مسیریابی چندپخشی کارا در شبکه‌های روی تراشه بازپیکربند

فاطمه نصیری، حمید سربازی آزاد و احمد خادم‌زاده

چکیده: الگوریتم‌های مسیریابی متعددی برای ارتباطات تک‌پخشی و چندپخشی در سیستم‌های روی تراشه چندپردازنده‌ای ارائه شده است. پروتکل‌های چندپخشی برای شبکه‌های روی تراشه در سال‌های اخیر، در هماهنگ‌سازی ساعت، دسترسی‌های مکرر به حافظه‌های مشترک توزیع‌شده، هم‌زمانی و همسان‌سازی حافظه‌های نهان مورد استفاده قرار می‌گیرند. الگوریتم‌های مسیریابی تک‌پخشی برای هدایت بسته‌های چندپخشی مناسب نیستند چون احتمال بروز مشکلاتی از قبیل افزایش ترافیک، ازدحام و بن‌بست را در شبکه روی تراشه بالا می‌برند. از جمله راهکارهای برجسته برای ارتباطات چندپخشی در سیستم‌های چندکامپیوتری عبارتند از الگوریتم‌های مبتنی بر مسیر و الگوریتم‌های مبتنی بر درخت که در سال‌های اخیر این راهکارها به شبکه‌های روی تراشه نیز تعمیم داده شده‌اند. در این مقاله، طرح پیشنهادی با استفاده از شبکه بازپیکربند، سعی در کاهش توان مصرفی و تأخیر بسته‌های چندپخشی در طول مسیر دارد. به طور دقیق‌تر، چنین ساختاری با استفاده از سوییچ‌های ساده در همبندی بازپیکربند به جای مسیریاب‌ها و با بخش‌بندی شبکه به اندازه‌های کوچک‌تر، درخت‌هایی را برای هدایت بسته‌های چندپخشی می‌سازد که منجر به بهبود توان مصرفی و تأخیر ارسال پیام می‌شود. نتایج به دست آمده از شبیه‌سازی بر روی ترافیک‌های واقعی و ساختگی، نشان از برتری روش پیشنهادی ارائه‌شده در مقایسه با روش‌های قبلی مبتنی بر درخت با حداکثر کاهش ۳۳٪ توان مصرفی و ۵۱٪ تأخیر متوسط بسته‌ها دارد.

کلیدواژه: **شبکه روی تراشه، شبکه بازپیکربند، مسیریابی چندپخشی، توان مصرفی، تأخیر متوسط بسته‌ها.**

۱- مقدمه

به دلیل مقیاس‌پذیری ضعیف مساحت و کارایی، معماری‌های سنتی ارتباطی در سیستم روی تراشه[1] مبتنی بر گذرگاه مشترک و اتصالات اختصاصی نقطه به نقطه برای پیاده‌سازی سیستم‌های پیچیده و در مقیاس بزرگ امروزی مناسب نیستند. این چالش از یک سو و مقیاس‌پذیری و کارایی بالای شبکه‌های میان‌ارتباطی در سیستم‌های چندپردازنده‌ای و چندکامپیوتری از سوی دیگر، منجر به طراحی ساختار مشابهی به عنوان جایگزین روش‌های سنتی ارتباطات داخلی با نام شبکه روی تراشه[2] شده است [۱] تا [۳]. همبندی[3] و روش‌های مسیریابی[4] از جمله پارامترهای مهم در معماری شبکه‌های روی تراشه هستند. هدف الگوریتم‌های مسیریابی می‌تواند انتخاب کوتاه‌ترین مسیر، جلوگیری از بن‌بست[5] و سرگردانی[6] بسته‌ها، کاهش توان مصرفی و تأخیر ارسال بسته‌ها باشد. الگوریتم‌های مسیریابی به دو دسته کلی تک‌پخشی[7] و چندپخشی[8] تقسیم‌بندی می‌شوند. در الگوریتم‌های تک‌پخشی، بسته‌ها از یک مبدأ به سمت یک مقصد ارسال می‌شوند در حالی که الگوریتم‌های چندپخشی برای هدایت بسته‌ها از یک مبدأ به چندین مقصد می‌باشند. بنابراین ارتباطات تک‌پخشی زیرمجموعه‌ای از ارتباطات چندپخشی می‌باشند و راهکارهای آنها برای هدایت بسته‌های چندپخشی مناسب نیستند. به طور کلی سه روش مبتنی بر تک‌پخشی[9]، مبتنی بر درخت[10] و مبتنی بر مسیر[11] [۴] تا [۷] برای ارسال بسته‌های چندپخشی ارائه شده است. در روش مبتنی بر تک‌پخشی، مبدأ چندین کپی جداگانه از بسته را به چندین مقصد ارسال می‌نماید که این عمل موجب ایجاد چندین کپی در شبکه می‌شود که منجر به افزایش ترافیک در شبکه خواهد شد و همچنین وجود چندین کپی در مبدأ و در داخل شبکه موجب تأخیر قابل ملاحظه‌ای خواهد شد. به همین دلیل این روش منسوخ شده و از آن در شبکه‌های واقعی استفاده نمی‌شود. در روش مبتنی بر مسیر، ارسال هر بسته از مبدأ به مقصدها از طریق مسیر همیلتون[12] صورت می‌گیرد. در واقع آدرس‌های مقصدها بر اساس موقعیت مکانی‌شان در مسیر همیلتون در سرآیند هر بسته قرار می‌گیرند و در طول مسیر به مقصدها تحویل داده می‌شوند. این روش با افزایش نرخ تزریق در گره مبدأ، منجر به افزایش رقابت در گره مبدأ، کاهش استفاده از لینک‌های مشترک و ارتباطات موازی می‌شود. همچنین به دلیل پیمایش مسیر همیلتون می‌تواند باعث افزایش در تأخیر ارسال پیام گردد. در حالی که در روش مبتنی بر درخت، درخت بر اساس الگوریتم درخت پوشای کمینه[13] ساخته می‌شود که ریشه[14] آن مبدأ و فرزندان[15] درخت در واقع مقصدهای ارتباط چندپخشی می‌باشند. هدف از ساخت درخت هدایت بسته‌ها از مبدأ به مقصدهاست که این امر توسط گره‌های میانی صورت می‌پذیرد. از جمله عیب‌های این روش این است که اگر شاخه‌ای از درخت مسدود شود، کل درخت مسدود خواهد شد. در این حالت بسته‌ها برای چندین سیکل زمانی تعدادی کانال ارتباطی را اشغال کرده‌اند که منجر به افزایش رقابت و درگیری در شبکه خواهد شد و بنابراین این روش برای نرخ ترافیک کم در شبکه مناسب است. به منظور غلبه بر این مشکل می‌توان از مزایای شبکه روی تراشه بازپیکربند

این مقاله در تاریخ ۱۲ تیر ماه ۱۳۹۵ دریافت و در تاریخ ۹ آبان ماه ۱۳۹۶ بازنگری شد.

فاطمه نصیری، دانشگاه آزاد اسلامی واحد علوم و تحقیقات، تهران، (email: fatemeh_nsr@yahoo.com).

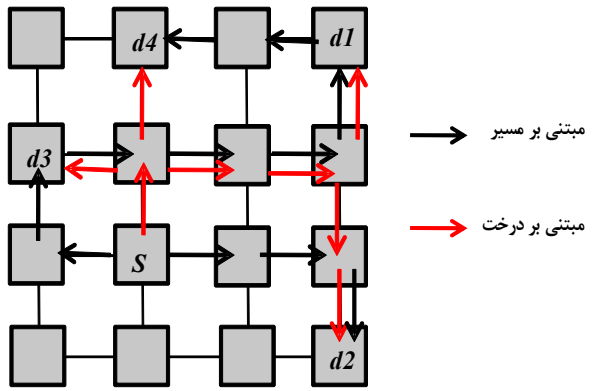
حمید سربازی آزاد، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، (email: azad@sharif.edu).

احمد خادم‌زاده، مرکز تحقیقات مخابرات ایران، تهران، (email: zadeh@itrc.ac.ir).

1. System on Chip
2. Network on Chip
3. Topology
4. Routing
5. Deadlock
6. Livelock
7. Unicast
8. Multicast
9. Unicast-Based
10. Tree-Based
11. Path-Based
12. Hamiltonian Path
13. Minimal Spanning Tree
14. Root
15. Children

Archive of SID

www.SID.ir



شکل ۱: الگوریتم‌های مسیریابی چندپخشی مبتنی بر مسیر و مبتنی بر درخت. خطوط قرمزرنگ، مسیریابی چندپخشی مبتنی بر درخت و خطوط سیاه‌رنگ، مسیریابی چندپخشی مبتنی بر مسیر را نمایش می‌دهند [۸].

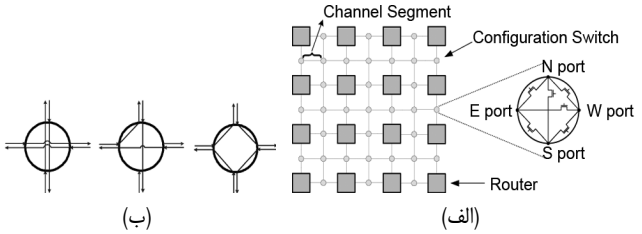
بهره برد. در معماری شبکه روی تراشه بازپیکربند سوئیچ‌های ساده‌ای در مجاورت روترها قرار دارند که می‌توانند در هدایت ترافیک چندپخشی با ساخت درخت پوشای کمینه مؤثر باشند. در الگوریتم STBA و OSTBA تنها با استفاده از سوییچ‌ها درخت‌ها ساخته می‌شوند در حالی که الگوریتم CTBA با استفاده از ترکیب سوییچ‌ها و روترهای میانی بسته‌های چندپخشی را هدایت می‌کند. در این مقاله، الگوریتم پیشنهادی با استفاده از الگوریتم‌های STBA، OSTBA و CTBA [۸] و بخش‌بندی شبکه به سایزهای کوچک‌تر، تلاش در ساخت درخت بیشتر برای هدایت ترافیک چندپخشی روی شبکه روی تراشه بازپیکربند دارد. لذا الگوریتم پیشنهادی برای شبکه‌هایی با اندازه‌های بزرگ‌تر نیز بهینه است. در واقع ساختار الگوریتم پیشنهادی دو گام اصلی است. گام نخست، ساخت درخت از گره مبدأ به یک یا چند گره میانی و گام دوم، ساخت درخت از گره‌های میانی به گره‌های مقصد است. در هر گام برای ساخت درخت، تلاش اول استفاده از سوییچ‌ها به تنهایی همانند الگوریتم STBA و تلاش دوم استفاده از ترکیب سوییچ‌ها و روترها همانند الگوریتم CTBA است. بدین ترتیب با استفاده از سوییچ‌ها به جای روترها و ساخت درخت‌های بیشتر بر روی ساختار بازپیکربند، می‌توان کارایی سیستم را افزایش داد. نتایج حاصل از شبیه‌سازی در ترافیک‌های ساختگی و واقعی، کاهش توان مصرفی و تأخیر ارسال پیام را نسبت به الگوریتم‌های ارائه‌شده پیشین نشان می‌دهد.

