Archive of SID

کاهش مصرف توان در سوئیچ‌های Openflow با روش تنظیم فرکانس

علی قیاسیان [۱]، استادیار

1- دانشکده فنی و مهندسی- دانشگاه شهرکرد-شهرکرد- ایران – ghiasian.ali@eng.sku.ac.ir

چکیده: با توجه به افزایش نقش صنایع مرتبط با فناوری اطلاعات و ارتباطات در تولید گازهای گلخانه‌ای، مدیریت مصرف توان تجهیزات سخت‌افزاری در بستر شبکه‌های انتقال داده بسیار موردتوجه قرارگرفته است. میزان مصرف توان تراشه‌های استفاده‌شده در این تجهیزات به‌طور مستقیم با فرکانس کار آن تراشه‌ها مرتبط است. ازاین‌رو یکی از راهکارهای کاهش مصرف توان این وسایل، تنظیم فرکانس کار آن‌ها بر اساس بار ترافیک شبکه می‌باشد. در این مقاله، چنین راهکاری به‌صورت یک مسئله بهینه‌سازی MILP در چارچوب معماری شبکه SDN و برای مجموعه‌ای متشکل از سوئیچ‌های Openflow که بر روی FPGA پیاده‌سازی شده‌اند، اعمال‌شده است. برای این منظور ابتدا با انجام آزمایش‌های میدانی نمودار مصرف توان یک سوئیچ برحسب نرخ ترافیک و فرکانس کار آن استخراج گردیده است. سپس با حل مسئله بهینه‌سازی ارائه‌شده بر روی یک شبکه زیرساخت با داده‌های واقعی استخراج‌شده از SNDlib و مقایسه نتایج حاصل با دو روش متداول دیگر در این حوزه، نشان داده‌شده است که روش پیشنهادی بیش از ۳۷٪ مصرف توان را کاهش می‌دهد.

واژه‌های کلیدی: توان، انرژی، سوئیچ، بهینه‌سازی، Openflow، SDN، FPGA، شبکه کامپیوتری، تنظیم فرکانس

Frequency scaling approach to reduce power consumption of Openflow switches

Ali Ghiasian[1], Assistant Professor

1- Faculty of Technology and Engineering, Shahrekord University, Shahrekord, Iran, Email: ghiasian.ali@eng.sku.ac.ir

Abstract: The significant impact of information and communication technology (ICT) on Greenhouse Gas emissions has caused a considerable research interest in energy management of network devices in recent years. As the amount of power consumption in routers and switches is a function of their working frequency, one approach to energy saving is to adapt working frequency of such devices dynamically and proportional to their traffic load. This paper describes this approach in Software Defined Networking (SDN) framework by proposing a Mixed Integer Linear Programming (MILP) formulation for a network of FPGA based interconnected Openflow switches. The prerequisite information of switch power profile is obtained by running experimental tests. The proposed method is applied to an infrastructural dataset extracted from the standard SNDlib database. Simulation results show that frequency scaling technique reduces power consumption by more than 37% in comparison to other well-known power-saving approaches.

Keywords: Power, energy, openflow, switch, router, software defined networking, FPGA, computer network, frequency scaling

تاریخ ارسال مقاله: ۱۳۹۶/۰۸/۲۴
تاریخ اصلاح مقاله: ۱۳۹۷/۰۳/۰۶
تاریخ پذیرش مقاله: ۱۳۹۷/۰۳/۱۹
نام نویسنده مسئول: علی قیاسیان
نشانی نویسنده مسئول: ایران - شهرکرد - دانشگاه شهرکرد- دانشکده مهندسی فنی و مهندسی

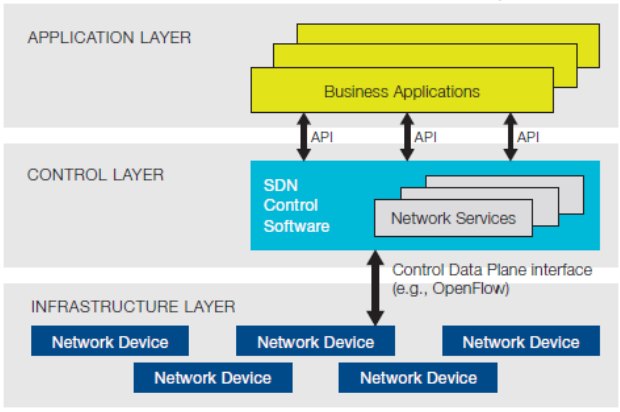
www.SID.ir

Archive of SID

۱- مقدمه

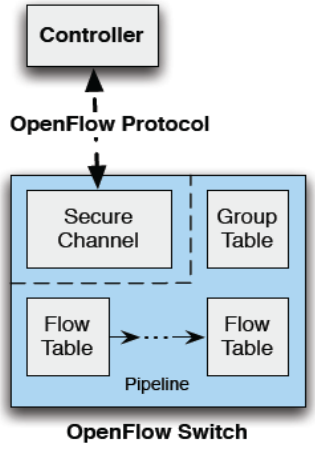
حفاظت از محیط‌زیست یکی از چالش‌های اصلی بشر در دنیای امروز است. دانشمندان و محققان در حوزه‌های مختلف علوم، توجه ویژه‌ای به این مسئله داشته‌اند. علوم مرتبط با فناوری اطلاعات و ارتباطات (ICT) نیز از این قاعده مستثنا نبوده‌اند. تحقیقات اخیر نشان داده است که نقش ICT در تولید گاز CO_2 و سایر گازهای آلاینده بسیار قابل‌توجه است [۳-۱]. در سال ۲۰۱۲ حدود ۴/۷ درصد از مصرف انرژی الکتریکی در جهان مربوط به صنایع ICT بوده است [۱]. این مقدار منجر به تولید ۵۳۰ مگا تن گاز CO_2 در اتمسفر شده که معادل ۱/۷ درصد از کل تولید گاز CO_2 در دنیا بوده است. نکته قابل‌توجه افزایش روزافزون استفاده از فنّاوری‌های مرتبط با ICT در جوامع بشری می‌باشد که الزام در توجه ویژه به این صنعت را برای حفظ محیط‌زیست یادآوری می‌سازد.

ساختار سنتی شبکه‌های انتقال داده اعم از چیدمان فیزیکی تجهیزات، نحوه آدرس‌دهی و مسیریابی بسته‌های داده مبتنی بر ساختار سلسله مراتبی و درختی می‌باشد. این ساختار توانایی پوشش دادن نیازهای رو به رشد سازمان‌های بزرگ، توسعه‌دهندگان شبکه زیرساخت و کاربران نهایی را نخواهند داشت. این نیازها اغلب ناشی از رشد روزافزون کاربران سیار، نیاز به پردازش ابری، مجازی‌سازی سرویس‌های شبکه، تغییر الگوی ترافیکی و التزام به کاهش مصرف انرژی بوده که طراحان شبکه را وادار به بازنگری در ساختارهای سنتی کرده است. بر همین اساس معماری جدیدی با مطرح کردن مفهوم شبکه‌های تعریف‌شده با نرم‌افزار (SDN) ارائه‌شده است که بتواند تا حدی نیازهای نسل جدید سامانه‌های مبتنی بر شبکه را تأمین نماید. گستره تحت پوشش SDN غالباً به سوئیچ‌های داده و تنظیم نحوه عملکرد آن‌ها محدود می‌شود. ایده اصلی در SDN جداسازی صفحه نرم‌افزاری بستر کنترلی از بستر سخت‌افزاری انتقال داده‌ها می‌باشد. در سوییچ‌های سنتی، بستر انتقال داده و بستر نرم‌افزار کنترل سوییچ به‌صورت متمرکز در یک سامانه پیاده‌سازی شده است. در معماری SDN که در شکل (۱) نشان داده‌شده است، یک واحد کنترل‌کننده متمرکز و جدای از بستر سخت‌افزار وظیفه تنظیم و پیکربندی سوئیچ‌ها را بر عهده دارد. نرم‌افزارهای کاربری در لایه بالایی کنترل‌کننده و بستر ارتباطی شبکه در لایه زیرین قرارگرفته‌اند به‌گونه‌ای که جزئیات لایه پایینی از دید نرم‌افزارهای لایه بالاتر مخفی می‌باشد [۴]. تجهیزات انتقال بسته در این معماری سوئیچ‌های Openflow هستند. بستر سخت‌افزاری برای پیاده‌سازی فانکشن سوئیچ، تراشه‌های منظور خاص ASIC[۱] یا تراشه‌های منظور عام FPGA هستند. در [۵] یک نمونه از پیاده‌سازی سوئیچ Openflow بر روی FPGA ارائه‌شده است. همان‌گونه که در شکل ۱ مشخص است مهم‌ترین پروتکل ارتباطی تعریف‌شده بین نرم‌افزار کنترل‌کننده و سوئیچ‌های زیرساخت، پروتکل متن‌باز Openflow می‌باشد.



