قوانین برای الگوریتم‌های درختی معرفی شده است. این پارامتر، می‌تواند تعادلی در توزیع قانون‌ها بین برگ‌های درخت ایجاد نماید. با استفاده از این معیار، دو هدف در طرح پیشنهادی ذکر شده برآورده می‌شود؛ هدف اول، کاهش تکرار قانون‌ها و هدف دوم، توازن در توزیع قانون‌ها در برگ‌های درخت است. برآورده شدن این دو هدف استفاده بهینه از حافظه‌های TCAM را تضمین می‌نماید. فرض کنید، مجموعه قوانین فعلی در فضای ژئومتریک در اثر تعداد مشخصی برش به k زیرمجموعه تقسیم شود. در هر زیرمجموعه در فضای ژئومتریک، تعداد $t_1,t_2,\ldots t_k$ قانون وجود دارد. اگر یک قانون در هنگام برش به دو بخش تقسیم شده باشد، در دو زیرمجموعه ظاهر می‌شود و در نتیجه در تعداد قوانین دو مقدار t_i و t_j ، $i\neq j$ تأثیر می‌گذارد. بنابراین مجموعه برش‌های بهینه، برش‌هایی هستند که قابلیت تقسیم مساوی مجموعه های برش خورده را دارا باشند. نگاهی به پارامتر ارائه شده در [۱۲] نشان می‌دهد، این معیار می‌تواند پارامتر مناسبی برای برش در نظر گرفته شود. این پارامتر با نام پارامتر pref در رابطه (۱) تعریف می‌شود.

$$pref = \sqrt{\sum_{i=1}^{k} {t_i}^2} \qquad (۱)$$

رابطه فوق، در هر گام برش هر بار محاسبه می‌شود. در رابطه (۱)، مقدار بهینه برش به ازای کمینه شدن این رابطه قابل دستیابی می‌باشد. مقدار کمینه زمانی ظاهر می‌شود که توزیع قوانین در زیرمجموعه‌های حاصل از برش به صورت متوازن باشد و همه زیرمجموعه‌های ایجاد شده در فضای ژئومتریک تعداد مساوی از قوانین را در خود جای داده باشند. برای مثال، در شکل (۲-الف)، مقدار pref برابر $\sqrt{3^2+3^2+1^2+1^2}=\sqrt{20}$ می‌باشد. به صورت مشابه در شکل (۲-ب) تا (۲-د) مقدار این پارامتر $\sqrt{12}$،$\sqrt{5}$ و $\sqrt{5}$ می‌باشد. واضح است که بهترین تصمیم‌گیری برای برش با کمینه بودن مقدارpref بدست می‌آید. در این مثال حالت کمینه به ازای برش‌های (ج) و (د) محقق می‌شود.



شکل ۲. مثالی از تاثیر برش‌ها برای محاسبه pref [۱۲]

۲-۲. معماری دسته‌بند پیشنهادی

در این مقاله معماری دسته بند شکل (۱) با استفاده از معیار prefاصلاح شده و به عنوان معماری پیشنهادی ارائه می‌شود. در نتیجه، ویژگی معماری پیشنهادی آن است که بلوک‌های TCAM در آن به صورت متوازن و بهینه استفاده می‌شود. در طرح نمایش داده شده در شکل (۱)، ابتدا آدرس IP مبدأ و مقصد باید در Index TCAM ذخیره شود. زمانی که بسته‌ای به عنوان

ورودی وارد دسته‌بند می‌گردد، فیلدهای آدرس IP مبدأ و مقصد از سرآیند این بسته، با درایه‌های Index TCAM مقایسه می‌شود. بررسی تطابق در این جست‌وجو بر اساس اولویت قانون‌ها انجام می‌گیرد. متناظر با بهترین تطابق یافت شده، آدرسی جهت دسترسی به یک خانه از حافظه RAM به دست می‌آید. در این خانه، با توجه به دسته‌بندی انجام شده در مرحله پیش‌پردازش، با استفاده از الگوریتم درختی، عددی نوشته شده است. این عدد، شماره بلوکی از بلوک‌های TCAM است. در بلوک TCAM مذکور، شماره پورت‌های مبدأ و مقصد قانون‌ها ذخیره شده است. در دسته‌بندی انجام شده در مرحله پیش پردازش توسط الگوریتم درختی، بیت‌هایی برای برش انتخاب می‌شوند. ویژگی قانون‌هایی که در یک بلوک TCAM قرار می‌گیرند آن است که این قانون‌ها در بیت‌های انتخابی توسط الگوریتم درختی با یکدیگر مشترک هستند. بنابراین جست‌وجو به دو مرحله شکسته می‌شود و در مرحله دوم بلوک‌های یاد شده برای مقایسه و یافتن بهترین قانون باید بررسی شوند.