ادامه این مقاله به صورت زیر سازمان‌دهی شده است. در بخش ۲ مقدماتی در مورد الگوریتم‌های مسیریابی مبتنی بر درخت و معماری شبکه بر تراشه ارائه شده است. در بخش ۳ کارهای مرتبط پیشین با موارد پیشنهادی مورد بررسی قرار گرفته است. الگوریتم پیشنهادی در بخش ۴ ارائه خواهد شد. در بخش ۵ کارایی الگوریتم پیشنهادی با الگوریتم‌های قبلی مقایسه می‌شوند. در بخش ۶ نیز نتیجه‌گیری مقاله آمده است.

۲- پیش‌زمینه

۲-۱ الگوریتم‌های مسیریابی چندپخشی

همان طور که در بخش قبل نیز اشاره شد اگر برای ارسال بسته‌های چندپخشی با ایجاد کپی‌های متعدد از بسته اصلی استفاده شود، کارایی شبکه به شکل محسوسی افت پیدا می‌کند [۴]. الگوریتم‌های مسیریابی چندپخشی به منظور حل این مشکل ارائه شده‌اند. در الگوریتم‌های مبتنی بر مسیر، بسته‌ها بر اساس موقعیت گره‌های مقصد در مسیر همیلتون هدایت و ارسال می‌شوند. در این روش شبکه بر اساس موقعیت گره مبدأ، به دو زیرشبکه بالایی و پایینی تقسیم‌بندی می‌شود. اگر برچسب گره‌های



شکل ۲: (الف) معماری شبکه بازپیکربند و ساختار داخلی پیکربندی سوییچ‌ها و (ب) سه پیکربندی امکان‌پذیر برای سوییچ‌ها [۹].

مقصد از گره مبدأ بزرگ‌تر باشد، بسته‌ها در زیرشبکه بالایی و در غیر این صورت در زیرشبکه پایینی هدایت می‌شوند. مزیت این روش سادگی آن است ولی اگر گره‌های مقصد در شبکه پراکنده باشند باعث افزایش تأخیر ارسال پیام در مقایسه با الگوریتم‌های مبتنی بر درخت خواهد شد. در الگوریتم‌های مسیریابی چندپخشی مبتنی بر درخت، بسته‌های چندپخشی به کمک درختی که بر اساس روش درخت پوشای کمینه ساخته می‌شود، هدایت می‌شوند. در این روش اگر یکی از شاخه‌های درخت مسدود شود، کل درخت مسدود خواهد شد. این حالت در زمانی که میزان ترافیک در شبکه افزایش یابد محسوس‌تر خواهد شد. نحوه عملکرد الگوریتم‌های مبتنی بر درخت و مبتنی بر مسیر در شکل ۱ قابل مشاهده است.

کاهش انسداد در درخت‌های ساخته‌شده برای هدایت بسته‌های چندپخشی و افزایش کارایی درخت‌های پوشای کمینه، انگیزه‌ای شد تا از سوییچ‌های ساده در شبکه روی تراشه بازپیکربند برای ساخت درخت پوشای کمینه استفاده نماییم. در واقع کنارزدن[1] روترهای میانی و کاهش انسداد در درخت‌های ساخته‌شده باعث افزایش کارایی در شبکه روی تراشه خواهد شد.

۲-۲ معماری بازپیکربند در شبکه روی تراشه

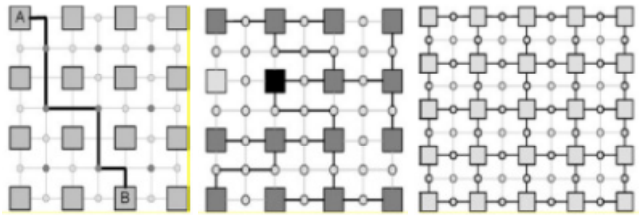
همبندی یکی از مهم‌ترین پارامترها در معماری شبکه روی تراشه است. همبندی‌های متفاوتی برای شبکه روی تراشه ارائه شده‌اند تا بتوانند توان مصرفی و تأخیر ارسال پیام را بهبود بخشند [۹] و [۱۰]. همبندی ارائه‌شده در [۹] و [۱۰] مصالحه مناسبی بین انعطاف‌پذیری در طراحی و سربار مساحت می‌باشد. در این معماری روترهای میانی مستقیم به یکدیگر متصل نشده‌اند بلکه اتصال آنها از طریق سوییچ‌های ساده‌ای به نام سوییچ‌های پیکربند که بین هر دو مسیریاب مجاور در یک شبکه توری قرار گرفته‌اند صورت می‌گیرد. شکل ۲- الف نمایی کلی از معماری شبکه روی تراشه بازپیکربند را نشان می‌دهد. در این شکل، اجزای مربع‌شکل گره‌های شبکه (مسیریاب‌ها و هسته‌های پردازشی متصل به آنها) و اجزای دایره‌ای‌شکل سوییچ‌ها هستند. ساختار داخلی یک سوییچ پیکربند در شکل ۲- الف آمده است. در این شکل برای رعایت سادگی تنها یک اتصال بین هر دو درگاه سوییچ پیکربند رسم شده است اما در واقع این اتصال دوطرفه است. در حقیقت ساختار داخلی یک سوییچ پیکربند متشکل از یک کراس‌بار است اما این کراس‌بار بسیار ساده‌تر از کراس‌بار داخل مسیریاب‌ها می‌باشد. زیرا به دلیل نداشتن درگاه به پردازنده محلی، هر سوییچ پیکربند دارای ۴ درگاه است و لذا کراس‌بار آن ۴×۴ است. به علاوه از آنجا که هیچ اتصال ورودی سوییچ به خودش متصل نمی‌شود برخی از نقاط اتصال کراس‌بار قابل حذف هستند.

بدین ترتیب کراس‌بار به جای ۱۶ نقطه اتصال دارای ۱۲ نقطه است. به عنوان مثال، درگاه خروجی پورت جنوبی می‌تواند به درگاه ورودی پورت

1. Bypass

Archive of SID

www.SID.ir



(ج) (ب) (الف)

شکل ۳: شبکه بازپیکربند با پیکربندی، (الف) مش معمولی، (ب) یک درخت باینری که گره سیاه‌رنگ نشان‌دهنده ریشه درخت است و (ج) سوییچ‌های پیکربندی که فلیت‌ها در آنها بافر می‌شود با رنگ تیره نمایش داده شده است [۹].

شرقی متصل باشد در حالی که درگاه ورودی پورت جنوبی می‌تواند به درگاه خروجی غربی متصل باشد. شکل ۲- ب، سه حالت مختلف از اتصالات داخلی یک سوییچ پیکربند را به تصویر می‌کشد. در این همبندی، چندین میانگیر در فواصل معینی از یک اتصال طولانی قرار می‌گیرند تا به این وسیله داده‌ها به صورت لوله‌ای بر روی اتصال حرکت کرده و زمان زیاد پیمایش اتصال، کلاک سیستم را کاهش ندهد. به هر قسمت از اتصالات که بین دو سوییچ پیکربند و یا یک سوییچ پیکربند و یک مسیریاب قرار داشته باشد یک قطعه اتصال گفته می‌شود. هر اتصال بین دو مسیریاب از دو قطعه اتصال تشکیل شده است. میانگیرها در سوییچ‌هایی قرار می‌گیرند که در همسایگی هیچ مسیریابی قرار نداشته باشند. شکل ۳ سوییچ‌های دربردارنده میانگیر را با رنگ تیره نمایش می‌دهد. در این شکل مشخص است که تمام اتصالات ممکن با بیش از دو قطعه اتصال، به یک میانگیر ختم می‌شوند. خط پررنگ در شکل ۳ یک اتصال بلند ساخته‌شده بین گره‌های A و B را نمایش می‌دهد.

برای توضیح بیشتر، شکل ۴ بخشی از یک سوییچ پیکربندی را که درگاه ورودی شرق به درگاه خروجی غرب را به هم متصل می‌سازد، نمایش می‌دهد. برای وضوح بیشتر، کراس‌بار داخل سوییچ به صورت مجموعه‌ای از مالتی‌پلاکسرها نشان داده شده است. در شکل ۴- الف اتصال بین این دو درگاه در حالت عادی نمایش داده شده است اما شکل ۴- ب، اتصال با استفاده از مالتی‌پلاکسرهای دوسطحی را نشان می‌دهد که بار خازنی کوچک‌تری در مسیر بین دو درگاه قرار می‌دهد و عبور از آن مستلزم صرف توان کمتری است. در این مقاله سوییچ‌های حالت ۴- ب فرض شده‌اند [۱۱].

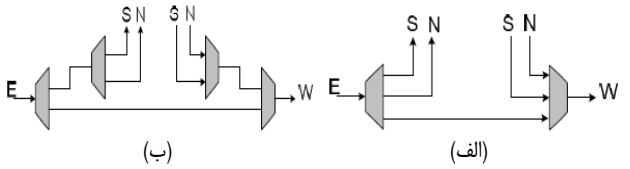
از همبندی شبکه روی تراشه بازپیکربند می‌توان در ساخت درخت برای هدایت ترافیک چندپخشی بهره گرفت. نتایج حاصل از شبیه‌سازی نشان می‌دهد معماری شبکه روی تراشه بازپیکربند ۴۰٪- ۱۲٪ سربار مساحت (با توجه به اندازه‌های مختلف شبکه روی تراشه و سایز بافرها) در مقایسه با شبکه روی تراشه معمولی ایجاد می‌کند [۱۰].

به طور خلاصه ما بر روی طراحی یک روش مبتنی بر سخت‌افزار برای هدایت بسته‌های چندپخشی در شبکه روی تراشه تمرکز می‌کنیم. بهره‌گیری از سوییچ‌های ساده در ساخت درخت به جای روترهای میانی و بخش‌بندی شبکه به اندازه‌های کوچک‌تر به منظور ساخت درخت‌های کارا، امکان بهبود توان مصرفی و تأخیر ارسال پیام را در مقایسه با روش‌های قبلی مبتنی بر درخت فراهم کرده است.

۳- کارهای مرتبط

ارتباطات چندپخشی از موضوعاتی است که در شبکه‌های میان‌ارتباطی[1]

1. Interconnection



(ب) (الف)

شکل ۴: بخشی از یک سوییچ پیکربندی بین درگاه ورودی شرق و درگاه خروجی غرب در (الف) کراس‌بار معمولی و (ب) کراس‌بار بهینه‌شده [۱۱].