شکل ۱: معماری SDN[۴]

ویژگی‌های سخت‌افزاری و نرم‌افزاری سوئیچ‌های مبتنی بر OpenFlow در [۶] آورده شده است. با توجه به شکل (۲) هر سوییچ OpenFlow، دارای یک یا چند جدول جریان و یک جدول گروه برای تعریف گروهی از جریان‌ها می‌باشد. از جدول‌های مذکور برای انجام عملیات مسیریابی و هدایت بسته‌ها استفاده می‌شود. همچنین هر سوئیچ OpenFlow دارای یک رابط امن برای ارتباط با کنترل‌کننده مرکزی تحت پروتکل Openflow می‌باشد. پروتکل OpenFlow با ارسال پیام‌های کنترلی به سوئیچ جدول‌های جریان را مقداردهی می‌کند.



شکل ۲: معماری نوعی سوئیچ OpenFlow [۵]

هرکدام از جدول‌های جریان، مجموعه‌ای از مدخل‌ها را شامل می‌شوند. هر مدخل، دارای یک فیلد تطبیق، یک یا تعدادی شمارنده و تعدادی دستورالعمل برای اعمال بر روی بسته‌های تطبیق یافته، می‌باشد. روند تطبیق بسته با مدخل جدول‌ها از اولین جدول جریان شروع می‌شود و برای جداول بعدی به ترتیب اولویت ادامه می‌یابد. در صورت پیدا شدن انطباقی بین بسته و یک مدخل، دستورالعمل‌های متناظر با آن مدخل اجرا می‌شوند. در صورت پیدا نشدن انطباق یا بسته حذف می‌شود و یا از طریق رابط کنترلی به کنترل‌کننده ارسال می‌گردد. معماری نوین SDN برای آن‌که بتواند جایگاه خود را در

www.SID.ir

Archive of SID

دنیای شبکه‌های کامپیوتری تثبیت کند باید بتواند پاسخ گوی چالش‌های جدید باشد. همان‌گونه که اشاره شد، مدیریت مصرف انرژی یکی از اساسی‌ترین مسائل مطرح در حوزه حفاظت از محیط‌زیست است.

بنابراین SDN باید قادر به پاسخ‌گویی به این نیاز باشد. در این مقاله، روشی برای کنترل مصرف انرژی سوئیچ‌های داده معرفی می‌شود که ساختار معماری SDN در پیاده‌سازی آن نقش کلیدی دارد. ایده اصلی آن است که فرکانس کار تراشه سوئیچ با بار ترافیک ورودی به آن منطبق شود. مطالعات انجام‌شده نشان می‌دهد که مصرف انرژی یک تراشه دیجیتال رابطه مستقیم با فرکانس پالس ساعت آن دارد. مثلاً توان مصرفی یک CPU چندهسته‌ای با مربع فرکانس کار آن متناسب است [۷]. لذا اگر بتوان بستر سخت‌افزاری که عملکرد سوئیچ شبکه بر روی آن پیاده‌سازی شده است را باقابلیت تنظیم فرکانس مجهز کرد، می‌توان مصرف توان سوئیچ را کنترل کرد. در این مقاله بستر سخت‌افزاری مورداستفاده تراشه FPGA است. ابتدا با اندازه‌گیری‌های دقیق رابطه بین توان مصرفی FPGA نسبت به فرکانس کار و بار ترافیکی آن به دست می‌آید. برای اعمال نتایج به‌دست‌آمده به شبکه‌ای از سوئیچ‌های متصل‌به‌هم لازم است از معماری SDN کمک گرفته شود. به عبارتی فرض می‌شود شبکه‌ای از سوئیچ‌های Openflow پیاده‌سازی شده در FPGA در اختیار است و تعداد مشخصی جریان داده با نرخ‌های تعریف‌شده از مبدأ تا مقصد معرفی‌شده‌اند. یک مسئله بهینه‌سازی از نوع MILP باهدف کمینه کردن کل انرژی مصرفی شبکه طراحی می‌شود. این مسئله در کنترلر SDN حل‌شده و حاصل آن مسیریابی جریان‌های داده از مبدأ به مقصد و تنظیم فرکانس کار سوئیچ‌های Openflow خواهد بود.

۲- مرور مقالات

تحقیقات اخیر نقش چشم‌گیر صنایع مرتبط با فناوری اطلاعات و ارتباطات (ICT) را در تولید گازهای گلخانه‌ای آشکارتر ساخته است. در سال ۲۰۱۲ حدود ۴/۷ درصد (معادل ۹۲۰ TWh) از کل مصرف انرژی برق در دنیا را صنعت ICT به خود اختصاص داده است [۸]. این امر باعث شده که ۱/۷ درصد از کل گاز CO_2 انتشاریافته به اتمسفر بر عهده این صنعت باشد. این در حالی است که ۳۷ درصد از تولید گاز CO_2 در دنیا به‌واسطه صنایع تولید انرژی برق حاصل می‌شود. این آمار اهمیت مدیریت مصرف انرژی الکتریکی برای کاهش حجم گازهای گلخانه‌ای را نشان می‌دهد. برای کاهش مصرف انرژی الکتریکی، مقاله [۹] مسئله مسیریابی جریان‌های داده را باهدف حداقل کردن تعداد هسته‌های پردازنده مورداستفاده به‌علاوه تعداد لینک‌های فعال شبکه پیشنهاد کرده است. نتایج شبیه‌سازی در مقاله حاضر نشان می‌دهد که روش پیشنهادی ما برای تنظیم فرکانس نتایج بهتری نسبت به روش کاهش تعداد لینک‌های مورداستفاده خواهد داشت. با بهره‌گیری از معماری SDN و پروتکل Openflow، در مقاله [۱۰] نویسندگان روشی برای تخصیص قوانین در جداول جریان معرفی کرده‌اند که با ارائه یک مدل ILP[۵] در زیرساخت شبکه و حل آن مصرف انرژی کاهش‌یافته است. در این روش فرض می‌شود که جداول جریان در حافظه‌های TCAM[۶] با حجم محدود پیاده‌سازی می‌شوند. محدودیت حجم حافظه به‌صورت یک قید در مسئله بهینه‌سازی ظاهرشده است.