در مرحله دوم جست‌وجو در بلوک‌ها، فیلدهای پورت مبدأ و مقصد از سرآیند بسته ورودی با درایه‌های بلوک TCAM انتخاب شده مقایسه می‌شوند و در صورت تطابق، قانونی با بالاترین تطابق به عنوان خروجی انتخاب مشخص می‌شود. در این معماری، هم‌زمان با بررسی بلوک انتخابی TCAM، بلوک دیگری نیز مورد بررسی قرار می‌گیرد. این بلوک، بلوک عمومی نامیده می‌شود. در بلوک عمومی، اطلاعات پورت‌های مبدأ و مقصد قانون‌هایی نوشته می‌شود که در هنگام برش در فضای ژئومتریک دو بعدی متناظر با آدرس‌هایIP مبدأ و مقصد، در زیربخش‌ها تکرار شده‌اند. مهم‌ترین علت مواجه با چنین قوانینی در هنگام برش، آن است که ممکن است بیتی برای برش انتخاب شود که در قانون مذکور مقدارش بی اهمیت (یا X) باشد. واضح است که در زیرمجموعه‌های ایجاد شده در اثر برش، این قانون در دو زیرمجموعه قرار می‌گیرد. درنتیجه، در هنگام ذخیره‌سازی قوانین در بلوک‌های TCAM، این قانون‌ها باید در دو بلوک TCAM نوشته شود. بنابراین، برای جلوگیری از این مشکل و استفاده بهینه از حافظه، این قانون‌ها در بلوک عمومی نوشته می‌شود.

۳. پیاده‌سازی و ارزیابی

در این بخش، ابتدا نحوه پیاده‌سازی الگوریتم پیشنهادی توضیح داده می‌شود. پس از معرفی زبان برنامه نویسی و ساختار کد، ابزار استفاده شده برای تولید بسته‌ها و مجموعه قوانین لازم برای دسته‌بند معرفی می‌شود. در نهایت، پس از معرفی پارامترهای ارزیابی، نتایج پیاده‌سازی الگوریتم پیشنهادی با معماری ژاهو روان مقایسه می‌شود.

۳-۱. پیاده‌سازی معماری کلاس‌بند دو مرحله ای

فیلدهای مورد جست‌وجو در پیاده‌سازی این الگوریتم شامل چهار فیلد آدرس IP مبدأ، آدرس IP مقصد، پورت مبدأ و پورت مقصد می‌باشد. در پیاده‌سازی این الگوریتم، افزایش تعداد برگ‌های درخت منجر به افزایش تعداد برش‌ها می‌شود. افزایش تعداد برش‌ها نیز خود سبب افزایش تعداد قانون‌های موجود در بلوک عمومی می‌شود. با افزایش تعداد قانون‌های بلوک عمومی، مصرف توان نیز زیاد می‌شود. به همین دلیل انجام برش‌های زیاد خیلی مناسب نخواهد بود. در این پیاده‌سازی، تعداد برگ های درخت حداکثر هشت عدد در نظر گرفته شده است. این هشت برگ به هشت بلوک TCAM نگاشت می‌گردد. یک بلوک TCAM اضافی نیز به منظور ذخیره قانون‌های عمومی در نظر گرفته شده است. برای پیاده‌سازی الگوریتم درختی، از سه آرایه به طول ۲۵۶ استفاده شده است. عدد ۲۵۶ تعداد حداکثری گره‌های درخت کامل با ارتفاع هفت می‌باشد. اولین آرایه، شماره بیت‌های برش را در گره‌های انتخاب شده برای برش مشخص می‌نماید. شماره بیت‌های برش از صفر تا ۳۱ می‌باشد. اگر مقدار خانه‌ای از این آرایه برابر ۳۲ باشد به این معنی است که آن گره برشی نداشته است. بنابراین، می‌توان نتیجه گرفت که آن گره یک برگ درخت است یا اینکه مورد استفاده قرار نگرفته است. برای انتخاب بیت مناسب برای برش، دو فیلد آدرس IP مبدأ و آدرس IP مقصد ابتدا باید بررسی شوند. برای این منظور از آرایه‌ای دیگر برای نشان دادن فیلد برش استفاده می‌گردد. آرایه دوم نیز همانند آرایه اول ۲۵۶ خانه دارد. اگر مقدار خانه‌ای از این آرایه برابر یک باشد به این معنی است که برش روی آدرس IP مقصد انجام گرفته است و اگر برابر صفر باشد بدان معنی است که برش بر روی آدرس IP مبدأ انجام گرفته است. وجود مقادیر دیگر در خانه‌های این آرایه نشان از عدم انتخاب آن گره برای برش می‌باشد. آرایه دیگری نیز به منظور نشان دادن برگ‌ها و همچنین تخصیص شماره بلوک به آن‌ها مورد استفاده قرار گرفته است. تعداد خانه‌های این آرایه نیز همانند دو آرایه دیگر ۲۵۶ عدد می‌باشد. آرایه چهارمی نیز با تعداد خانه‌هایی معادل با تعداد قانون-های مجموعه قوانین تعریف شده است. وظیفه این آرایه، ذخیره شماره بلوک ذخیره‌سازی قانون‌ها می‌باشد. هر قانون، متناسب با جایی که در مجموعه قوانین دارد خانه‌ای از این آرایه را در اختیار می‌گیرد. با انجام هر برش، قانونی که در بیت مورد برش مقدار یک را دارد به مجموعه قوانین فرزند راست گره جاری انتساب می‌یابد و شماره بلوک آن در این آرایه با توجه به این که در چندمین مرحله از برش قرار دارد تغییر خواهد کرد. اگر مقدار این بیت در زمان برش برابر با صفر باشد آن قانون به مجموعه قوانین فرزند سمت چپ گره جاری انتساب می‌یابد و در این آرایه شماره بلوک قانون مورد نظر تغییر نمی‌کند. اگر مقدار این بیت در قانونی برابر X باشد، آن قانون به بلوک عمومی منتقل شده و درنتیجه، شماره بلوک منتسب به آن هشت خواهد بود.