و کامپیوتری مورد بررسی قرار گرفته است [۷]. به طور کلی، سه روش عمده برای ارتباطات چندپخشی ارائه شده‌اند که عبارتند از روش‌های مبتنی بر تک‌پخشی، روش‌های مبتنی بر درخت و روش‌های مبتنی بر مسیر. در روش مبتنی بر تک‌پخشی، از یک پیام کپی‌های متعددی از مبدأ برای ارسال به مقصدهایش ایجاد می‌گردد که این امر منجر به افزایش ترافیک در سطح شبکه و در نتیجه افزایش تأخیر در ارسال پیام خواهد شد. امروزه این روش منسوخ شده و در ارتباطات چندپخشی مورد استفاده قرار نمی‌گیرد [۵]. از سویی دیگر، در روش مبتنی بر مسیر برای ارسال بسته‌ها از مسیر همیلتون استفاده می‌گردد. در این روش با توجه به موقعیت گره مبدأ، شبکه به دو زیرشبکه بالایی[2] و پایینی[3] تقسیم‌بندی می‌شود. در زیرشبکه بالایی، تمامی گره‌ها، شماره برچسبی بزرگ‌تر از گره مبدأ دارند. در مقابل، در زیرشبکه پایینی، تمامی گره‌ها شماره برچسبی کوچک‌تر از گره مبدأ دارند. گره مبدأ، شماره گره‌های مقصد را به ترتیب صعودی یا نزولی با توجه به زیرشبکه بالایی و پایینی در سرآیند هر بسته قرار می‌دهد. هر بسته در حین پیمودن مسیر همیلتون، وقتی به روتری می‌رسد شماره مشخصه روتر با شماره اولین گره در لیست سرآیند مقایسه می‌شود. در صورت برابربودن آنها، یک کپی از بسته به گره مربوطه تحویل داده شده، شماره گره از لیست حذف گردیده و بسته به مسیر خود ادامه می‌دهد. چون بسته مجبور به حرکت در مسیر همیلتون است پس این روش عاری از بن‌بست و سرگردانی است. به منظور بهبود این روش، راهکارهای متعددی مانند روش CP[4] [۱۲]، MP[5] [۷]، HAMMUM[6] [۱۳] و HOE[7] [۱۴] ارائه شده‌اند. در روش MP شبکه به چهار زیرشبکه بر اساس موقعیت گره مبدأ تقسیم می‌شود. در این بخش‌بندی، گره‌های دو زیرشبکه پایینی دارای شماره برچسب‌هایی کوچک‌تر از گره مبدأ و بالعکس گره‌ها در دو شبکه بالایی دارای شماره برچسب‌هایی بزرگ‌تر از گره مبدأ می‌باشند. در هر زیرشبکه، مسیر حرکت بسته‌ها بر اساس استراتژی همیلتون از گره مبدأ به سمت گره‌های مقصد است. در روش CP، شبکه با توجه به موقعیت گره مبدأ و موقعیت گره‌های مقصد، به ستون‌های بالایی و پایینی تقسیم‌بندی می‌شود. در صورت وجود گره یا گره‌های مقصد در هر یک از ستون‌ها، یک کپی از بسته از گره مبدأ به آنها ارسال خواهد شد. این دو الگوریتم بر پایه الگوریتم‌های مسیریابی قطعی می‌باشند و عاری از بن‌بست و سرگردانی هستند. اما الگوریتم HAMMUM با استفاده از محدودیت‌های مشخص و به کمک الگوریتم مسیریابی وفقی تلاش در ایجاد الگوریتمی عاری از بن‌بست دارد. الگوریتم HOE نیز عاری از بن‌بست است و نسبت به روش HAMMUM از درجه وفقیت بیشتری برخوردار است. این الگوریتم نیز بر پایه الگوریتم‌های

2. High-Channel
3. Low-Channel
4. Column Path
5. Multi-Path
6. Hamiltonian Adaptive Multicast Unicast Method
7. HAMMUM Odd Even

Archive of SID

www.SID.ir

مبتنی بر مسیر است که از محدودیت‌هایی در چرخش مانند الگوریتم odd-even استفاده می‌کند. این امر انعطاف‌پذیری بیشتری در انتخاب مسیر مناسب در یک شبکه پرازدحام را فراهم می‌کند. دو روش ذکرشده باعث کاهش تأخیر در ارسال پیام و توان مصرفی می‌شوند. روش‌های مبتنی بر مسیر، دارای ساختار سخت‌افزاری ساده‌ای در طراحی هستند و تا زمانی که گره‌های مقصد در نزدیکی یکدیگر قرار دارند عملکرد مناسبی خواهند داشت. در صورتی که گره‌های مقصد به طور پراکنده از یکدیگر در شبکه باشند به دلیل پیمایش مسیر همیلتون، این دسته از الگوریتم‌ها عملکرد مناسبی ندارند و از سوی دیگر منجر به افزایش رقابت در گره مبدأ و کاهش استفاده از لینک‌های مشترک خواهند شد.

روش MDND[1] [۱۵] با بخش‌بندی شبکه به چهار بخش شمالی، جنوبی، شرقی و غربی با توجه به گره مبدأ تلاش در ارسال بسته‌های چندپخشی دارد. در صورت وجود گره یا گره‌های مقصد در راستای شرقی و غربی و همچنین در راستای شمالی و جنوبی یک کپی از بسته ارسال می‌شود. روترهای میانی در صورت نیاز به کپی از همین بسته‌ها استفاده می‌کنند و ممکن است عملیات کپی در گره‌هایی به جز گره‌های مقصد انجام گیرد. در این روش از الگوریتم XY استفاده می‌شود تا عاری از بن‌بست باشد ولی در الگوریتم SMDP[2] [۱۶] با توجه به موقعیت گره مبدأ، شبکه به هشت بخش شمالی، شمال شرقی، شرقی، جنوب شرقی، جنوبی، جنوب غربی، غربی و شمال غربی بخش‌بندی می‌شوند. گره مبدأ به یکی از گره‌های مقصد در هر بخش کپی ارسال می‌نماید. سپس با استفاده از مسیر همیلتون، کپی به تمامی مقصدها تحویل داده می‌شود. الگوریتم ارائه‌شده در [۱۷] باز هم با بخش‌بندی شبکه به هشت بخش، با استفاده از الگوریتم مسیریابی وفقی، تلاش به ارسال بسته‌های چندپخشی دارد. در این راهکار با توجه به ظرفیت بافر روترها در میان چندین مسیر، مسیری که روترها بار ترافیکی کمتری را تحمل می‌کنند برگزیده می‌شود. دو الگوریتم PBSP[3] و APSP[4] در [۱۸] که مبتنی بر مسیر هستند با بخش‌بندی شبکه به چهار بخش و استفاده از کمترین تعداد گام‌ها از گره مبدأ به گره‌های مقصد، ارسال بسته‌های چندپخشی را انجام می‌دهند. این دو الگوریتم با کاهش تعداد گام‌ها باعث کاهش تأخیر در ارسال بسته‌ها می‌شوند.

در روش‌های مبتنی بر درخت، هدایت بسته‌های چندپخشی از طریق درخت پوشای کمینه انجام می‌گیرد. شاخه‌های درخت بر اساس موقعیت گره مبدأ که همان ریشه درخت است و گره‌های مقصد که فرزندان آن به حساب می‌آیند شکل می‌گیرند. هر بسته به سمت یک یا چند کانال خروجی ارسال می‌شود تا شاخه‌های درخت شکل گرفته و بدین ترتیب به تمامی گره‌های مقصد دست یابند. در این متد اگر یکی از شاخه‌های درخت مسدود شود، کل درخت مسدود می‌گردد. انسداد درخت، باعث اشغال‌ماندن چندین کانال ارتباطی برای چندین سیکل زمانی و افزایش رقابت در سطح شبکه برای دستیابی به کانال‌های ارتباطی می‌شود، در نتیجه ازدحام در سطح شبکه افزایش می‌یابد. بنابراین الگوریتم‌های مسیریابی مبتنی بر درخت برای نرخ ترافیک کم در شبکه مناسب می‌باشند. الگوریتم‌های مسیریابی مبتنی بر درخت متعددی برای شبکه روی تراشه ارائه شده است. روش VCTM[5] [۴] از ارسال چندین کپی در سطح شبکه جلوگیری می‌کند ولی به دلیل این که ارسال پیام‌ها بر اساس الگوریتم XY و از جداول مسیریابی مسیریاب صورت می‌گیرد منجر به توان مصرفی نسبتاً بالا و سربار مساحت اشغالی خواهد شد [۴]. دو الگوریتم OPT[6] و LXYROPT[7] [۱۹] به منظور کاهش توان مصرفی نسبت به VCTM ارائه شده‌اند. الگوریتم OPT تلاش در ساخت درخت به کمک الگوریتم غرب‌ابتدا[8] دارد و بدین ترتیب باعث افزایش تأخیر در ارسال پیام می‌شود. الگوریتم LXYROPT گره‌های مقصد را به دو دسته کلی تقسیم‌بندی می‌کند. دسته اول، گره‌هایی هستند که در سمت راست گره مبدأ قرار گرفته‌اند و دسته دوم، بقیه گره‌ها که در سمت چپ گره مبدأ قرار دارند. برای ساخت درخت در سمت چپ از الگوریتم XY و در سمت راست از الگوریتم OPT استفاده می‌شود. دو الگوریتم اشاره‌شده با ساخت درخت به کمک الگوریتم‌های وفقی باعث کاهش توان مصرفی می‌شوند. سه الگوریتم STBA، OSTBA و CTBA [۸] با استفاده از شبکه روی تراشه بازپیکربند و بهره‌مندی از سوییچ‌های ساده تلاش در ساخت درخت دارند. دو الگوریتم STBA و OSTBA تنها از سوییچ‌های ساده در ساخت درخت استفاده می‌کنند ولی الگوریتم CTBA در صورت اشغال‌بودن سوییچ‌ها از روترهای میانی نیز برای ساخت درخت استفاده می‌کند. این سه الگوریتم برای ارسال بسته‌های چندپخشی با کنارزدن روترهای میانی و استفاده از سوییچ‌های ساده باعث کاهش توان مصرفی و تأخیر ارسال پیام می‌شوند. جدول ۱ الگوریتم‌های مسیریابی چندپخشی را به صورت طبقه‌بندی شده، نمایش می‌دهد.

٤- الگوریتم پیشنهادی

همان طور که قبلاً اشاره شد الگوریتم‌های مسیریابی مبتنی بر مسیر مانند CP، MP و DP به کمک الگوریتم‌های قطعی هدایت پیام‌های چندپخشی را انجام می‌دهند در حالی که الگوریتم HAMMUM با استفاده از الگوریتم قطعی و با ایجاد یک سری محدودیت‌هایی در چرخش، ارسال بسته‌های پخشی را انجام می‌دهد. الگوریتم HOE با تغییر الگوریتم قطعی به الگوریتم وفقی در HAMMUM تلاش در هدایت پیام‌های چندپخشی دارد. این الگوریتم‌ها باعث افزایش رقابت در گره مبدأ و کاهش استفاده از لینک‌های اشتراکی در سطح شبکه می‌شوند اما در مجموع این دسته از الگوریتم‌ها، زمانی که گره‌های مقصد در نزدیکی یکدیگر قرار دارند عملکرد مناسبی دارند.

الگوریتم‌های مسیریابی مبتنی بر درخت همچون VCTM، OPT و LXYROPT با ساخت درختی که گره مبدأ به عنوان ریشه و گره‌های مقصد فرزندان آن هستند، پیام‌های چندپخشی را هدایت می‌کنند.