در [۲] مروری بر مقالات مختلف انجام‌شده است که هرکدام در قالب یک مدل بهینه‌سازی به تحلیل مسئله کاهش انرژی مصرفی پرداخته‌اند. در صنایع زیرساخت شبکه و مخابرات، تجهیزاتی که بیشترین مصرف توان را به خود اختصاص می‌دهند سوئیچ‌ها و روترها می‌باشند. بررسی نمودارهای مصرف توان این تجهیزات نشان می‌دهد که توان مصرفی را می‌توان به دو مؤلفه اصلی به نام‌های مصرف توان ایستا و مصرف توان پویا تفکیک کرد [۱۱]. مصرف توان ایستا مربوط به حالت بدون بار تجهیز می‌شود و شامل توان مصرفی سامانه خنک‌کننده، المان‌های الکترونیکی، شاسی و غیره می‌باشد درحالی‌که مصرف توان پویا مرتبط و متناسب با بار ترافیک دستگاه خواهد بود. در ساعات اوج بار ترافیک، به دلیل نیاز به پردازش بیشتر و دسترسی بیشتر به حافظه‌ها و سایر المان‌ها مصرف توان پویا بیشتر می‌شود. ازآنجاکه مصرف توان ایستا بخش عمده‌ای از مصرف توان کل در تجهیزات سوئیچ و رورتر را به خود اختصاص می‌دهد [۲]، پروفایل مصرف این تجهیزات به‌صورت روشن-خاموش مدل می‌شود. به عبارتی یا تجهیز روشن و مصرف توان ثابتی دارد و یا خاموش با مصرف توان صفر در نظر گرفته می‌شود. تحت این مدل، تلاش زیادی برای تنظیم ساعات روشن بودن تجهیزات بر اساس مدل رفتاری ترافیک صورت گرفته است. این تلاش‌ها به معرفی اصطلاح LPI ختم شده است که هدف آن قرار دادن تجهیز در حالت خاموش یا بیکار است به شرطی که ترافیکی از آن عبور نکند [۱۲، ۱۳]. این امر نیاز به مسیریابی هوشمندانه ترافیک شبکه و اطلاع از رفتار آماری آن دارد. در سوئیچ‌های واقع در هسته شبکه، جایی که کارت‌های خط تجهیزات قابل جداسازی از شاسی هستند، هر کارت خط به‌طور مستقل می‌تواند خاموش یا روشن شود. در [۱۴] سه الگوریتم فرا اکتشافی برای تشخیص آن‌که کدام کارت خط باید خاموش شود معرفی‌شده است. ایده اصلی آن است که با حل یک مسئله برنامه‌ریزی خطی (LP) به‌صورت دوری سعی در خاموش کردن لینک‌های فعال شود با این قید که شبکه دچار ازدحام نگردد. نویسندگان مقاله [۱۵] مسئله خاموش کردن کارت‌های خط را با ارائه دو الگوریتم فرا اکتشافی باهدف پیدا کردن K کوتاه‌ترین مسیر برای هر جریان داده بررسی کرده‌اند. برخلاف مدل‌های مورداستفاده در [۱۵-۱۳] در مقاله حاضر علاوه بر امکان خاموش کردن تجهیز در شرایط بی‌باری، امکان تنظیم فرکانس کار برای بهینه کردن مصرف توان متناسب با بار ترافیک تجهیز در نظر گرفته می‌شود. این قابلیت به کاهش بیشتر مصرف توان کمک شایانی خواهد کرد.

www.SID.ir

Archive of SID

برای کاهش مصرف توان پویا، متداول‌ترین شیوه پیشنهادشده تنظیم فرکانس و ولتاژ کار CPU می‌باشد [۱۶]. میزان اثربخشی این روش که به نام DVFS شناخته می‌شود برای حالت‌های مختلف در [۱۷] بررسی و تأییدشده است. درحالی‌که ابتدا روش DVFS برای پردازنده‌های منظور عام یا همان CPU ها معرفی‌شده بود، ایده مشابهی با عنوان تنظیم نرخ (AR [۱]) برای روترها و سوئیچ‌ها در [۱۲] مطرح گردید. هدف اصلی درروش AR آن است که فرکانس هسته پردازشگر شبکه یا نرخ ارسال داده روی لینک‌ها بسته به بار ترافیک واردشده به تجهیز به‌صورت وقفی تنظیم شود. مقالات ارائه‌شده تاکنون اکثراً به جنبه تنظیم وقفی نرخ ارسال یا ALR [۱] توجه داشته‌اند [۱۲، ۱۸، ۱۹]. ایده انطباق نرخ ارسال داده برای لینک‌های اترنت برای نخستین بار در [۲۰] مطرح شد. نویسندگان آن مقاله پیشنهاد دادند که نرخ ارسال روی لینک‌های اترنت تابعی از طول صف بسته‌ها در سوئیچ‌ها باشد. این ایده در کمیته استانداردسازی اترنت برای اعمال نرخ‌های تطبیقی ۱۰۰ Mbps، ۱ Gbps و ۱۰ Gbps پیگیری شد. لیکن به دلیل آن‌که اعمال ALR با محدودیت‌های عملیاتی همراه بود در نسخه نهایی استاندارد IEEE 802.3az گنجانده نشد [۲۱]. علاوه بر زمان موردنیاز برای تغییر نرخ ارسال داده، درروش ALR لازم است هردو گره انتهای یک لینک در خصوص نرخ ارسال و دریافت با یکدیگر هماهنگ باشند. این محدودیت‌ها منجر به عدم مقیاس‌پذیری روش ALR می‌گردند.

برخلاف کارهای گذشته در [۱۲] و [۱۳]، در مقاله حاضر بین مفهوم ALR و آنچه ACR نامیده می‌شود تفکیک قائل شده است. در ALR نرخ ارسال داده برحسب بار ترافیک تجهیز انطباق می‌یافت درحالی‌که در ACR فرکانس هسته پردازشگر بر اساس بار ترافیک تنظیم می‌شود. در این مقاله تکنیک ACR که کاربرد آن در CPU ها قبلاً بررسی‌شده است، برای کنترل توان سوئیچ Openflow بر روی FPGA اعمال می‌گردد. یک دلیل مهم برای آن‌که روش ACR برای کاهش توان مصرفی سوئیچ‌ها و روترها کمتر موردتوجه قرارگرفته است عدم وجود قابلیت تنظیم فرکانس پالس ساعت در محصولات تجاری موجود می‌باشد. برای رفع این مشکل در مقاله حاضر از تراشه‌های قابل‌برنامه‌ریزی FPGA به‌عنوان بستر پیاده‌سازی سوئیچ داده استفاده می‌شود. به دلیل معماری قابل‌انعطاف و قابلیت‌های موازی‌سازی و اجرای خط لوله‌ای، تراشه‌های FPGA کاربرد وسیعی در صنایع ICT به خود اختصاص داده‌اند [۲۲، ۲۳]. در [۲۳] معماری سوئیچ حافظه مشترک با ظرفیت بالابر روی FPGA پیاده‌سازی شده است. در بسیاری از کاربردها، از FPGA به‌عنوان شتاب‌دهنده برای کمک به کاهش بار CPU کامپیوتر استفاده می‌شود. به‌عنوان‌مثال، با توجه به این‌که کاربردهای پردازش ویدئو و تصویر حجم محاسبات زیادی را در بردارند، در [۲۴] الگوریتمی برای آنالیز ویدئو با پیاده‌سازی بر روی Xilinx Virtes 5 FPGA ارائه‌شده است. در [۲۵] نیز یک راهکار مبتنی بر FPGA برای انجام پردازش‌های داده‌کاوی معرفی‌شده است. در مرجع [۲۶] رمز کننده سخت‌افزاری برای لینک‌های پرسرعت بر روی FPGA پیاده‌سازی شده است. در مقاله حاضر از FPGA به‌عنوان بستر سخت‌افزاری برای پیاده‌سازی سوئیچ Openflow استفاده می‌شود. تراشه مورداستفاده از خانواده Virtex 5 کمپانی Xilinx و مدل XC5VTX240TFFG1759 -2 با ۱۷۵۹ پایه می‌باشد. سایر جزئیات این تراشه در [۲۷] در دسترس می‌باشد.