جهت پیاده‌سازی الگوریتم پیشنهادی و الگوریتم ژاهو روان از زبان VHDL، مبتنی بر استاندارد IEEE 1994، در محیط توسعه ISim نسخه 13.1 استفاده شده است.

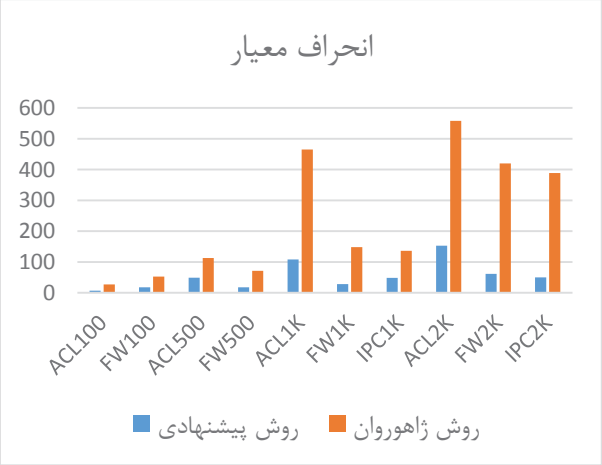
مجموعه قانون‌ها و بسته‌های آزمونی که برای ارزیابی و تست معماری پیشنهادی لازم است با استفاده از ابزار Classbench [۱۶] ایجاد شده است. با استفاده از این ابزار می‌توان مجموعه قوانینی تولید کرد که خاصیتی مشابه با مجموعه قوانین واقعی در دسته‌بندی بسته‌های IP دارا باشند. این ابزار از دو ماژول استفاده می‌نماید. ماژول اول برای تولید مجموعه قوانین با تعداد دلخواه مورد استفاده قرار می‌گیرد؛ ماژول دوم، مجموعه ای از بسته‌های تصادفی را با توجه به ویژگی‌های آماری قوانین تولید شده توسط ماژول اول، تولید می‌نماید.

سه نوع کلی از مجموعه قانون‌ها توسط ابزار Classbench تولید

می‌شود. این سه نوع عبارتند از: لیست کنترل دسترسی (ACL)، دیواره آتش (FW) و زنجیره IP (Ipc). هر یک از مجموعه قوانین با توجه به نوع آن‌ها و اندازه مجموعه تولید شده نامگذاری می‌شوند. به عنوان مثال، ACL1K به مجموعه قانون‌های دیواره آتش شامل، مشتمل برحدود ۱۰۰۰ قانون اشاره دارد. لازم به ذکر است در این پژوهش، برای ارزیابی این الگوریتم مجموعه قوانینی با اندازه‌های ۱۰۰، ۵۰۰، ۱۰۰۰ و ۲۰۰۰ با نوع‌های متفاوت ایجاد شده‌اند.