در الگوریتم VCTM ساخت درخت به کمک الگوریتم قطعی XY صورت می‌گیرد اما دو الگوریتم OPT و LXYROPT از الگوریتم وفقی غرب‌ابتدا برای ساخت درخت استفاده می‌کنند. این دسته از الگوریتم‌ها باعث کاهش توان مصرفی می‌شوند ولی اگر یکی از شاخه‌های درخت مسدود شود کل درخت مسدود خواهد شد و بنابراین عملکرد خوبی در میزان ترافیک بالا در شبکه ندارند.

سه الگوریتم STBA، OSTBA و CTBA از شبکه بازپیکربند برای ساخت درخت استفاده می‌کنند. درخت‌ها به کمک سوییچ‌ها و با کنارگذاشتن روترها ساخته می‌شوند. دو الگوریتم STBA و OSTBA

1. Message Duplication in Non Destination
2. Single Message Duplicate in Partition
3. Path Based Shortest Path
4. All Pair Shortest Path
5. Virtual Circuit Tree Multicasting
6. Optimize Tree
7. Left XY Right Optimize Tree
8. West-First

Archive of SID

www.SID.ir

جدول ۱: طبقه‌بندی الگوریتم‌های مسیریابی چندپخشی ارائه‌شده.

پیکربندی	فاکتورهای بهبودیافته		الگوریتم‌های مسیریابی چندپخشی	الگوریتم‌های ارائه‌شده در مقالات تحقیقاتی گذشته
	تأخیر پیام	توان مصرفی		
مش معمولی	√	√	مبتنی بر مسیر	MP.۱۹۹۳ [۵]
مش معمولی	√	√	مبتنی بر مسیر	DP.۱۹۹۳ [۱۲]
مش معمولی	√	√	مبتنی بر مسیر	CP.۱۹۹۸ [۱۲]
مش معمولی	√	–	مبتنی بر درخت	VCTM.۲۰۰۸ [۴]
مش معمولی	√	√	مبتنی بر مسیر	HAMMUM.۲۰۱۰ [۱۳]
مش معمولی	–	√	مبتنی بر درخت	LXYROPT.۲۰۱۱ [۱۹]
مش معمولی	–	√	مبتنی بر درخت	OPT.۲۰۱۱ [۱۹]
مش معمولی	√	√	مبتنی بر مسیر	BPSP.۲۰۱۲ [۲۶]
مش معمولی	√	√	مبتنی بر مسیر	APSP.۲۰۱۲ [۲۶]
مش معمولی	√	√	مبتنی بر مسیر	HOE.۲۰۱۴ [۱۴]
مش بازپیکربند	√	√	مبتنی بر درخت	STBA.۲۰۱۶ [۸]
مش بازپیکربند	√	√	مبتنی بر درخت	OSTBA.۲۰۱۶ [۸]
مش بازپیکربند	√	√	مبتنی بر درخت	CTBA.۲۰۱۶ [۸]
مش معمولی	√	√	مبتنی بر مسیر	SMDP.۲۰۱۶ [۱۶]
مش معمولی	√	√	مبتنی بر مسیر	MDND.۲۰۱۶ [۱۵]

فقط از سوییچ‌ها برای ساخت درخت استفاده می‌کنند ولی الگوریتم CTBA در صورت اشغال‌بودن سوییچ‌ها از یک یا چندین روتر میانی نیز استفاده می‌کند. در این الگوریتم‌ها تلاش برای هدایت ترافیک چندپخشی بر روی شبکه بازپیکربند است. این دسته از الگوریتم‌ها باعث کاهش تأخیر ارسال پیام و توان مصرفی می‌شوند و برای شبکه‌هایی با سایز کوچک‌تر مناسب می‌باشند.

الگوریتم پیشنهادی در این بخش، مبتنی بر درخت است که از سوییچ‌های ساده معماری بازپیکربند استفاده می‌کند که در بخش ۲-۲ به آن اشاره شد. این الگوریتم دارای پنج مرحله اصلی است. گام اول، تفکیک ترافیک تک‌پخشی و چندپخشی است. ترافیک تک‌پخشی بر روی شبکه مش معمولی و با استفاده از الگوریتم قطعی XY هدایت می‌شود. ترافیک چندپخشی بر روی شبکه بازپیکربند با ساخت درخت بر روی سوئیچ‌های میانی ارسال می‌شود. روند ارسال ترافیک چندپخشی و تک‌پخشی با تفکیک پهنای بیتی به دو بخش انجام می‌شود. گام دوم، بخش‌بندی شبکه به چهار بخش مساوی است. گام سوم، ساخت درخت از گره مبدأ به گره میانی در هر بخش است. البته در صورت وجود گره‌های مقصد در آن بخش، درخت‌ها بر اساس دو الگوریتم Kruskal [۲۰] و غرب‌ابتدا ساخته می‌شوند. گره مبدأ به عنوان ریشه درخت در نظر گرفته می‌شود و گره‌های مقصد که در واقع گره‌های میانی درخت اصلی در نظر گرفته می‌شوند، در هر بخش به صورت رندم انتخاب می‌شوند. گام چهارم، ساخت درخت از گره‌های میانی در هر بخش به گره‌های مقصد همان بخش است که باز هم به کمک دو الگوریتم Kruskal و غرب‌ابتدا [۲۱] انجام می‌گیرد. الگوریتم غرب‌ابتدا کمک به ساخت درخت با وفقیت بیشتر می‌کند و الگوریتم Kruskal سعی در ساختن درختی با کمترین وزن دارد. گام آخر، تنظیم‌کردن سوییچ‌ها بر اساس درخت‌های ساخته‌شده است. لازم به ذکر است که در ساخت درخت‌ها از سوییچ‌های ساده استفاده می‌کنیم ولی در صورت اشغال‌بودن سوییچ‌ها از روترهای میانی نیز استفاده می‌شود. این الگوریتم برای شبکه‌هایی با مقیاس‌پذیری بالا عملکرد مناسبی خواهد داشت.

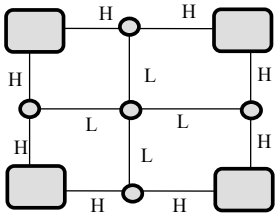
۴-۱ رویکرد الگوریتم پیشنهادی

هر ارتباط چندپخشی با یک گره مبدأ و چندین گره مقصد مشخص می‌شود. هدف ما در اینجا ساخت درخت برای هر ارتباط چندپخشی بر روی شبکه روی تراشه بازپیکربند است. ممکن است که درخت‌های مربوط به ارتباطات چندپخشی مختلف در استفاده از سوییچ‌ها با یکدیگر هم‌پوشانی داشته باشند ولی نباید با یکدیگر در تناقض باشند. به عنوان مثال اگر پورت شرقی یک سوییچ به پورت غربی آن متصل شود، پورت شمالی دیگر نمی‌تواند به پورت غربی اتصال یابد. این واقعیت در ساخت تمامی درخت‌ها باید صادق باشد. هدف در الگوریتم پیشنهادی هدایت بیشتر ترافیک چندپخشی با ساخت تعداد بیشتری درخت بر روی معماری بازپیکربند است که این درخت‌ها با کمترین تعداد لینک ساخته می‌شوند.

برای این منظور الگوریتم پیشنهادی در این مقاله از سوییچ‌های ساده شبکه بازپیکربند، تفکیک پهنای باند، تقسیم‌بندی ترافیک، ساخت درخت به روش وفقی و تقسیم‌بندی شبکه بهره می‌برد. این روش در مقایسه با روش‌های قبلی، تلاش در ساخت درخت بیشتر بر روی شبکه بازپیکربند دارد که باعث کاهش توان مصرفی و تأخیر در ارسال پیام می‌شود. همان طور که قبلاً اشاره شد، ترافیک به دو بخش کلی تک‌پخشی و چندپخشی تقسیم‌بندی می‌شود. ترافیک تک‌پخشی بر روی شبکه مش معمولی با استفاده از الگوریتم قطعی XY و ترافیک چندپخشی بر روی شبکه بازپیکربند هدایت می‌شود. این روند با استفاده از تفکیک پهنای بیتی به دو بخش تک‌پخشی و چندپخشی انجام می‌شود. البته پهنای بیتی بیشتر به ترافیک تک‌پخشی اختصاص دارد چون در کاربردهای واقعی میزان نرخ ارتباطات چندپخشی از تک‌پخشی بیشتر است [۲۲]. بعد از تقسیم‌بندی شبکه به چهار بخش مساوی، نوبت به ساخت درخت می‌رسد. ساخت درخت از گره مبدأ که به عنوان ریشه درخت در نظر گرفته می‌شود به گره‌های میانی در هر بخش انجام می‌گیرد. گره‌های میانی به صورت رندم انتخاب می‌شوند. البته اگر گره‌های مقصد در آن بخش وجود داشته باشند، گره‌ای به صورت رندم به عنوان گره میانی در نظر گرفته می‌شود. در ساخت درخت از الگوریتم‌های kruskal و غرب‌ابتدا استفاده می‌گردد. در گام بعدی، ساخت درخت از گره میانی هر بخش به گره‌های مقصد در آن

Archive of SID

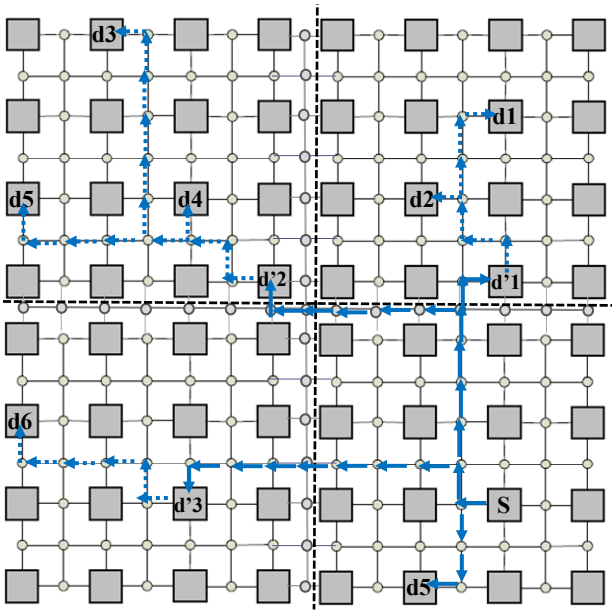
www.SID.ir



شکل ۵: وزن لینک‌ها در شبکه بازپیکربند [۸].

بخش انجام می‌شود. در این مرحله، گره میانی در هر بخش به عنوان ریشه درخت در نظر گرفته می‌شود و درخت از گره مبدأ به گره‌های مقصد ساخته می‌شود. در این مرحله نیز از الگوریتم‌های kruskal و غرب‌ابتدا برای ساخت درخت استفاده می‌شود. تفکیک ساخت درخت به دو مرحله موجب استفاده بهتر از سوییچ‌ها و در نهایت هدایت ترافیک چندپخشی بیشتری بر روی شبکه بازپیکربند می‌شود. در هر دو مرحله ذکرشده در بالا، درخت از گره مبدأ به عنوان ریشه با افزودن شاخه‌های اضافی به کمک دو الگوریتم ذکرشده به سمت گره‌های مقصد حرکت می‌کند. ساخت درخت با استفاده از الگوریتم غرب‌ابتدا که الگوریتمی وفقی و عاری از بن‌بست است باعث می‌شود تا الگوریتم پیشنهادی وفقیت بیشتری در ساخت درخت با توجه به شرایط موجود در شبکه داشته باشد. از طرف دیگر، الگوریتم kruskal کمک می‌کند تا درختی با کمترین وزن ساخته شود. برای نیل به این مقصود، به لینک‌هایی که مابین اتصال دو سوییچ قرار گرفته وزن کمتر و به لینک‌هایی که از یک سو به سوییچ و از سوی دیگر به روتر متصل شده باشند وزن بیشتری اختصاص می‌دهیم. شکل ۵ نمایی از وزن‌های تعبیه‌شده در شبکه را نشان می‌دهد. همان طور که در شکل ملاحظه می‌کنید لینک‌های با وزن کمتر با حرف L و لینک‌های با وزن بیشتر با حرف H مشخص شده‌اند.