نوآوری‌های این مقاله به شرح ذیل می‌باشند:

- اعمال روش ACR بر روی FPGA و استخراج نمودار مصرف توان آن برحسب فرکانس و بار ترافیک ورودی زمانی که طرح سوئیچ Openflow بر روی تراشه پیاده‌سازی شده است
- اعمال روش تنظیم فرکانس کار سوئیچ‌های Openflow در شبکه‌ای از سوئیچ‌های متصل‌به‌هم به‌صورت یک مسئله بهینه‌سازی MILP و حل آن به کمک ابزار CPLEX در نرم‌افزار GAMS
- شبیه‌سازی روش پیشنهادی بر روی توپولوژی و داده‌های واقعی شبکه و نمایش بهبود ایجادشده در مصرف توان در مقایسه با دو روش متداول دیگر

۳- استخراج نمودار مصرف توان FPGA

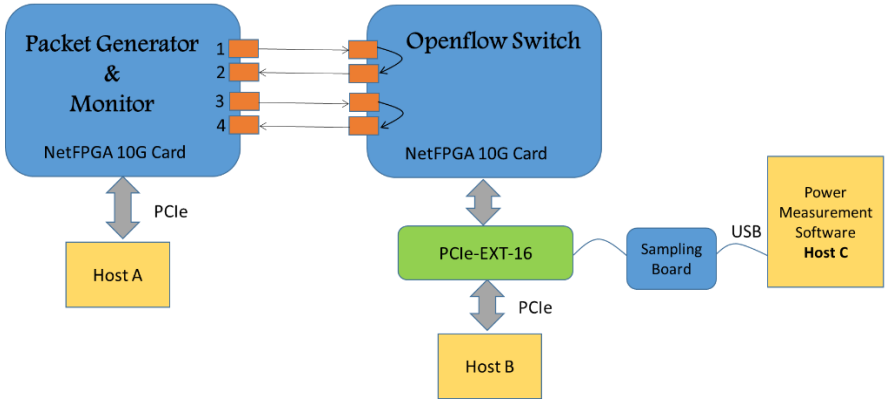
برای استخراج نمودار مصرف توان FPGA از بستر NetFPGA 10G [۲۴] که پروژه سوئیچ Openflow [۲۵] بر روی آن پیاده‌سازی شده است استفاده می‌گردد (شکل ۳). برای این منظور ماژولی برای تغییر فرکانس کار FPGA به‌صورت بلادرنگ به پروژه سوئیچ Openflow اضافه‌شده است. این پروژه یک سوئیچ متن‌باز ۴×۱۰ Gbps با ظرفیت سوئیچینگ ۴۰Gbps است [۲۸]. ازآنجاکه بورد الکترونیکی NetFPGA 10G یک کارت PCIe می‌باشد، برای اندازه‌گیری توان مصرفی آن از کارت توسعه درگاه PCIe تحت عنوان PCIeEXT-16 Smart [۲۹] استفاده شد.

شکل ۳: بورد NetFPGA 10G

برای اندازه‌گیری توان مصرفی بورد در شرایط مختلف بار ترافیکی لازم است یک مولد بسته با نرخ متغیر و طول بسته‌های متفاوت در اختیار باشد. برای این منظور یک ماژول مولد بسته طراحی و بر روی یک بورد NetFPGA 10G مجزا پیاده‌سازی شد. چهار پورت بورد مولد بسته به‌صورت یک‌به‌یک به چهار پورت بورد سوئیچ مطابق شکل (۴) متصل گردید.

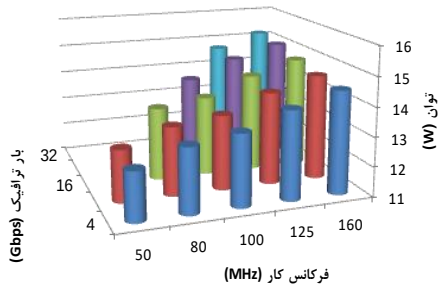
www.SID.ir

Archive of SID



شکل ۴: بلوک دیاگرام سناریوی اندازه‌گیری توان

ماژول مولد بسته قادر است بسته‌های با اندازه‌های مختلف تولید کند. بازه‌های زمانی بین هر دو بسته متوالی قابل تنظیم است. با تنظیم این دو پارامتر نرخ تولید بسته از ۱Gbps تا ۸Gbps برای هر پورت و درمجموع از ۴Gbps تا ۳۲ Gbps برای چهار پورت قابل‌تغییر است. ظرفیت گذردهی سوئیچ با تغییر فرکانس مرکزی FPGA تغییر می‌کند. حداکثر گذردهی با فرکانس ۱۶۰Mhz و با توجه به عرض باس داده ۲۵۶ بیت معادل ۴۰Gbps می‌باشد. با کاهش فرکانس گذردهی سوئیچ به‌صورت خطی کم می‌شود. برای افزایش دقت اندازه‌گیری، افزایش بار ترافیک تا جایی انجام می‌شود که حذف بسته اتفاق نیفتد. اندازه‌گیری‌ها بیش از ده مرتبه تکرار و میانگین نتایج به‌دست‌آمده در شکل (۵) نشان داده‌شده است.



شکل ۵: نتایج حاصل از اندازه‌گیری توان FPGA

در اندازه‌گیری‌های انجام‌شده مشخص گردید که توان ایستای FPGA معادل ۱۱/۸ وات می‌باشد. با درنظرگرفتن این نکته و درون‌یابی انجام‌شده روی داده‌ها رابطه (۱) برای توان مصرفی FPGA حاصل می‌گردد. نحوه حصول این رابطه این‌گونه است که ابتدا به ازای هرکدام از چهار فرکانس ۸۰، ۱۰۰، ۱۲۵ و ۱۶۰ مگاهرتز یک رابطه خطی بین بار و توان مصرفی به دست آمد. شیب هر چهار خط مقدار ۰/۰۲۴ است ولی عرض از مبدأ آن‌ها تفاوت دارد. به عبارتی با افزایش فرکانس فقط عرض از مبدأ افزایش می‌یابد. رابطه بین افزایش توان مصرفی با افزایش فرکانس با یک خط با رگرسیون ۰/۹۸ تقریب زده شد.

$$pwr = 0.017f + .024l + 11.8m \qquad (۱)$$

در رابطه (۱) f فرکانس کار FPGA برحسب مگاهرتز است که به‌صورت کسری از فرکانس حداکثر ظاهر می‌شود، l بار ترافیک برحسب گیگابیت بر ثانیه و m یک عدد باینری نشان‌گر روشن یا خاموش بودن بورد است. اگر بورد روشن باشد $m = 1$ و اگر خاموش باشد $m = 0$ در نظر گرفته می‌شود. هر سه پارامتر l، m و f در تنظیم توان مصرفی FPGA تأثیرگذارند. به عبارتی توان مصرفی FPGA برابر است با یک مقدار توان پایه ثابت ۱۱/۸ وات به‌علاوه توان مصرفی متغیری که تابعی از بار ترافیک ورودی و فرکانس کار FPGA می‌باشد. افزایش توان مصرفی با افزایش فرکانس کار رابطه خطی با شیب ۰/۰۱۷ watt/MHz دارد. همچنین نتایج آزمایش‌ها نشان می‌دهد که افزایش توان مصرفی FPGA برحسب بار ترافیکی ورودی رابطه خطی با شیب ۰/۰۲۴ watt/Gpbs دارد.

در ادامه با طرح یک مسئله بهینه‌سازی MILP نشان داده می‌شود که چگونه می‌توان با تنظیم این سه پارامتر، ضمن مسیریابی جریان‌های داده توان مصرفی کل شبکه را کاهش داد.