همان‌گونه که قبلاً گفته شد، در این شبیه‌سازی فرض شده است که هشت بلوک TCAM در مرحله دوم طرح استفاده می‌شود. جدول (۱) توزیع قوانین در بین بلوک‌ها را برای چند مجموعه قانون نشان می‌دهد. در این جدول برای دسته‌بندهایی با تعداد تقریبی ۱۰۰ قانون، محدوده هر بلوک مشتمل بر حداکثر ۱۳ قانون در نظر گرفته شده است. همچنین، برای دسته‌بندهایی با تعداد ۵۰۰، یک هزار (1K) و دوهزار (2K) قانون، به ترتیب محدوده هر بلوک مشتمل بر حداکثر ۶۵، ۱۳۰ و ۲۶۰ در نظر گرفته شده است. همان‌گونه که در جدول (۱) قابل مشاهده است، تعداد حداکثر قوانین مجاز در بلوک عمومی، برای بعضی از دسته‌بندها در طرح ژاهو روان کمتر از حداکثر تعداد مجاز در طرح پیشنهادی است. نتایج روش ژاهو روان نشان می‌دهد که در بلوک‌های ایجاد شده توزیع قانون‌ها به صورت نامتوازن می‌باشد. به عبارت دیگر، در بعضی از بلوک‌ها خیلی بیشتر از حداکثر ظرفیت مجاز، قانون انتساب داده شده است. همچنین، در روش ژاهو روان مانند روش پیشنهادی، قانون‌هایی که بیش از حد مجاز هر بلوک به آن انتساب داده شده‌اند، به بلوک عمومی منتقل می‌شوند. نگاهی به جدول (۱) نشان می‌دهد تعداد قانون‌های بلوک عمومی در روش ژاهو روان به اندازه قابل توجهی بیشتر از تعداد قانون‌ها در بلوک عمومی در طرح پیشنهادی این مقاله است. بنابراین، توزیع قانون‌ها بین بلوک‌ها در روش پیشنهادی بسیار متوازن‌تر از روش ژاهو روان است.

متوازن بودن توزیع قانون‌ها در برگ‌های درخت یا همان بلوک‌های TCAM هدف و معیار اصلی نشان‌دهنده کارایی الگوریتم پیشنهادی است. بنابراین، در ارزیابی کارایی الگوریتم پیشنهادی، از پارامتر انحراف معیار جهت نشان دادن چگونگی توزیع قانون‌ها در بلوک‌ها استفاده شده است. همان‌طور که قبلاً بحث شد، تعداد قانون‌های عمومی یک عامل کلیدی تأثیرگذار بر روی کارایی دسته‌بندی بسته‌های مبتنی بر TCAM دومرحله‌ای می‌باشد. اما، توزیع قانون‌ها بین بلوک‌های TCAM نیز یک فاکتور کلیدی برای استفاده بهینه از فضای TCAM در این طرح می‌باشد و باید مورد توجه قرار گیرد. همان‌گونه که قبلاً گفته شد، در این شبیه‌سازی فرض شده است که هشت بلوک TCAM در مرحله دوم طرح استفاده می‌شود. بدیهی است که کم بودن انحراف معیار تعداد قوانین توزیع شده در بلوک‌های هشتگانه TCAM در اثر اجرای یک الگوریتم، نشان دهنده توزیع متوازن‌تر قوانین توسط آن الگوریتم است و متوازن بودن توزیع قوانین در بلوک‌های هشتگانه TCAM مهم‌ترین شاخص برای کارایی معماری سخت‌افزاری دسته‌بند می‌باشد. نمودار ارائه شده در شکل (۳)، انحراف معیار توزیع مجموعه قوانین در بلوک‌های هشتگانه را برای روش پیشنهادی و روش ژاهو روان نشان می‌دهد. واضح است که در این نمودار، در همه حالات، مقدار انحراف معیار تعداد قوانین توزیع شده در بلوک‌های هشتگانه در اثر اعمال طرح پیشنهادی، به طرز قابل توجهی از انحراف معیار تعداد قوانین توزیع شده در بلوک‌های هشتگانه در اثر اعمال طرح ژاهو روان کمتر است. این مشاهده موید کارایی بالاتر روش پیشنهادی نسبت به روش ژاهو روان است.