اجزای دایره‌ای مشخص‌کننده سوییچ‌های ساده و اجزای مستطیل‌شکل مشخص‌کننده روتر و عناصر پردازشی می‌باشند. درخت با استفاده از الگوریتم kruskal، با کمترین تعداد لینک‌های کم‌وزن ساخته می‌شود و در عین حال دور یا چرخشی هم در درخت وجود ندارد. در هنگام ساخت درخت، از گره مبدأ به گره‌های میانی و یا از گره‌های میانی به گره‌های مقصد، در گام اول ساخت درخت فقط از سوییچ‌ها استفاده می‌شود. همانند عملکرد الگوریتم STBA که در [۸] ذکر شده و در واقع فقط از لینک‌های L برای ساخت درخت استفاده می‌شود. البته لینک‌هایی که به گره مبدأ و یا گره‌های مقصد و میانی از یک سو متصل می‌شوند از نوع H می‌باشند. برای هدایت بسته‌های چندپخشی بیشتر زمانی که لینک‌ها اشغال شدند، در گام دوم ساخت درخت امکان هم‌پوشانی بر روی درخت‌ها نیز امکان‌پذیر است. همان طور که قبلاً اشاره شد به دلیل نرخ بالای ترافیک تک‌پخشی در کاربردهای واقعی، ۷۵٪ از پهنای بیتی هر لینک به ترافیک تک‌پخشی و ۲۵٪ از پهنای بیتی هر لینک به ترافیک چندپخشی اختصاص می‌یابد. بنابراین در لینک‌های از نوع L که فقط ارتباطات چندپخشی از آنها استفاده می‌کنند امکان هم‌پوشانی چهار درخت متفاوت وجود دارد. این عملکرد همانند الگوریتم OSTBA [۸] است و با چنین هم‌پوشانی‌ای امکان ساخت درخت بیشتری بر روی همبندی بازپیکربند فراهم می‌شود. در این مرحله نیز مانند قبل از دو الگوریتم غرب‌ابتدا و kruskal برای ساخت درخت استفاده می‌شود. بدین ترتیب درخت‌های به صورت وفقی و با کمترین وزن ساخته می‌شوند. در گام سوم با اشغال‌شدن سوییچ‌ها و استفاده از لینک‌ها به صورت اشتراکی، به منظور ساخت درخت بیشتر، علاوه بر سوییچ‌ها از روترهای میانی هم استفاده می‌کنیم. پیام‌های



شکل ۶: روند کار الگوریتم پیشنهادی (خطوط ساده مسیریابی از گره مبدأ به گره‌های میانی و خطوط نقطه‌چین مسیریابی از گره‌های میانی به گره‌های مقصد را نمایش می‌دهند).

چندپخشی با عبور از دو یا چند روتر میانی به سوییچ‌هایی دست می‌یابند که می‌توانند ازآنجا روند ساخت درخت را ادامه دهند. در این مرحله نیز از دو الگوریتم غرب‌ابتدا و kruskal استفاده می‌شود. در مرحله آخر، تنظیم‌کردن پورت‌های ورودی و خروجی سوییچ‌ها بر اساس درخت‌های ساخته‌شده است. بعد از اتمام این مراحل که امکان ساخت درخت بیشتر بر روی معماری بازپیکربند وجود ندارد از شبکه مش معمولی و الگوریتم VCTM برای هدایت بسته‌های چندپخشی استفاده می‌شود. لازم به ذکر است اگر گره‌های مقصد و گره مبدأ در یک بخش قرار داشته باشند دیگر در ساخت درخت نیاز به گره‌های میانی نخواهد بود و درخت در همان ناحیه ساخته می‌شود.

شکل ۶ نمونه‌ای از مسیریابی توسط الگوریتم پیشنهادی را نشان می‌دهد. مسیریابی در دو مرحله کلی انجام می‌گیرد: در مرحله اول ارسال پیام‌ها از گره مبدأ به گره‌های میانی و در مرحله دوم ارسال پیام‌ها از گره‌های میانی به گره‌های مقصد است که در هر دو مرحله ساخت درخت بر اساس سوییچ‌ها و در صورت لزوم بر اساس روترهای میانی صورت می‌گیرد. لازم به ذکر است اگر گره‌های مقصد در همسایگی گره مبدأ، یعنی در همان بخش باشند، از گره‌های میانی برای ساخت درخت استفاده نمی‌شود.

شکل ۷ شبه‌کد مربوط به الگوریتم پیشنهادی را نمایش می‌دهد. L مجموعه لینک‌های بین سوییچ‌ها و H مجموعه لینک‌های بین سوییچ و روتر را مشخص می‌کند. مجموعه گره‌های مبدأ و گره‌های میانی در P و مجموعه گره‌های میانی و گره‌های مقصد در M مشخص می‌شوند. در مرحله اول به کمک الگوریتم kruskal و غرب‌ابتدا درخت‌های مربوط به گره‌های مبدأ و گره‌های میانی (p_i, Q_i) با استفاده از لینک‌های کم‌وزن ساخته می‌شوند. امکان چهار هم‌پوشانی بر روی لینک‌های کم‌وزن وجود دارد ($CountL = ۴$). بعد از اشغال‌شدن لینک‌های کم‌وزن، از روترهای میانی استفاده می‌شود. در صورت عدم ساخت درخت با استفاده از همبندی بازپیکربند از مش معمولی استفاده می‌شود. در مرحله دوم برای ساخت درخت از گره‌های میانی به گره‌های مقصد نیز به همان ترتیب که گفته شد با کمک الگوریتم kruskal و غرب‌ابتدا درخت‌های مربوط به گره‌های

Archive of SID

www.SID.ir

```
1.  Start to make tree by the source node's label;
2.  L = Set of links between two switches;
3.  H = Set of links between one switch and one router;
4.  P = Set of sources of different trees to middle destinations;
5.  M = Set of middle sources of different trees to end destinations;
    //each middle source/destination per section select randomly
6.  CountL = 4;
    //each link in L can be used by 4 different trees
7.  FOR all p_i ∈ P
8.      Q_i = all the set of destination nodes of p_i ;
9.      K = Kruskal (p_i, Q_i, "West-first"", L union H, CountL);
10.     IF (Count L > 0 for L)
        //K is the set of used links
11.         L = L – K ;
12.         H = H – K ;
13.         FOR each L
14.             CountL – – ;
15.     ELSE Use VCTM;
        //cannot find any tree for P_i source and Q_i destinations
16. End FOR
17. FOR all m_i ∈ M
18.     Q_i = all the set of destination nodes of m_i ;
19.     K = Kruskal (m_i, Q_i, "West-first", L union H, CountL);
20.     IF (Count L > 0 for L)
        //K is the set of used links
21.         L = L – K ;
22.         H = H – K ;
23.         FOR each L
24.             CountL – – ;
25.     ELSE Use VCTM;
        //cannot find any tree for M_i source and Q_i destinations
26. End FOR
```

شکل ۷: شبه‌کد الگوریتم پیشنهادی.

میانی و گره‌های مقصد (m_i, Q_i) با استفاده از لینک‌های کم‌وزن ساخته می‌شوند. امکان چهار هم‌پوشانی بر روی لینک‌های کم‌وزن وجود دارد $(CountL = ۴)$. بعد از اشغال‌شدن لینک‌های کم‌وزن، از روترهای میانی استفاده می‌شود. در صورت عدم ساخت درخت با استفاده از همبندی بازپیکربند از مش معمولی استفاده می‌شود.

۴-۲ عاری از بن‌بست‌بودن الگوریتم پیشنهادی

به این نکته توجه شود که الگوریتم پیشنهادی عاری از بن‌بست است. زمانی که برای ساخت درخت فقط از سوییچ‌ها استفاده می‌شود، به دلیل این که داوری در سوییچ‌ها برای هدایت پیام‌ها صورت نمی‌گیرد، الگوریتم عاری از بن‌بست است. ولی در حالتی که از روترهای میانی نیز برای هدایت بسته‌های چندپخشی استفاده شود، امکان بروز بن‌بست وجود دارد. زمانی که چندین درخت بر روی شبکه ساخته شود، ممکن است بن‌بست به وجود آید و در این حالت برای حل مشکل از روش [۹] استفاده می‌شود. این متد با بررسی تمام درخت‌های ساخته‌شده به دنبال مسیرهایی است که امکان بروز بن‌بست در آنها وجود دارد. در صورت وجود چنین مسیرهایی یک کانال مجازی به روتر مربوط اختصاص می‌دهد. در الگوریتم‌های مبتنی بر درخت امکان بروز بن‌بست زمانی که روتر منتظر به دست آوردن چندین پورت خروجی به طور هم‌زمان برای ارسال بسته‌ها می‌ماند وجود دارد. اگر وابستگی چرخشی در یک روتر برای دو یا چند درخت ایجاد شود احتمال بروز بن‌بست وجود دارد. برای حل این مشکل، در معماری استفاده‌شده در این مقاله، روتر با آزادشدن هر پورت خروجی، ارسال بسته به آن پورت را انجام می‌دهد و دیگر منتظر آزادشدن سایر پورت‌ها نمی‌ماند. روتر یک کپی از بسته ارسالی نگه می‌دارد تا در صورت آزادشدن به آخرین پورت مربوطه ارسال نماید.

۵- ارزیابی کارایی الگوریتم پیشنهادی

در این بخش، ابتدا متد ارزیابی و زیرساخت استفاده‌شده مورد بررسی قرار می‌گیرد و سپس ارزیابی کارایی الگوریتم پیشنهادی و مقایسه آن با الگوریتم‌های قبلی ارائه می‌شود.

۵-۱ مسایل مورد بررسی به عنوان محک ارزیابی

۵-۱-۱ زیرساخت

برای شبیه‌سازی از شبیه‌ساز Xmulator [۱۸] و [۲۳] که مجهز به کتابخانه توان مصرفی Orion-II است، استفاده شده است [۲۴] و [۲۵]. در این شبیه‌ساز از دو زبان نسل جدید یعنی C# و XML استفاده گردیده و نام آن مخفف دو کلمه XML و Simulator هست. این ابزار دارای قابلیت توسعه بالایی است و به کاربران اجازه می‌دهد که پیکربندی و تغییرات مورد نیاز را با داشتن آگاهی از ساختار داخلی شبیه‌ساز پیاده‌سازی نمایند.