۴- کاهش توان در شبکه

در این قسمت یک مسئله بهینه‌سازی برای کاهش مصرف توان در شبکه‌ای از سوئیچ‌های Openflow متصل‌به‌هم ارائه می‌شود. توان مصرفی کل شبکه برابر است با مجموع مصرف توان گره‌های آن. جدول (۱) علائم اختصاری بکار رفته با توضیح هر یک را نشان می‌دهد.

شبکه موردنظر به‌صورت یک گراف جهت‌دار $G(\aleph, \mathcal{L})$ نمایش داده می‌شود که در آن $\aleph$ مجموعه رئوس یا گره‌های شبکه و $\mathcal{L}$ مجموعه یال‌ها یا لینک‌های شبکه هستند. بار ترافیک سوئیچ i همان‌طور که در معادله (۲) نشان داده‌شده برابر است با مجموع بارهای ترافیک ورودی و خروجی آن سوئیچ.

$$l_i = \sum_{sd\in\mathcal{F}} \{ \sum_{a\in N^+(i)} J_{i,a}^{sd} + \sum_{a\in N^-(i)} J_{a,i}^{sd} \} \qquad (۲)$$

پارامتر $J_{x,y}^{sd}$ برابر است با مقدار فلوی sd که از لینک (x,y) عبور می‌کند و از رابطه (۳) محاسبه می‌گردد.

www.SID.ir

Archive of SID

$$J_{x,y}^{sd} = b_{x,y}^{sd}.r^{sd} \quad (۳)$$

جدول ۱: علائم اختصاری در مسئله بهینه‌سازی

علامت	محدوده تغییرات	توضیحات
$\aleph$	$\|\aleph\| = N$	مجموعه گره‌های شبکه
$\mathcal{L}$	$\|\mathcal{L}\| = L$	مجموعه لینک‌های شبکه
I	$i \in \{1,2,\dots,N\}$	اندیس گره‌ها
pwr_i	$pwr_i \in [0, pwr_{max}]$	توان مصرفی گره i
f_i	$f_i \in [0, f_{max}]$	فرکانس کار گره i
α_i	$\alpha_i \in \{0.0625, 0.125, 0.2, 0.25, 0.34, 0.4, 0.5, 0.67, 1\}$	ضریب مقیاس فرکانس گره i
$\mathcal{F}$	$\mathcal{F}=\{\forall sd, s\in\mathcal{F} \; Src, d \in Des\}, \|\mathcal{F}\|=F$	مجموعه جریان‌های داده
^{6}Src	$Src \subseteq \aleph$	مجموعه گره‌های مبدأ
Des	$Des \subseteq \aleph$	مجموعه گره‌های مقصد
$N^+(i)$	$N^+(i) \subseteq \aleph$	مجموعه همسایه‌های خروجی گره i
$N^-(i)$	$N^-(i) \subseteq \aleph$	مجموعه همسایه‌های ورودی گره i
(x,y)	$(x,y) \in \mathcal{L}$	لینک بین گره x و گره y
r^{sd}	$r^{sd} \in [0, r^{max}]$	نرخ جریان sd
$J_{x,y}^{sd}$	$J_{x,y}^{sd} \in \{0, r^{sd}\}$	نرخ جریان sd روی لینک (x,y)
$b_{x,y}^{sd}$	$b_{x,y}^{sd} \in \{0,1\}$	شاخص باینری نشان‌گر عبور جریان sd روی لینک (x,y)
$C_{x,y}$	$C_{x,y} \in (0, C^{max}]$	ظرفیت لینک (x,y)
λ	$\lambda \in (0,1]$	ضریب ازدحام
μ	$\mu \in (0,1]$	ضریب پردازش
T_i	$T_i \in [0, T^{max}]$	گذردهی گره i
m_i	$m_i \in \{0,1\}$	شاخص باینری نشان‌گر روشن یا خاموش بودن گره i
$y_{x,y}$	$y_{x,y} \in \{0,1\}$	شاخص باینری نشان‌گر استفاده یا عدم استفاده از لینک (x,y)

توان مصرفی یک سوئیچ توسط رابطه (۱) مدل می‌شود. تعاریف سایر پارامترهای معادلات مذکور در جدول (۱) تجمیع شده است. میزان گذردهی و ظرفیت پردازشی یک سوئیچ تابعی از فرکانس کار آن سوئیچ است. با توجه به توضیحات ارائه‌شده در بخش (۳) رابطه بین گذردهی سوئیچ با فرکانس کار آن، یک رابطه خطی به‌صورت (۴) در نظر گرفته می‌شود.

$$T_i = kf_i \quad (۴)$$

تابع هدف در مسئله بهینه‌سازی کمینه کردن مجموع توان مصرفی در کل شبکه است. ازآنجایی‌که توان مصرفی تابعی از فرکانس کار و بار ترافیکی سوئیچ است، هدف کمینه کردن توان مصرفی سوئیچ‌ها با تنظیم فرکانس کار و بار هر سوئیچ است به‌نحوی‌که همه جریان‌های داده از مبدأ به مقصد مسیریابی شوند. لذا تابع هدف به‌صورت رابطه (۵) بیان می‌شود.

$$\min_{m_i,\alpha_i,b_{x,y}^{sd}} \sum_i Pwr_i \quad \forall i \in \aleph, \forall (x,y) \in \mathcal{L}, \forall sd \in \mathcal{F} \quad (۵)$$

که در آن m_i یک متغیر باینری است به‌نحوی‌که یک بودن آن به معنای روشن بودن سوئیچ i و صفر بودن آن به معنای عدم استفاده از سوئیچ i و لذا خاموش بودن آن است. برای دست یافتن به کارآمدترین نتیجه، اجازه داده می‌شود که سوئیچ‌های بدون بار خاموش شوند. همان‌گونه که در رابطه (۵) مشاهده می‌شود، این کار با تعریف m_i به‌عنوان یکی از متغیرهای مسئله بهینه‌سازی حاصل می‌گردد. $b_{x,y}^{sd}$ متغیر باینری مسیریابی است و یک بودن آن به معنای استفاده از لینک (x,y) برای عبور جریان sd خواهد بود. متغیر α_i نیز فرکانس کار سوئیچ را طبق رابطه (۶) مشخص می‌سازد:

$$f_i = \alpha_i f_{max} \; \forall \, i \in \aleph \quad (۶)$$

ضریب α_i مقادیر گسسته‌ای را با توجه به مشخصات بلوک DCM یا DLL[1] در FPGA اختیار می‌کند.

به‌عنوان‌مثال برای تراشه Virtex 5، α_i مقادیر خود را از مجموعه {۰/۰۶۲۵، ۰/۱۲۵، ۰/۲، ۰/۲۵، ۰/۳۴، ۰/۴، ۰/۵، ۰/۶۷، ۱} اختیار خواهد کرد [۲۷].