شکل ۳. مقایسه انحراف معیار توزیع قوانین در بلوک‌های TCAM

جدول ۱. توزیع قانون‌ها در بلوک‌های TCAM

نوع مجموعه	شماره بلوک طرح پیشنهادی									شماره بلوک طرح ژاهوروان									
	۰	۱	۲	۳	۴	۵	۶	۷	ع	۰	۱	۲	۳	۴	۵	۶	۷	ع
ACL100	۱۱	۱۰	۱۱	۹	۱۵	۱۱	۱۰	۸	۱۵	۰	۱۰	۰	۲۳	۲۰	۵	۲۵	۵	۱۲
FW100	۸	۱۲	۱۳	۶	۱۲	۱۲	۹	۰	۲۳	۱۷	۲	۰	۰	۱۸	۰	۱	۱	۵۶
ACL500	۴۰	۶۹	۶۱	۸۷	۴۰	۵۰	۵۲	۲۹	۵۷	۶۱	۱۰	۵۸	۱۲۰	۵۳	۱۴	۱۰۴	۷	۵۸
FW500	۵۱	۵۰	۵۲	۵۰	۵۳	۵۰	۴۷	۴۵	۶۷	۵۸	۴۸	۵۲	۱	۹۷	۵۳	۵۲	۳۸	۶۶
ACL1K	۱۰۶	۱۰۲	۹۰	۱۱۱	۱۲۱	۱۱۰	۱۱۳	۱۳۱	۱	۰	۱۴۸	۱۴۷	۰	۸۸	۰	۵۰۲	۰	۰
FW1K	۱۰۴	۱۰۵	۱۰۵	۱۰۴	۱۰۵	۱۰۵	۱۰۴	۱۰۴	۱۳۴	۱۰۵	۱۰۵	۱۰۵	۲	۲۰۸	۱۰۵	۱۰۵	۱۰۳	۱۳۲
IPC1K	۱۳۷	۱۰۵	۱۲۳	۹۶	۱۲۲	۱۱۴	۱۰۷	۹۹	۷۹	۱۴۶	۱۸۳	۹۴	۷۲	۹۸	۱۵۸	۲۹	۱۲۴	۷۸
ACL2K	۲۱۵	۱۹۷	۲۰۸	۲۱۷	۲۴۰	۲۱۹	۲۲۷	۲۰۲	۵۸	۰	۰	۲۴۵	۲۴۳	۱۴۶	۵۰۲	۵۰۷	۱۲۱	۱۹
FW2K	۲۰۹	۲۰۹	۲۰۹	۲۰۸	۲۱۰	۲۱۰	۲۰۸	۲۰۹	۲۷۴	۲۰۹	۲۰۹	۳	۴۱۷	۱	۴۱۹	۲۱۰	۲۰۸	۲۷۰
IPC2K	۱۸۳	۱۸۱	۱۹۰	۱۸۱	۱۸۶	۱۹۲	۱۸۲	۱۸۰	۲۳۶	۳۴۶	۷۰	۱۱	۱۹۱	۰	۳۶۲	۲۴۹	۱۸۶	۲۹۶

۴. نتیجه‌گیری

در این مقاله روشی دو مرحله‌ای برای دسته‌بندی بسته‌های IP پیشنهاد شد. این روش با استفاده از انتخاب‌ها، بیت‌های مناسب برای برش در فضای ژئومتریک قوانین، سعی در توزیع متوازن قانون‌ها و گروه‌بندی آن‌ها در ساختار برگ‌های یک درخت تصمیم می‌نماید. در معماری پیشنهادی هر برگ درخت با یک بلوک TCAM متناظر است. همچنین یک بلوک اضافی به نام بلوک عمومی در نظر گرفته شده است که در بردارنده قوانین تکراری است. انتخاب برش پویایی که بتواند تعداد قوانین تکراری کمتری را در زمان برش بین بلوک‌های TCAM توزیع نماید به عنوان پردازش کلیدی در الگوریتم پیشنهادی لحاظ شده است. کاهش تعداد قوانین تکراری در بلوک عمومی با توجه به فعال شدن بلوک‌های عمومی در هر بار جست و جوی بسته‌ها، یکی از فاکتورهای کلیدی برای کاهش مصرف توان در این نوع معماری دسته‌بند می‌باشد.