کاربر امکان اشکال‌زدایی و ورود به هسته شبیه‌ساز را داراست و به علاوه، جهت محاسبه توان مصرفی شبکه‌های روی تراشه، کتابخانه توان Orion-II به این شبیه‌ساز افزوده شده است. نتایج شبیه‌سازی انجام‌شده توسط این ابزار با نتایج مدل‌های ریاضی و نتایج شبیه‌سازهای دیگر مقایسه و تأیید شده است. برای محاسبه توان مصرفی توسط Orion-II، فناوری مدار را ۴۵ نانومتر و فرکانس کاری را ۲ GHz در نظر گرفته‌ایم. این شبیه‌ساز امکان تعریف جزئیات ریز سخت‌افزاری و الگوریتم‌ها را ضمن حفظ کارایی داراست. همچنین امکان استفاده از معماری شیءگرا و انواع سوئیچینگ را داراست و به سادگی تعریف بسته‌های جدید، اجزای جدید برای بسته‌های موجود و یا تعریف الگوریتم‌های مسیریابی را فراهم می‌کند. ما از مسیریاب‌های معمولی خط لوله‌ای ۵ مرحله‌ای برای بخش راهگزینی خزشی شبکه روی تراشه استفاده می‌نماییم.

۵-۱-۲ کاربردهای بررسی‌شده

به منظور ارزیابی کارایی الگوریتم پیشنهادی از کاربردهای ساختگی و واقعی استفاده می‌کنیم. در کاربردهای ساختگی، هر عنصر پردازشی می‌تواند با نرخ توزیع یکنواخت پیام تولید کند. آدرس‌های مقصد در ترافیک چندپخشی و تک‌پخشی به صورت رندم به طور یکنواخت از گره‌های موجود در شبکه انتخاب می‌شوند. برای مشاهده اثر الگوریتم پیشنهادی، میزان ترافیک چندپخشی ۳۰٪ و میزان ترافیک تک‌پخشی را ۷۰٪ در نظر گرفته‌ایم. در این حالت تعداد مقصدها در ارتباطات چندپخشی بین ۳ تا ۲۰ مقصد مختلف در سایزهای مختلف شبکه روی تراشه در نظر گرفته شده‌اند.

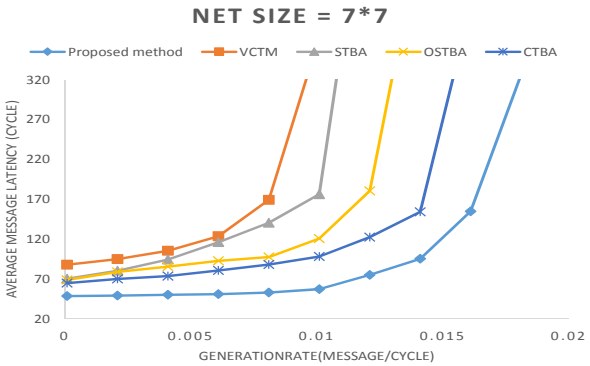
در کاربردهای واقعی، ما از برنامه‌های ترافیک SPLASH-۲ [۲۲] استفاده می‌کنیم. پس از بررسی برنامه‌های SPLASH-۲ دیده شد تنها ۳٪ تا ۱۰٪ از ارتباطات به صورت چندپخشی می‌باشند. به منظور مشاهده اثر الگوریتم پیشنهادی، میزان ارتباطات چندپخشی را با افزودن ارتباطات چندپخشی به طور رندم به ۳۰٪ در برنامه‌های water-spatial، water-nsquared، barnes، raytrace و fft رساندیم.

۵-۱-۳ پارامترهای شبیه‌سازی

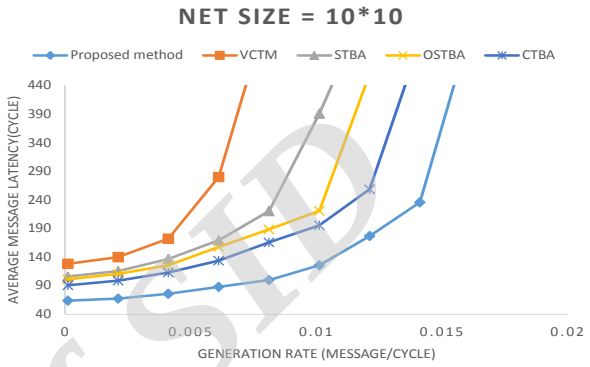
ما پارامترهای شبیه‌سازی را بر شبکه روی تراشه بازپیکربند با سایز ۷×۷، ۱۰×۱۰ و ۱۲×۱۲ برای کاربردهای ساختگی و برای کاربردهای واقعی ۸×۸ در نظر گرفته‌ایم.

Archive of SID

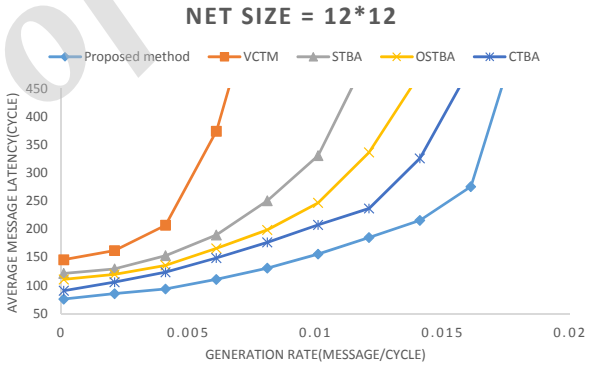
www.SID.ir



شکل ۸: میانگین تأخیر ارسال پیام‌های چندپخشی در شبکه بازپیکربند ۷×۷.



شکل ۹: میانگین تأخیر ارسال پیام‌های چندپخشی در شبکه بازپیکربند ۱۰×۱۰.



شکل ۱۰: میانگین تأخیر ارسال پیام‌های چندپخشی در شبکه بازپیکربند ۱۲×۱۲.

به منظور شبیه‌سازی الگوریتم پیشنهادی، بخش راه‌گزینی خزشی شبکه‌های روی تراشه بازپیکربند از پارامترهای یکسان ۲ کانال مجازی در هر درگاه، میانگیرهای ۸ فلیتی و اتصالات ۱۲۸ بیتی استفاده شده است. برای بازیابی از بن‌بست از روش کانال فرار استفاده می‌شود. در تمامی آزمایش‌ها، پیام‌های تک‌پخشی ۲ و یا ۵ فلیت و پیام‌های چندپخشی ۵ فلیت در نظر گرفته شده‌اند. تمامی پیام‌های تک‌پخشی به کمک الگوریتم xy بر روی مش معمولی هدایت و ارسال می‌شوند و الگوریتم پیشنهادی تنها برای ارسال پیام‌های چندپخشی کاربرد دارد. برای مقایسه الگوریتم پیشنهادی با سایر الگوریتم‌های ارائه‌شده از دو پارامتر توان مصرفی و تأخیر ارسال پیام استفاده می‌نماییم. پارامترهای مربوط به شبیه‌سازی در جدول ۲ قابل مشاهده است.

توان مصرفی: نتایج به دست آمده توان از کتابخانه Orion-II برای شبکه روی تراشه‌هایی که با فرکانس GHz ۲ کار می‌کنند گزارش شده است. نتایج گزارش‌شده از الگوریتم پیشنهادی، مقایسه الگوریتم مسیریابی مبتنی بر درخت بر روی شبکه روی تراشه معمولی و بازپیکربند است که برای هر دو شبکه پارامترهایی یکسان در نظر گرفته شده است.

جدول ۲: پارامترهای شبیه‌سازی.

پارامتر	مقادیر
اندازه شبکه روی تراشه	کاربردهای واقعی ۸×۸ کاربردهای ساختگی ۷×۷ – ۱۰×۱۰ – ۱۲×۱۲
اندازه بسته‌ها	۲ و ۵ فلیت برای تک‌پخشی ۵ فلیت برای چندپخشی
عمق بافر	۸ فلیت
تعداد کانال‌های مجازی	۲
پهنای فلیت	۱۲۸ بیت
روش راه‌گزینی	خزشی

تأخیر ارسال پیام: منظور از تأخیر در ارسال پیام مدت‌زمانی است که اولین فلیت پیام به داخل شبکه ارسال می‌شود تا زمانی که آخرین فلیت آن توسط تمامی گره‌های مقصد دریافت می‌شود.

نتایج آزمایش‌های انجام‌شده بر روی کاربردهای واقعی نشان می‌دهد مجموعه گره‌های مقصد در بازه‌های زمانی متفاوت تکرار می‌شوند. بدین ترتیب زمانی که سوییچ‌ها را برای ساخت درختی برای گره‌های مقصد خاص بر روی شبکه روی تراشه بازپیکربند تنظیم می‌نماییم، می‌توانیم در زمان‌های مختلف به صورت ایستا از آن استفاده کنیم.

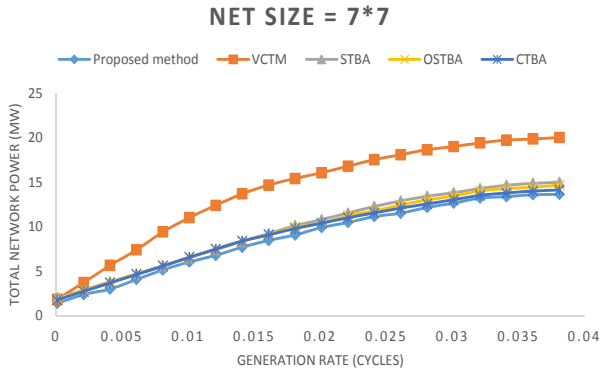
۵-۲ نتایج ارزیابی کارایی الگوریتم پیشنهادی