قیود مسئله بهینه‌سازی فوق به شرح ذیل می‌باشند:

۴-۱- قید تعادل جریان

قید تعادل جریان که در رابطه (۷) نوشته‌شده است اطمینان ایجاد می‌کند که مجموع جریان ورودی به یک گره و جریان تولیدشده در آن گره برابر است با مجموع جریان خروجی از آن گره. درنتیجه هر جریان حتماً از مبدأ به مقصد مسیریابی می‌شود.

$$\sum_{a\in N^-(i)} b_{a,i}^{sd} - \sum_{a\in N^+(i)} b_{i,a}^{sd} = \begin{cases} -1 & if \;\; i = s \\ +1 & if \;\; i = d \\ 0 & o.w. \end{cases} \quad (۷)$$

۴-۲- قید ظرفیت لینک

این قید برای دو منظور در نظر گرفته می‌شود. نخست آن‌که مجموع نرخ جریان‌های عبوری از یک لینک از ظرفیت لینک تجاوز نکند. دوم آن‌که برای جلوگیری از ازدحام، با معرفی ضریب λ کسری از ظرفیت نامی لینک برای عبور جریان‌ها در نظر گرفته می‌شود و قسمتی از ظرفیت به‌صورت ذخیره بدون استفاده باقی می‌ماند. ازدحام در عمل ممکن است در شرایطی که هجومی از بسته‌های جریان‌های مختلف وارد سوئیچ می‌شود به وجود آید. قید ظرفیت در معادله (۸) بیان‌شده است.

$$\sum_{sd\in\mathcal{F}} J_{x,y}^{sd} \le \lambda y_{x,y} C_{x,y} \quad \forall (x,y) \in \mathcal{L} \quad (۸)$$

۴-۳- قیود عدم استفاده از سوئیچ‌های خاموش

دو قید (۹) و (۱۰) الزام می‌کنند که هیچ جریانی اجازه عبور از سوئیچ‌های خاموش را ندارد. همچنین عدم استفاده از لینک‌های متصل به سوئیچ‌های خاموش توسط دو قید (۱۱) و (۱۲) تضمین می‌گردد.

$$b_{i,a}^{sd} \le m_i \; \forall \, i \in \aleph, \forall a \in N^+(i), \forall sd \in \mathcal{F} \quad (۹)$$

www.SID.ir

Archive of SID

$$b_{a,i}^{sd} \le m_i \quad \forall i \in \aleph, \forall a \in N^-(i), \forall sd \in \mathcal{F} \qquad (۱۰)$$

$$\forall i \in \aleph, \forall a \in N^+(i) y_{i,a} \le m_i \qquad (۱۱)$$

$$\forall i \in \aleph, \forall a \in N^-(i) y_{a,i} \le m_i \qquad (۱۲)$$

۴-۴- قید گذردهی گره

ظرفیت پردازش و گذردهی هر گره محدود است. این قید الزام می‌کند که جمع نرخ جریان‌های ورودی به گره و خروجی از گره باید کمتر از گذردهی آن گره باشد. برای جلوگیری از ایجاد ازدحام زمانی که هجومی از بسته‌ها به‌طور هم‌زمان وارد سوئیچ می‌شوند، ضریب $\boldsymbol{\mu}$ در نظر گرفته می‌شود.

$$l_i \le \mu T_i \; \forall i \in \aleph \qquad (۱۳)$$

لازم به ذکر است که مجموع دو رابطه (۴) و (۱۳) ارتباط بین فرکانس و نرخ ارسال داده را تعریف می‌کنند.

۵- حل مسئله بهینه‌سازی و تحلیل نتایج

برای حل مسئله MILP ارائه‌شده در بخش ۴، از نرم‌افزار GAMS و تحلیل‌گر CPLEX بر روی یک کامپیوتر با مشخصات CPU Intel Corei5 6200, 2.4Ghz, 8G RAM بهره گرفته شد. برای داده‌های شبیه‌سازی از کتابخانه SNDLib [۳۰] که یک پایگاه داده پر استناد شامل سناریوهای مختلف با داده‌های واقعی می‌باشد استفاده‌شده است. سناریوی انتخاب‌شده شبکه اروپایی nobel-eu است که مشخصات آن به همراه سایر پارامترها در جدول ۲ جمع‌آوری گردیده است. این شبکه ارتباط بین ۲۸ گره در کشورهای مختلف اروپایی را شکل می‌دهد. توپولوژی این شبکه در شکل (۶) نمایش داده‌شده است. مقدار k بر اساس مشخصات سخت‌افزار تعیین‌شده است. هرکدام از پورت‌های بورد NetFPGA دارای نرخ ۱۰Gbps و مجموع آن‌ها معادل ظرفیت سوئیچینگ ۴۰Gbps است. K نشان‌دهنده عرض بأس داخلی در معماری NetFPGA برای هر پورت و برحسب بایت است. با لحاظ کردن مقدار k برابر ۸ بایت (یا ۶۴ بیت) نرخ داده هر پورت معادل ۱۶۰Mhz*۶۴=۱۰/۲۴Gbps حاصل می‌گردد.

جدول ۲: مشخصات شبکه nobel-eu و پارامترهای شبیه‌سازی[۳۰]

تعداد گره‌ها	۲۸
تعداد لینک‌های جهت‌دار	۸۲
ظرفیت لینک نرمال شده به واحد	۳۰۰
تعداد جریان‌های داده	۳۷۸
جمع نرخ فلوها نرمال شده به واحد	۱۸۹۸
۰/۹	λ
۰/۹	μ
۱۶۰	f_{max}
۸	k

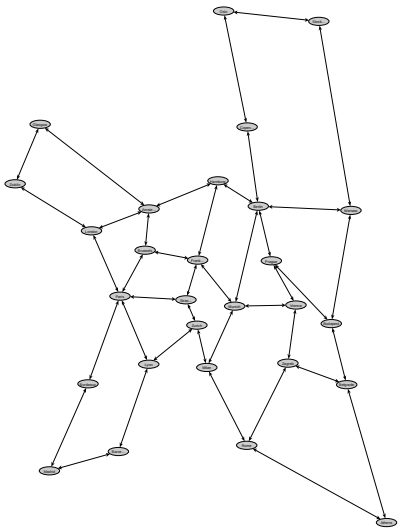
روش پیشنهادی ما که نام FreqScal به آن اطلاق می‌شود با دو روش دیگر به نام‌های MinNode [۱۱]و MinLink [۱۵،۱۴] مقایسه می‌شود. کنترل مصرف توان تجهیزات شبکه عمدتاً به یکی از دو روش خاموش کردن تجهیز بلااستفاده (Minnode) یا خاموش کردن کارت لینک بلااستفاده (Minlink) انجام می‌شود که روش پیشنهادی با هردوی آن‌ها مقایسه شده است. با توجه به اینکه روش پیشنهادی مبتنی بر استفاده از تراشه‌های FPGA و بوردهای مربوطه می‌باشد و اندازه‌گیری‌های توان‌ها نیز بر همین مبنا انجام‌شده است، عملاً امکان مقایسه عادلانه با روش‌های مشابه انجام‌شده در مقالات دیگر که از داده‌های مربوط به مصرف توان روترهای تجاری استفاده می‌کنند میسر نمی‌باشد. به‌عنوان‌مثال توان استاتیک بسیاری از روترهای تجاری از توان استاتیک بوردهای مورداستفاده ما بسیار بیشتر است[۳۱]. لذا مقایسه نتایج مصرف توان روش پیشنهادی ما با این مقالات که منجر به مصرف توان کمتر روش ما می‌شود نتیجه‌گیری صحیحی نخواهد بود. به همین دلیل روش‌های متداول دیگر نظیر Minnode و Minlink بر روی همان بستر FPGA اعمال‌شده است.