بررسی نتایج شبیه‌سازی نشان داد که روش پیشنهادی موفق شده است نسبت به روش ژاهو روان، قوانین را به صورت متوازن‌تری بین بلوک‌های TCAM توزیع نماید. همچنین کمتر بودن ظرفیت بلوک عمومی به عنوان یک معیار دیگر برای مقایسه بهینگی روش توزیع قوانین استفاده شد. بررسی معیارهای بیان شده نشان داد که روش پیشنهادی برای برش در فضای ژئومتریک، باعث توزیع متوازن‌تر و کم تکرارتر قوانین در بلوک‌های TCAM نسبت به روش ژاهو روان شده است. بنابراین، می‌توان نتیجه گرفت که استفاده از الگوریتم برش پویا، در دسته‌بند سخت‌افزاری مبتنی بر TCAM، استفاده بهینه‌تر از فضای حافظه بلوک‌های TCAM و کاهش توان مصرفی را در معماری دسته بندIP تضمین می‌نماید.

۵. مراجع

[1] Taylor, D. E. “Survey And Taxonomy of Packet Classification Techniques”; ACM Comput. Surv. 2005, 37, 238-275.

[2] Yeim-Kuan, C.; Kai-Yang, L. “An Efficient TCAM Update Scheme for Packet Classification”; Advanced Information Networking and Applications (AINA), 2013 IEEE 27th Int. Conf. On 2013, 1017 - 1024.

[3] Zhao, R.; Xianfeng, L.; Wenjun, L. “An Energy-Efficient TCAM-Based Packet Classification with Decision-Tree Mapping”; TENCON 2013 - 2013 IEEE Region 10 Conf. (31194) 2013, 1- 5.

[4] Zane, F.; Narlikar, G.; Basu, A. “Coolcams: Power-Efficient Tcams For Forwarding Engines”; INFOCOM 2003. Twenty-Second Annual Joint Conf. of the IEEE Computer and Communications. IEEE Societies 2003, 42-52.

[5] Bremler-Barr, A.; Hay, D.; Hendler, D. “Layered Interval Codes For TCAM-Based Classification”; Computer Networks 2012, 56, 3023-3039.

[6] Ma, Y.; Banerjee, S. “A Smart Pre-Classifier To Reduce Power Consumption of Tcams For Multi-Dimensional Packet Classification”; SIGCOMM Comput. Commun. Rev. 2012, 42, 335-346.

[7] Bremler-Barr, A.; Hendler, D. “Space-Efficient TCAM-Based Classification Using Gray Coding”; Computers, IEEE Transactions on 2012, 61, 18-30.

[8] Kai, Z.; Chengchen, H.; Hongbin, L.; Bin, L. “An Ultra High Throughput And Power Efficient TCAM-Based IP Lookup Engine”; Twenty-Third Annualjoint Conf. of the IEEE Computer and Communications, INFOCOM 2004, 1984 - 1994.

[9] Panigrahy, R.; Samar, S. “Reducing TCAM Power Consumption and Increasing Throughput”; High Performance Interconnects, Proceedings. 10th Symposium on 2002, 107 - 112.

[10] Spitznagel, E.; Taylor, D.; Turner, J. “Packet Classification Using Extended Tcams”; Network Protocols, Proceedings. 11th IEEE Int. Conf. on 2003, 120 - 131.

[11] Vamanan, B.; Vijaykumar, T. N. “Treecam: Decoupling Updates and Lookups in Packet Classification”; Proceedings of the Seventh Conf. on Emerging Networking Experiments and Technologies 2011, 1-12.

[12] Haoyu, S.; Turner, J. S. “ABC: Adaptive Binary Cuttings for Multidimensional Packet Classification”; Networking, IEEE/ACM Transactions on 2013, 21, 98 - 109.

[13] Gupta, P.; Mckeown, N. “Classifying Packets with Hierarchical Intelligent Cuttings”; Micro, IEEE 2000, 20, 34 - 41

[14] Sumeet, S.; Florin, B.; George, V.; Jia, W. “Packet Classification Using Multidimensional Cutting”; Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications 2003, 213-224.

[15] Woo, T. Y. C. “A Modular Approach to Packet Classification: Algorithms and Results”; INFOCOM 2000. Nineteenth Annual Joint Conf. of the IEEE Computer and Communications Societies Proceedings. IEEE 2000, 1213 - 1222.

[16] Taylor, D. E.; Turner, J. S. “Classbench: A Packet Classification Benchmark”; INFOCOM 2005. 24th Annual Joint Conf. of the IEEE Computer and Communications Societies Proceedings IEEE 2005, 2068 - 2079.