۵-۲-۱ تحلیل کاربردهای ساختگی

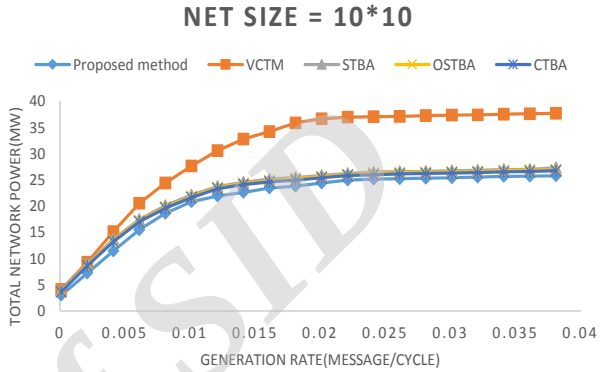
برای ارزیابی کارایی الگوریتم پیشنهادی از سه سایز متفاوت ۷×۷، ۱۰×۱۰ و ۱۲×۱۲ برای کاربردهای ساختگی در شبکه روی تراشه بازپیکربند استفاده می‌کنیم. شکل‌های ۸ تا ۱۰ نتایج حاصل از شبیه‌سازی تأخیر ارسال پیام را برای اندازه‌های مختلف شبکه روی تراشه بازپیکربند نمایش می‌دهند. همان طور که در شکل ۷ مشاهده می‌کنید با فرض اندازه شبکه ۷×۷، الگوریتم STBA، ۱۷٪ بهبود در تأخیر ارسال پیام نسبت به الگوریتم VCTM ایجاد می‌کند. همچنین ۲ الگوریتم OSTBA و CTBA به ترتیب ۱۰٪ و ۱۶٪ بهبود در تأخیر ارسال پیام نسبت به STBA ایجاد می‌کنند. الگوریتم پیشنهادی ۵۲٪ بهبود در تأخیر ارسال پیام نسبت به الگوریتم VCTM و ۱۸٪ بهبود در تأخیر ارسال پیام نسبت به الگوریتم CTBA ایجاد می‌نماید. الگوریتم پیشنهادی ۵۳٪ بهبود در تأخیر ارسال پیام نسبت به الگوریتم VCTM ایجاد می‌کند. در واقع می‌توان گفت الگوریتم پیشنهادی ۲۰٪ بهبود در تأخیر ارسال پیام نسبت به الگوریتم CTBA ایجاد می‌کند. در حالی که الگوریتم STBA ۱۴٪ و الگوریتم OSTBA ۱۹٪ بهبود در تأخیر ارسال پیام نسبت به الگوریتم VCTM ایجاد می‌کنند که این نتایج در شکل ۹ برای اندازه شبکه ۱۰×۱۰ قابل مشاهده است. در شکل ۱۰ با اندازه شبکه ۱۲×۱۲، الگوریتم STBA ۱۷٪ بهبود در تأخیر ارسال پیام نسبت به الگوریتم VCTM ایجاد کرده است. الگوریتم‌های OSTBA و CTBA به ترتیب ۲۵٪ و ۳۸٪ بهبود در تأخیر ارسال پیام به وجود آورده‌اند و ۱۱٪ بهبود در تأخیر ارسال پیام الگوریتم پیشنهادی نسبت به الگوریتم CTBA را نشان می‌دهد. نتیجه بهبود به طور متوسط ۵۱٪ الگوریتم پیشنهادی در مقایسه با الگوریتم VCTM و یا بهبود به طور متوسط ۱۷٪ الگوریتم پیشنهادی در مقایسه با الگوریتم CTBA به دلیل استفاده از سوییچ‌ها به جای روترهای میانی است و از طرف دیگر به دلیل تقسیم‌بندی شبکه به چهار بخش مساوی و ارسال پیام‌ها به گره‌های میانی و از آنجا به گره‌های مقصد، مدیریتی مناسب در ساخت درخت شکل می‌گیرد. بدین ترتیب

Archive of SID

www.SID.ir



شکل ۱۱: مقایسه توان مصرفی در شبکه بازپیکربند ۷×۷.



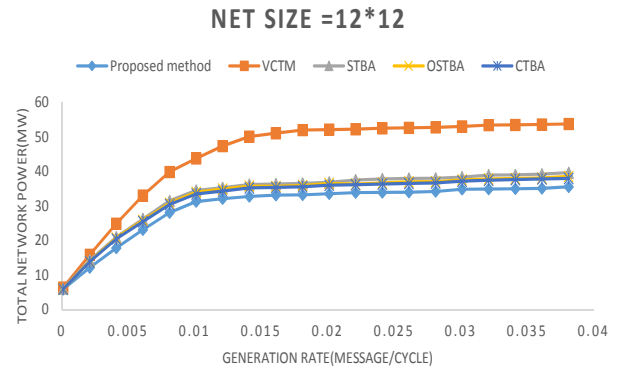
شکل ۱۲: مقایسه توان مصرفی در شبکه بازپیکربند ۱۰×۱۰.

درخت‌های بیشتری بر روی معماری بازپیکربند ساخته می‌شود که منجر به بهبود در ارسال پیام می‌گردد.

شکل‌های ۱۱ تا ۱۳ مقایسه توان مصرفی الگوریتم پیشنهادی را با سایر الگوریتم‌ها نشان می‌دهند. این مقایسه در سایزهای شبکه با تعداد گره‌های ۴۹، ۱۰۰ و ۱۴۴ انجام گرفته است. در این سه شکل، الگوریتم STBA به طور میانگین ۲۲٪ بهبود در توان مصرفی نسبت به الگوریتم VCTM دارد. الگوریتم‌های OSTBA و CTBA به ترتیب ۲۴٪ و ۲۶٪ بهبود در توان مصرفی نسبت به الگوریتم VCTM دارند در حالی که الگوریتم پیشنهادی ۳۳٪ بهبود در توان مصرفی نسبت به الگوریتم VCTM دارد. دلیل کاهش توان مصرفی در الگوریتم پیشنهادی به خاطر استفاده از سوییچ‌ها به جای روترهاست و کاهش تعداد روترهای میانی در هدایت پیام‌های پخشی از گره مبدأ به گره‌های مقصد است.

۵-۲-۲ تحلیل کاربردهای واقعی

برای ارزیابی الگوریتم پیشنهادی از محک چندنخی SPLASH-۲ استفاده می‌کنیم. ارتباطات چندپخشی در محک SPLASH-۲ در پروتکل حافظه‌های نهان دیده شده‌اند [۱۵]. ما از ۵ برنامه SPLASH-۲ به نام‌های water-spatial، water-nsquared، barnes، raytrace و fft استفاده می‌کنیم. شبیه‌سازی در شبکه روی تراشه بازپیکربند ۸×۸ انجام گرفته است. برای هدایت پیام‌های تک‌پخشی از الگوریتم xy بر روی ساختار شبکه روی تراشه معمولی استفاده شده است در حالی که پیام‌های چندپخشی با استفاده از الگوریتم پیشنهادی بر روی شبکه روی تراشه بازپیکربند هدایت می‌شوند. شکل‌های ۱۴ و ۱۵ به ترتیب نرمال‌شده میانگین تأخیر ارسال پیام‌های چندپخشی و نرمال‌شده توان مصرفی شبکه را برای برنامه‌های مختلف SPLASH-۲ نمایش می‌دهند. نتایج حاصل از شبیه‌سازی در شکل ۱۴ نشان می‌دهد الگوریتم پیشنهادی حداکثر ۵۵٪ (و به طور میانگین ۴۹٪) در مقایسه با الگوریتم VCTM بهبود در تأخیر



شکل ۱۳: مقایسه توان مصرفی در شبکه بازپیکربند ۱۲×۱۲.

ارسال پیام ایجاد می‌کند. در حالی که به طور میانگین ۱۵٪، ۲۶٪ و ۳۶٪ به ترتیب الگوریتم‌های STBA، OSTBA و CTBA بهبود در تأخیر ارسال پیام در مقایسه با الگوریتم VCTM را نشان می‌دهند. پس می‌توان گفت الگوریتم پیشنهادی به طور میانگین ۱۳٪ بهبود در تأخیر ارسال پیام در مقایسه با الگوریتم CTBA ایجاد می‌کند.

الگوریتم پیشنهادی، حداکثر ۳۵٪ (به طور میانگین ۳۱٪) بهبود در توان مصرفی نسبت به الگوریتم VCTM داراست. الگوریتم‌های STBA، OSTBA و CTBA به طور میانگین ۲۲٪ بهبود در توان مصرفی در مقایسه با الگوریتم VCTM را نشان می‌دهند.

۶- نتیجه‌گیری

در این مقاله، الگوریتمی کارا برای مسیریابی پیام‌های چندپخشی در شبکه روی تراشه ارائه شده که مبتنی بر درخت است و با بخش‌بندی شبکه به چهار بخش مساوی و ساخت درخت از گره مبدأ به گره‌های میانی و از گره‌های میانی به گره‌های مقصد هدایت پیام‌های چندپخشی را انجام می‌دهد. البته ساخت درخت با استفاده از سوییچ‌ها به جای روترها است. نتایج حاصل از شبیه‌سازی روی کاربردهای واقعی و ساختگی نشان از برتری الگوریتم پیشنهادی بر الگوریتم‌های پیشین دارد.

شایان ذکر است که ساخت درخت‌ها در الگوریتم پیشنهادی به صورت ایستا صورت می‌پذیرد. برای بهبود کارایی الگوریتم پیشنهادی می‌توان ساخت درخت‌ها را به صورت پویا نیز مورد بررسی قرار داد و همچنین بررسی تفکیک پهنای بیتی بهبودیافته برای هدایت پیام‌های تک‌پخشی و چندپخشی بر روی شبکه مش معمولی و بازپیکربند می‌تواند در آینده مورد بازبینی قرار گیرند.

مراجع

[1] W. J. Dally and B. Towles, "Route packets, not wires: on-chip interconnection networks," in *Proc. Design Automation Conf,. DAC'01*, pp. 684-689, Las Vegas, NV, USA, 22-22 Jun. 2001.

[2] W. O. Cesario, *et al*., "Multiprocessor SoC platforms: a component-based design approach," *IEEE Design & Test of Computers*, vol. 19, no.6, pp. 52-63, Nov.-Dec. 2002.

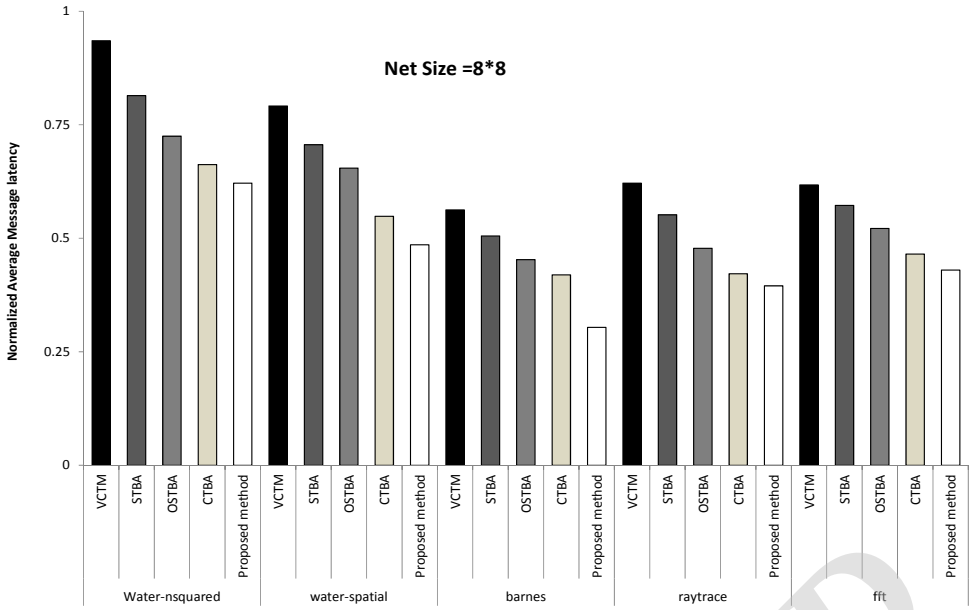
[3] L. Benini and G. De Micheli, "Networks on chips: a new SoC paradigm," *Computer*, vol. 35, no. 1, pp. 70-78, Jan. 2002.

[4] N. E. Jerger, L. S. Peh, and M. H. Lipasti, "Virtual circuit tree multicasting: a case for onchip hardware multicast support," in *Proc. of the Int. Conf. on Computer Architecture,* pp. 229-240, Beijing, China, 21-25 Jun. 2008.