درروش MinNod تابع هدف کمینه کردن تعداد گره‌های روشن در شبکه می‌باشد. لذا در این روش رابطه (۵) با رابطه (۱۴) جایگزین می‌شود.

$$min \sum_i m_i \; \forall i \in \aleph \qquad (۱۴)$$

درروش MinLink تابع هدف کمینه کردن تعداد لینک‌های استفاده‌شده می‌باشد [۱۴، ۱۵]. به عبارتی در این شیوه رابطه (۵) با رابطه (۱۵) جایگزین می‌شود.

$$min \sum_{(x,y)} y_{x,y} \; \forall (x,y) \in \mathcal{L} \qquad (۱۵)$$



شکل ۶: توپولوژی شبکه nobel-eu

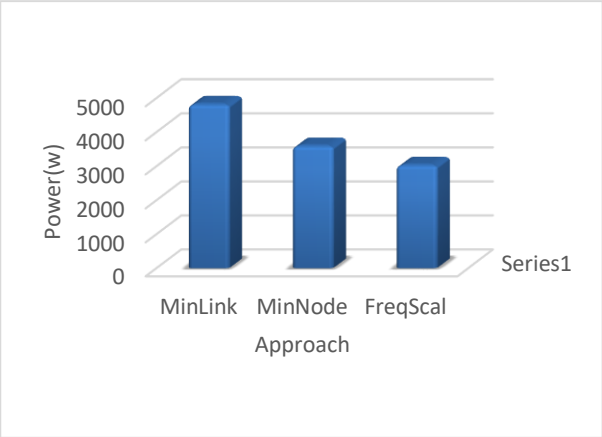
نتایج شبیه‌سازی نمایش داده‌شده در شکل (۷) از حاصل جمع توان مصرفی تک‌تک سوئیچ‌های شبکه nobel-eu به‌دست‌آمده است. توان مصرفی هر سوئیچ با مدل رابطه (۱) برحسب فرکانس کار و بار

www.SID.ir

Archive of SID

سوئیچ محاسبه‌شده است. فرکانس کار و بار هر سوئیچ نیز با حل مسئله بهینه‌سازی به‌دست‌آمده است. نتایج شکل (۷) نشان می‌دهد که روش FreqScal نسبت به دو روش MinNode و MinLink مصرف توان شبکه را بیشتر کاهش می‌دهند. بهبود حاصل‌شده نسبت به روش MinLink ۳۷٪ و نسبت به روش MinNode معادل ۱۶٪ می‌باشد. دلیل این امر آن است که فرکانس کار سوئیچ‌ها متناسب با بار ترافیک آن‌ها تنظیم می‌شود. این امر در شکل (۸) قابل‌مشاهده است. با توجه به اینکه توان مصرفی سوئیچ با فرکانس کار آن رابطه مستقیم دارد، تنظیم فرکانس به کاهش چشم‌گیر توان منجر می‌شود.

لازم به ذکر است که با توجه به رابطه (۶) فرکانس منتخب برابر است با حاصل‌ضرب ضریب α_i در فرکانس بیشینه که در شبیه‌سازی ۱۶۰ مگاهرتز درنظرگرفته شده است. درنتیجه فرکانس‌های گسسته یکی از مقادیر مجموعه {۱۰، ۲۰، ۳۲، ۴۰، ۵۴، ۶۴، ۸۰، ۱۰۷، ۱۶۰} می‌باشند. شکل (۸) فرکانس‌های انتخاب‌شده را برای هرکدام از گره‌های شبکه نشان می‌دهد. توجه کنید که فقط دو گره شماره ۵ و ۱۸ لازم بوده است که با فرکانس بیشینه کار کنند.



شکل ۷: مصرف توان شبکه nobel-eu با سه استراتژی MinLink، MinNode و FreqScal

مدت‌زمانی که CPU برای حل مسئله بهینه‌سازی ارائه‌شده نیاز دارد در جدول (۳) نمایش داده‌شده است. همان‌گونه که مشخص است، روش پیشنهادی در مقایسه با روش MinNode زمان بیشتری برای حل نیاز دارد ولی در عوض مصرف انرژی شبکه را نیز به‌طور مؤثری کاهش می‌دهد. در مقایسه با روش MInLink روش پیشنهادی نه‌تنها به کاهش مصرف توان شبکه کمک زیادی می‌کند، پیچیدگی محاسباتی کمتری نیز دارد به‌نحوی‌که پیچیدگی زمانی MinLink بیش از ۴۴ برابر روش FreqScal می‌باشد. این در حالی است که مصرف توان MinLink نیز حدود ۱/۶ برابر مصرف توان FreqScal است.

جدول ۳: زمان حل مسئله

	۱۰۱۹
	۲۴
	۶۵

۶- نتیجه‌گیری

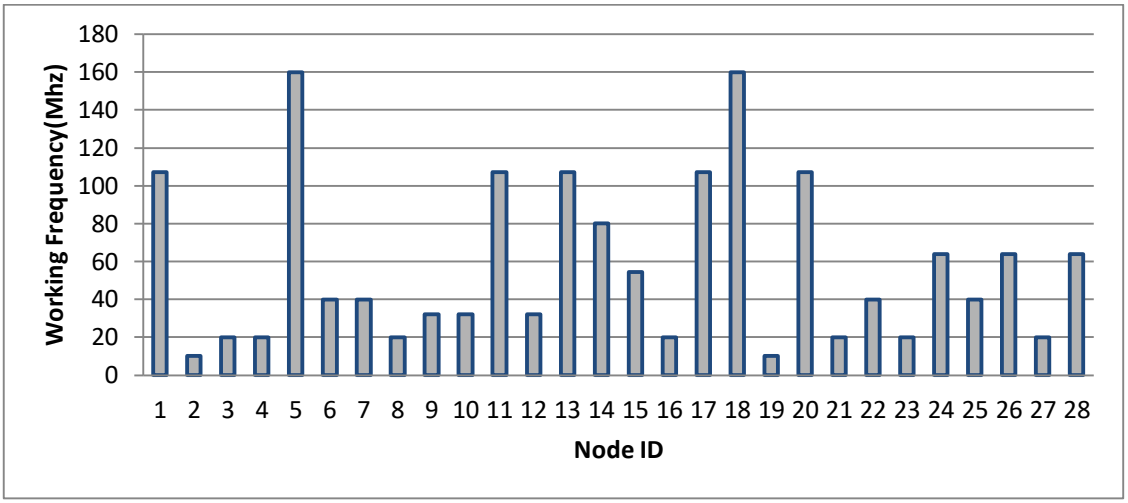
در این مقاله روشی برای کاهش مصرف توان سوئیچ‌های Openflow در معماری شبکه SDN ارائه گردید که مبتنی بر تنظیم فرکانس کار سوئیچ با توجه به بار ترافیکی آن می‌باشد. با توجه به کاربرد روزافزون FPGA در زیرساخت‌های شبکه، پیاده‌سازی سوئیچ Openflow بر روی تراشه‌های FPGA در نظر گرفته شد. لذا با انجام آزمایش‌های متعدد پروفایل توان مصرفی FPGA برحسب بار و فرکانس کار سوئیچ استخراج گردید. ایده تنظیم فرکانس با مسئله استاندارد مسیریابی جریان‌های داده تلفیق گردید تا به‌صورت یک مسئله بهینه‌سازی MILP برای شبکه‌ای از سوئیچ‌های به هم متصل اعمال گردد. با حل این مسئله بر روی یک توپولوژی استاندارد حاصل از کتاب‌خانه SNDlib و تنظیم فرکانس کار سوئیچ‌ها، کارایی روش پیشنهادی در مقایسه با دو روش متداول دیگر نشان داده شد.

نکته قابل‌ذکر این است که روش پیشنهادشده در این مقاله با تکیه‌بر قابلیت تنظیم فرکانس موجود در تراشه‌های FPGA ارائه‌شده است. بنابراین تجهیزاتی که از تراشه‌های ASIC با فرکانس کار ثابت در ساخت سوئیچ‌های داده استفاده می‌کنند نمی‌توانند از این روش بهره‌مند شوند.

در آینده پیشنهاد تعمیم پروتکل Openflow برای انتقال پیام‌های کنترلی تنظیم فرکانس بین کنترل‌کننده مرکزی و سوئیچ‌های داده ارائه خواهد شد. همچنین ارائه یک الگوریتم بلادرنگ برای حل مسئله بهینه‌سازی در پژوهش‌های بعدی مدنظر قرار می‌گیرد.

www.SID.ir

Archive of SID



شکل ۸: فرکانس کار گره‌های شبکه درروش FreqScal

مراجع

[1] L. Chiaraviglio, M. Mellia, and F. Neri, "Minimizing ISP Network Energy Cost: Formulation and Solutions," *IEEE/ACM Trans. Netw.*, vol. 20, no. 2, pp. 463–476, Apr. 2012.

[2] B. Addis, A. Capone, G. Carello, L. G. Gianoli, and B. Sansò, "Energy management in communication networks: a journey through modelling and optimization glasses," *arXiv Prepr. arXiv1507.02636*, vol. 92, pp. 76–94, 2015.

[3] E. Gelenbe and Y. Caseau, "The impact of information technology on energy consumption and carbon emissions," *Ubiquity*, vol. 2015, no. June, pp. 1–15, 2015.

[4] Open Networking Foundation, "*Software-Defined Networking: The New Norm for Networks*", Internet: https://www.opennetworking.org/images/stories/downloads/openflow/wp-sdn-newnorm.pdf, Apr. 2012.

[5] "NetFPGA 10 G openflow switch." https://github.com/NetFPGA/NetFPGA-public/wiki/NetFPGA-10G-OpenFlow-Switch, 12 Jul 2012.

[6] OpenFlow, "*OpenFlow Switch Specification Version: 1.1.0*", Internet: http://www.openflow.org/documents/openflow-spec-v1.1.0.pdf, FEB 28, 2011

[7] I. Takouna, W. Dawoud, and C. Meinel, "Accurate mutlicore processor power models for power-aware resource management," *Proc. - IEEE 9th Int. Conf. Dependable, Auton. Secur. Comput. DASC 2011*, pp. 419–426, 2011.

[8] E. Gelenbe and Y. Caseau, "The impact of information technology on energy consumption and carbon emissions," *Ubiquity*, vol. 2015, no. June, pp. 1–15, Jun. 2015.

[9] B. Addis, D. Belabed, M. Bouet, and S. Secci, "Virtual network functions placement and routing optimization," *2015 IEEE 4th Int. Conf. Cloud Netw.*, pp. 171–177, 2015.

[10] F. Giroire, J. Moulierac, and T. K. Phan, "Optimizing rule placement in software-defined networks for energy-aware routing," in *2014 IEEE Global Communications Conference*, pp. 2523–2529, 2014.

[11] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright, "Power Awareness in Network Design and Routing," in *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*, pp. 457–465, 2008.

[12] R. Bolla *et al.*, "Cutting the energy bills of Internet Service Providers and telecoms through power management: An impact analysis," *Comput. Networks*, vol. 56, no. 10, pp. 2320–2342, Jul. 2012.

[13] R. Bolla, R. Bruschi, A. Carrega, and F. Davoli, "Green Networking With Packet Processing Engines: Modeling and Optimization," *IEEE/ACM Trans. Netw.*, vol. 22, no. 1, pp. 110–123, Feb. 2014.

[14] W. Fisher, M. Suchara, and J. Rexford, "Greening backbone networks," in *Proceedings of the first ACM SIGCOMM workshop on Green networking - Green Networking '10*, p. 29, 2010.

[15] G. Lin, S. Soh, K.-W. Chin, and M. Lazarescu, "Efficient heuristics for energy-aware routing in networks with bundled links," *Comput. Networks*, vol. 57, no. 8, pp. 1774–1788, 2013.

[16] G. Dhiman, G. Marchetti, and T. Rosing, "vGreen," in *Proceedings of the 14th ACM/IEEE international symposium on Low power electronics and design - ISLPED '09*, p. 243, 2009.

[17] I. Hwang and M. Pedram, "A Comparative Study of the Effectiveness of CPU Consolidation Versus Dynamic Voltage and Frequency Scaling in a Virtualized Multicore Server," *IEEE Trans. Very Large Scale Integr. Syst.*, pp. 1–14, 2015.

[18] Jian Tang, B. Mumey, Yun Xing, and A. Johnson, "On exploiting flow allocation with rate adaptation for green networking," in *2012 Proceedings IEEE INFOCOM*, pp. 1683–1691, 2012.

[19] J. Galán-Jiménez and A. Gazo-Cervero, "Using bio-inspired algorithms for energy levels assessment in energy efficient wired communication networks," *J. Netw. Comput. Appl.*, vol. 37, pp. 171–185, 2014.

[20] C. Gunaratne, K. Christensenf, and B. Nordman, "Managing energy consumption costs in desktop PCs and LAN switches with proxying, split TCP connections, and scaling of link speed," *Int. J. Netw. Manag.*, vol. 15, no. 5, pp. 297–310, Sep. 2005.

[21] "IEEE 802.3.az." [Online]. Available: http://standards.ieee.org/findstds/standard/802.3az-2010.html, 2010.

[22] Junqing Sun, G. D. Peterson, and O. O. Storaasli, "High-Performance Mixed-Precision Linear Solver for FPGAs," *IEEE Trans. Comput.*, vol. 57, no. 12, pp. 1614–1623, Dec. 2008.

[23] M. Ejlali, M. A. Montazeri, H. Saidi, and A. Ghiasian, "Design and implementation of a shared memory switch fabric," *6th Int. Symp. Telecommun.*, pp. 721–727, Nov. 2012.

[24] "NetFPGA." [Online]. Available: http://netfpga.org, 2008.

[25] J. Naous, D. Erickson, G. A. Covington, G. Appenzeller, and N. McKeown, "Implementing an OpenFlow switch on the NetFPGA platform," in *Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications Systems - ANCS '08*, pp. 1–9, 2008.

www.SID.ir

Archive of SID

[۲۶] پ دری، ع قیاسیان، ح سعیدی، «طراحی و پیاده‌سازی برای خطوط پرسرعت»، مجله FPGA در بستر AES رمزنگار مهندسی برق دانشگاه تبریز، دوره ۴۶، شماره ۱، ۱۵۳-۱۶۷، ۱۳۹۵.

[27] D. S. May, V. Fpgas, and V.- Fpga, "Virtex-5 FPGA Data Sheet: DC and Switching Characteristics," vol. 152, pp. 1–65, 2013.

[28] "NetFPGA 10G openflow switch." [Online]. Available: https://github.com/NetFPGA/NetFPGA-public/wiki/NetFPGA-10G-OpenFlow-Switch, 12 Jul 2012.

[29] PCIEXT-64U-Live insertion PCI bus extender. [Online]. Available: https://datasheets.globalspec.com/ps/4183/ULTRAVIEW/65FDC0D4-20E9-409D-8910-CB3B50654744, 2016.

[30] "SndLib.", Library of test instances for survivable fixed telecommunications network design, http://sndlib.zib.de/home.action?show=/dynamicmatrices.overview.action%3Fframeset, 2011.

[31] R. G. Garroppo, S. Giordano, G. Nencioni, and M. G. Scutellà, "Mixed Integer Non-Linear Programming models for Green Network Design," *Comput. Oper. Res.*, vol. 40, no. 1, pp. 273–281, Jan. 2013.

زیرنویس‌ها

[۱] Application Specific Integrated Circuit
[۲] Field Programmable Gate Array
[۳] Flow table
[۴] Mixed Integer Linear Programming
[۵] Integer Linear Program
[۶] Ternary Content Addressable Memory
[۷] Static Power Consumption
[۸] Dynamic Power Consumption
[۹] Low Power Idle
[۱۰] Dynamic Voltage and Frequency Scaling
[۱۱] Adaptive Rate
[۱۲] Adaptive Link Rate
[۱۳] Adaptive Core Rate
[۱۴] Packet Generator
[۱۵] Digital Clock Management
[۱۶] Delay Locked Loop

www.SID.ir