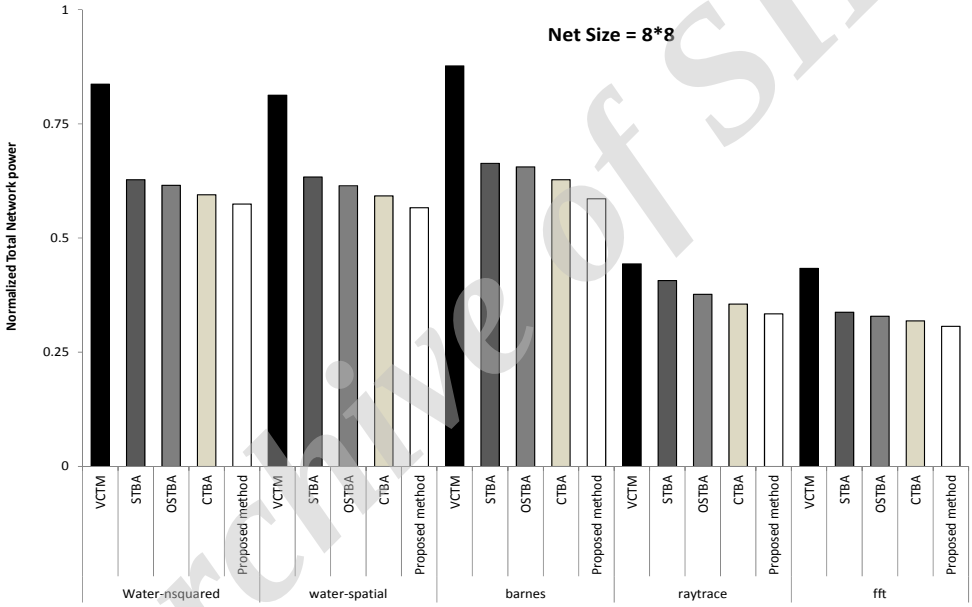
[5] P. McKinely, H. Xu, A. H. Esfahanian, and L. Ni, "Unicast-based multicast communication in wormhole-routed networks," *IEEE Trans. of Parallel and Distributed Systems*, vol. 5, no. 12, pp. 1252-1265, Dec. 1994.

Archive of SID

www.SID.ir



شکل ۱۴: نرمال‌شده میانگین تأخیر ارسال پیام‌های چندپخشی در شبکه بازپیکربند ۸×۸.



شکل ۱۵: نرمال‌شده توان مصرفی در شبکه بازپیکربند ۸×۸.

algorithms," *IEEE Trans. on Parallel and Distributed Systems*, vol. 9, no. 6, pp. 535-549, Jun. 1998.

[13] M. Ebrahimi, M. Danestalab, P. Liljeberg, and H. Tenhunen, "HAMUM-a novel routing protocol for unicast and multicast traffic in MPSocs," in *Proc. 18th Euro Micro Conf. on Parallel, Distributed and Network Based Processing*, *PDP'10,* pp. 525-532, Pisa, Italy, 17-19 Feb. 2010.

[14] P. Bahrebar and D. Stroobandt, "Improving hamiltonian-based routing methods for on-chip networks: a turn model approach," in *Proc. 14th Conf. on Design, Automation & Test in Europe*, *DATE'14,* 4 pp., Dresden, Germany, 24-28 Mar. 2014.

[15] M. R. Arun, P. A. Jisha, and J. Jose, "A novel energy efficient multicasting approach for mesh NoCs," *Procedia Computer Science*, vol. 93, pp. 283-291, 2016.

[16] M. R. Arun and P. A. Jisha, "SMDP-single message duplicate in partition, a multicast routing method in mesh 2D NoC," in *Proc. 3rd Int. Conf. on Innovations in Information Embedded and Communication Systems*, pp. 676-680, 2016.

[17] Z. Wang, *et al.*, "An adaptive partition-based multicast routing scheme for mesh-based networks-on-chip," *Computers & Electrical Engineering*, vol. 51, pp. 235-251, Apr. 2016.

[18] *Xmulator NoC Simulator*, Available: www.xmulator.org, Apr. 2012.

[19] W. Hu, Z. Lu, A. Jantsch, and H. Liu, "Power-efficient tree based multicast support for networks-on-chip," in Proc. *16th Asia and*

[6] E. A. Carara and F. G. Moraes, "Deadlock-free multicast routing algorithm for wormhole-switched mesh networks-on-chip," in *Proc. Computer Society Annual Symp. on the VLSI, ISVLSI'08*, pp. 341-346, Montpellier, France, 7-9 Apr. 2008.

[7] X. Lin and L. M. Ni, "Multicast communication in multicomputer networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. 4, no. 10, pp. 1105-1117, Oct. 1993.

[8] F. Nasiri, H. Sarbazi-Azad, and A. Khademzadeh, "Reconfigurable multicast routing for networks on chip," *Microprocessors and Microsystems*, vol. 42, pp. 180-189, May 2016.

[9] M. Modarressi, A. Tavakkol, and H. Sarbazi-Azad, "Application-aware topology reconfiguration for on-chip networks," *IEEE Trans. on Very Large-Scale Integrated Circuits and Systems*, vol. 19, no. 11, pp. 2010-2022, Nov. 2011.

[10] M. Modarressi and H. Sarbazi-Azad, "Reconfigurable cluster-based networks-on-chip for application-specific MPSoCs," in *Proc. IEEE 23rd Int. Conf. on Application-Specific Systems, Architectures, and Pricessors*, *ASAP'12,* pp. 153-156, Delft, Netherlands, 9-11 Jul. 2012.

[11] Y. Choi and T. M. Pinkston, "Evaluation of crossbar architectures for deadlock recovery routers," *J. of Parallel and Distributed Computing*, vol. 61, no. 1, pp. 49-78, Jan. 2001.

[12] R. V. Boppana, S. Chalasani, and C. S. Raghavendra, "Resource deadlock and performance of wormhole multicast routing

Archive of SID

www.SID.ir

Pacific Design Automation Conf., ASP-DAC'11, pp. 363-368, Yokohama, Japan, 25-28 Jan. 2011.
[20] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, 2001.
[21] C. J. Glass and L. M. Ni, "The turn model for adaptive routing," in *Proc. Int. Conf. Computer Architecture, ISCA'92*, pp. 278-287, Gold Coast, Australia, 19-21 May 1992.
[22] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The SPLASH-2 programs: characterization and methodological considerations," in *Proc. Int. Conf. Computer Architecture, ISCA'95*, vol. 23, pp. 24-36, S. Margherita Ligure, Italy, 22-24 Jun. 1995.
[23] A. Nayebi, S. Meraji, A. Shamaei, and H. Sarbazi-Azad, "XMulator: a listener-based integrated simulation platform for interconnection networks," in *Proc. Int. Conf. on Modelling and Simulation, AMS'07*, pp. 128-132, Phuket, Thailand, 27-30 Mar. 2007.
[24] W. Hang-Sheng, Z. Xinping, P. Li-Shiuan, and S. Malik, "Orion: a power-performance simulator for interconnection networks," in *Proc. 35th Annual IEEE/ACM Int. Symp. on Microarchitecture, MICRO'02*, pp. 294-305, Istanbul, Turkey, 18-22 Nov. 2002.
[25] H. Kahng, B. Li, L. Peh, and K. Samadi, "ORION 2.0: a fast and accurate NoC power and area model for early-stage design space exploration," in *Proc. 9th Conf. on Design, Automation & Test in Europe*, *DATE'09,* pp. 423-428, Nice, France, 20-24 Apr. 2009.

فاطمه نصیری تحصیلات کارشناسی خود را در رشته مهندسی کامپیوتر- سخت‌افزار در سال ۱۳۸۲ از دانشگاه آزاد اسلامی واحد تهران مرکز به پایان رسانده است و همچنین تحصیلات کارشناسی ارشد و دکتری خود را در رشته مهندسی کامپیوتر- معماری کامپیوتر به‌ترتیب در سال‌های ۱۳۸۵ و ۱۳۹۶ از دانشگاه آزاد اسلامی واحد علوم و تحقیقات تهران به پایان رسانده است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: شبکه روی تراشه، سیستم روی تراشه، الگوریتم‌های مسیریابی، سیستم‌های روی تراشه چند هسته‌ای و معماری سیستم‌های چندپردازنده‌ای.

حمید سربازی آزاد تحصیلات خود را در مقطع کارشناسی در رشته مهندسی کامپیوتر سخت‌افزار در سال ۱۳۶۹ در دانشگاه شهید بهشتی ودر سال ۱۳۷۲مقطع کارشناسی ارشد خود را در رشته مهندسی کامپیوتر- معماری کامپیوتر در دانشگاه صنعتی شریف به پایان رسانده است. دکتری علوم کامپیوتر را در دانشگاه گلاسکو انگلستان در سال ۱۳۸۰ اخذ نموده است. دکتر سربازی آزاد استاد دانشگاه صنعتی شریف در گروه مهندسی کامپیوتر و همچنین رئیس بخش علوم کامپیوتر پژوهشگاه دانش‌های بنیادی است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از : معماری کامپیوترهای با کارایی بالا، شبکه روی تراشه و سیستم روی تراشه، سیستم‌های توزیع شده و موازی و ارزیابی و کارایی سیستم‌های ذخیره‌سازی. دکتر سربازی آزاد بیش از ۳۰۰ مقاله علمی در کنفرانس‌ها و ژورنال‌های بین‌المللی به چاپ رسانده است. ایشان در سال ۱۳۸۴ موفق به دریافت جایزه خوارزمی و در سال‌های ۱۳۸۲، ۱۳۸۵، ۱۳۸۶ ، ۱۳۸۸ و ۱۳۹۱ به عنوان پژوهشگر برتر در دانشگاه صنعتی شریف معرفی شدند. ایشان همچنین به عنوان عضو هیأت داوران در مجلات بین‌المللی متعددی همچون CSI و IEEE مشغول به فعالیت هستند.

احمد خادم‌زاده متولد شهر مشهد می‌باشد. ایشان تحصیلات کارشناسی خود را در رشته فیزیک کاربردی در دانشگاه فردوسی مشهد در سال ۱۳۴۷به پایان رسانده‌اند. ایشان تحصیلات کارشناسی ارشد و دکتری خود را به‌ترتیب در رشته‌های اطلاعات و مخابرات دیجیتالی و تئوری اطلاعات و کدینگ از دانشگاه کنت انگلستان اخذ نموده‌اند. دکتر خادم‌زاده به عنوان مشاور و سرپرست گروه مهندسی تست پژوهشگاه ارتباطات و فناوری اطلاعات (مرکز تحقیقات مخابرات ایران) مشغول به همکاری می باشند. دکتر خادم‌زاده به عنوان مدرس در دانشگاه‌های تهران و عضو پایدار هیأت بررسی کنفرانس مهندسی برق ایران می‌باشند. ایشان همچنین چهار مرتبه موفق به دریافت جوایز بین‌المللی خوارزمی و همچنین به عنوان پژوهشگر برتر ملی در وزارت فناوری ارتباطات و اطلاعات انتخاب شده‌اند.

Archive of SID

www.SID.ir